SIMULIA

3DEXPERIENCE

ABAQUS 2025 FD04
EXAMPLE PROBLEMS GUIDE

DASSAULT SYSTEMES

Contents

Trademarks and Legal Notices....9
Abaqus Example Problems Guide....10
What's New....11
Static Stress/Displacement Analyses....16
Static and quasi-static stress analyses....17
Axisymmetric analysis of bolted pipe flange connections....18
Elastic-plastic collapse of a thin-walled elbow under in-plane bending and internal pressure....32
Parametric study of a linear elastic pipeline under in-plane bending....42
Indentation of an elastomeric foam specimen with a hemispherical punch....48
Collapse of a concrete slab....61
Jointed rock slope stability....69
Notched beam under cyclic loading....75
Uniaxial ratcheting under tension and compression....81
Hydrostatic fluid elements: modeling an airspring....88
Shell-to-solid submodeling and shell-to-solid coupling of a pipe joint....99
Stress-free element reactivation....109
Transient loading of a viscoelastic bushing....113
Indentation of a thick plate 117
Damage and failure of a laminated composite plate....125
Analysis of an automotive boot seal....133
Pressure penetration analysis of an air duct kiss seal....137
Self-contact in rubber/foam components: jounce bumper....151
Self-contact in rubber/foam components: rubber gasket....157
Submodeling of a stacked sheet metal assembly....160
Axisymmetric analysis of a threaded connection....166
Direct cyclic analysis of a cylinder head under cyclic thermomechanical loadings....171
Erosion of material (sand production) in an oil wellbore....183
Submodel stress analysis of pressure vessel closure hardware....193
Using a composite layup to model a yacht hull....209
Energy computations in a contact analysis....216
Bending of a beam in a sleeve....224
Fluid pressure penetration analysis of an O-ring seal of a pipe connection....227
Buckling and collapse analyses....237
Snap-through buckling analysis of circular arches....238
Laminated composite shells: buckling of a cylindrical panel with a circular hole....247
Buckling of a column with spot welds....262
Elastic-plastic K-frame structure....268
Unstable static problem: reinforced plate under compressive loads....273
Buckling of an imperfection-sensitive cylindrical shell....278
Forming analyses....285
Upsetting of a cylindrical billet: quasi-static analysis with mesh-to-mesh solution mapping (Abaqus/Standard) and adaptive meshing (Abaqus/Explicit)....286
Superplastic forming of a rectangular box....296
Stretching of a thin sheet with a hemispherical punch....303

Deep drawing of a cylindrical cup....310
Extrusion of a cylindrical metal bar with frictional heat generation....319
Rolling of thick plates....327
Axisymmetric forming of a circular cup....336
Cup/trough forming....340
Forging with sinusoidal dies....347
Forging with multiple complex dies....354
Flat rolling: transient and steady-state....360
Section rolling....370
Ring rolling....377
Axisymmetric extrusion: transient and steady-state....382
Two-step forming simulation....393
Upsetting of a cylindrical billet: coupled temperature-displacement and adiabatic analysis....400
Unstable static problem: thermal forming of a metal sheet....414
Inertia welding simulation using Abaqus/Standard and Abaqus/CAE....417
Moldflow translation examples....431
Fracture and damage....435
A plate with a part-through crack: elastic line spring modeling....436
Contour integrals for a conical crack in a linear elastic infinite half space....441
Elastic-plastic line spring modeling of a finite length cylinder with a part-through axial flaw....461
Crack growth in a three-point bend specimen....469
Analysis of skin-stiffener debonding under tension....475
Failure of blunt notched fiber metal laminates....478
Debonding behavior of a double cantilever beam....497
Debonding behavior of a single leg bending specimen....511
Postbuckling and growth of delaminations in composite panels....521
Import analyses....530
Springback of two-dimensional draw bending....531
Deep drawing of a square box....536
Dynamic Stress/Displacement Analyses....549
Dynamic stress analyses....550
Nonlinear dynamic analysis of a structure with local inelastic collapse....551
Detroit Edison pipe whip experiment....562
Rigid projectile impacting eroding plate....566
Eroding projectile impacting eroding plate....571
Tennis racket and ball....576
Pressurized fuel tank with variable shell thickness....583
Modeling of an automobile suspension....588
Explosive pipe closure....593
Knee bolster impact with general contact....597
Crimp forming with general contact....602
Collapse of a stack of blocks with general contact....607
Cask drop with foam impact limiter....611
Oblique impact of a copper rod....622
Water sloshing in a baffled tank....626
Seismic analysis of a concrete gravity dam....630
Progressive failure analysis of thin-wall aluminum extrusion under quasi-static and dynamic loads....643
Impact analysis of a pawl-ratchet device....654

High-velocity impact of a ceramic target....660
Confined compression of multiple two-dimensional cylinders with different material properties....672
Mode-based dynamic analyses....675
Analysis of a rotating fan using substructures and cyclic symmetry....676
Linear analysis of the Indian Point reactor feedwater line....683
Response spectra of a three-dimensional frame building....697
Brake squeal analysis....700
Dynamic analysis of antenna structure utilizing residual modes....704
Steady-state dynamic analysis of a vehicle body-in-white model....708
Eulerian and co-simulation analyses....714
Rivet forming....715
Impact of a water-filled bottle....725
Dynamic impact of a scooter with a bump....738
Tire and Vehicle Analyses....748
Tire analyses....749
Symmetric results transfer for a static tire analysis....750
Steady-state rolling analysis of a tire....757
Subspace-based steady-state dynamic tire analysis....765
Steady-state dynamic analysis of a tire substructure....770
Coupled acoustic-structural analysis of a tire filled with air....772
Import of a steady-state rolling tire....780
Analysis of a solid disc with Mullins effect and permanent set....785
Tread wear simulation in Abaqus/Standard....794
Dynamic analysis of an air-filled tire with rolling transport effects....802
Acoustics in a circular duct with flow....809
Vehicle analyses....813
Inertia relief in a pick-up truck....814
Substructure analysis of a pick-up truck model....819
Display body analysis of a pick-up truck model....828
Continuum modeling of automotive spot welds....832
Occupant safety analyses....845
Seat belt analysis of a simplified crash dummy....846
Side curtain airbag impactor test....855
Side curtain airbag analysis using the lumped kinetic molecular method....860
Mechanism Analyses....864
Resolving overconstraints in a multi-body mechanism model....865
Geometry and model....865
Loading and boundary conditions....866
Understanding overconstraint messages....866
Correcting the overconstrained model....868
Results and discussion....868
Input files....868
Figures....869
Crank mechanism....872
Problem description....872
Results and discussion....873
Files....873
Tables....873

Figures....874
Snubber-arm mechanism....876
Geometry and model....876
Model interactions....876
Results and discussion....876
Input files....877
Figures....877
Flap mechanism....880
Geometry and model....880
Model interactions....880
Results and discussion....881
Input files....881
Figures....881
Tail-skid mechanism....884
Geometry and model....884
Model interaction....884
Results and discussion....885
Input files....885
Tables....886
Figures....886
Cylinder-cam mechanism....889
Geometry and model....889
Model interactions....889
Results and discussion....889
Files....890
Figures....890
Driveshaft mechanism....893
Geometry and model....893
Model interactions....893
Results and discussion....894
Input files....894
Figures....894
Geneva mechanism....896
Geometry and model....896
Model interactions....896
Results and discussion....896
Input files....897
Figures....897
Trailing edge flap mechanism....899
Geometry and model....899
Model interactions....899
Results and discussion....899
Input files....899
Figures....900
Substructure analysis of a one-piston engine model....901
Geometry and material....901
Models....901
Results and discussion....902

Input files....902
Figures....904
Application of bushing connectors in the analysis of a three-point linkage....907
Geometry and materials....907
Models....907
Results and discussion....912
Files....913
Figures....914
Gear assemblies....920
Application description....920
Abaqus modeling approaches and simulation techniques....921
Case 1: Spur gears....921
Case 2: Rack and pinion....922
Case 3: Screw gear....922
Discussion of results and comparison of cases....923
Input files....924
Figures....924
Electrochemical and Electromagnetic Analyses....928
Fully coupled thermal-electrochemical-structural analysis in batteries....929
Application description....929
Abaqus modeling approaches and simulation techniques....930
Input files....931
References....931
Figures....931
Electromagnetic analyses....933
Eigenvalue analysis of a piezoelectric transducer....934
Transient dynamic nonlinear response of a piezoelectric transducer....940
Thermal-electrical modeling of an automotive fuse....946
Modeling of a silicon piezoresistive pressure sensor....951
Heat Transfer and Thermal-Stress Analyses and Particle Methods....957
Heat transfer and thermal-stress analyses....958
Thermal-stress analysis of a disc brake....959
A sequentially coupled thermomechanical analysis of a disc brake with an Eulerian approach....969
Exhaust manifold assemblage....978
Coolant manifold cover gasketed joint....984
Conductive, convective, and radiative heat transfer in an exhaust manifold....993
Thermal-stress analysis of a reactor pressure vessel bolted closure....998
Coupled thermomechanical analysis of viscoelastic dampers....1009
Particle methods....1019
Mixing of granular media in a drum mixer....1020
Particle cluster drop 1028
Additive Manufacturing Process Simulation....1032
Sequential thermomechanical analysis of a laser powder bed fusion build....1033
Application description....1033
Abaqus modeling approaches and simulation techniques....1034
Heat transfer analysis....1034
Structural analysis....1035

Discussion of results and comparison of cases....1036
Acknowledgements....1037
Input files....1037
References....1037
Tables....1038
Figures....1038
Sequential thermomechanical analysis of a directed energy deposition build....1042
Application description....1042
Abaqus modeling approaches and simulation techniques....1043
Heat transfer analysis....1044
Static structural analysis....1045
Discussion of results and comparison of cases....1045
Files....1046
References....1046
Tables....1046
Figures....1047
Cure Process Simulation....1052
Modeling the Watts test of adhesives....1053
Application description....1053
Abaqus modeling approaches and simulation techniques....1054
Input files....1055
References....1055
Tables....1056
Figures....1056
Prediction of cure-induced spring-in of an angle bracket....1061
Application description....1061
Abaqus modeling approaches and simulation techniques....1062
Input files....1064
References....1064
Tables....1064
Figures....1066
Mass Diffusion and Soils Analyses....1074
Mass diffusion analyses....1075
Hydrogen diffusion in a vessel wall section....1076
Diffusion toward an elastic crack tip....1080
Soils analyses....1087
Plane strain consolidation....1088
Calculation of phreatic surface in an earth dam....1092
Axisymmetric simulation of an oil well....1097
Analysis of a pipeline buried in soil....1104
Hydraulically induced fracture in a wellbore....1108
Permafrost thawing–pipeline interaction....1114
Acoustic and Shock and Abaqus/Aqua Analyses....1122
Acoustic and shock analyses....1123
Fully and sequentially coupled acoustic-structural analysis of a muffler....1124
Coupled acoustic-structural analysis of a speaker....1136
Analysis of a speaker using Abaqus-Dymola co-simulation....1140

Response of a submerged cylinder to an underwater explosion shock wave....1148
Convergence studies for shock analyses using shell elements....1158
UNDEX analysis of a detailed submarine model 1170
Coupled acoustic-structural analysis of a pick-up truck....1183
Long-duration response of a submerged cylinder to an underwater explosion....1193
Deformation of a sandwich plate under CONWEP blast loading....1197
Abaqus/Aqua analyses....1208
Jack-up foundation analyses....1209
Riser dynamics....1213
Structural Optimization Analyses....1218
Topology, shape, and bead optimization analyses....1219
Topology optimization of an automotive control arm....1220
Shape optimization of a connecting rod....1226
Bead optimization of a plate....1233
Sizing optimization analyses....1237
Sizing optimization of a gear shift control holder....1238
Sizing optimization of a car door....1244
Design Sensitivity Analyses....1250
Using Abaqus/CAE to compute shape variations....1251
Defining a shape variation....1251
Design sensitivity analysis of a composite centrifuge....1253
Geometry and model....1253
Materials....1253
Design parameters and normalization....1253
Results and discussion....1254
Input files....1255
Tables....1255
Figures....1256
Design sensitivities for tire inflation, footprint, and natural frequency analysis....1261
Geometry and model....1261
Perturbation sizes for the finite differencing operations....1261
Normalization of sensitivity results....1262
Results and discussion....1262
Input files....1263
Tables....1263
Figures....1264
Design sensitivity analysis of a windshield wiper....1272
Geometry, model properties, and design parameters....1272
Loading, boundary conditions, and design responses....1272
Results and discussion....1273
Input files....1274
Tables....1274
Figures....1275
Design sensitivity analysis of a rubber bushing....1280
Geometry, model properties, and design parameters....1280
Design sensitivity analysis and design parameters....1280
Results and discussion....1280
Input files....1281

Tables....1281
Figures....1282
Postprocessing of Abaqus Results....1285
About user postprocessing of Abaqus results files....1286
Initialization....1286
Data processing....1287
Postprocessing examples....1288
Joining data from multiple results files and converting file format: FJOIN....1289
Calculation of principal stresses and strains and their directions: FPRIN....1292
Creation of a perturbed mesh from original coordinate data and eigenvectors: FPERT....1294
Output radiation view factors and facet areas: FRAD....1297
Creation of a data file to facilitate the postprocessing of elbow element results: FELBOW....1299
Visualizing beam outputs with actual beam cross-section representation for contact....1301
Translating Abaqus data to modal neutral file format for analysis in MSC.ADAMS....1306
Product Index....1311
Abaqus/Standard....1312
Abaqus/Explicit....1318
Abaqus/CAE....1321
Abaqus/Viewer....1322
Abaqus/AMS....1323
Abaqus/Aqua....1324
Abaqus/Design....1325

Trademarks and Legal Notices

Trademarks

Abaqus, **3D**EXPERIENCE®, the 3DS logo, the Compass icon, IFWE, 3DEXCITE, 3DVIA, BIOVIA, CATIA, CENTRIC PLM, DELMIA, ENOVIA, GEOVIA, MEDIDATA, NETVIBES, OUTSCALE, SIMULIA and SOLIDWORKS are commercial trademarks or registered trademarks of Dassault Systèmes, a European company (Societas Europaea) incorporated under French law, and registered with the Versailles trade and companies registry under number 322 306 440, or its subsidiaries in the United States and/or other countries. All other trademarks are owned by their respective owners. Use of any Dassault Systèmes or its subsidiaries trademarks is subject to their express written approval.

Legal Notices

Abaqus and this documentation may be used or reproduced only in accordance with the terms of the software license agreement signed by the customer, or, absent such agreement, the then current software license agreement to which the documentation relates.

This documentation and the software described in this documentation are subject to change without prior notice.

Dassault Systèmes or its Affiliates shall not be responsible for the consequences of any errors or omissions that may appear in this documentation.

© Dassault Systèmes Americas Corp., 2025.

For a full list of the third-party software contained in this release, please go to the Legal Notices in the Abaqus 2025 HTML documentation, which can be obtained from a documentation installation, or in the SIMULIA Established Products 2025 Program Directory, which is available on *www.3ds.com*.

Abaqus Example Problems Guide

The Abaqus Example Problems Guide contains many solved examples from which users can learn how to run simulations involving nontrivial physics.
Some of the problems are quite difficult and require combinations of the capabilities in the code. These examples are intended to provide guidance to users who must work on a class of problems with which they are relatively unfamiliar. In each worked example the discussion in the guide states why the example is included and leads the reader through the standard approach to an analysis: element and mesh selection, material model, and a discussion of the results. Many of these problems are worked with different element types, mesh densities, and other variations.

This guide is a part of the Abaqus® documentation collection, which describes all the capabilities of the Abaqus finite element analysis technology used in SIMULIA® applications.

Input files referred to in this guide are included with the Abaqus release in compressed archive files. To reproduce the graphical representation of the solution reported in some of the problems, modifications to various options (such as step time, display options, output requests, solver precision, output frequency, etc.) may be required.

The **abaqus fetch** utility is used to extract these input files for use. For example, to fetch input file *boltpipeflange_3d_cyclsym.inp*, type

```
abaqus fetch job=boltpipeflange_3d_cyclsym.inp
```

Parametric study script (`.psf`) and user subroutine (`.f`) files can be fetched in the same manner. All files for a particular problem can be obtained by leaving off the file extension. The **abaqus fetch** utility is explained in detail in *Fetching Sample Input Files*.

It is sometimes useful to search the input files. The **findkeyword** utility is used to locate input files that contain user-specified input. This utility is defined in *Querying the Keyword/Problem Database*.

In addition to this guide, there are two other guides that contain worked problems. The *Abaqus Benchmarks Guide* contains benchmark problems that provide evidence that the software can produce a result from a benchmark defined by an external body or institution such as NAFEMS. The tests in this guide are sufficient to show accuracy and convergence compared to benchmark data. The *Abaqus Verification Guide* contains a large number of tests that are intended to provide evidence that the implementation of the numerical model produces the expected results for one or several well-defined options in the code.

The qualification process for new Abaqus releases includes running and verifying results for all problems in this guide, the *Abaqus Benchmarks Guide*, and the *Abaqus Verification Guide*.

What's New

This page describes recent changes in Abaqus Example Problems.

2025 FD03

Fully Coupled Thermal-Electrochemical-Structural Analysis in Batteries

A new example demonstrates the capability of a fully coupled thermal-electrochemical-structural analysis for modeling coupled multiphysics-multiscale behaviors in lithium-ion batteries under mechanical constraints.

Benefits: The test demonstrates the capability of a fully coupled thermal-electrochemical-structural analysis in Abaqus/Standard.

For more information, see *Fully coupled thermal-electrochemical-structural analysis in batteries*.

Tread Wear Simulation in Abaqus/Standard

The example is updated to illustrate a new, simpler approach to modeling tread wear in a steady-state transport procedure.

Benefits: The new approach demonstrates the ease of setting up the model for wear calculations in a steady-state transport procedure and of visualizing wear distances.
In the new approach, you specify wear properties on contact surfaces as usual. However, wear calculations in the steady-state transport procedure are made internally; you do not need to code a user subroutine as you did in the approach available in earlier releases. In addition, the new approach enables you to visualize wear output.

For more information, see *Tread wear simulation in Abaqus/Standard*.

2025 FD02

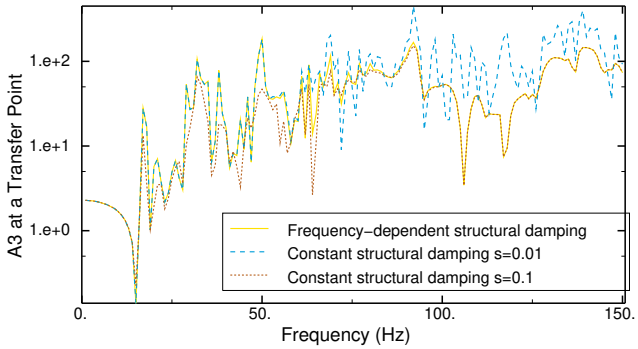
Steady-State Dynamic Analysis of a Vehicle Body-in-White Model

A new example demonstrates the use of frequency-dependent structural damping in the steady-state dynamic analysis of a vehicle body-in-white model.

Benefits: The test demonstrates the use of frequency-dependent structural damping in a steady-state dynamic analysis in Abaqus/Standard.

In the new example, the frequency-dependence of the structural damping is such that s = 0.01 for frequencies lower than 50 Hz and s = 0.1 for frequencies higher than 100 Hz, with a linear transition between the two regions.

The figure below shows the transfer inertance at the node located at the bottom of the driver's seat. As expected, the transfer inertance with frequency-dependent structural damping is identical to that for a constant s = 0.01 structural damping for frequencies lower than 50 Hz and that for a constant s = 0.1 structural damping for frequencies higher than 100 Hz.

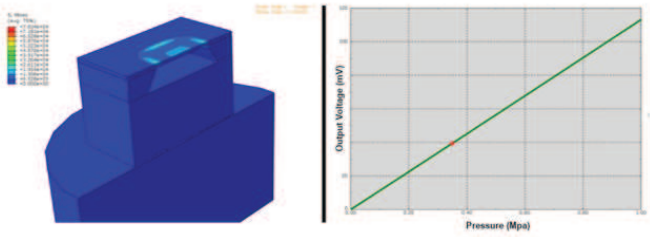


For more information, see *Steady-state dynamic analysis of a vehicle body-in-white model*.

Modeling of a Silicon Piezoresistive Pressure Sensor

A new example illustrates the use of a piezoresistive material model to simulate a silicon piezoresistive pressure sensor.

Benefits: The test demonstrates the use of a piezoresistive material model in Abaqus/Standard.



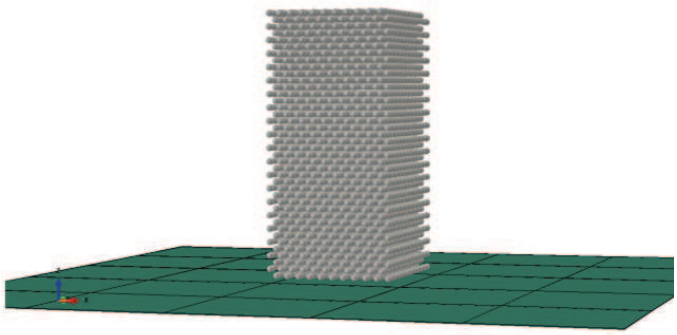
For more information, see *Modeling of a silicon piezoresistive pressure sensor*.

2025 FD01

Particle Cluster Drop

A new example demonstrates the use of particle clustering to describe the nonspherical shape of limestone aggregates that are dropped onto a rigid floor under gravity loading.

Benefits: The test demonstrates the use of clustering in the discrete element method in Abaqus/Standard.



For more information, see *Particle cluster drop*.

PDF Guides Available

You can download PDF versions of the Abaqus guides from each guide overview.

Benefits: You can easily access the PDF versions of the guides.
In earlier releases, PDF versions of the guides were updated only for the GA (General Availability) release and could be downloaded from the Dassault Systèmes Knowledge Base. As of 2025 FD01, the PDF guides will be updated with each release of the HTML versions of the guides.

For more information, see *Abaqus Example Problems Guide*.

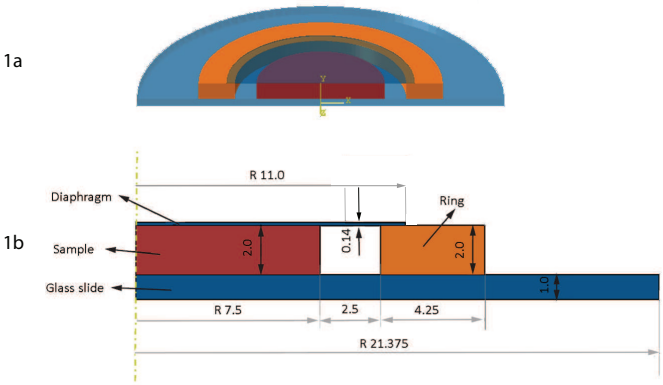
2024 FD04

Modeling the Watts Test of Adhesives

A new example demonstrates how to model the curing process of adhesives.

Benefits: The tests demonstrate the adhesive curing process modeling capability in Abaqus/Standard.

The Watts test measures the polymerization shrinkage of materials. In the Watts test, a disc-shaped specimen is sandwiched between two glass plates, as shown below.



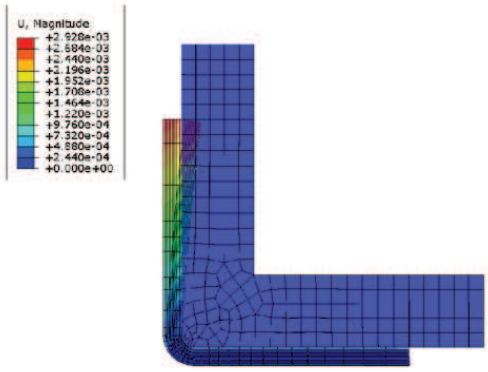
For more information, see *Modeling the Watts test of adhesives*.

2024 FD03

Cure Process Simulation: Prediction of Cure-Induced Spring-In of an Angle Bracket

A new example demonstrates the use of multiscale technology combined with cure modeling in Abaqus/Standard to model the cure-induced spring-in deformation of an angle bracket.

Benefits: The tests demonstrate the composite curing process modeling capability in Abaqus/Standard.



For more information, see *Prediction of cure-induced spring-in of an angle bracket*.

2024 FD01

Failure of Blunt Notched Fiber Metal Laminates

A new test simulates failure and damage in a fiber metal laminate containing a blunt notch subjected to quasi-static loading.

Benefits: The tests demonstrate the capability in Abaqus/Standard.
Multiscale modeling is used.

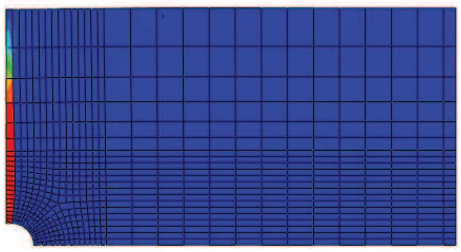


Figure 1: Fiber damage pattern in the 0° fiber-reinforced epoxy layer for the 0° loading direction (using the Hashin damage model, DAMAGEFT contour plot).

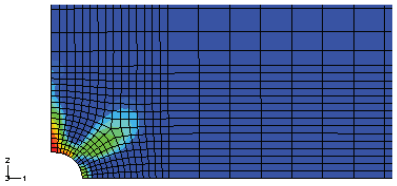


Figure 2: Matrix damage pattern in the 0° fiber-reinforced epoxy layer for the 0° loading direction (using the Hashin damage model, DAMAGEMT contour plot).

For more information, see *Failure of blunt notched fiber metal laminates*.

Static Stress/Displacement Analyses

In this section:

- *Static and quasi-static stress analyses*
- *Buckling and collapse analyses*
- *Forming analyses*
- *Fracture and damage*
- *Import analyses*

Static and quasi-static stress analyses

In this section:

- *Axisymmetric analysis of bolted pipe flange connections*
- *Elastic-plastic collapse of a thin-walled elbow under in-plane bending and internal pressure*
- *Parametric study of a linear elastic pipeline under in-plane bending*
- *Indentation of an elastomeric foam specimen with a hemispherical punch*
- *Collapse of a concrete slab*
- *Jointed rock slope stability*
- *Notched beam under cyclic loading*
- *Uniaxial ratcheting under tension and compression*
- *Hydrostatic fluid elements: modeling an airspring*
- *Shell-to-solid submodeling and shell-to-solid coupling of a pipe joint*
- *Stress-free element reactivation*
- *Transient loading of a viscoelastic bushing*
- *Indentation of a thick plate*
- *Damage and failure of a laminated composite plate*
- *Analysis of an automotive boot seal*
- *Pressure penetration analysis of an air duct kiss seal*
- *Self-contact in rubber/foam components: jounce bumper*
- *Self-contact in rubber/foam components: rubber gasket*
- *Submodeling of a stacked sheet metal assembly*
- *Axisymmetric analysis of a threaded connection*
- *Direct cyclic analysis of a cylinder head under cyclic thermomechanical loadings*
- *Erosion of material (sand production) in an oil wellbore*
- *Submodel stress analysis of pressure vessel closure hardware*
- *Using a composite layup to model a yacht hull*
- *Energy computations in a contact analysis*
- *Bending of a beam in a sleeve*
- *Fluid pressure penetration analysis of an O-ring seal of a pipe connection*

Axisymmetric analysis of bolted pipe flange connections

Products: Abaqus/Standard

Overview

This example shows how to perform a design analysis of a bolted pipe flange connection using an economical axisymmetric model and how to assess the accuracy of the asixymmetric model.

In addition, several three-dimensional models that use multiple levels of substructures are analyzed to demonstrate the use of substructures with a large number of retained degrees of freedom. Finally, a three-dimensional model containing stiffness matrices is analyzed to demonstrate the use of the matrix input functionality.

A bolted pipe flange connection is a common and important part of many piping systems. Such connections are typically composed of hubs of pipes, pipe flanges with bolt holes, sets of bolts and nuts, and a gasket. These components interact with each other in the tightening process and when operation loads such as internal pressure and temperature are applied. Experimental and numerical studies on different types of interaction among these components are frequently reported. The studies include analysis of the bolt-up procedure that yields uniform bolt stress (*Bibel and Ezell, 1992*), contact analysis of screw threads (*Fukuoka, 1992*; *Chaaban and Muzzo, 1991*), and full stress analysis of the entire pipe joint assembly (*Sawa et al., 1991*). To establish an optimal design, a full stress analysis determines factors such as the contact stresses that govern the sealing performance, the relationship between bolt force and internal pressure, the effective gasket seating width, and the bending moment produced in the bolts.

Geometry and model

The bolted joint assembly being analyzed is depicted in *Figure 1*. The geometry and dimensions of the various parts are taken from Sawa et al. (1991), modified slightly to simplify the modeling. The inner wall radius of both the hub and the gasket is 25 mm. The outer wall radii of the pipe flange and the gasket are 82.5 mm and 52.5 mm, respectively. The thickness of the gasket is 2.5 mm. The pipe flange has eight bolt holes that are equally spaced in the pitch circle of radius 65 mm. The radius of the bolt hole is modified in this analysis to be the same as that of the bolt: 8 mm. The bolt head (bearing surface) is assumed to be circular, and its radius is 12 mm.

The Young's modulus is 206 GPa and the Poisson's ratio is 0.3 for both the bolt and the pipe hub/flange. The gasket is modeled with either solid continuum or gasket elements. When continuum elements are used, the gasket's Young's modulus, E, equals 68.7 GPa and its Poisson's ratio, ν, equals 0.3.

When gasket elements are used, a linear gasket pressure/closure relationship is used with the effective “normal stiffness,” S_n, equal to the material Young's modulus divided by the thickness so that $S_n =$ 27.48 GPa/mm. Similarly a linear shear stress/shear motion relationship is used with an effective shear stiffness, S_t, equal to the material shear modulus divided by the thickness so that $S_t =$ 10.57 GPa/mm. The membrane behavior is specified with a Young's modulus of 68.7 GPa and a Poisson's ratio of 0.3. Sticking contact conditions are assumed in all contact areas: between the bearing surface and the flange and between the gasket and the hub. Contact between the bolt shank and the bolt hole is ignored.

The finite element idealizations of the symmetric half of the pipe joint are shown in *Figure 2* and *Figure 3*, corresponding to the axisymmetric and three-dimensional analyses, respectively. The mesh used for the axisymmetric analysis consists of a mesh for the pipe hub/flange and gasket and a separate mesh for the bolts. In *Figure 2* the top figure shows the mesh of the pipe hub and flange, with the bolt hole area shown in a lighter shade; and the bottom figure shows the overall mesh with the gasket and the bolt in place.

For the axisymmetric model second-order elements with reduced integration, CAX8R, are used throughout the mesh of the pipe hub/flange. The gasket is modeled with either CAX8R solid continuum elements or GKAX6

gasket elements. Contact between the gasket and the pipe hub/flange is modeled with contact pairs between surfaces defined on the faces of elements in the contact region or between such element-based surfaces and node-based surfaces. In an axisymmetric analysis the bolts and the perforated flange must be modeled properly. The bolts are modeled as plane stress elements since they do not carry hoop stress. Second-order plane stress elements with reduced integration, CPS8R, are employed for this purpose. The contact surface definitions, which are associated with the faces of the elements, account for the plane stress condition automatically. To account for all eight bolts used in the joint, the combined cross-sectional areas of the shank and the head of the bolts must be calculated and redistributed to the bolt mesh appropriately using the area attributes for the solid elements. The contact area is adjusted automatically.

Figure 4 illustrates the cross-sectional views of the bolt head and the shank. Each plane stress element represents a volume that extends out of the *x*–*y* plane. For example, element *A* represents a volume calculated as $(H_A) \times (Area_A)$. Likewise, element *B* represents a volume calculated as $(H_B) \times (Area_B)$. The sectional area in the *x*–*z* plane pertaining to a given element can be calculated as

$$Area = 2\int_{X_1}^{X_2}\left[\left(R^2 - x^2\right)^{\frac{1}{2}}\right]dx = \left[x\left(R^2 - x^2\right)^{\frac{1}{2}} + R^2 \arcsin\left(\frac{x}{|R|}\right)\right]\Bigg|_{X_1}^{X_2},$$

where *R* is the bolt head radius, R_{bolthead}, or the shank radius, R_{shank} (depending on the element location), and X_1 and X_2 are *x*-coordinates of the left and right side of the given element, respectively.

If the sectional areas are divided by the respective element widths, W_A and W_B, we obtain representative element thicknesses. Multiplying each element thickness by eight (the number of bolts in the model) produces the thickness values that are found in the solid section definition.

Sectional areas that are associated with bolt head elements located on the model's contact surfaces are used to calculate the surface areas of the nodes used in defining the node-based surfaces of the model. Referring again to *Figure 4*, nodal contact areas for a single bolt are calculated as follows:

$$A_1 = \frac{A_C}{4}, \quad A_9 = \frac{A_F}{4},$$

$$A_2 = \frac{A_C}{2}, \quad A_4 = \frac{A_D}{2}, \quad A_6 = \frac{A_E}{2}, \quad A_8 = \frac{A_F}{2},$$

$$A_3 = (A_C + A_D)/4, \quad A_5 = (A_D + A_E)/4, \quad A_7 = (A_E + A_F)/4,$$

where A_1 through A_9 are contact areas that are associated with contact nodes 1–9 and A_c through A_F are sectional areas that are associated with bolt head elements *C*–*F*. Multiplying the above areas by eight (the number of bolts in the model) provides the nodal contact areas found in the contact property definitions.

A common way of handling the presence of the bolt holes in the pipe flange in axisymmetric analyses is to smear the material properties used in the bolt hole area of the mesh and to use inhomogeneous material properties that correspond to a weaker material in this region. General guidelines for determining the effective material properties for perforated flat plates are found in ASME Section VIII Div 2 Article 4–9. For the type of structure under study, which is not a flat plate, a common approach to determining the effective material properties is to calculate the elasticity moduli reduction factor, which is the ratio of the ligament area in the pitch circle to the annular area of the pitch circle. In this model the annular area of the pitch circle is given by $AA =$ 6534.51 mm^2, and the total area of the bolt holes is given by $AH = 8\pi 8^2 =$ 1608.5 mm^2. Hence, the reduction factor is simply $1 - AH/AA =$ 0.754. The effective in-plane moduli of elasticity, $E1'$ and $E2'$, are obtained by multiplying the respective moduli, $E1$ and $E2$, by this factor. We assume material isotropy in the *r*–*z* plane; thus, $E1' = E2' = E'$. The modulus in the hoop direction, $E3'$, should be very small and is chosen such that $E'/E3' =$ 10^6. The in-plane shear modulus is then calculated based on the effective elasticity modulus: $G'_{12} = E'/2(1+\nu)$. The shear moduli in the hoop direction are also calculated similarly but with ν set to zero (they are not used in

an axisymmetric model). Hence, we have $E1' = E2' =$ 155292 MPa, $E3' =$ 0.155292 MPa, $G'_{12} =$ 59728 MPa, and $G'_{13} = G'_{23} =$ 0.07765 MPa. These orthotropic elasticity moduli are specified using engineering constants for the bolt hole part of the mesh.

The mesh for the three-dimensional analysis without substructures, shown in *Figure 3*, represents a 22.5° segment of the pipe joint and employs second-order brick elements with reduced integration, C3D20R, for the pipe hub/flange and bolts. The gasket is modeled with C3D20R elements or GK3D18 elements. The top figure shows the mesh of the pipe hub and flange, and the bottom figure shows both the gasket and bolt (in the lighter color). Contact is modeled by the interaction of contact surfaces defined by grouping specific faces of the elements in the contacting regions. For three-dimensional contact where both the main and secondary surfaces are deformable, the small-sliding contact pair formulation must be used to indicate that small relative sliding occurs between contacting surfaces. No special adjustments need be made for the material properties used in the three-dimensional model because all parts are modeled appropriately.

Four different meshes that use substructures to model the flange are tested. A first-level substructure is created for the entire 22.5° segment of the flange shown in *Figure 3*, while the gasket and the bolt are meshed as before. The nodes on the flange in contact with the bolt cap form a node-based surface, while the nodes on the flange in contact with the gasket form another node-based surface. These node-based surfaces will form contact pairs with the main surfaces on the bolt cap and on the gasket, which are defined using the surface definition options. The retained degrees of freedom on the substructure include all three degrees of freedom for the nodes in these node-based surfaces as well as for the nodes on the 0° and 22.5° faces of the flange. Appropriate boundary conditions are specified at the substructure usage level.

A second-level substructure of 45° is created by reflecting the first-level substructure with respect to the 22.5° plane. The nodes on the 22.5° face belonging to the reflected substructure are constrained in all three degrees of freedom to the corresponding nodes on the 22.5° face belonging to the original first-level substructure. The half-bolt and the gasket sector corresponding to the reflected substructure are also constructed by reflection. The retained degrees of freedom include all three degrees of freedom of all contact node sets and of the nodes on the 0° and 45° faces of the flange. MPC-type CYCLSYM is used to impose cyclic symmetric boundary conditions on these two faces.

A third-level substructure of 90° is created by reflecting the original 45° second-level substructure with respect to the 45° plane and by connecting it to the original 45° substructure. The remaining part of the gasket and the bolts corresponding to the 45°–90° sector of the model is created by reflection and appropriate constraints. In this case it is not necessary to retain any degrees of freedom on the 0° and 90° faces of the flange because this 90° substructure will not be connected to other substructures and appropriate boundary conditions can be specified at the substructure creation level.

The final substructure model is set up by mirroring the 90° mesh with respect to the symmetry plane of the gasket perpendicular to the *y*-axis. Thus, an otherwise large analysis ($\approx$ 750,000 unknowns) when no substructures are used can be solved conveniently ($\approx$ 80,000 unknowns) by using the third-level substructure twice. The sparse solver is used because it significantly reduces the run time for this model.

Finally, a three-dimensional matrix-based model is created by replacing elements for the entire 22.5° segment of the flange shown in *Figure 3* with stiffness matrices, while the gasket and the bolt are meshed as before. Contact between the flange and gasket and the flange and bolt cap is modeled using node-based secondary surfaces just as for the substructure models. Appropriate boundary conditions are applied as in the three-dimensional model without substructures.

Loading and boundary conditions

The only boundary conditions are symmetry boundary conditions. In the axisymmetric model $u_z = 0$ is applied to the symmetry plane of the gasket and to the bottom of the bolts. In the three-dimensional model $u_y = 0$ is applied to the symmetry plane of the gasket as well as to the bottom of the bolt. The $\theta = 0°$ and $\theta = 22.5°$ planes are also symmetry planes. On the $\theta = 22.5°$ plane, symmetry boundary conditions are enforced by invoking

suitable nodal transformations and applying boundary conditions to local directions in this symmetry plane. These transformations are implemented using a local coordinate system definition. On both the symmetry planes, the symmetry boundary conditions $u_z = 0$ are imposed everywhere except for the dependent nodes associated with the C BIQUADMPC and nodes on one side of the contact surface. The second exception is made to avoid overconstraining problems, which arise if there is a boundary condition in the same direction as a Lagrange multiplier constraint associated with the rough friction specification.

In the models where substructures are used, the boundary conditions are specified depending on what substructure is used. For the first-level 22.5° substructure the boundary conditions and constraint equations are the same as for the three-dimensional model shown in *Figure 3*. For the 45° second-level substructure the symmetry boundary conditions are enforced on the $\theta = 45°$ plane with the constraint equation $u_z + u_x = 0$. A transform could have been used as well. For the 90° third-level substructure the face $\theta = 90°$ is constrained with the boundary condition $u_x = 0$.

For the three-dimensional model containing matrices, nodal transformations are applied for symmetric boundary conditions. Entries in the stiffness matrices for these nodes are also in local coordinates.

A clamping force of 15 kN is applied to each bolt by associating the pre-tension node with a pre-tension section. The pre-tension section is identified by means of a surface definition. The pre-tension is then prescribed by applying a concentrated load to the pre-tension node. In the axisymmetric analysis the actual load applied is 120 kN since there are eight bolts. In the three-dimensional model with no substructures the actual load applied is 7.5 kN since only half of a bolt is modeled. In the models using substructures all half-bolts are loaded with a 7.5 kN force. For all of the models the pre-tension section is specified about halfway down the bolt shank.

Sticking contact conditions are assumed in all surface interactions in all analyses and are simulated with rough friction and no-separation contact.

Results and discussion

All analyses are performed as small-displacement analyses.

Figure 5 shows a top view of the normal stress distributions in the gasket at the interface between the gasket and the pipe hub/flange predicted by the axisymmetric (bottom) and three-dimensional (top) analyses when solid continuum elements are used to model the gasket. The figure shows that the compressive normal stress is highest at the outer edge of the gasket, decreases radially inward, and changes from compression to tension at a radius of about 35 mm, which is consistent with findings reported by Sawa et al. (1991). The close agreement in the overall solution between axisymmetric and three-dimensional analyses is quite apparent, indicating that, for such problems, axisymmetric analysis offers a simple yet reasonably accurate alternative to three-dimensional analysis.

Figure 6 shows a top view of the normal stress distributions in the gasket at the interface between the gasket and the pipe hub/flange predicted by the axisymmetric (bottom) and three-dimensional (top) analyses when gasket elements are used to model the gasket. Close agreement in the overall solution between the axisymmetric and three-dimensional analyses is also seen in this case. The gasket starts carrying compressive load at a radius of about 40 mm, a difference of 5 mm with the previous result. This difference is the result of the gasket elements being unable to carry tensile loads in their thickness direction. This solution is physically more realistic since, in most cases, gaskets separate from their neighboring parts when subjected to tensile loading. Removing the no-separation contact from the gasket/flange contact surface definition in the input files that model the gasket with continuum elements yields good agreement with the results obtained in *Figure 6* (since, in that case, the solid continuum elements in the gasket cannot carry tensile loading in the gasket thickness direction).

The models in this example can be modified to study other factors, such as the effective seating width of the gasket or the sealing performance of the gasket under operating loads. The gasket elements offer the advantage of allowing very complex behavior to be defined in the gasket thickness direction. Gasket elements can also use any of the small-strain material models provided in Abaqus including user-defined material models. *Figure 7* shows a comparison of the normal stress distributions in the gasket at the interface between the gasket and the pipe hub/flange predicted by the axisymmetric (bottom) and three-dimensional (top) analyses when isotropic

material properties are prescribed for gasket elements. The results in *Figure 7* compare well with the results in *Figure 5* from analyses in which solid and axisymmetric elements are used to simulate the gasket.

Figure 8 shows the distribution of the normal stresses in the gasket at the interface in the plane $z = 0$. The results are plotted for the three-dimensional model containing only solid continuum elements and no substructures, for the three-dimensional model with matrices, and for the four models containing the substructures described above.

An execution procedure is available to combine model and results data from two substructure output databases into a single output database. For more information, see *Combining Output from Substructures*.

This example can also be used to demonstrate the effectiveness of the quasi-Newton nonlinear solver. This solver utilizes an inexpensive, approximate stiffness matrix update for several consecutive equilibrium iterations, rather than a complete stiffness matrix factorization each iteration as used in the default full Newton method. The quasi-Newton method results in an increased number of less expensive iterations, and a net savings in computing cost.

Input files

boltpipeflange_axi_solidgask.inp	Axisymmetric analysis containing a gasket modeled with solid continuum elements.
boltpipeflange_axi_node.inp	Node definitions for boltpipeflange_axi_solidgask.inp and boltpipeflange_axi_gkax6.inp.
boltpipeflange_axi_element.inp	Element definitions for boltpipeflange_axi_solidgask.inp.
boltpipeflange_3d_solidgask.inp	Three-dimensional analysis containing a gasket modeled with solid continuum elements.
boltpipeflange_axi_gkax6.inp	Axisymmetric analysis containing a gasket modeled with gasket elements.
boltpipeflange_3d_gk3d18.inp	Three-dimensional analysis containing a gasket modeled with gasket elements.
boltpipeflange_3d_substr1.inp	Three-dimensional analysis using the first-level substructure (22.5° model).
boltpipeflange_3d_substr2.inp	Three-dimensional analysis using the second-level substructure (45° model).
boltpipeflange_3d_substr3_1.inp	Three-dimensional analysis using the third-level substructure once (90° model).
boltpipeflange_3d_substr3_2.inp	Three-dimensional analysis using the third-level substructure twice (90° mirrored model).
boltpipeflange_3d_gen1.inp	First-level substructure generation data referenced by boltpipeflange_3d_substr1.inp and boltpipeflange_3d_gen2.inp.
boltpipeflange_3d_gen2.inp	Second-level substructure generation data referenced by boltpipeflange_3d_substr2.inp and boltpipeflange_3d_gen3.inp.
boltpipeflange_3d_gen3.inp	Third-level substructure generation data referenced by boltpipeflange_3d_substr3_1.inp and boltpipeflange_3d_substr3_2.inp.
boltpipeflange_3d_node.inp	Nodal coordinates used in boltpipeflange_3d_substr1.inp, boltpipeflange_3d_substr2.inp, boltpipeflange_3d_substr3_1.inp, boltpipeflange_3d_substr3_2.inp, boltpipeflange_3d_cyclsym.inp, boltpipeflange_3d_gen1.inp, boltpipeflange_3d_gen2.inp, and boltpipeflange_3d_gen3.inp.
boltpipeflange_3d_cyclsym.inp	Same as file boltpipeflange_3d_substr2.inp except that CYCLSYM type MPCs are used.

boltpipeflange_3d_missnode.inp	Same as file boltpipeflange_3d_gk3d18.inp except that the option to generate missing nodes is used for gasket elements.
boltpipeflange_3d_isomat.inp	Same as file boltpipeflange_3d_gk3d18.inp except that gasket elements are modeled as isotropic using the **MATERIAL* option.
boltpipeflange_3d_ortho.inp	Same as file boltpipeflange_3d_gk3d18.inp except that gasket elements are modeled as orthotropic and the **ORIENTATION* option is used.
boltpipeflange_axi_isomat.inp	Same as file boltpipeflange_axi_gkax6.inp except that gasket elements are modeled as isotropic using the **MATERIAL* option.
boltpipeflange_3d_usr_umat.inp	Same as file boltpipeflange_3d_gk3d18.inp except that gasket elements are modeled as isotropic with user subroutine *UMAT*.
boltpipeflange_3d_usr_umat.f	User subroutine *UMAT* used in boltpipeflange_3d_usr_umat.inp.
boltpipeflange_3d_solidnum.inp	Same as file boltpipeflange_3d_gk3d18.inp except that solid element numbering is used for gasket elements.
boltpipeflange_3d_matrix.inp	Three-dimensional analysis containing matrices and a gasket modeled with solid continuum elements.
boltpipeflange_3d_stiffPID4.inp	Matrix representing stiffness of a part of the flange segment for three-dimensional analysis containing matrices.
boltpipeflange_3d_stiffPID5.inp	Matrix representing stiffness of the remaining part of the flange segment for three-dimensional analysis containing matrices.
boltpipeflange_3d_qn.inp	Same as file boltpipeflange_3d_gk3d18.inp except that the quasi-Newton nonlinear solver is used.

References

- Bibel, G. D., and R. M. Ezell, "An Improved Flange Bolt-Up Procedure Using Experimentally Determined Elastic Interaction Coefficients," *Journal of Pressure Vessel Technology*, vol. 114, pp. 439–443, 1992.
- Chaaban, A., and U. Muzzo, "Finite Element Analysis of Residual Stresses in Threaded End Closures," *Transactions of ASME*, vol. 113, pp. 398–401, 1991.
- Fukuoka, T., "Finite Element Simulation of Tightening Process of Bolted Joint with a Tensioner," *Journal of Pressure Vessel Technology*, vol. 114, pp. 433–438, 1992.
- Sawa, T., N. Higurashi, and H. Akagawa, "A Stress Analysis of Pipe Flange Connections," *Journal of Pressure Vessel Technology*, vol. 113, pp. 497–503, 1991.

Figures

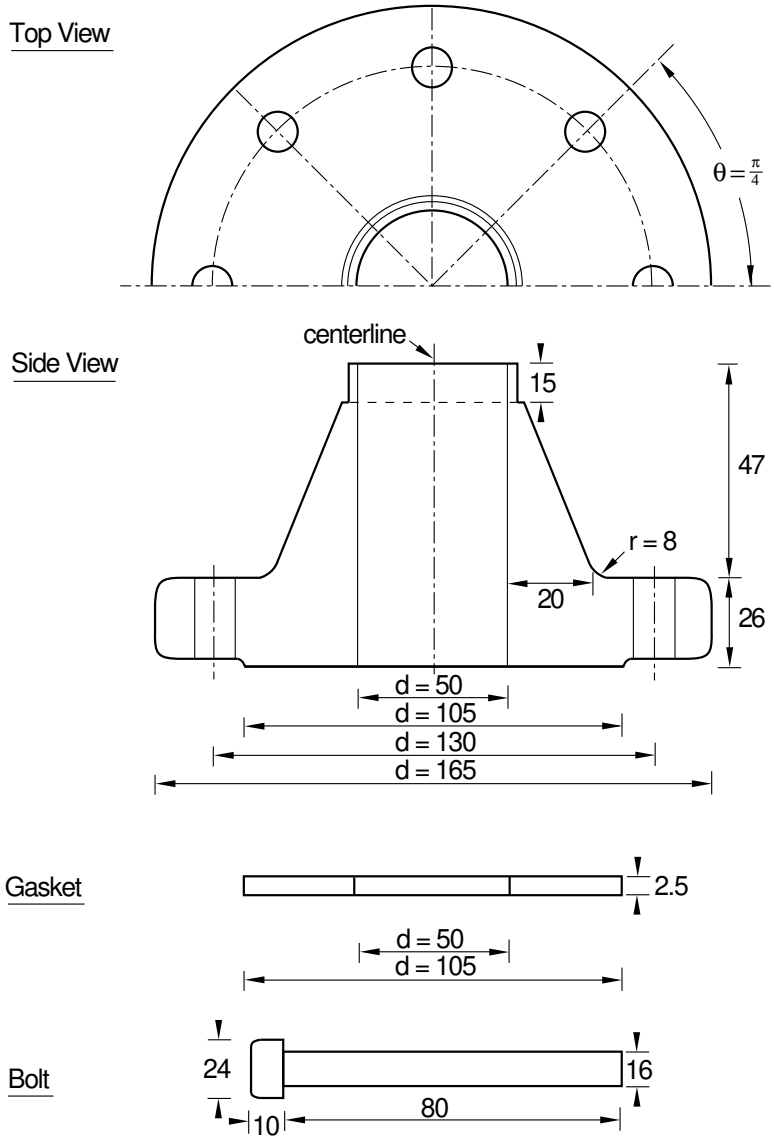


Figure 1: Schematic of the bolted joint. All dimensions in mm.

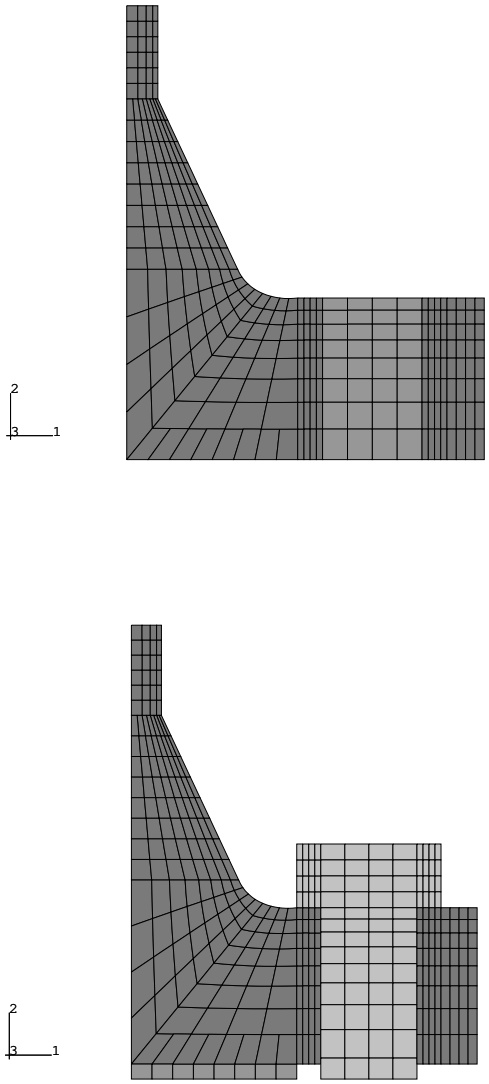


Figure 2: Axisymmetric model of the bolted joint.

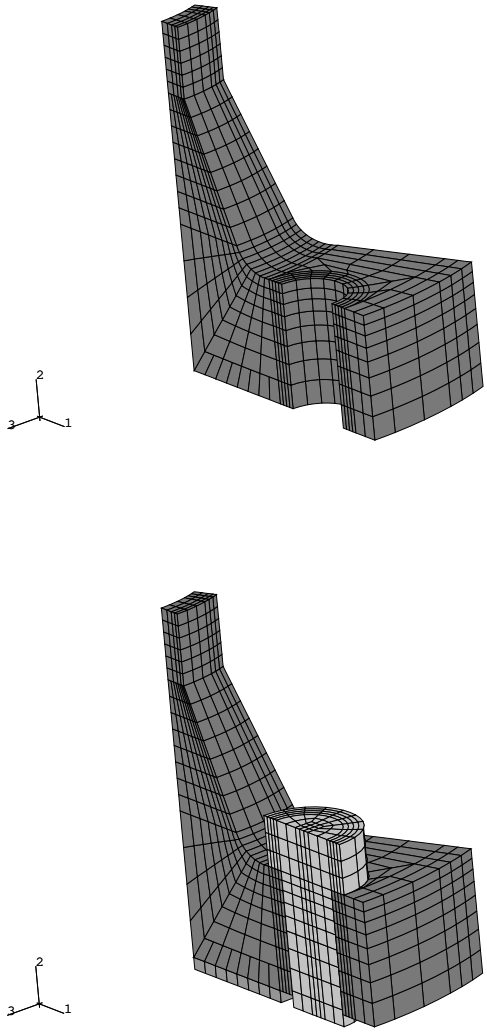


Figure 3: 22.5° segment three-dimensional model of the bolted joint.

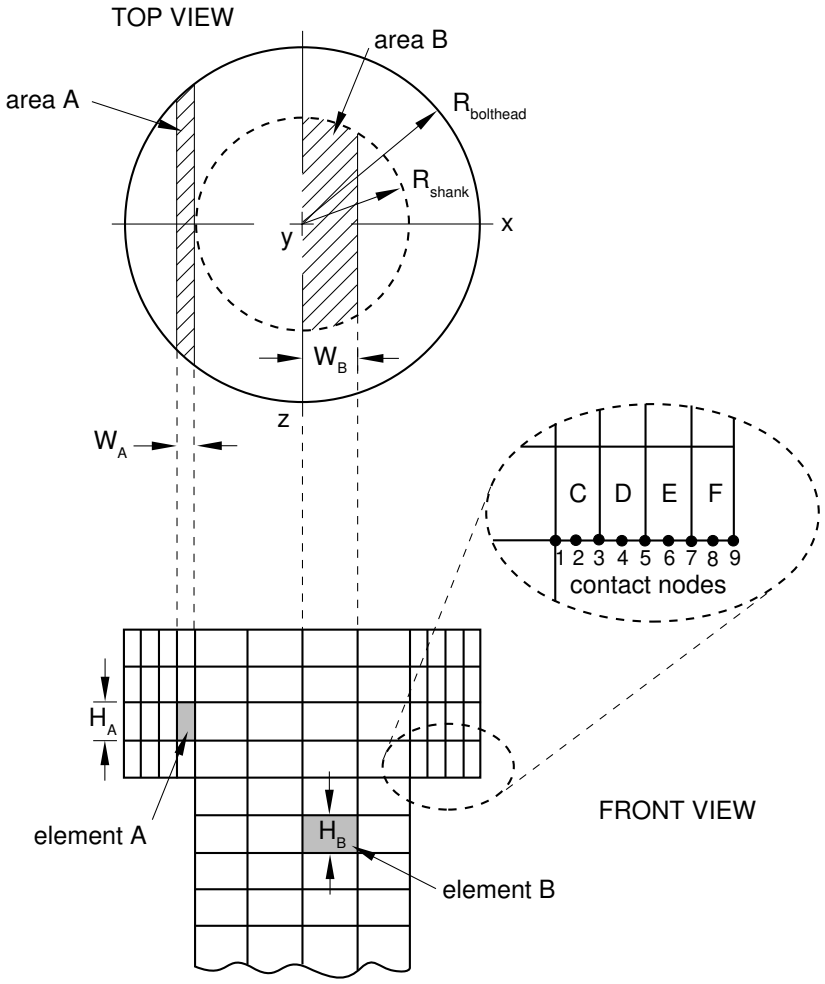
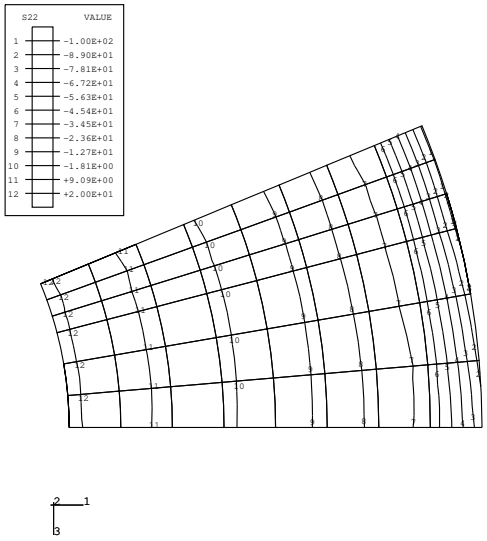


Figure 4: Cross-sectional views of the bolt head and the shank.



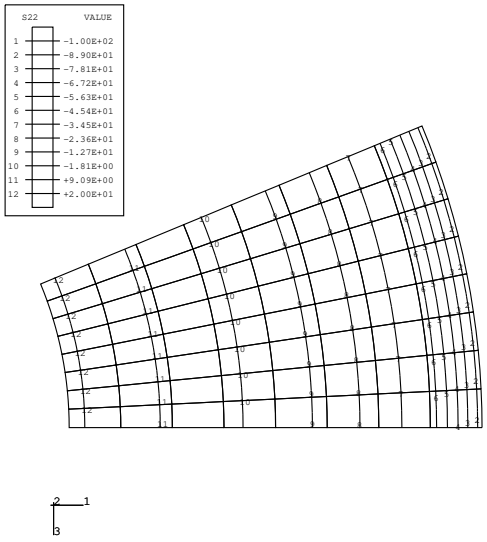


Figure 5: Normal stress distribution in the gasket contact surface when solid elements are used to model the gasket: three-dimensional versus axisymmetric results.

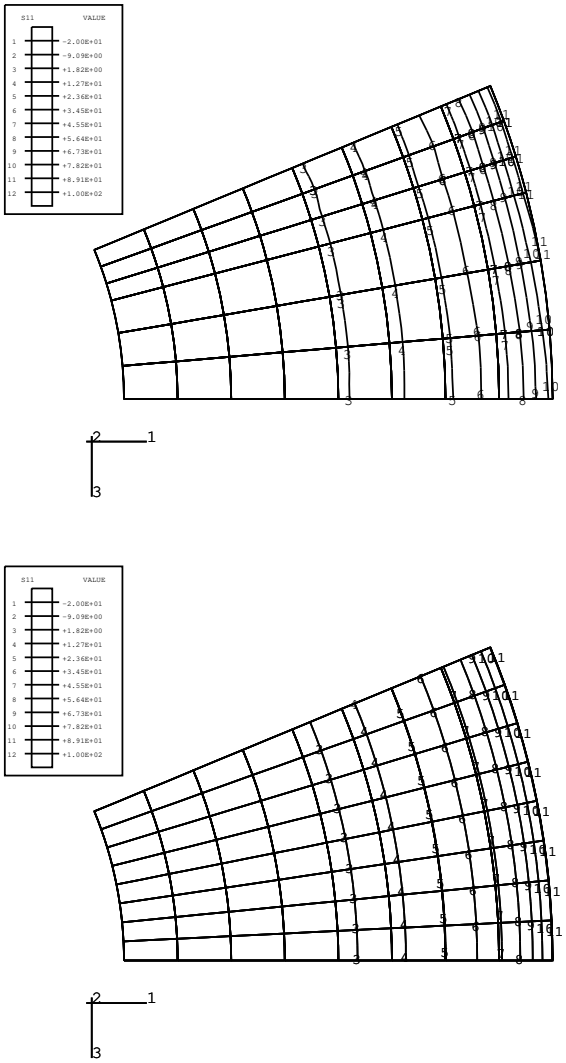


Figure 6: Normal stress distribution in the gasket contact surface when gasket elements are used with direct specification of the gasket behavior: three-dimensional versus axisymmetric results.

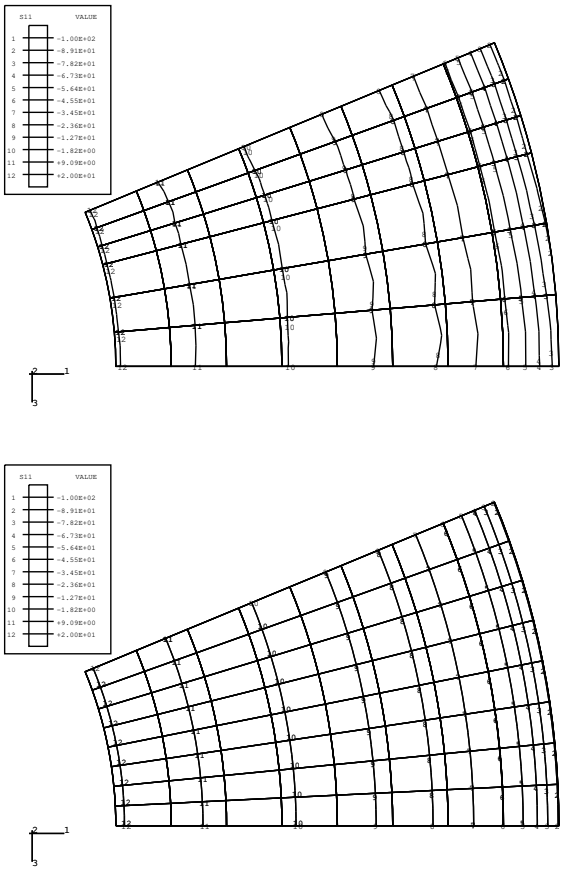


Figure 7: Normal stress distribution in the gasket contact surface when gasket elements are used with isotropic material properties: three-dimensional versus axisymmetric results.

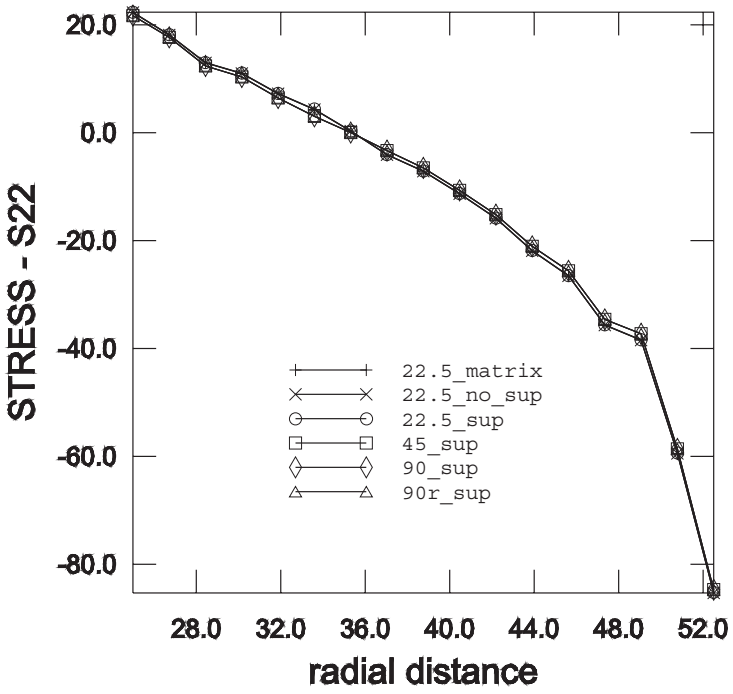


Figure 8: Normal stress distribution in the gasket contact surface along the line $z = 0$ for the models with and without substructures.

Elastic-plastic collapse of a thin-walled elbow under in-plane bending and internal pressure

Products: Abaqus/Standard

Overview

This example verifies the use of shell and elbow element modeling of ovalization effects.

Elbows are used in piping systems because they ovalize more readily than straight pipes and, thus, provide flexibility in response to thermal expansion and other loadings that impose significant displacements on the system. Ovalization is the bending of the pipe wall into an oval—i.e., noncircular—configuration. The elbow is, thus, behaving as a shell rather than as a beam. Straight pipe runs do not ovalize easily, so they behave essentially as beams. Thus, even under pure bending, complex interaction occurs between an elbow and the adjacent straight pipe segments; the elbow causes some ovalization in the straight pipe runs, which in turn tend to stiffen the elbow. This interaction can create significant axial gradients of bending strain in the elbow, especially in cases where the elbow is very flexible. This example provides verification of shell and elbow element modeling of such effects, through an analysis of a test elbow for which experimental results have been reported by Sobel and Newman (1979). An analysis is also included with elements of type ELBOW31B (which includes ovalization but neglects axial gradients of strain) for the elbow itself and beam elements for the straight pipe segments. This provides a comparative solution in which the interaction between the elbow and the adjacent straight pipes is neglected. The analyses predict the response up to quite large rotations across the elbow, so as to investigate possible collapse of the pipe and, particularly, the effect of internal pressure on that collapse.

Geometry and model

The elbow configuration used in the study is shown in *Figure 1*. It is a thin-walled elbow with elbow factor

$$\lambda = \frac{Rt}{r^2\sqrt{1-\nu^2}} = 0.167$$

and radius ratio $R/r =$ 3.07, so the flexibility factor from Dodge and Moore (1972) is 10.3. (The flexibility factor for an elbow is the ratio of the bending flexibility of an elbow segment to that of a straight pipe of the same dimensions, for small displacements and elastic response.) This is an extremely flexible case because the pipe wall is so thin.

To demonstrate convergence of the overall moment-rotation behavior with respect to meshing, the two shell element meshes shown in *Figure 2* are analyzed. Since the loading concerns in-plane bending only, it is assumed that the response is symmetric about the midplane of the system so that in the shell element model only one-half of the system need be modeled. Element type S8R5 is used, since tests have shown this to be the most cost-effective shell element in Abaqus (input files using element types S9R5, STRI65, and S8R for this example are included with the Abaqus release). The elbow element meshes replace each axial division in the coarser shell element model with one ELBOW32 or two ELBOW31 elements and use 4 or 6 Fourier modes to model the deformation around the pipe. Seven integration points are used through the pipe wall in all the analyses. This is usually adequate to provide accurate modeling of the progress of yielding through the section in such cases as these, where essentially monotonic straining is expected.

The ends of the system are rigidly attached to stiff plates in the experiments. These boundary conditions are easily modeled for the ELBOW elements and for the fixed end in the shell element model. For the rotating end of the shell element model the shell nodes must be constrained to a beam node that represents the motion of the end plate using a kinematic coupling constraint as described below.

The material is assumed to be isotropic and elastic-plastic, following the measured response of type 304 stainless steel at room temperature, as reported by Sobel and Newman (1979). Since all the analyses give results that are stiffer than the experimentally measured response, and the mesh convergence tests (results are discussed below) demonstrate that the meshes are convergent with respect to the overall response of the system, it seems that this stress-strain model may overestimate the material's actual strength.

Loading

The load on the pipe has two components: a “dead” load, consisting of internal pressure (with a closed end condition), and a “live” in-plane bending moment applied to the end of the system. The pressure is applied to the model in an initial step and then held constant in the second analysis step while the bending moment is increased. The pressure values range from 0.0 to 3.45 MPa (500 lb/in^2), which is the range of interest for design purposes. The equivalent end force associated with the closed-end condition is applied as a follower force because it rotates with the motion of the end plane.

Kinematic boundary conditions

The fixed end of the system is assumed to be fully built-in. The loaded end is fixed into a very stiff plate. For the ELBOW element models this condition is represented by the `NODEFORM` boundary condition applied at this node. In the shell element model this rigid plate is represented by a single node, and the shell nodes at the end of the pipe are attached to it by using a kinematic coupling constraint and specifying that all degrees of freedom at the shell nodes are constrained to the motion of the single node.

Results and discussion

The moment-rotation responses predicted by the various analysis models and measured in the experiment, all taken at zero internal pressure, are compared in *Figure 3*. The figure shows that the two shell models give very similar results, overestimating the experimentally measured collapse moment by about 15%. The 6-mode ELBOW element models are somewhat stiffer than the shell models, and those with 4 Fourier modes are much too stiff. This clearly shows that, for this very flexible system, the ovalization of the elbow is too localized for even the 6-mode ELBOW representation to provide accurate results.

Since we know that the shell models are convergent with respect to discretization, the most likely explanation for the excessive stiffness in comparison to the experimentally measured response is that the material model used in the analyses is too strong. Sobel and Newman (1979) point out that the stress-strain curve measured and used in this analysis, shown in *Figure 1*, has a 0.2% offset yield that is 20% higher than the Nuclear Systems Materials Handbook value for type 304 stainless steel at room temperature, which suggests the possibility that the billets used for the stress-strain curve measurement may have been taken from stronger parts of the fabrication. If this is the case, it points out the likelihood that the elbow tested is rather nonuniform in strength properties despite the care taken in its manufacture. We are left with the conclusion that discrepancies of this magnitude cannot be eliminated in practical cases, and the design use of such analysis results must allow for them.

Figure 4 compares the moment-rotation response for opening and closing moments under 0 and 3.45 MPa (500 lb/in^2) internal pressure and shows the strong influence of large-displacement effects. If large-displacement effects were not important, the opening and closing moments would produce the same response. However, even with a 1° relative rotation across the elbow assembly, the opening and closing moments differ by about 12%; with a 2° relative rotation, the difference is about 17%. Such magnitudes of relative rotation would not normally be considered large; in this case it is the coupling into ovalization that makes geometric nonlinearity significant. As the rotation increases, the cases with closing moment loading show collapse, while the opening moment curves do not. In both cases internal pressure shows a strong effect on the results, which is to be expected in such a thin-walled pipeline. The level of interaction between the straight pipe and the elbows is well illustrated by the strain distribution on the outside wall, shown in *Figure 5*. The strain contours are slightly discontinuous at the ends of the curved elbow section because the shell thickness changes at those sections.

Figure 6 shows a summary of the results from this example and *Uniform collapse of straight and curved pipe segments*. The plot shows the collapse value of the closing moment under in-plane bending as a function of internal pressure. The strong influence of pressure on collapse is apparent. In addition, the effect of analyzing the elbow by neglecting interaction between the straight and curved segments is shown: the "uniform bending" results are obtained by using elements of type ELBOW31B in the bend and beams (element type B31) for the straight segments. The importance of the straight/elbow interaction is apparent. In this case the simpler analysis neglecting the interaction is conservative (in that it gives consistently lower values for the collapse moment), but this conservatism cannot be taken for granted. The analysis of Sobel and Newman (1979) also neglects interaction and agrees quite well with the results obtained here.

For comparison the small-displacement limit analysis results of Goodall (1978), as well as his large-displacement, elastic-plastic lower bound (Goodall, 1978a), are also shown in this figure. Again, the importance of large-displacement effects is apparent from that comparison.

Detailed results obtained with the model that uses ELBOW31 elements are shown in *Figure 7* through *Figure 9*. *Figure 7* shows the variation of the von Mises stress along the length of the piping system. The length is measured along the centerline of the pipe starting at the loaded end. The figure compares the stress distribution at the intrados (integration point 1) on the inner and outer surfaces of the elements (section points 1 and 7, respectively). *Figure 8* shows the variation of the von Mises stress around the circumference of two elements (451 and 751) that are located in the bend section of the model; the results are for the inner surface of the elements (section point 1). *Figure 9* shows the ovalization of elements 451 and 751. A nonovalized, circular cross-section is included in the figure for comparison. From the figure it is seen that element 751, located at the center of the bend section, experiences the most severe ovalization. These three figures were produced with the aid of the elbow element postprocessing program `felbow.f` (*Creation of a data file to facilitate the postprocessing of elbow element results: FELBOW*), written in Fortran. The postprocessing programs `felbow.C` (*A C++ version of FELBOW*) and `felbow.py` (*An Abaqus Scripting Interface version of FELBOW*), written in C++ and Python, respectively, are also available for generating the data for figures such as *Figure 8* and *Figure 9*. The user must ensure that the output variables are written to the output database to use these two programs.

Shell-to-solid submodeling

One particular case is analyzed using the shell-to-solid submodeling technique. This problem verifies the interpolation scheme in the case of double curved surfaces. A solid submodel using C3D27R elements is created around the elbow part of the pipe, spanning an angle of 40°. The finer submodel mesh has three elements through the thickness, 10 elements around half of the circumference of the cylinder, and 10 elements along the length of the elbow. Both ends are driven from the global shell model made of S8R elements. The time scale of the static submodel analysis corresponds to the arc length in the global Riks analysis. The submodel results agree closely with the shell model. The total force and the total moment in a cross-section through the submodel are written to the results (`.fil`) file.

Shell-to-solid coupling

A model using the shell-to-solid coupling capability in Abaqus is included. Such a model can be used for a careful study of the stress and strain fields in the elbow. The entire elbow is meshed with C3D20R elements, and the straight pipe sections are meshed with S8R elements (see *Figure 10*). At each shell-to-solid interface illustrated in *Figure 10*, an element-based surface is defined on the edge of the solid mesh and an edge-based surface is defined on the edge of the shell mesh. A shell-to-solid coupling constraint is used in conjunction with these surfaces to couple the shell and solid meshes.

Edge-based surfaces are defined at the end of each pipe segment. These surfaces are coupled to reference nodes that are defined at the center of the pipes using a distributing coupling constraint. The loading and fixed boundary conditions are applied to the reference points. The advantage of using this method is that the pipe cross-sectional areas are free to deform; thus, ovalization at the ends is not constrained. The moment-rotation response of the shell-to-solid coupling model agrees very well with the results shown in *Figure 4*.

Input files

In all the following input files (with the exception of *elbowcollapse_elbow31b_b31.inp*, *elbowcollapse_s8r5_fine.inp*, and *elbowcolpse_shl2sld_s8r_c3d20r.inp*) the step concerning the application of the pressure load is commented out. To include the effects of the internal pressure in any given analysis, uncomment the step definition in the appropriate input file.

elbowcollapse_elbow31b_b31.inp	ELBOW31B and B31 element model.
elbowcollapse_elbow31_6four.inp	ELBOW31 model with 6 Fourier modes.
elbowcollapse_elbow32_6four.inp	ELBOW32 model with 6 Fourier modes.
elbowcollapse_s8r.inp	S8R element model.
elbowcollapse_s8r5.inp	S8R5 element model.
elbowcollapse_s8r5_fine.inp	Finer S8R5 element model.
elbowcollapse_s9r5.inp	S9R5 element model.
elbowcollapse_stri65.inp	STRI65 element model.
elbowcollapse_submod.inp	Submodel using C3D27R elements.
elbowcolpse_shl2sld_s8r_c3d20r.inp	Shell-to-solid coupling model using S8R and C3D20R elements.

References

- Dodge, W. G., and S. E. Moore, "Stress Indices and Flexibility Factors for Moment Loadings on Elbows and Curved Pipes," *Welding Research Council Bulletin*, no. 179, 1972.
- Goodall, I. W., "Lower Bound Limit Analysis of Curved Tubes Loaded by Combined Internal Pressure and In-Plane Bending Moment," *Research Division Report RD/B/N4360, Central Electricity Generating Board, England*, 1978.
- Goodall, I. W., "Large Deformations in Plastically Deforming Curved Tubes Subjected to In-Plane Bending," *Research Division Report RD/B/N4312, Central Electricity Generating Board, England*, 1978a.
- Sobel, L. H., and S. Z. Newman, "Elastic-Plastic In-Plane Bending and Buckling of an Elbow: Comparison of Experimental and Simplified Analysis Results," *Westinghouse Advanced Reactors Division, Report WARD–HT–94000–2*, 1979.

Figures

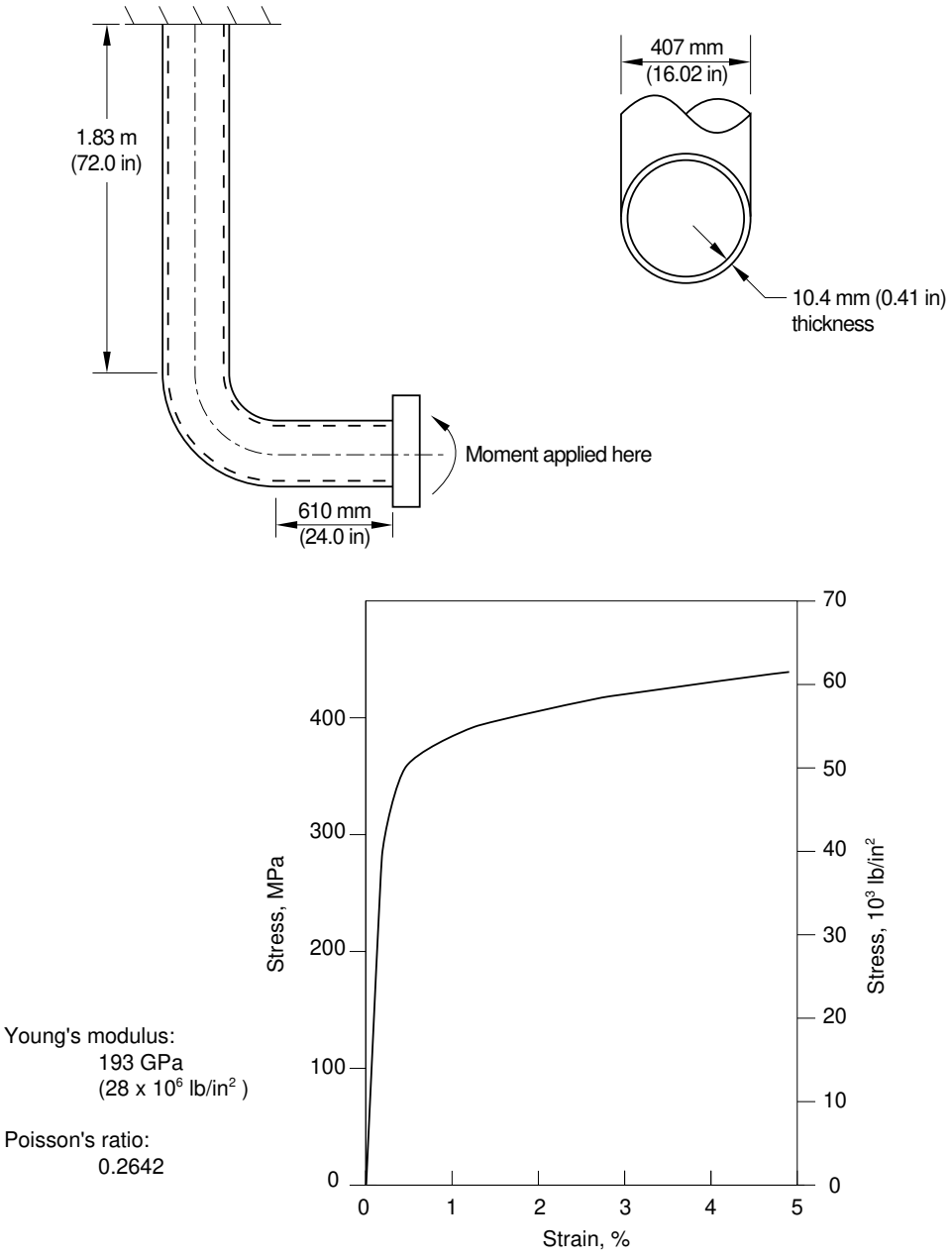


Figure 1: MLTF elbow: geometry and measured material response.

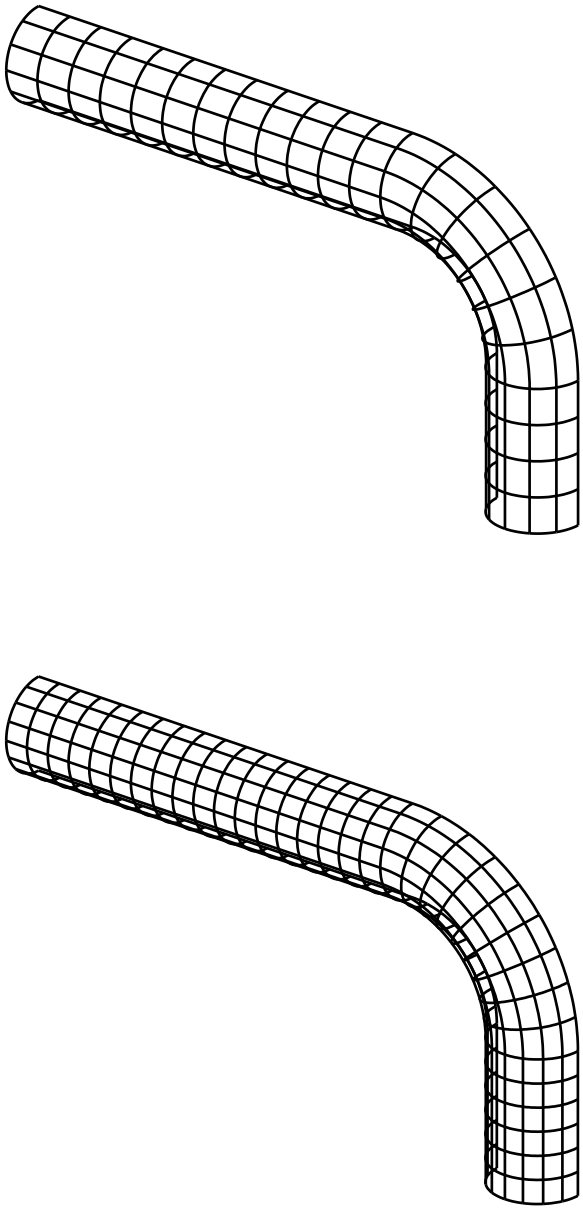


Figure 2: Models for elbow/pipe interaction study.

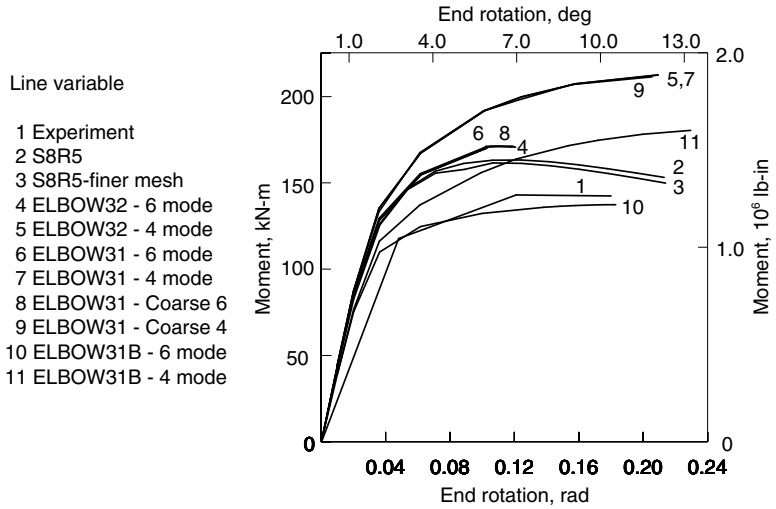


Figure 3: Moment-rotation response: mesh convergence studies.

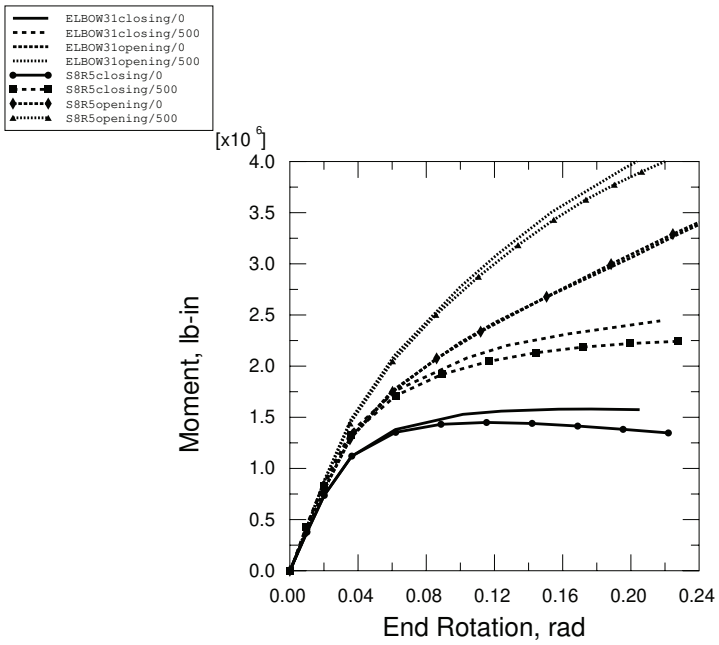
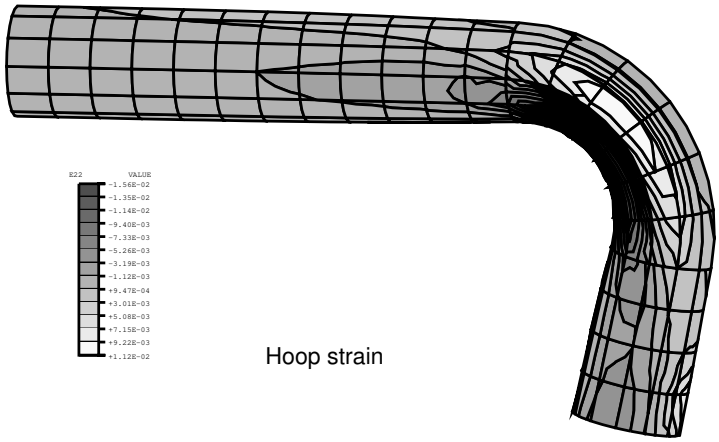
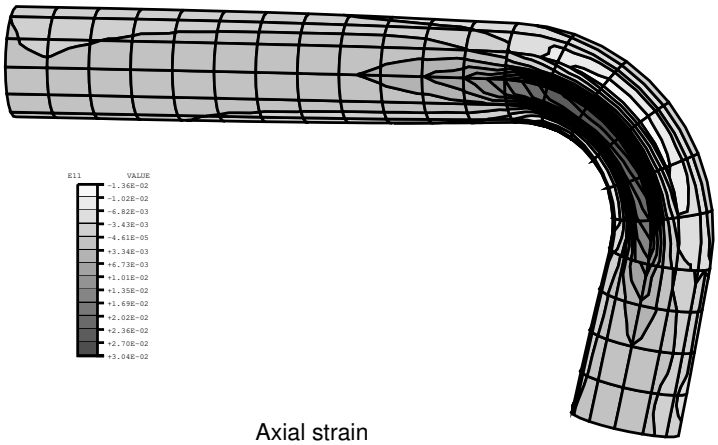


Figure 4: Moment-rotation response: pressure dependence.



Hoop strain



Axial strain

Figure 5: Strain distribution on the outside surface: closing moment case.

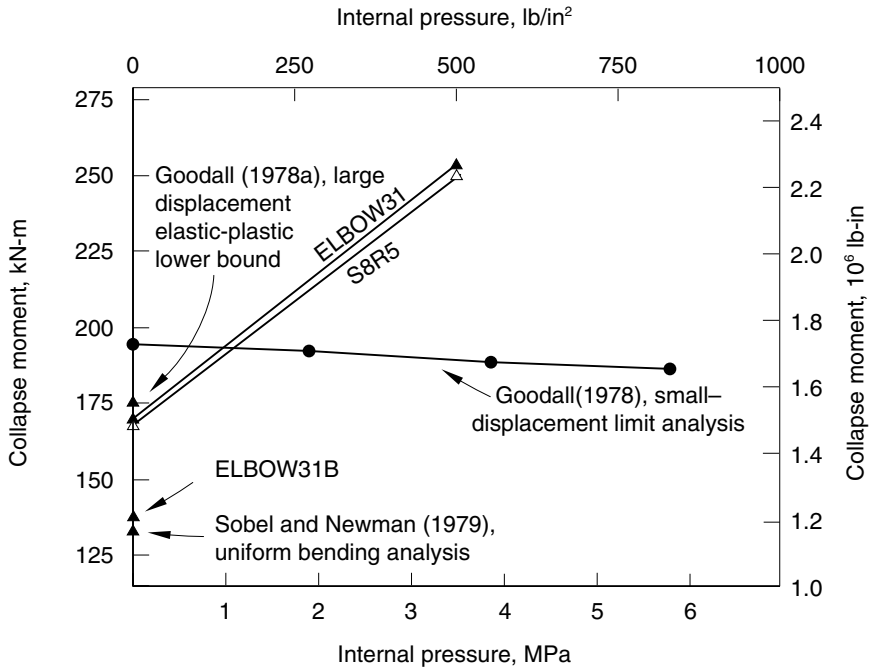


Figure 6: In-plane bending of an elbow, elastic-plastic collapse moment results.

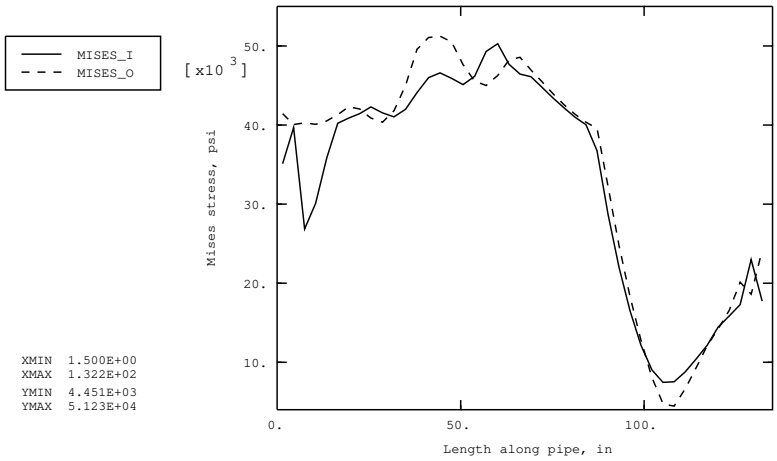


Figure 7: Von Mises stress distribution along the length of the piping system.

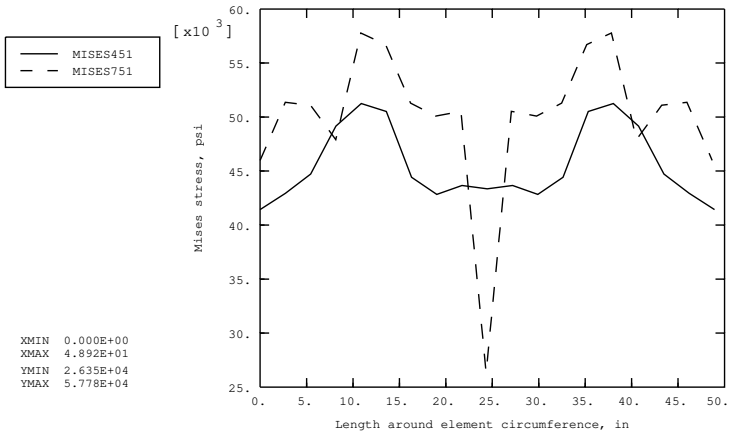


Figure 8: Von Mises stress distribution around the circumference of elements 451 and 751.

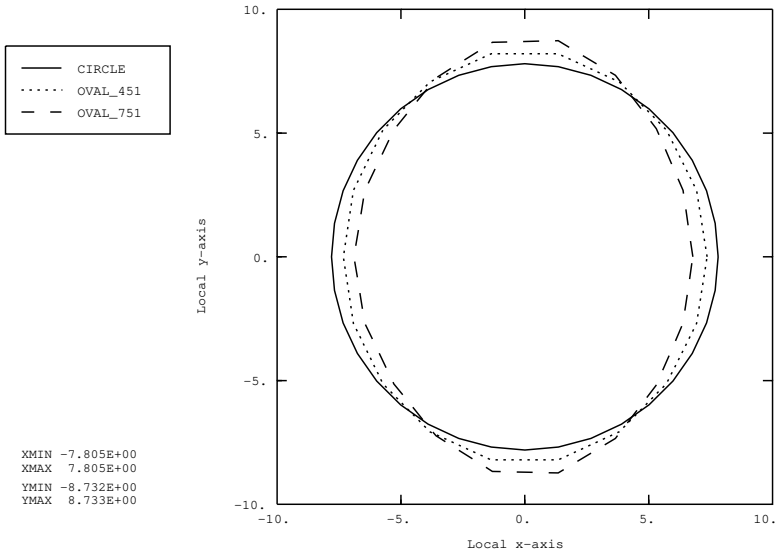


Figure 9: Ovalization of elements 451 and 751.

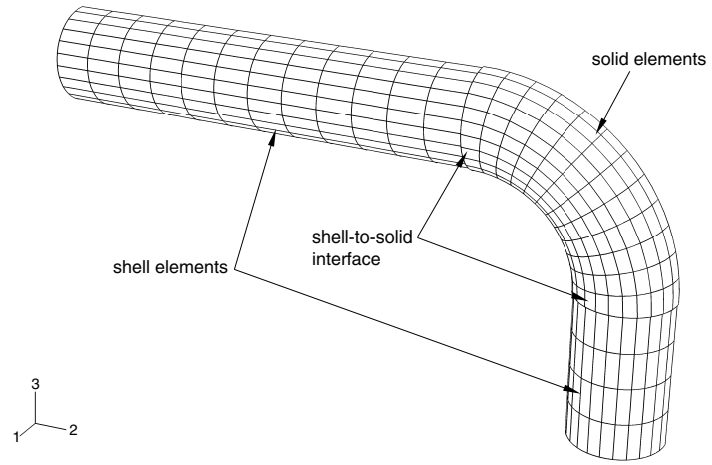


Figure 10: Shell-to-solid coupling model study.

Parametric study of a linear elastic pipeline under in-plane bending

Products: Abaqus/Standard Abaqus/Explicit

References:

- *Pipes and Pipebends with Deforming Cross-Sections: Elbow Elements*

Overview

This example demonstrates the ability of elbow elements to model the nonlinear response of initially circular pipes and pipebends accurately when the distortion of the cross-section by ovalization is significant.
It also provides some guidelines on the importance of including a sufficient number of Fourier modes in the elbow elements to capture the ovalization accurately. In addition, this example illustrates the shortcomings of using "flexibility knockdown factors" with simple beam elements in an attempt to capture the effects of ovalization in an ad hoc manner for large-displacement analyses. Similar analyses involving pipe elements in Abaqus/Explicit are included.

Elbows are used in piping systems because they ovalize more readily than straight pipes and, thus, provide flexibility in response to thermal expansion and other loadings that impose significant displacements on the system. Ovalization is the bending of the pipe wall into an oval—i.e., noncircular—configuration. The elbow is, thus, behaving as a shell rather than as a beam.

Geometry and model

The pipeline configuration used in the study is shown in *Figure 1*. It is a simple model with two straight pipe sections connected by a 90° elbow. The straight pipes are 25.4 cm (10.0 inches) in length, the radius of the curved section is 10.16 cm (4.0 inches), and the outer radius of the pipe section is 1.27 cm (0.5 inches). The wall thickness of the pipe is varied from 0.03175 cm to 0.2032 cm (0.0125 inches to 0.08 inches) in a parametric study, as discussed below. The pipe material is assumed to be isotropic linear elastic with a Young's modulus of 194 GPa (28.1×10^6 psi) and a Poisson's ratio of 0.0. The straight portions of the pipeline are assumed to be long enough so that warping at the ends of the structure is negligible.

Two loading conditions are analyzed. The first case is shown in *Figure 1* with unit inward displacements imposed on both ends of the structure. This loading condition has the effect of closing the pipeline in on itself. In the second case the sense of the applied unit displacements is outward, opening the pipeline. Both cases are considered to be large-displacement/small-strain analyses.

A parametric study comparing the results obtained with different element types (shells, elbows, and pipes) over a range of flexibility factors, k, is performed. As defined in Dodge and Moore (1972), the flexibility factor for an elbow is the ratio of the bending flexibility of the elbow segment to that of a straight pipe of the same dimensions, assuming small displacements and an elastic response. When the internal (gauge) pressure is zero, as is assumed in this study, k can be approximated as

$$k = \frac{1.66}{\lambda},$$

where

$$\lambda = \frac{Rt}{r^2\sqrt{1-\nu^2}},$$

R is the bend radius of the curved section, r is the mean radius of the pipe, t is the wall thickness of the pipe, and ν is Poisson's ratio. Changes in the flexibility factor are introduced by varying the wall thickness of the pipe.

The pipeline is modeled with three different element types: S4 shell elements, ELBOW31 elbow elements, and PIPE31 pipe elements. The S4 shell element model consists of a relatively fine mesh of 40 elements about the circumference and 75 elements along the length. This mesh is deemed fine enough to capture the true response of the pipeline accurately, although no mesh convergence studies are performed. Two analyses are conducted with the shell mesh: one with automatic stabilization using a constant damping factor (see *Automatic Stabilization of Static Problems with a Constant Damping Factor*), and one with adaptive automatic stabilization (see *Adaptive Automatic Stabilization Scheme*). The pipe and elbow element meshes consist of 75 elements along the length; the analyses with these element types do not use automatic stabilization.

The results of the shell element model with automatic stabilization using a constant damping factor are taken as the reference solution. The reaction force at the tip of the pipeline is used to evaluate the effectiveness of the pipe and elbow elements. In addition, the ovalization values of the pipeline cross-section predicted by the elbow element models are compared.

The elbow elements are tested with 0, 3, and 6 Fourier modes, respectively. In general, elbow element accuracy improves as more modes are used, although the computational cost increases accordingly. In addition to standard pipe elements, tests are performed on pipe elements with a special flexibility knockdown factor. Flexibility knockdown factors (Dodge and Moore, 1972) are corrections to the bending stiffness based upon linear semianalytical results. They are applied to simple beam elements in an attempt to capture the global effects of ovalization. The knockdown factor is implemented in the PIPE31 elements by scaling the true thickness by the flexibility factor; this is equivalent to scaling the moment of inertia of the pipe element by $\mathbf{1/k}$.

Results and discussion

The results obtained with the shell element model with automatic stabilization using a constant damping factor are taken as the reference solution. Very similar results are obtained with the same mesh using the adaptive automatic stabilization scheme.

The tip reaction forces due to the inward prescribed displacements for the various analysis models are shown in *Figure 2*. The results are normalized with respect to those obtained with the shell model. The results obtained with the ELBOW31 element model with 6 Fourier modes show excellent agreement with the reference solution over the entire range of flexibility factors considered in this study. The remaining four models generally exhibit excessively stiff response for all values of k. The PIPE31 element model, which uses the flexibility knockdown factor, shows a relatively constant error of about 20% over the entire range of flexibility factors. The 0-mode ELBOW31 element model and the PIPE31 element model without the knockdown factor produce very similar results for all values of k.

The normalized tip reaction forces due to the outward unit displacement for the various analysis models are shown in *Figure 3*. Again, the results obtained with the 6-mode ELBOW31 element model compare well with the reference shell solution. The 0-mode and 3-mode ELBOW31 and the PIPE31 (without the flexibility knockdown factor) element models exhibit overly stiff response. The PIPE31 element model with the knockdown factor has a transition region near $k = 1.5$, where the response changes from being too stiff to being too soft. The results in Abaqus/Explicit for pipe elements are consistent with those obtained in Abaqus/Standard.

Figure 4 and *Figure 5* illustrate the effect of the number of included Fourier modes (0, 3, and 6) on the ability of the elbow elements to model the ovalization in the pipebend accurately in both load cases considered in this study. By definition, the 0-mode model cannot ovalize, which accounts for its stiff response. The 3-mode and the 6-mode models show significant ovalization in both loading cases. *Figure 6* compares the ovalization of the 6-mode model in the opened and closed deformation states. It clearly illustrates that when the ends of the pipe are displaced inward (closing mode), the height of the pipe's cross-section gets smaller, thereby reducing the overall stiffness of the pipe; the reverse is true when the pipe ends are displaced outward: the height of the pipe's cross-section gets larger, thereby increasing the pipe stiffness. These three figures were produced with the aid of the elbow element postprocessing program `felbow.f` (*Creation of a data file to facilitate the postprocessing of elbow element results: FELBOW*), written in Fortran. The postprocessing programs `felbow.C` (*A C++ version of FELBOW*) and `felbow.py` (*An Abaqus Scripting Interface version of FELBOW*), written in C++

and Python, respectively, are also available for generating the data for these figures. The user must ensure that the output variables are written to the output database to use these two programs.

Parametric study

The performance of the pipe and elbow elements investigated in this example is analyzed conveniently in a parametric study using the Python scripting capabilities of Abaqus (*Scripting Parametric Studies*). We perform a parametric study in which eight analyses are executed automatically for each of the three element types (S4, ELBOW31, and PIPE31) discussed above; these parametric studies correspond to wall thickness values ranging from 0.03175 cm to 0.2032 cm (0.0125 inches to 0.08 inches).

The Python script file *elbowtest.psf* is used to perform the parametric study. The function `customTable` (shown below) is an example of advanced Python scripting (Lutz and Ascher, 1999), which is used in *elbowtest.psf*. Such advanced scripting is not routinely needed, but in this case a dependent variable such as k cannot be included as a column of data in an XYPLOT file. `customTable` is designed to overcome this limitation by taking an XYPLOT file from the parametric study and converting it into a new file of reaction forces versus flexibility factors (k).

```
##############################################################
#
def customTable(file1, file2):
    for line in file1.readlines():
        print line
        nl = string.split(line,',')

        disp = float(nl[0])
        bend_radius = float(nl[1])
        wall_thick  = float(nl[2])
        outer_pipe_radius = float(nl[3])
        poisson = float(nl[4])
        rf = float(nl[6])

        mean_rad = outer_pipe_radius - wall_thick/2.0
        k = bend_radius*wall_thick/mean_rad**2
        k = k/sqrt(1.e0 - poisson**2)
        k = 1.66e0/k

        outputstring = str(k) + ', ' + str(rf) + '\n'
        file2.write(outputstring)
#
##############################################################
```

Input files

elbowtest_shell.inp	S4 model.
elbowtest_shell_stabil_adap.inp	S4 model with adaptive stabilization.
elbowtest_elbow0.inp	ELBOW31 model with 0 Fourier modes.
elbowtest_elbow3.inp	ELBOW31 model with 3 Fourier modes.
elbowtest_elbow6.inp	ELBOW31 model with 6 Fourier modes.
elbowtest_pipek.inp	PIPE31 model with the flexibility knockdown factor.
elbowtest_pipek_xpl.inp	PIPE31 model with the flexibility knockdown factor in Abaqus/Explicit.
elbowtest_pipe.inp	PIPE31 model without the flexibility knockdown factor.

elbowtest_pipe_xpl.inp	PIPE31 model without the flexibility knockdown factor in Abaqus/Explicit.
elbowtest.psf	Python script file for the parametric study.

References

- Dodge, W. G., and S. E. Moore, "Stress Indices and Flexibility Factors for Moment Loadings on Elbows and Curved Pipes," *Welding Research Council Bulletin*, no. 179, 1972.
- Lutz, M., and D. Ascher, Learning Python, O'Reilly, 1999.

Figures

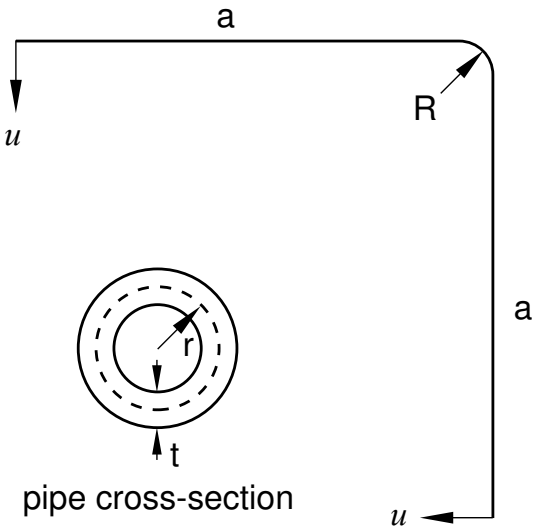


Figure 1: Pipeline geometry with inward prescribed tip displacements.

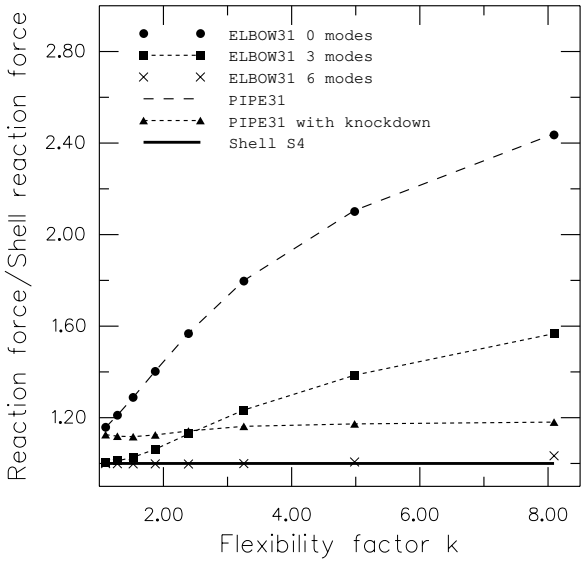


Figure 2: Normalized tip reaction force: closing displacement case.

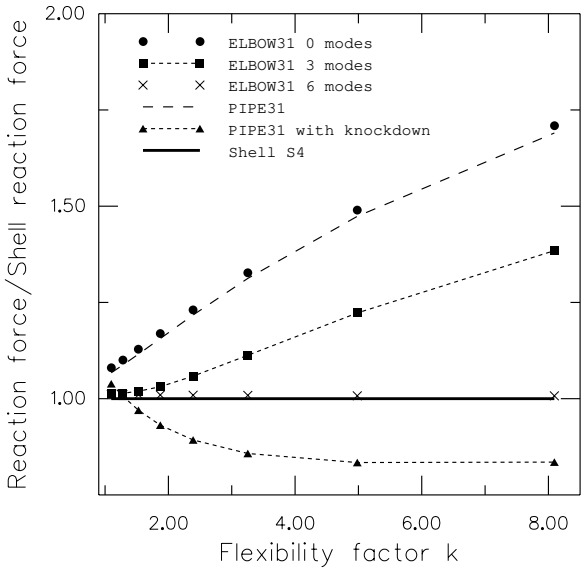


Figure 3: Normalized tip reaction force: opening displacement case.

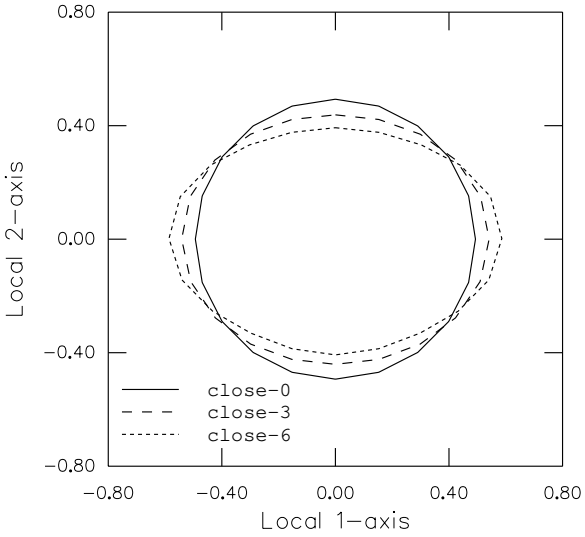


Figure 4: Ovalization of the ELBOW31 cross-sections for 0, 3, and 6 Fourier modes: closing displacement case.

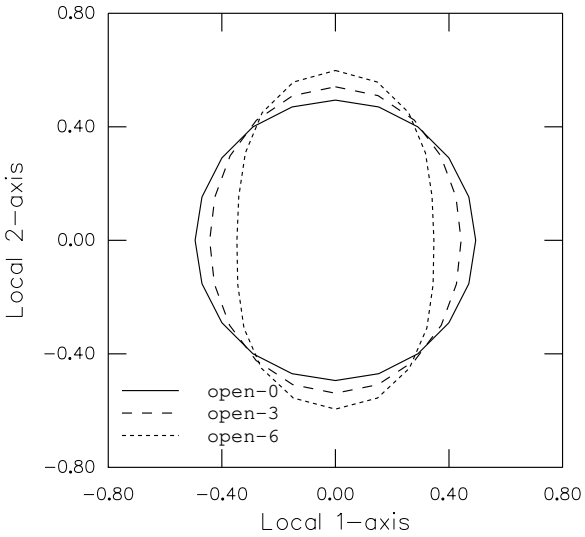


Figure 5: Ovalization of the ELBOW31 cross-sections for 0, 3, and 6 Fourier modes: opening displacement case.

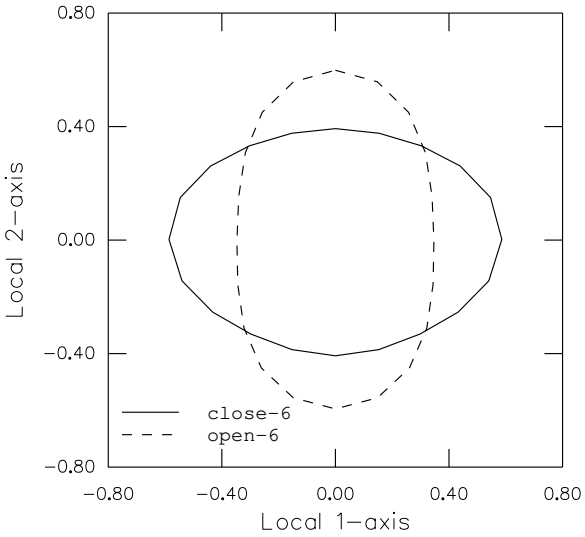


Figure 6: Ovalization of the ELBOW31 cross-sections for 6 Fourier modes: opening and closing displacement cases.

Indentation of an elastomeric foam specimen with a hemispherical punch

Products: Abaqus/Standard Abaqus/Explicit Abaqus/Design

Overview

This example shows an application of elastomeric foam materials when used in energy absorption devices.

We consider a cylindrical specimen of an elastomeric foam, indented by a rough, rigid, hemispherical punch. Examples of elastomeric foam materials are cellular polymers such as cushions, padding, and packaging materials. This example uses the same geometry as the crushable foam model of *Simple tests on a crushable foam specimen*, but with a slightly different mesh.

In addition, design sensitivity analysis is carried out for a shape design parameter and a material design parameter to illustrate the usage of design sensitivity analysis for a problem involving contact.

Geometry and model

The axisymmetric model (135 linear 4-node elements) analyzed is shown in *Figure 1*. The mesh refinement is biased toward the center of the foam specimen where the largest deformation is expected. The foam specimen has a radius of 600 mm and a thickness of 300 mm. The punch has a radius of 200 mm. The bottom nodes of the mesh are fixed, while the outer boundary is free to move.

A contact pair is defined between the punch, which is modeled by a rough spherical rigid surface, and a secondary surface composed of the faces of the axisymmetric elements in the contact region. The friction coefficient between the punch and the foam is 0.8. A point mass of 200 kg representing the weight of the punch is attached to the rigid body reference node. The model is analyzed in both Abaqus/Standard and Abaqus/Explicit.

Material

The elastomeric foam material is defined using experimental test data. The uniaxial compression and simple shear data stress-strain curves are shown in *Figure 2*. Other available test data options are biaxial test data, planar test data, and volumetric test data. The test data are defined in terms of nominal stress and nominal strain values.

Abaqus performs a nonlinear least-squares fit of the test data to determine the hyperfoam coefficients μ_i, α_i, and β_i.

Details of the formulation and usage of the hyperfoam model are given in *Hyperelastic Behavior in Elastomeric Foams*, *Hyperelastic material behavior*, and *Fitting of hyperelastic and hyperfoam constants*. *Fitting of elastomeric foam test data* illustrates the fitting of elastomeric foam test data to derive the hyperfoam coefficients.

For the material used in this example, β_i is zero, since the effective Poisson's ratio, ν, is zero as specified by the POISSON parameter. The order of the series expansion is chosen to be $N = 2$ since this fits the test data with sufficient accuracy. It also provides a more stable model than the $N = 3$ case.

The viscoelastic properties in Abaqus are specified in terms of a relaxation curve (shown in *Figure 3*) of the normalized modulus $M(t)/M_0$, where $M(t)$ is the shear or bulk modulus as a function of time and M_0 is the instantaneous modulus as determined from the hyperfoam model. This requires Abaqus to calculate the Prony series parameters from data taken from shear and volumetric relaxation tests. The relaxation data are specified as part of the definition of shear test data but actually apply to both shear and bulk moduli when used in conjunction with the hyperfoam model. Abaqus performs a nonlinear least-squares fit of the relaxation data to a Prony series to determine the coefficients, $\overline{g}_i^P$, and the relaxation periods, τ_i. A maximum order of 2 is used for fitting the

Prony series. If creep data are available, you can specify normalized creep compliance data to compute the Prony series parameters.

A rectangular material orientation is defined for the foam specimen, so stress and strain are reported in material axes that rotate with the element deformation. This is especially useful when looking at the stress and strain values in the region of the foam in contact with the punch in the direction normal to the punch (direction "22").

The rough surface of the punch is modeled by specifying a friction coefficient of 0.8 for the contact surface interaction.

Procedure and loading definitions

Two cases are analyzed. In the first case the punch is displaced statically downward to indent the foam, and the reaction force-displacement relation is measured for both the purely elastic and viscoelastic cases. In the second case the punch statically indents the foam through gravity loading and is then subjected to impulsive loading. The dynamic response of the punch is sought as it interacts with the viscoelastic foam.

Case 1

In Abaqus/Standard the punch is displaced downward by a prescribed displacement boundary condition in the first step, indenting the foam specimen by a distance of 250 mm. Geometric nonlinearity should be accounted for in this step, since the response involves large deformation. In the second step the punch is displaced back to its original position. Two analyses are performed—one using the static procedure for both steps and the other using the quasi-static procedure for both steps. During a static step the material behaves purely elastically, using the properties specified with the hyperfoam model. The quasi-static, direct-integration implicit dynamic, or fully coupled thermal-stress procedure must be used to activate the viscoelastic behavior. In this case the punch is pushed down in a period of one second and then moved back up again in one second. The accuracy of the creep integration in the quasi-static procedure can be controlled and is typically calculated by dividing an acceptable stress error tolerance by a typical elastic modulus. In this problem we estimate a stress error tolerance of about 0.005 MPa and use the initial elastic modulus, $E_0 = 2 \sum \mu_i = 0.34$, to determine an accuracy tolerance of 0.01.

In Abaqus/Explicit the punch is also displaced downward by a prescribed displacement boundary condition, indenting the foam by a depth of 250 mm. The punch is then lifted back to its original position. In this case the punch is modeled as either an analytical rigid surface or a discrete rigid surface defined with RAX2 elements. The entire analysis runs for 2 seconds. The actual time period of the analysis is large by explicit dynamic standards. Hence, to reduce the computational time, the mass density of the elements is increased artificially to increase the stable time increment without losing the accuracy of the solution. The mass scaling factor is set to 10, which corresponds to a speedup factor of $\sqrt{10}$. The reaction force-displacement relation is measured for both the elastic and viscoelastic cases.

Case 2

The Abaqus/Standard analysis is composed of three steps. The first step is a quasi-static step, where gravity loading is applied to the point mass of the punch. The gravity loading is ramped up in two seconds, and the step is run for a total of five seconds to allow the foam to relax fully. In the second step, which is a direct-integration implicit dynamic step, an impulsive load in the form of a half sine wave amplitude with a peak magnitude of 5000 N is applied to the punch over a period of one second. In the third step, also a direct-integration implicit dynamic step, the punch is allowed to move freely until the vibration is damped out by the viscoelastic foam. For a dynamic analysis with automatic time incrementation, the value of the half-increment residual tolerance for the direct-integration implicit dynamic procedure controls the accuracy of the time integration. For systems that have significant energy dissipation, such as this heavily damped model, a relatively high value of this tolerance can be chosen. We choose the tolerance to be 100 times a typical average force that we estimate (and later confirm from the analysis results) to be on the order of 50 N. Thus, the half-increment residual tolerance

is 5000 N. For the second direct-integration implicit dynamic step we bypass calculation of initial accelerations at the beginning of the step, since there is no sudden change in load to create a discontinuity in the accelerations.

In the Abaqus/Explicit analysis the punch indents the foam quasi-statically through gravity loading and is then subjected to an impulsive loading. In the first step gravity loading is applied to the point mass of the punch, and the foam is allowed to relax fully. The mass scaling factor in this step is set to 10. In the second step a force in the form of a half sine wave is applied to the punch, and the dynamic response of the punch is obtained as it interacts with the viscoelastic foam. In the third step the load is removed, and the punch is allowed to move freely. Mass scaling is not used in Steps 2 and 3 since the true dynamic response is sought.

Design sensitivity analysis

For the design sensitivity analysis (DSA) carried out with static steps in Abaqus/Standard, the hyperfoam material properties are given using direct input of coefficients based on the test data given above. For $N = 2$, the coefficients are $\mu_1 = 0.16245$, $\mu_2 = 3.59734\text{E}{-}05$, $\alpha_1 = 8.89239$, $\alpha_2 = -4.52156$, and $\beta_i = 0.0$. Since the quasi-static procedure is not supported for DSA, it is replaced with the static procedure and the viscoelastic material behavior is removed. In addition, since a more accurate tangent stiffness leads to improved sensitivity results, the solution controls are used to tighten the residual tolerance.

The material parameter μ_1 is chosen as one of the design parameters. The other (shape) design parameter used for design sensitivity analysis, L, represents the thickness of the foam at the free end (see *Figure 1*). The z-coordinates of the nodes on the top surface are assumed to depend on L via the equation $z = 300 + (L - 300)\, r/600$. The r-coordinates are considered to be independent of L. To define this dependency in Abaqus, the gradients of the coordinates with respect to L

$$\frac{dr}{dL} = 0$$

$$\frac{dz}{dL} = \frac{r}{600}$$

are given as part of the specification of parameter shape variation.

Results and discussion

This problem tests the hyperfoam material model in Abaqus but does not provide independent verification of the model. The results for all analyses are discussed in the following paragraphs.

Case 1

Deformation and contour plots for oriented S22 stress and LE22 strain are shown for the viscoelastic foam in *Figure 4* through *Figure 6* for the Abaqus/Standard analysis and *Figure 7* through *Figure 9* for the Abaqus/Explicit analysis. Even though the foam has been subjected to large strains, only moderate distortions occur because of the zero Poisson's ratio. The maximum logarithmic strain is on the order of −1.85, which is equivalent to a stretch of $\lambda = e^{-1.85} = 0.16$ or a nominal compressive strain of 84%, indicating severe compression of the foam.

Figure 10 shows a comparison of the punch reaction force histories obtained with Abaqus/Standard and Abaqus/Explicit. In the viscoelastic case the stresses relax during loading and, consequently, lead to a softer response than in the purely elastic case. A comparison of the force-displacement responses obtained with Abaqus/Standard and Abaqus/Explicit is shown in *Figure 11*. The purely elastic material is reversible, while the viscoelastic material shows hysteresis.

Case 2

Figure 12 shows various displaced configurations during the Case 2 analysis for Abaqus/Standard and Abaqus/Explicit. Displacement, velocity, and acceleration histories for the punch are shown in *Figure 13*, *Figure*

14, and *Figure 15*, respectively. The displacement is shown to reach a steady value at the stress relaxation stage, followed by a severe drop due to the impulsive dynamic load. This is followed by a rebound and then finally by a rapid decay of the subsequent oscillations due to the strong damping provided by the viscoelasticity of the foam.

Abaqus/Design

Figure 16 and *Figure 17* show the contours of sensitivity of the displacement in the z-direction to the design parameters $\boldsymbol{L}$ and $\boldsymbol{\mu_1}$, respectively. *Figure 18* and *Figure 19* show the contours of sensitivity of S22 to the design parameters L and $\boldsymbol{\mu_1}$, respectively. To provide an independent assessment of the results provided by Abaqus, sensitivities were computed using the overall finite difference (OFD) technique. The central difference method with a perturbation size of 0.1% of the value of the design parameter was used to obtain the OFD results. *Table 1* shows that the sensitivities computed using Abaqus compare well with the overall finite difference results.

Input files

indentfoam_std_elast_1.inp	Case 1 of the Abaqus/Standard example using elastic properties of the foam, which is statically deformed in two **STATIC* steps.
indentfoam_std_elast_1_st.inp	Case 1 of the Abaqus/Standard example (CAX4R elements with hourglass control based on total stiffness) using elastic properties of the foam, which is statically deformed in two **STATIC* steps.
indentfoam_std_elast_1_eh.inp	Case 1 of the Abaqus/Standard example (CAX4R elements with enhanced hourglass control) using elastic properties of the foam, which is statically deformed in two **STATIC* steps.
indentfoam_std_visco_1.inp	Case 1 of the Abaqus/Standard example using viscoelastic properties of the foam, which is statically deformed in two **VISCO* steps.
indentfoam_std_visco_1_st.inp	Case 1 of the Abaqus/Standard example (CAX4R elements with hourglass control based on total stiffness) using viscoelastic properties of the foam, which is statically deformed in two **VISCO* steps.
indentfoam_std_visco_1_eh.inp	Case 1 of the Abaqus/Standard example (CAX4R elements with enhanced hourglass control) using viscoelastic properties of the foam, which is statically deformed in two **VISCO* steps.
indentfoam_std_visco_2.inp	Case 2 of the Abaqus/Standard example using viscoelastic properties of the foam.
indentfoam_std_visco_2_surf.inp	Case 2 of the Abaqus/Standard example using viscoelastic properties of the foam. Surface-to-surface contact is utilized.
indentfoam_xpl_elast_1.inp	Case 1 of the Abaqus/Explicit example using elastic properties of the foam with the punch modeled as an analytical rigid surface.
indentfoam_xpl_elast_1_subcyc.inp	Case 1 of the Abaqus/Explicit example using elastic properties of the foam with the punch modeled as an analytical rigid surface using the subcycling feature.
indentfoam_xpl_elast_fac_1.inp	Case 1 of the Abaqus/Explicit example using elastic properties of the foam with the punch modeled as a faceted rigid surface.
indentfoam_xpl_visco_1.inp	Case 1 of the Abaqus/Explicit example using viscoelastic properties of the foam with the punch modeled as an analytical rigid surface.
indentfoam_xpl_visco_2.inp	Case 2 of the Abaqus/Explicit example using viscoelastic properties of the foam with the punch modeled as an analytical rigid surface.
indentfoamhemipunch_dsa.inp	Design sensitivity analysis.

Tables

Table 1: Comparison of normalized sensitivities at the end of the analysis computed using Abaqus and the overall finite difference (OFD) method.

Normalized sensitivity	Abaqus	OFD
$\frac{\mu_1}{\lvert S22\rvert^{max}}\left\lvert\frac{d\,S22}{d\mu_1}\right\rvert_{max}$	0.4921	0.4922
$\frac{L}{\lvert S22\rvert^{max}}\left\lvert\frac{d\,S22}{dL}\right\rvert_{max}$	1.085	1.104
$\frac{\mu_1}{\lvert u_2\rvert^{max}}\left\lvert\frac{du_2}{d\mu_1}\right\rvert_{max}$	0.006925	0.006927
$\frac{L}{\lvert u_2\rvert^{max}}\left\lvert\frac{du_2}{dL}\right\rvert_{max}$	0.4059	0.4120
$\frac{\mu_1}{CPRESS^{max}}\left\lvert\frac{d\,CPRESS}{d\mu_1}\right\rvert_{max}$	0.5084	0.5085
$\frac{L}{CPRESS^{max}}\left\lvert\frac{d\,CPRESS}{dL}\right\rvert_{max}$	0.3252	0.3207

Figures

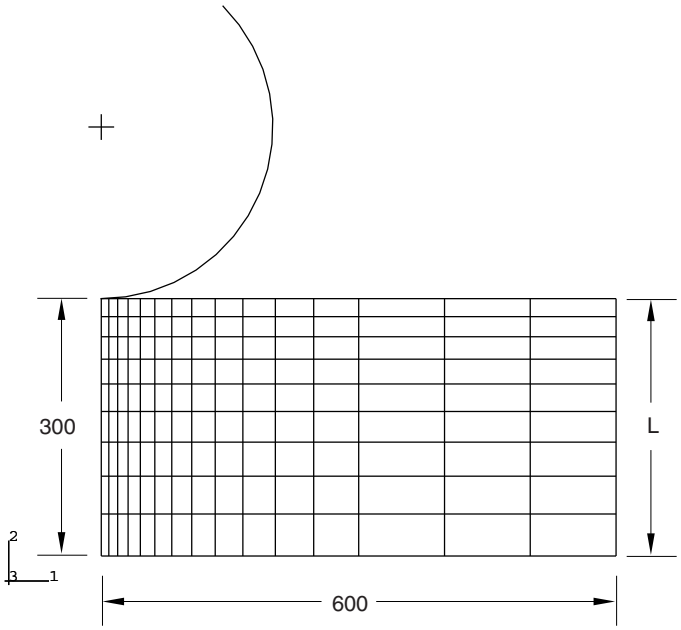


Figure 1: Model for foam indentation by a spherical punch.

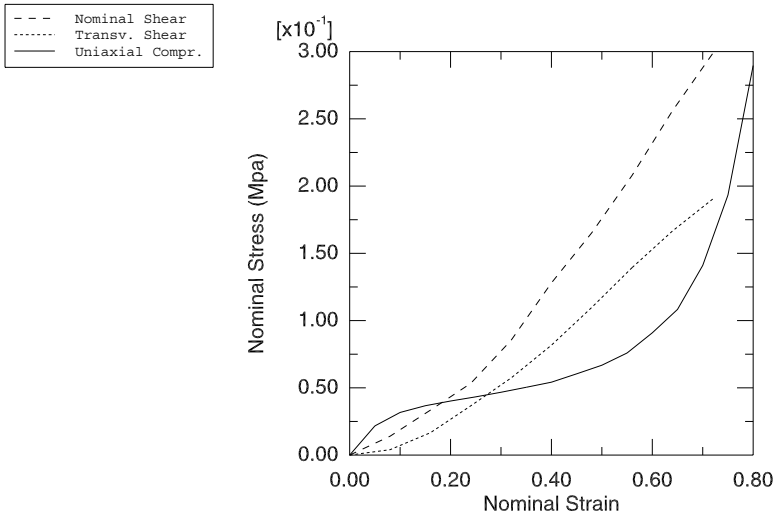


Figure 2: Elastomeric foam stress-strain curves.

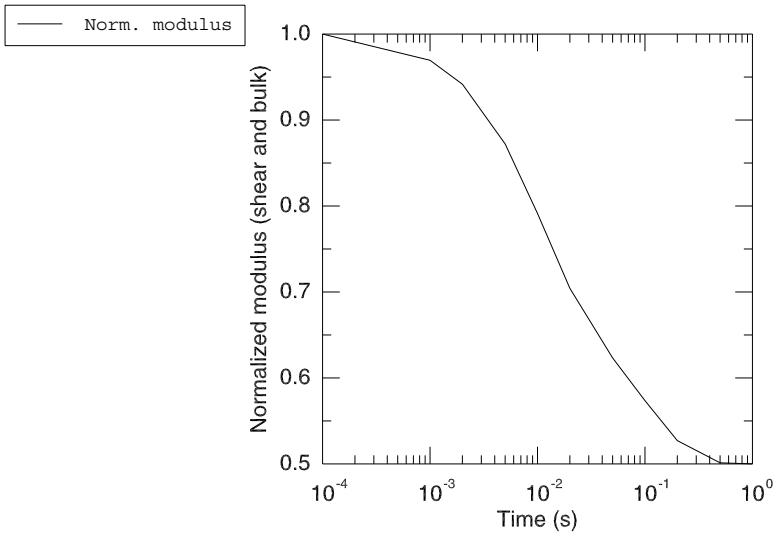


Figure 3: Elastic modulus relaxation curve.

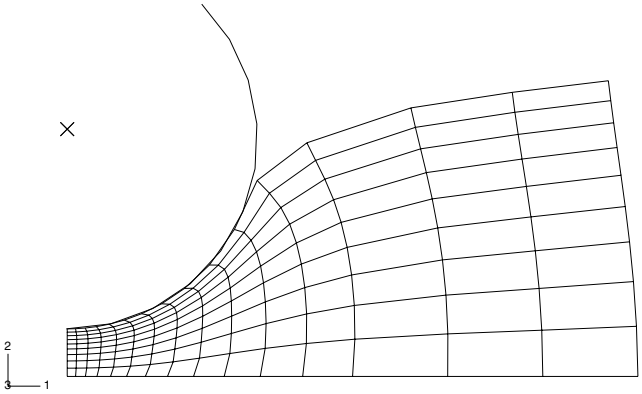


Figure 4: Maximum deformation of viscoelastic foam: Case 1, Abaqus/Standard.

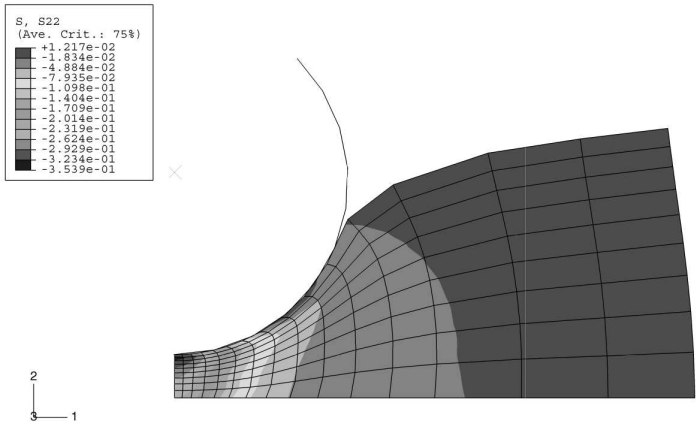


Figure 5: S22 contour plot of viscoelastic foam: Case 1, Abaqus/Standard.

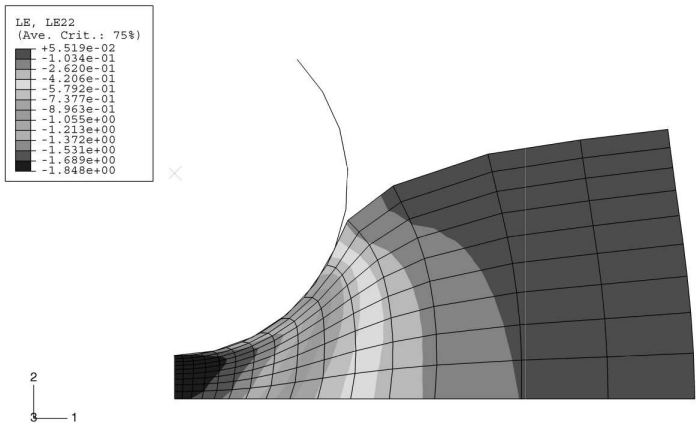


Figure 6: LE22 contour plot of viscoelastic foam: Case 1, Abaqus/Standard.

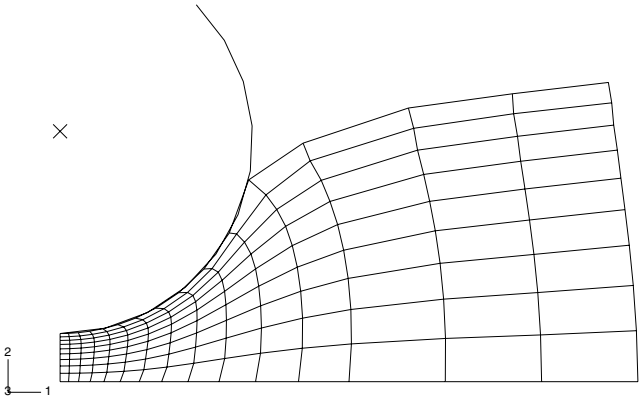


Figure 7: Deformed plot at 1.0 s: Case 1, Abaqus/Explicit.

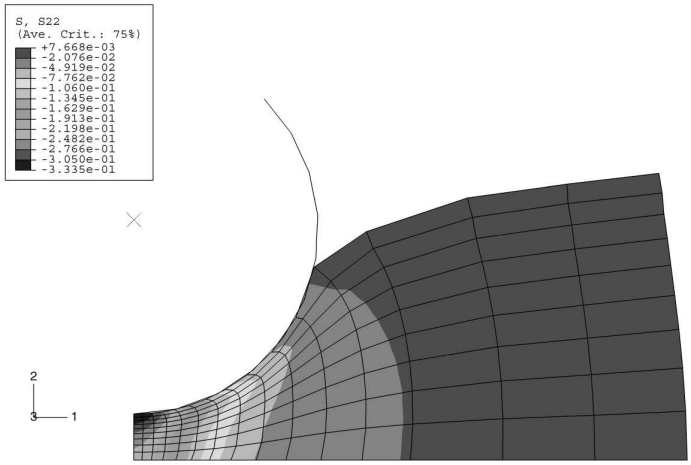


Figure 8: S22 contour plot of viscoelastic foam at 1.0 s: Case 1, Abaqus/Explicit.

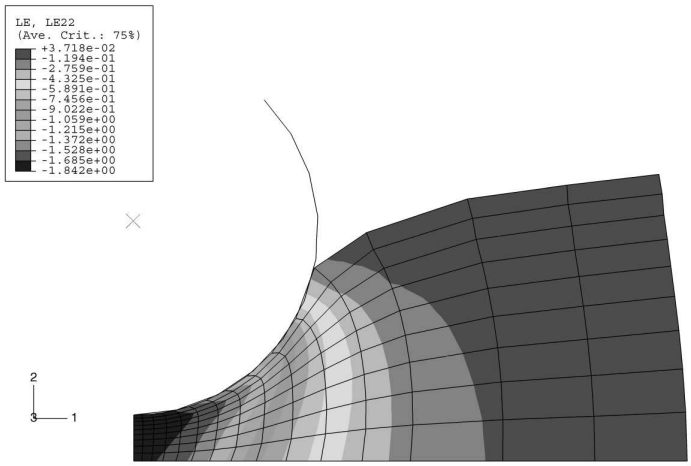


Figure 9: LE22 contour plot of viscoelastic foam at 1.0 s: Case 1, Abaqus/Explicit.

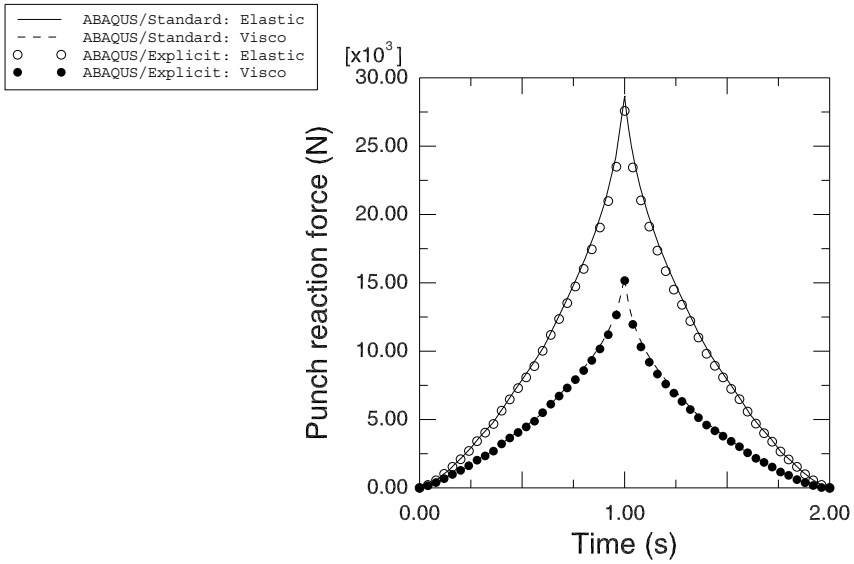


Figure 10: Punch reaction force history: Case 1.

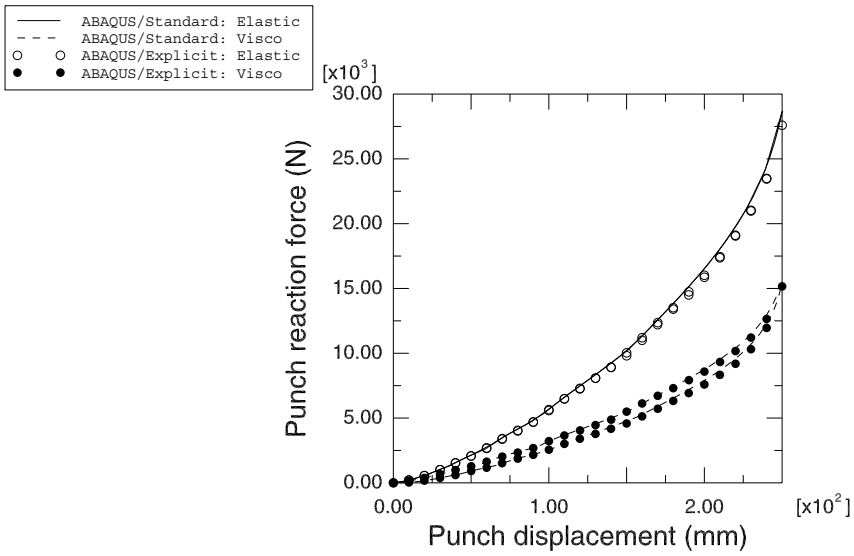


Figure 11: Punch reaction force versus displacement response (loading-unloading curves): Case 1.

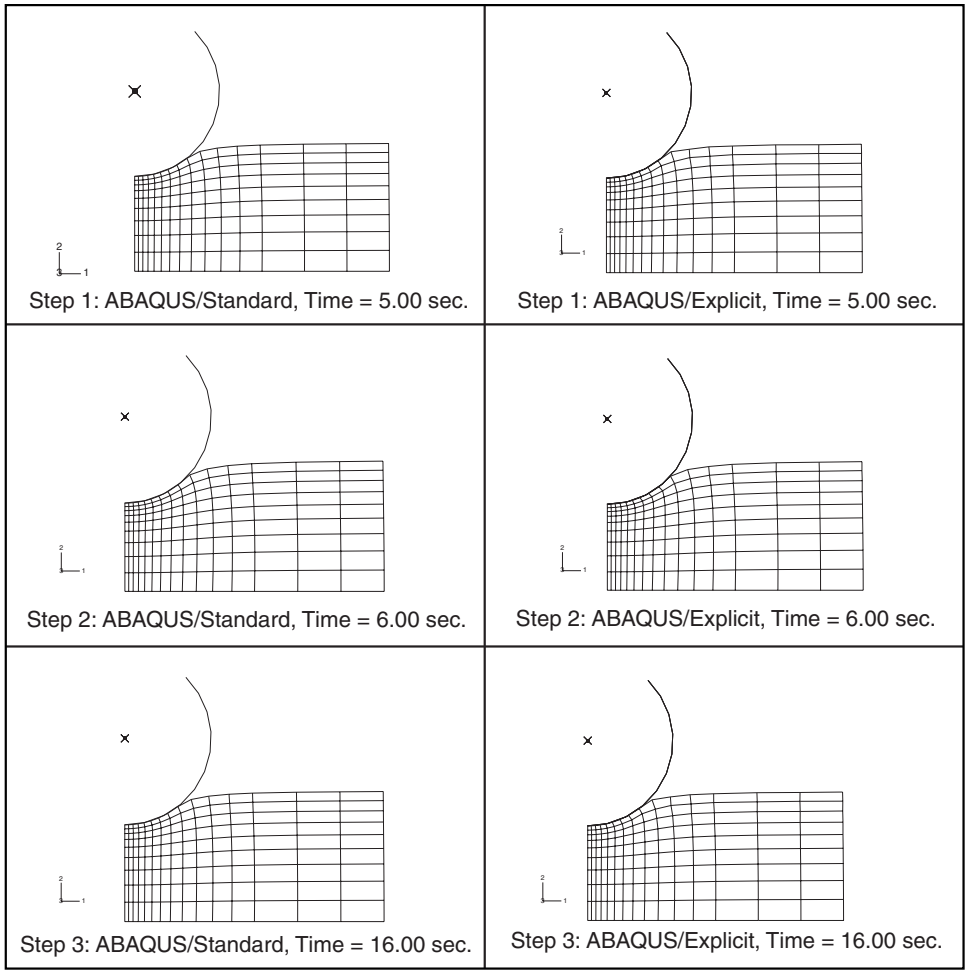


Figure 12: Deformed shape plots at the end of visco and dynamic steps: Case 2, Abaqus/Standard (left) and Abaqus/Explicit (right).

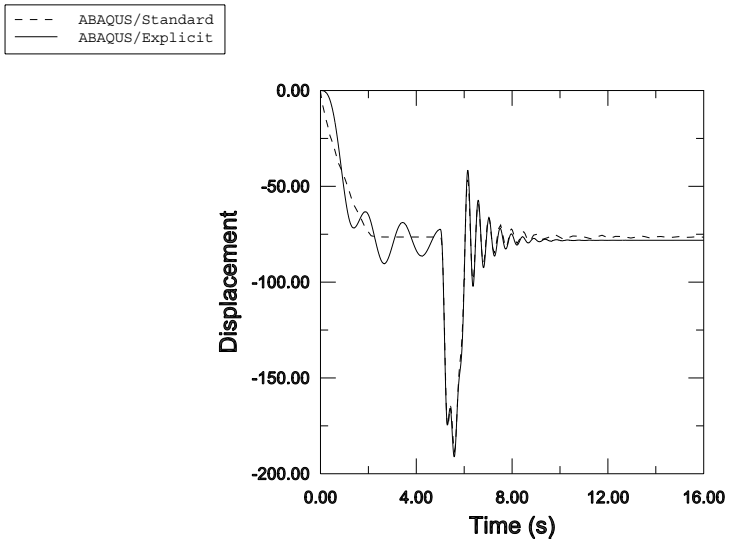


Figure 13: Displacement histories of the punch: Case 2, Abaqus/Standard and Abaqus/Explicit.

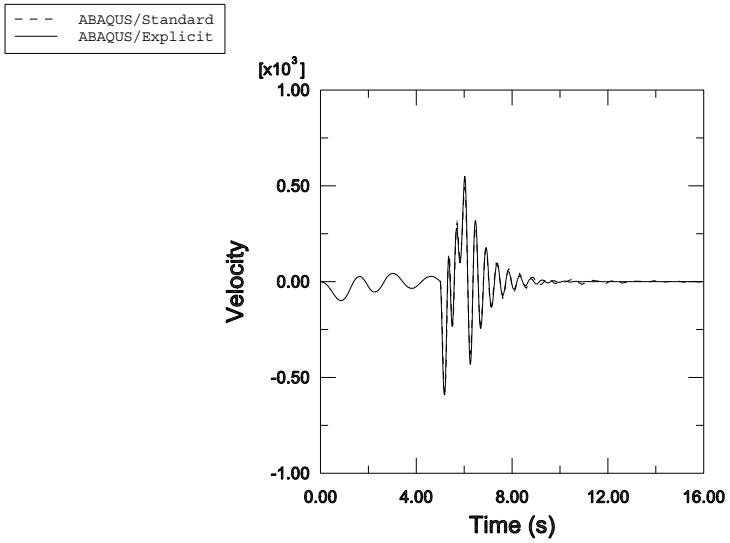


Figure 14: Velocity histories of the punch: Case 2, Abaqus/Standard and Abaqus/Explicit.

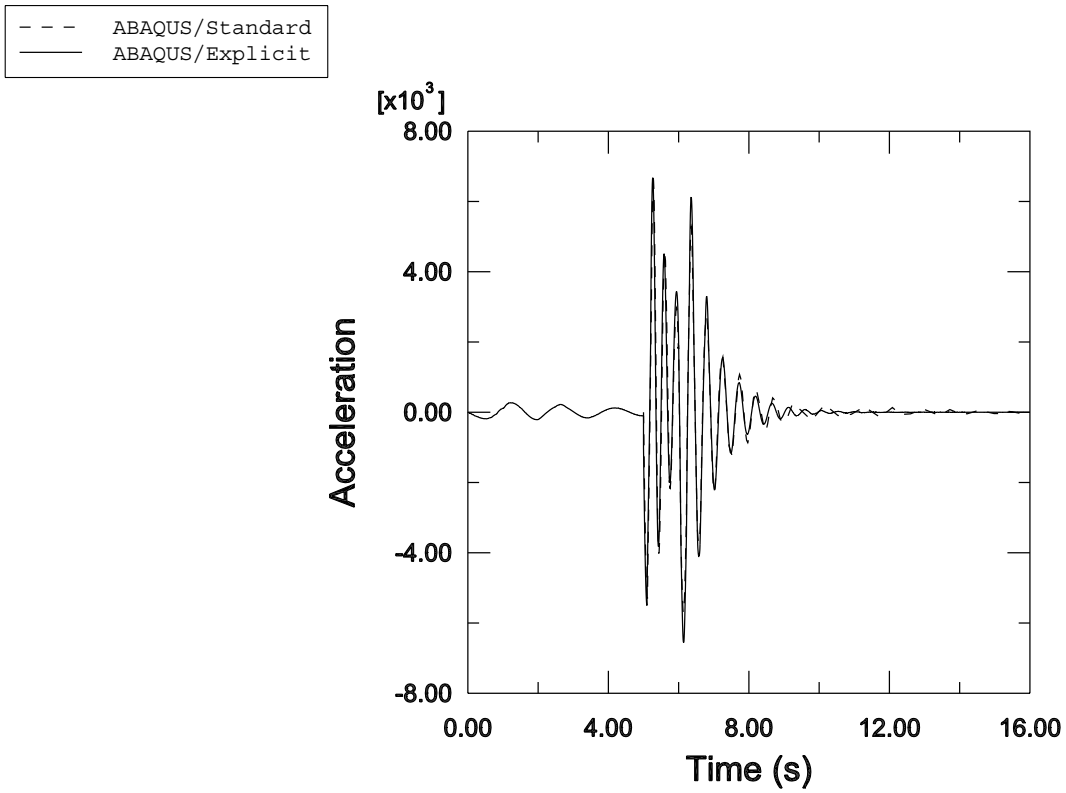


Figure 15: Acceleration histories of the punch: Case 2, Abaqus/Standard and Abaqus/Explicit.

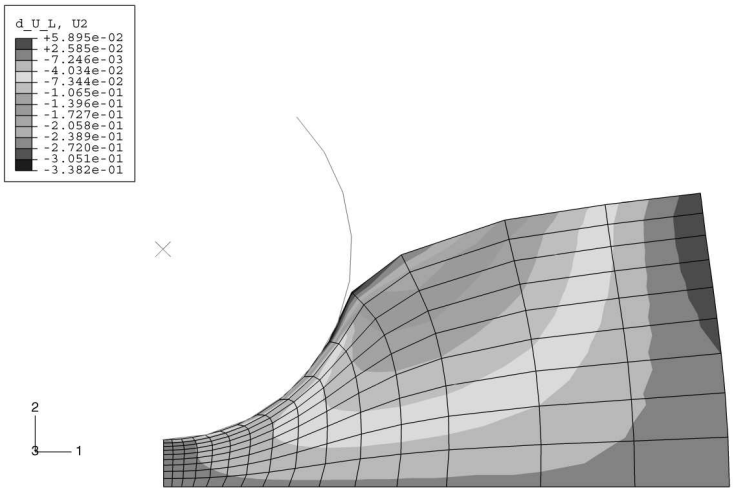


Figure 16: Sensitivities at the end of the analysis for displacement in the z-direction with respect to L.

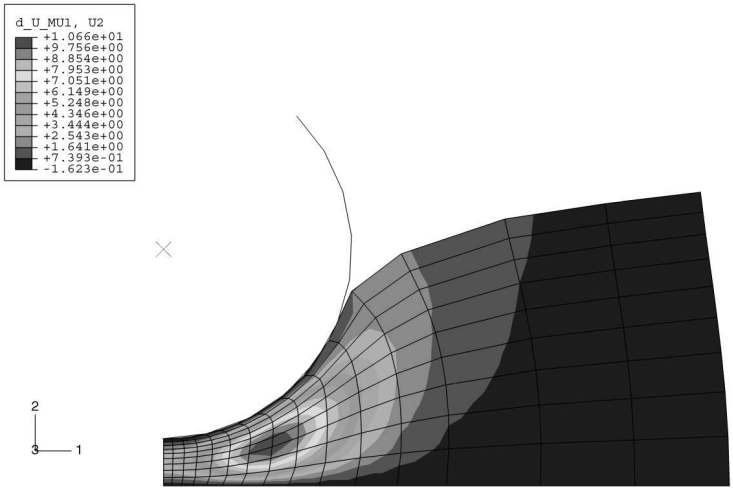


Figure 17: Sensitivities at the end of the analysis for displacement in the z-direction with respect to μ_1.

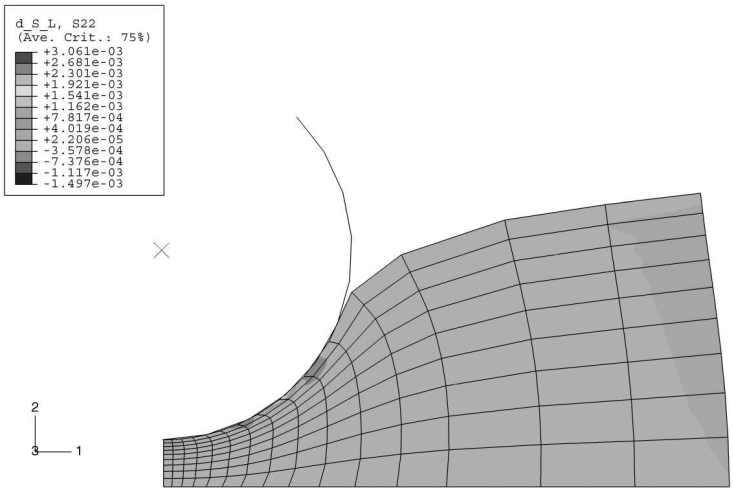


Figure 18: Sensitivities at the end of the analysis for stress S22 with respect to L.

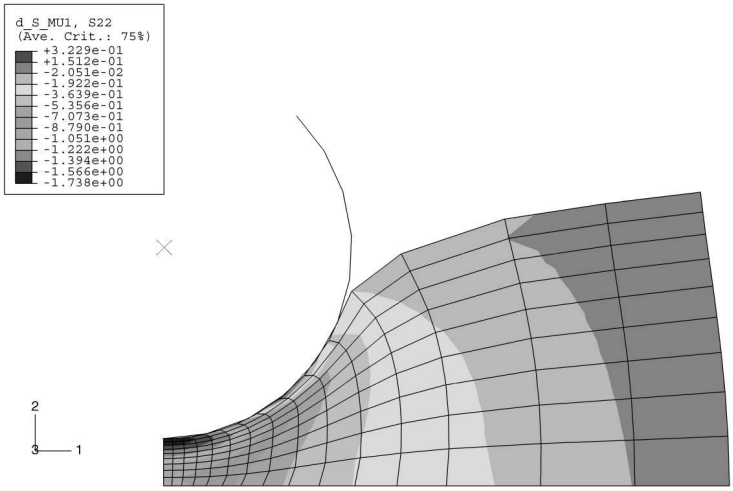


Figure 19: Sensitivities at the end of the analysis for stress S22 with respect to μ_1.

Collapse of a concrete slab

Products: Abaqus/Standard Abaqus/Explicit

References:

- *Concrete Smeared Cracking*
- *Cracking Model for Concrete*

Overview

This problem examines the use of the smeared crack model and the brittle cracking model for the analysis of reinforced concrete structures.

Geometric modeling

The geometry of the problem is defined in *Figure 1*. A square slab is supported in the transverse direction at its four corners and loaded by a point load at its center. The slab is reinforced in two directions at 75% of its depth. The reinforcement ratio (volume of steel/volume of concrete) is 8.5×10^{-3} in each direction. The slab was tested experimentally by McNeice (1967) and has been analyzed by a number of workers, including Hand et al. (1973), Lin and Scordelis (1975), Gilbert and Warner (1978), Hinton et al. (1981), and Crisfield (1982).

Symmetry conditions allow us to model one-quarter of the slab. A 3×3 mesh of 8-node shell elements is used for the Abaqus/Standard analysis. No mesh convergence studies have been performed, but the reasonable agreement between the analysis results and the experimental data suggests that the mesh is adequate to predict overall response parameters with usable accuracy. Three different meshes are used in Abaqus/Explicit to assess the sensitivity of the results to mesh refinement: a coarse 6×6 mesh, a medium 12×12 mesh, and a fine 24×24 mesh of S4R elements. Nine integration points are used through the thickness of the concrete to ensure that the development of plasticity and failure is modeled adequately. The two-way reinforcement is modeled using layers of uniaxial reinforcement (rebars). Symmetry boundary conditions are applied on the two edges of the mesh, and the corner point is restrained in the transverse direction.

Material properties

The material data are given in *Table 1*. The material properties of concrete are taken from Gilbert and Warner (1978). Some of these data are assumed values, because they are not available for the concrete used in the experiment. The assumed values are taken from typical concrete data. The compressive behavior of concrete in the cracking model in Abaqus/Explicit is assumed to be linear elastic. This is a reasonable assumption for a case such as this problem, where the behavior of the structure is dominated by cracking resulting from tension in the slab under bending.

The modeling of the concrete-reinforcement interaction and the energy release at cracking is of critical importance to the response of a structure such as this once the concrete starts to crack. These effects are modeled in an indirect way by adding "tension stiffening" to the plain concrete model. This approach is described in *A cracking model for concrete and other brittle materials*, *Concrete Smeared Cracking*, and *Cracking Model for Concrete*. The simplest tension stiffening model defines a linear loss of strength beyond the cracking failure of the concrete. In this example three different values for the strain beyond failure at which all strength is lost (5×10^{-4}, 1×10^{-3}, and 2×10^{-3}) are used to illustrate the effect of the tension stiffening parameters on the response.

Since the response is dominated by bending, it is controlled by the material behavior normal to the crack planes. The material's shear behavior in the plane of the cracks is not important. Consequently, the choice of shear retention has no significant influence on the results. In Abaqus/Explicit the shear retention chosen is exhausted

at the same value of the crack opening at which tension stiffening is exhausted. In Abaqus/Standard full shear retention is used because it provides a more efficient numerical solution.

Solution control

Since considerable nonlinearity is expected in the response, including the possibility of unstable regimes as the concrete cracks, the modified Riks method is used with automatic incrementation in the Abaqus/Standard analysis. With the Riks method the load data and solution parameters serve only to give an estimate of the initial increment of load. In this case it seems reasonable to apply an initial load of 1112 N (250 lb) to the quarter-model for a total initial load on the structure of 4448 N (1000 lb). This can be accomplished by specifying a load of 22241 N (5000 lb) and an initial time increment of 0.05 out of a total time period of 1.0. The analysis is terminated when the central displacement reaches 25.4 mm (1 in).

Since Abaqus/Explicit is a dynamic analysis program and in this case we are interested in static solutions, the slab must be loaded slowly enough to eliminate any significant inertia effects. The slab is loaded in its center by applying a velocity that increases linearly from 0 to 2.0 in/second such that the center displaces a total of 1 inches in 1 second. This very slow loading rate ensures quasi-static solutions; however, it is computationally expensive. The CPU time required for this analysis can be reduced in one of two ways: the loading rate can be increased incrementally until it is judged that any further increase in loading rate would no longer result in a quasi-static solution, or mass scaling can be used (see *Explicit Dynamic Analysis*). These two approaches are equivalent. Mass scaling is used here to demonstrate the validity of such an approach when it is used in conjunction with the brittle cracking model. Mass scaling is done by increasing the density of the concrete and the reinforcement by a factor of 100, thereby increasing the stable time increment for the analysis by a factor of 10 and reducing the computation time by the same amount while using the original slow loading rate. *Figure 4* shows the load-deflection response of the slab for analyses using the 12×12 mesh with and without mass scaling. The mass scaling used does not affect the results significantly; therefore, all subsequent analyses are performed using mass scaling.

Results and discussion

Results for each analysis are discussed in the following sections.

Abaqus/Standard results

The numerical and experimental results are compared in *Figure 2* on the basis of load versus deflection at the center of the slab. The strong effect of the tension stiffening assumption is very clear in that plot. The analysis with tension stiffening, such that the tensile strength is lost at a strain of 10^{-3} beyond failure, shows the best agreement with the experiment. This analysis provides useful information from a design viewpoint. The failure pattern in the concrete is illustrated in *Figure 3*, which shows the predicted crack pattern on the lower surface of the slab at a central deflection of 7.6 mm (0.3 in).

Abaqus/Explicit results

Figure 5 shows the load-deflection response of the slab for the three different mesh densities using a tension stiffening value of 2×10^{-3}. Since the coarse mesh predicts a slightly higher limit load than the medium and fine meshes do and the limit loads for the medium and fine mesh analyses are very close, the tension stiffening study is performed using the medium mesh only.

The numerical (12×12 mesh) results are compared to the experimental results in *Figure 6* for the three different values of tension stiffening. It is clear that the less tension stiffening used, the softer the load-deflection response is. A value of tension stiffening somewhere between the highest and middle values appears to match the experimental results best. The lowest tension stiffening value causes more sudden cracking in the concrete and, as a result, the response tends to be more dynamic than that obtained with the higher tension stiffening values.

Figure 7 shows the numerically predicted crack pattern on the lower surface of the slab for the medium mesh.

Input files

Abaqus/Standard input files

collapseconcslab_s8r.inp	S8R elements.
collapseconcslab_s9r5.inp	S9R5 elements.
collapseconcslab_postoutput.inp	**POST OUTPUT* analysis.

Abaqus/Explicit input files

mcneice_1.inp	Coarse (6 × 6) mesh; tension stiffening = 2×10^{-3}.
mcneice_2.inp	Medium (12 × 12) mesh; tension stiffening = 2×10^{-3}.
mcneice_3.inp	Fine (24 × 24) mesh; tension stiffening = 2×10^{-3}.
mcneice_4.inp	Medium (12 × 12) mesh; tension stiffening = 1×10^{-3}.
mcneice_5.inp	Medium (12 × 12) mesh; tension stiffening = 5×10^{-4}.
mcneice_6.inp	Medium (12 × 12) mesh; tension stiffening = 2×10^{-3}; no mass scaling.

References

- Crisfield, M. A., "Variable Step-Length for Nonlinear Structural Analysis," *Report 1049, Transport and Road Research Lab., Crowthorne, England*, 1982.
- Gilbert, R. I., and R. F. Warner, "Tension Stiffening in Reinforced Concrete Slabs," *Journal of the Structural Division, American Society of Civil Engineers*, vol. 104, ST12, pp. 1885–1900, 1978.
- Hand, F. D., D. A. Pecknold, and W. C. Schnobrich, "Nonlinear Analysis of Reinforced Concrete Plates and Shells," *Journal of the Structural Division, American Society of Civil Engineers*, vol. 99, ST7, pp. 1491–1505, 1973.
- Hinton, E., H. H. Abdel Rahman, and O. C. Zienkiewicz, "Computational Strategies for Reinforced Concrete Slab Systems," *International Association of Bridge and Structural Engineering Colloquium on Advanced Mechanics of Reinforced Concrete*, pp. 303–313, 1981.
- Lin, C. S., and A. C. Scordelis, "Nonlinear Analysis of Reinforced Concrete Shells of General Form," *Journal of the Structural Division, American Society of Civil Engineers*, vol. 101, pp. 523–238, 1975.
- McNeice, A. M., "Elastic-Plastic Bending of Plates and Slabs by the Finite Element Method," *Ph. D. Thesis, London University*, 1967.

Tables

Table 1: Material properties for the McNeice slab.

Concrete properties:

Properties are taken from Gilbert and Warner (1978) if available in that paper.

Properties marked with a * are not available and are assumed values.

Young's modulus	28.6 GPa (4.15×10^6 lb/in^2)
Poisson's ratio	0.15

Uniaxial compression values:

Yield stress	20.68 MPa (3000 lb/in^2)*
Failure stress	37.92 MPa (5500 lb/in^2)
Plastic strain at failure	1.5×10^{-3}*
Ratio of uniaxial tension to compression failure stress	8.36×10^{-2}
Ratio of biaxial to uniaxial compression failure stress	1.16*
Cracking failure stress	459.8 lb/in^2 (3.17 MPa)
Density (before mass scaling)	2.246×10^{-4} lb s^2/in^4 (2400 kg/m^3)

"Tension stiffening" is assumed as a linear decrease of the stress to zero stress, at a strain of 5×10^{-4}, at a strain of 10×10^{-4}, or at a strain of 20×10^{-4}.

Steel (rebar) properties:

Young's modulus	200 GPa (29×10^6 lb/in^2)
Yield stress	345 MPa (50×10^3 lb/in^2)
Density (before mass scaling)	7.3×10^{-4} lb s^2/in^4 (7800 kg/m^3)

Figures

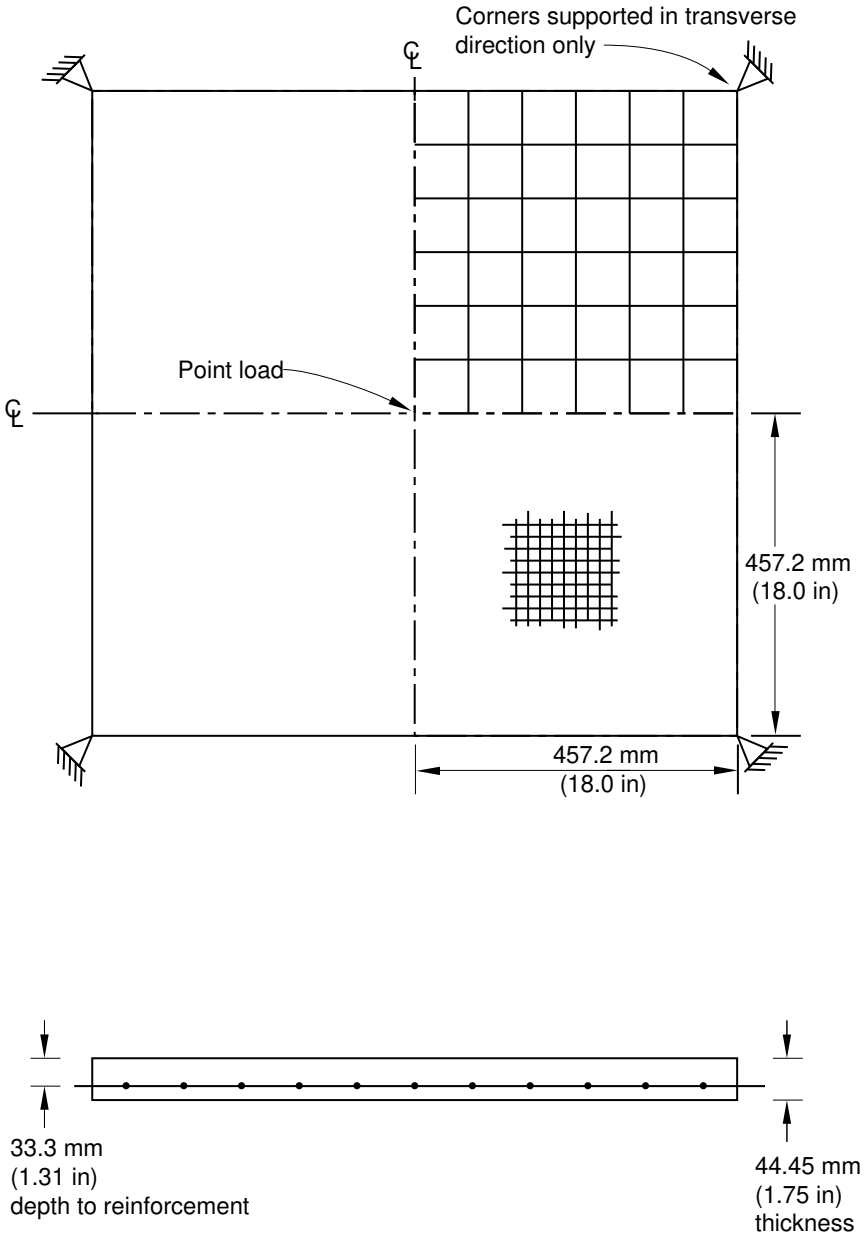


Figure 1: McNeice slab.

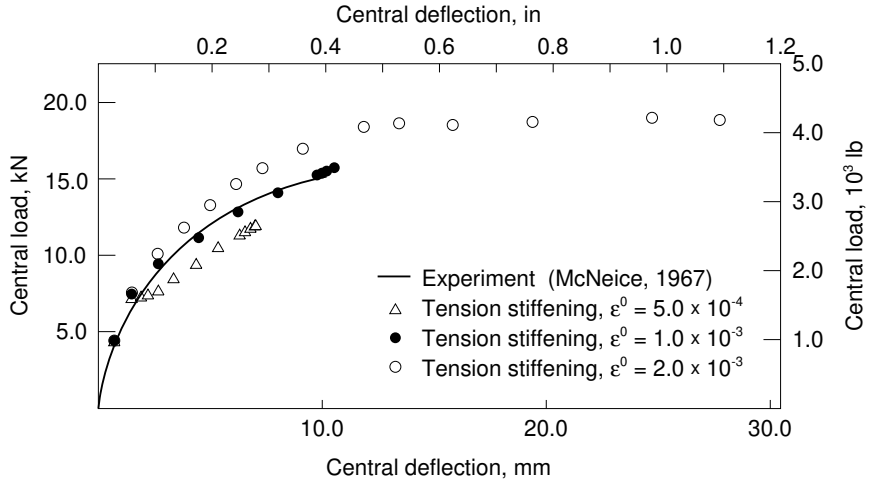


Figure 2: Load-deflection response of McNeice slab, Abaqus/Standard.

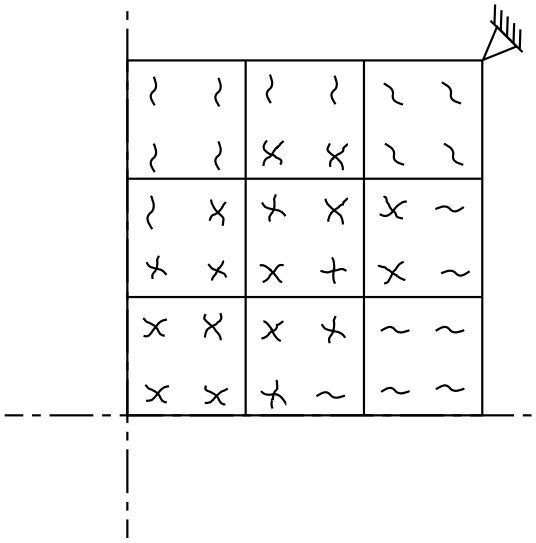


Figure 3: Crack pattern on lower surface of slab, Abaqus/Standard.

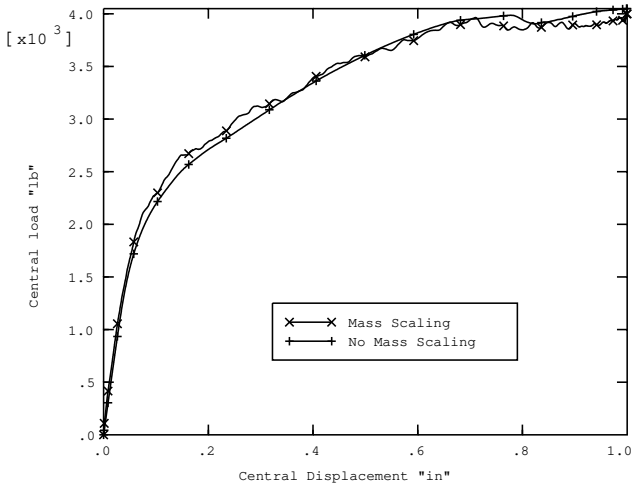


Figure 4: Load-deflection response of McNeice slab, Abaqus/Explicit; influence of mass scaling.

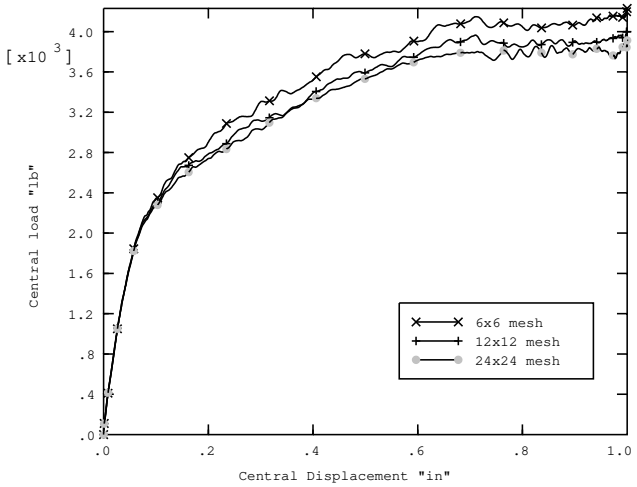


Figure 5: Load-deflection response of McNeice slab, Abaqus/Explicit; influence of mesh refinement.

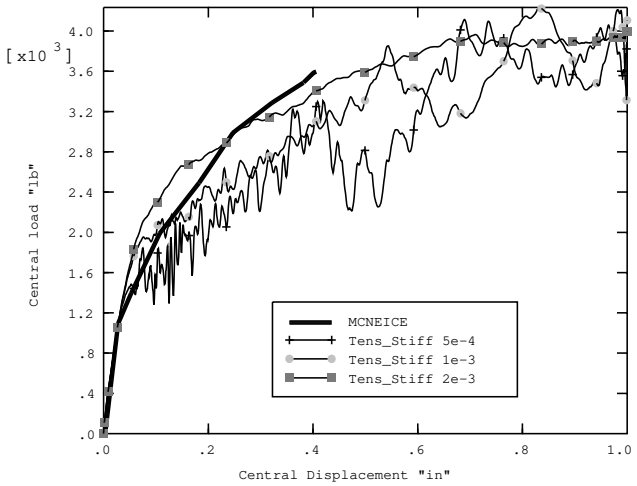


Figure 6: Load-deflection response of McNeice slab, Abaqus/Explicit; influence of tension stiffening.

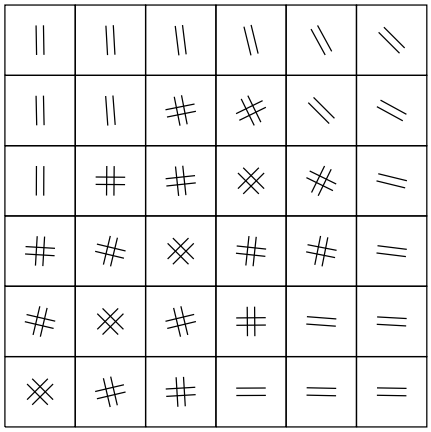


Figure 7: Crack pattern on lower surface of slab, Abaqus/Explicit.

Jointed rock slope stability

Products: Abaqus/Standard

Overview

This example illustrates the use of the jointed material model in the context of geotechnical applications.
In this example we examine the stability of the excavation of part of a jointed rock mass, leaving a sloped embankment. This problem is chosen mainly as a verification case because it has been studied previously by Barton (1971) and Hoek (1970), who used limit equilibrium methods, and by Zienkiewicz and Pande (1977), who used a finite element model. This example also has been extended to study the slope stability of excavated soil medium with the same geometry, by using the Mohr-Coulomb plasticity model with and without the tension cutoff feature.

Geometry and model

The plane strain model analyzed is shown in *Figure 1* together with the excavation geometry and material properties. The rock mass contains two sets of planes of weakness: one vertical set of joints and one set of inclined joints. We begin from a nonzero state of stress. In this problem this consists of a vertical stress that increases linearly with depth to equilibrate the weight of the rock and horizontal stresses caused by tectonic effects: such stress is quite commonly encountered in geotechnical engineering. The active “loading” consists of removal of material to represent the excavation. It is clear that, with a different initial stress state, the response of the system would be different. This illustrates the need of nonlinear analysis in geotechnical applications—the response of a system to external “loading” depends on the state of the system when that loading sequence begins (and, by extension, to the sequence of loading). We can no longer think of superposing load cases, as is done in a linear analysis.

Practical geotechnical excavations involve a sequence of steps, in each of which some part of the material mass is removed. Liners or retaining walls can be inserted during this process. Thus, geotechnical problems require generality in creating and using a finite element model: the model itself, and not just its response, changes with time—parts of the original model disappear, while other components that were not originally present are added. This example is somewhat academic, in that we do not encounter this level of complexity. Instead, following the previous authors' use of the example, we assume that the entire excavation occurs simultaneously.

Solution controls

The jointed material model includes a joint opening/closing capability. When a joint opens, the material is assumed to have no elastic stiffness with respect to direct strain across the joint system. Because of this, and also as a result of the fact that different combinations of joints may be yielding at any one time, the overall convergence of the solution is expected to be nonmonotonic. In such cases setting the time incrementation parameters automatically is generally recommended to prevent premature termination of the equilibrium iteration process because the solution may appear to be diverging.

As the end of the excavation process is approached, the automatic incrementation algorithm reduces the load increment significantly, indicating the onset of failure of the slope. In such analyses it is useful to specify a minimum time step to avoid unproductive iteration.

For the nonassociated flow case the unsymmetric equation solver should be used. This is essential for obtaining an acceptable rate of convergence since nonassociated flow plasticity has a nonsymmetric stiffness matrix.

Results and discussion

In this problem we examine the effect of joint cohesion on slope collapse through a sequence of solutions with different values of joint cohesion, with all other parameters kept fixed. *Figure 2* shows the variation of horizontal displacements as cohesion is reduced at the crest of the slope (point *A* in *Figure 1*) and at a point one-third of the way up the slope (point *B* in *Figure 1*). This plot suggests that the slope collapses if the cohesion is less than 24 kPa for the case of associated flow or less than 26 kPa for the case of nonassociated flow. These compare well with the value calculated by Barton (26 kPa) using a planar failure assumption in his limit equilibrium calculations. Barton's calculations also include "tension cracking" (akin to joint opening with no tension strength) as we do in our calculation. Hoek calculates a cohesion value of 24 kPa for collapse of the slope. Although he also makes the planar failure assumption, he does not include tension cracking. This is, presumably, the reason why his calculated value is lower than Barton's. Zienkiewicz and Pande assume the joints have a tension strength of one-tenth of the cohesion and calculate the cohesion value necessary for collapse as 23 kPa for associated flow and 25 kPa for nondilatant flow.

Figure 3 shows the deformed configuration after excavation for the nonassociated flow case and clearly illustrates the manner in which the collapse is expected to occur. *Figure 4* shows the magnitude of the frictional slip on each joint system for the nonassociated flow case. A few joints open near the crest of the slope.

The study of soil slope stability using the Mohr-Coulomb plasticity model is performed for two cases: one without tension cutoff and one including the tension cutoff feature. The tension cutoff feature limits the stress carrying capacity of soil in tension. It can be seen that the maximum principal stress without tension cutoff (see contour plot in *Figure 5*) is higher than the limiting maximum principal stress (see *Figure 6*) with tension cutoff as expected. With tension cutoff, one also observes the appearance of the equivalent plastic strain in tension, PEEQT in the region of maximum principal stress. In this case it is also seen that the equivalent plastic strain, PEEQ on the cohesion failure surface is higher compared to the case without tension cutoff. The contour plots for the equivalent plastic strains are not shown.

Input files

jointrockstabil_nonassoc_30pka.inp	Nonassociated flow case problem; cohesion = 30 kPa.
jointrockstabil_assoc_25kpa.inp	Associated flow case; cohesion = 25 kPa.
mc_slopestabil.inp	Slope stability analysis, Mohr-Coulomb plasticity without tension cutoff
mctc_slopestabil.inp	Slope stability analysis, Mohr-Coulomb plasticity with tension cutoff

References

- Barton, N., "Progressive Failure of Excavated Rock Slopes," *Stability of Rock Slopes*, Proceedings of the 13th Symposium on Rock Mechanics, Illinois, pp. 139–170, 1971.
- Hoek, E., "Estimating the Stability of Excavated Slopes in Open Cast Mines," *Trans. Inst. Min. and Metal.*, vol. 79, pp. 109–132, 1970.
- Zienkiewicz, O. C., and G. N. Pande, "Time-Dependent Multilaminate Model of Rocks – A Numerical Study of Deformation and Failure of Rock Masses," *International Journal for Numerical and Analytical Methods in Geomechanics*, vol. 1, pp. 219–247, 1977.

Figures

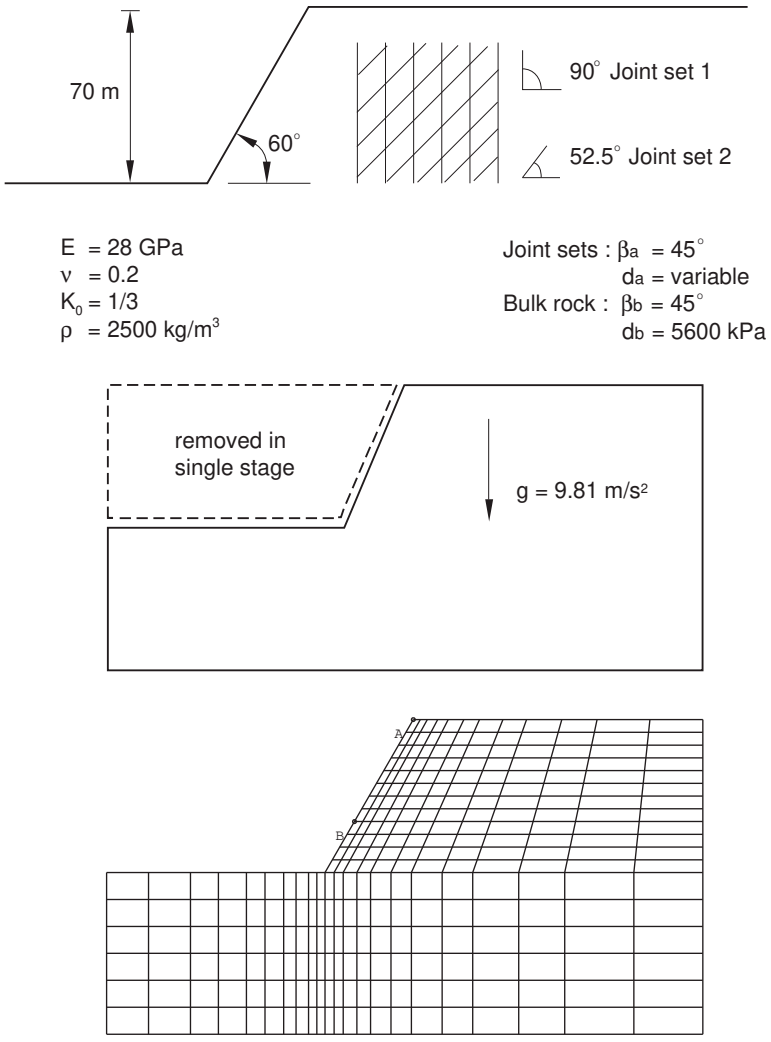


Figure 1: Jointed rock slope problem.

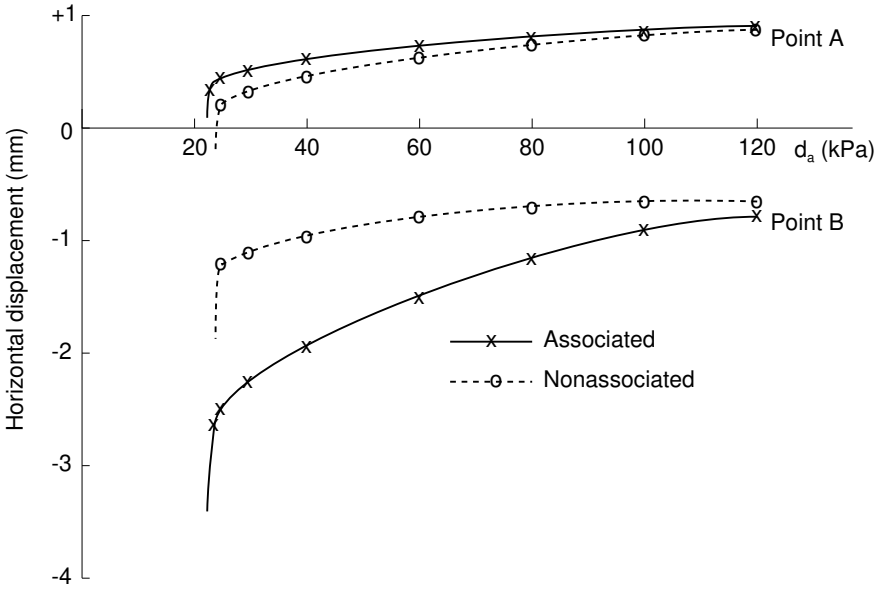


Figure 2: Horizontal displacements with varying cohesion.

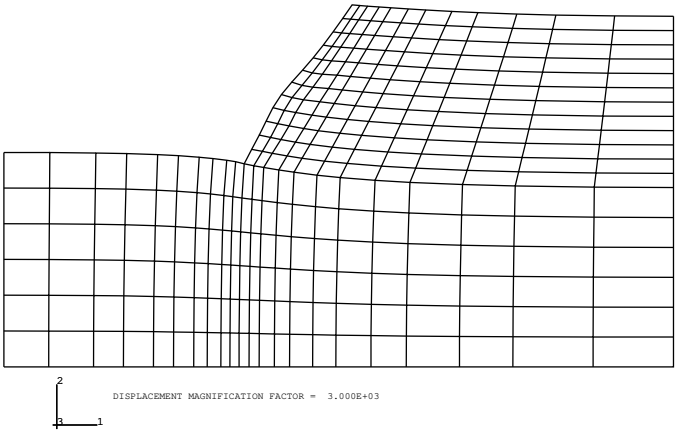
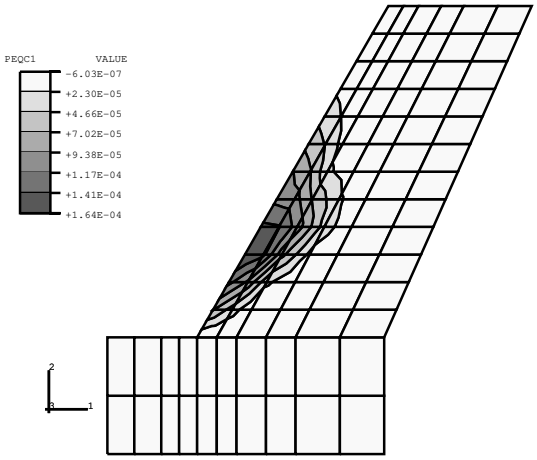
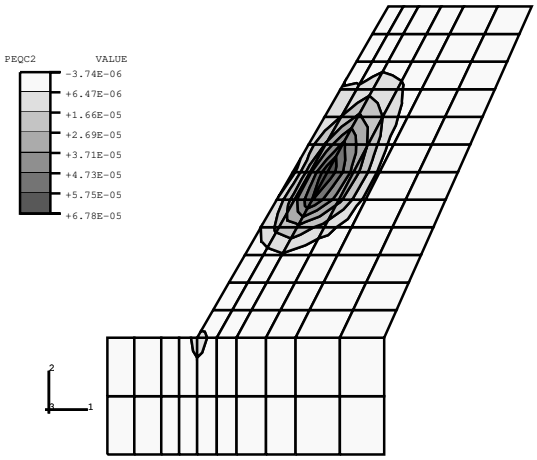


Figure 3: Deformed configuration (nonassociated flow).



Joint set 1 (vertical joints).



Joint set 2 (inclined joints).

Figure 4: Contours of frictional slip magnitudes (nonassociated flow).

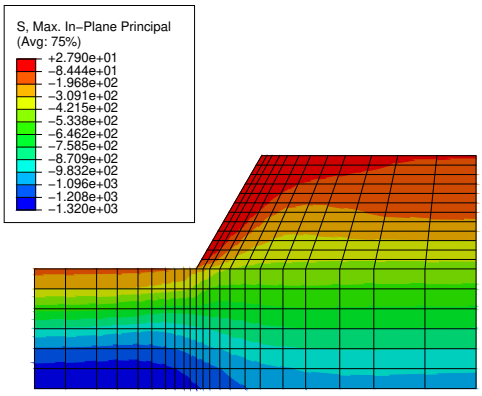


Figure 5: Maximum principal stress without tension cutoff.

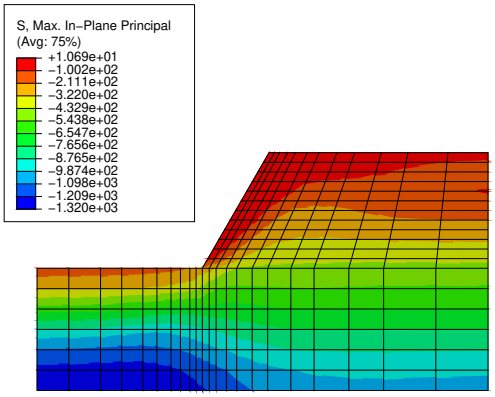


Figure 6: Maximum principal stress with tension cutoff.

Notched beam under cyclic loading

Products: Abaqus/Standard

References:

- *Models for Metals Subjected to Cyclic Loading*
- *Plasticity Corrections*

Overview

This example illustrates the use of the nonlinear isotropic/kinematic hardening material model to simulate the response of a notched beam under cyclic loading.

The model has two features to simulate plastic hardening in cyclic loading conditions: the center of the yield surface moves in stress space (kinematic hardening behavior), and the size of the yield surface evolves with inelastic deformation (isotropic hardening behavior). This combination of kinematic and isotropic hardening components is introduced to model the Bauschinger effect and other phenomena such as plastic shakedown, ratcheting, and relaxation of the mean stress.

The component investigated in this example is a notched beam subjected to a cyclic 4-point bending load. The results from the cyclic analysis are compared with the finite element results published by Benallal et al. (1988) and Doghri (1993).

This example also uses this component to demonstrate the plasticity corrections capabilities for estimating the plastic solution near a notch region based on the solution of a linear elastic analysis. In this case, the notched beam is subject to monotonic loading conditions. The results of the plasticity corrections are compared to those from a full nonlinear elastic-plastic analysis.

Geometry and model

The geometry and mesh are shown in *Figure 1*. *Figure 2* shows the discretization in the vicinity of the notch, which is the region of interest in this analysis. Only one-half of the beam is modeled since the geometry and loading are symmetric with respect to the x= 0 plane. All dimensions are given in millimeters. The beam is 1 mm thick and is modeled with plane strain, second-order, reduced-integration elements (type CPE8R). The mesh is chosen to be similar to the mesh used by Doghri (1993). No mesh convergence studies have been performed.

Material

The material properties reported by Doghri (1993) for a low-carbon (AISI 1010), rolled steel are used in this example.

A Young's modulus of E= 210 GPa and a Poisson's ratio of ν = 0.3 define the elastic response of the material.

The initial yield stress is $\sigma|_0$ = 200 MPa.

The nonlinear evolution of the center of the yield surface is defined by the equation

$$\dot{\boldsymbol{\alpha}} = C\frac{1}{\sigma^0}(\boldsymbol{\sigma} - \boldsymbol{\alpha})\dot{\bar{\varepsilon}}^{pl} - \gamma\boldsymbol{\alpha}\dot{\bar{\varepsilon}}^{pl},$$

where $\boldsymbol{\alpha}$ is the backstress, σ^0 is the size of the yield surface (size of the elastic range), $\bar{\varepsilon}^{pl}$ is the equivalent plastic strain, and C = 25.5 GPa and γ = 81 are the material parameters that define the initial hardening modulus

and the rate at which the hardening modulus decreases with increasing plastic strain, respectively. The quantity $\sqrt{2/3}\,C/\gamma =$ 257 MPa defines the limiting value of the equivalent backstress $\overline{\alpha} = \sqrt{\boldsymbol{\alpha}^{dev} : \boldsymbol{\alpha}^{dev}}$; further hardening is possible only through the change in the size of the yield surface (isotropic hardening).

The isotropic hardening behavior of this material is modeled with the exponential law

$$\sigma^0 = \sigma|_0 + Q_\infty \left(1 - e^{-b\,\bar{\varepsilon}^{pl}}\right),$$

where σ^0 is the size of the yield surface (size of the elastic range), Q_∞ = 2000 MPa is the maximum increase in the elastic range, and b = 0.26 defines the rate at which the maximum size is reached as plastic straining develops.

The material used for this simulation is cold rolled. This work hardened state is represented by specifying an initial equivalent plastic strain $\bar{\varepsilon}^{pl}\big|_0$ = 0.43 (so that σ^0 = 411 MPa) and an initial backstress tensor

$$\boldsymbol{\alpha}|_0 = \begin{bmatrix} 128 & 0 & 0 \\ 0 & -181 & 0 \\ 0 & 0 & 53 \end{bmatrix} \text{MPa.}$$

Loading and boundary conditions

The beam is subjected to a 4-point bending load. Since only half of the beam is modeled, the model contains one concentrated load at a distance of 26 mm from the symmetry plane (see *Figure 1*). The pivot point is 42 mm from the symmetry plane. The simulation runs 3 1/2 cycles over 7 time units. In each cycle the load is ramped from zero to 675 N and back to zero. An amplitude curve is used to describe the loading and unloading. The increment size is restricted to a maximum of 0.125 to force Abaqus to follow the prescribed loading/unloading pattern closely.

Results and discussion

Figure 3 shows the final deformed shape of the beam after the 3 1/2 cycles of load; the final load on the beam is 675 N.

The deformation is most severe near the root of the notch. The results reported in *Figure 4* and *Figure 5* are measured in this area (element 166, integration point 3). *Figure 4* shows the time evolution of stress versus strain. Several important effects are predicted using this material model. First, the onset of yield occurs at a lower absolute stress level during the first unloading than during the first loading, which is the Bauschinger effect. Second, the stress-strain cycles tend to shift and stabilize so that the mean stress decreases from cycle to cycle, tending toward zero. This behavior is referred to as the relaxation of the mean stress and is most pronounced in uniaxial cyclic tests in which the strain is prescribed between unsymmetric strain values. Third, the yield surface shifts along the strain axis with cycling, whereas the shape of the stress-strain curve tends to remain similar from one cycle to the next. This behavior is known as ratcheting and is most pronounced in uniaxial cyclic tests in which the stress is prescribed between unsymmetric stress values. Finally, the hardening behavior during the first half-cycle is very flat relative to the hardening curves of the other cycles, which is typical of work hardened metals whose initial hardened state is a result of a large monotonic plastic deformation caused by a forming process such as rolling. The low hardening modulus is the result of the initial conditions on backstress, which places the center of the yield surface at a distance of $\sqrt{\boldsymbol{\alpha}|_0^{dev} : \boldsymbol{\alpha}|_0^{dev}} =$ 228 MPa away from the origin of stress space. Since this distance is close to the maximum possible distance (257 MPa), most of the hardening during the first cycle is isotropic.

These phenomena are modeled in this example primarily by the nonlinear evolution of the backstress, since the rate of isotropic hardening is very small. This behavior can be verified by conducting an analysis in which the elastic domain remains fixed throughout the analysis.

Figure 5 shows the evolution of the direct components of the deviatoric part of the backstress tensor. The backstress components evolve most during the first cycle as the Bauschinger effect overcomes the initial hardening configuration. Only the deviatoric components of the backstress are shown so that the results obtained using Abaqus can be compared to those reported by Doghri (1993). Since Abaqus uses an extension of the Ziegler evolution law, a backstress tensor with nonzero pressure is produced, whereas the backstress tensor produced with the law used by Doghri (which is an extension of the linear Prager law) is deviatoric. Since the plasticity model considers only the deviatoric part of the backstress, this difference in law does not affect the other solution variables.

The results shown in *Figure 4* and *Figure 5* agree well with the results reported by Doghri (1993).

Plasticity corrections

This example demonstrates the Neuber and Glinka plasticity corrections for estimating the plastic solution near a notch region from the stress results of a linear elastic analysis. The plasticity correction capabilities require the definition of the plastic response of the material. This example uses an isotropic Mises plasticity definition that matches the uniaxial response of the original combined-hardening material. *Figure 6* shows the isotropic hardening definition obtained this way.

The notched beam is subject to a load that increases monotonically from 0 N to 300 N. The results of the plasticity corrections are compared to those from a full nonlinear elastic-plastic analysis.

Figure 7 and *Figure 8* show the distributions of stresses and equivalent plastic strains at the end of the analysis. In *Figure 7* the stresses obtained by the nonlinear elastic-plastic analysis are compared with the Neuber and Glinka stresses evaluated by performing purely elastic analysis, and *Figure 8* shows a similar comparison for equivalent plastic strains. The Neuber and Glinka stresses and plastic strains show good agreement with the values obtained by the elastic-plastic analysis as is expected for such localized plastic deformations. Neuber's rule predictions of stresses and strains are higher (more conservative) than Glinka's results as is usually the case.

Input files

cyclicnotchedbeam.inp	Input data for cyclic analysis.
notchedbeam.inp	Input data for elastic-plastic analysis.
notchedbeam_plasticity_correction.inp	Input data for plasticity correction analysis.
cyclicnotchedbeam_mesh.inp	Element and node data.

References

- Benallal, A., R. Billardon, and I. Doghri, "An Integration Algorithm and the Corresponding Consistent Tangent Operator for Fully Coupled Elastoplastic and Damage Equations," *Communications in Applied Numerical Methods*, vol. 4, pp. 731–740, 1988.
- Doghri, I., "Fully Implicit Integration and Consistent Tangent Modulus in Elasto-Plasticity," *International Journal for Numerical Methods in Engineering*, vol. 36, pp. 3915–3932, 1993.

Figures

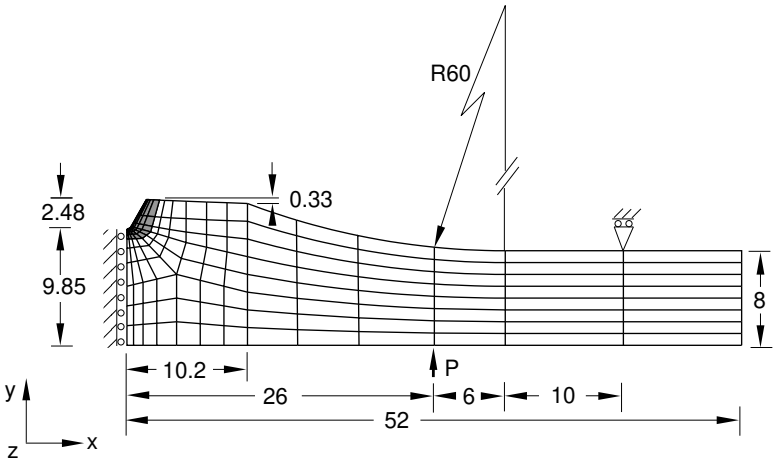


Figure 1: Undeformed mesh (dimensions in mm).

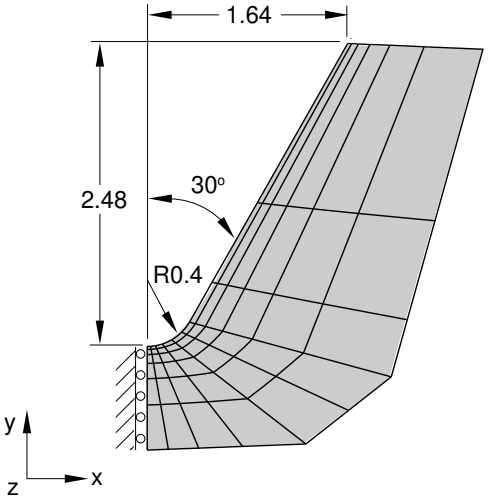


Figure 2: Magnified view of the root of the notch.

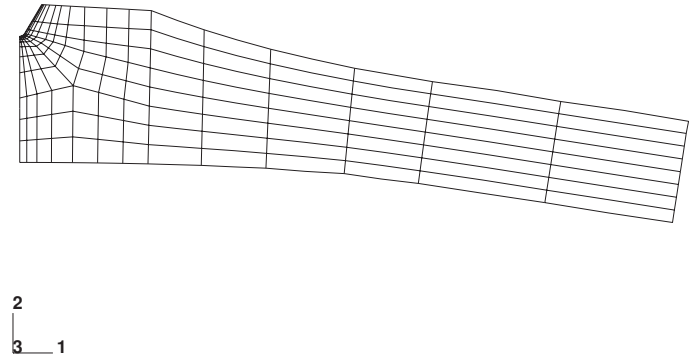


Figure 3: Deformed mesh at the conclusion of the simulation. Displacement magnification factor is 3.

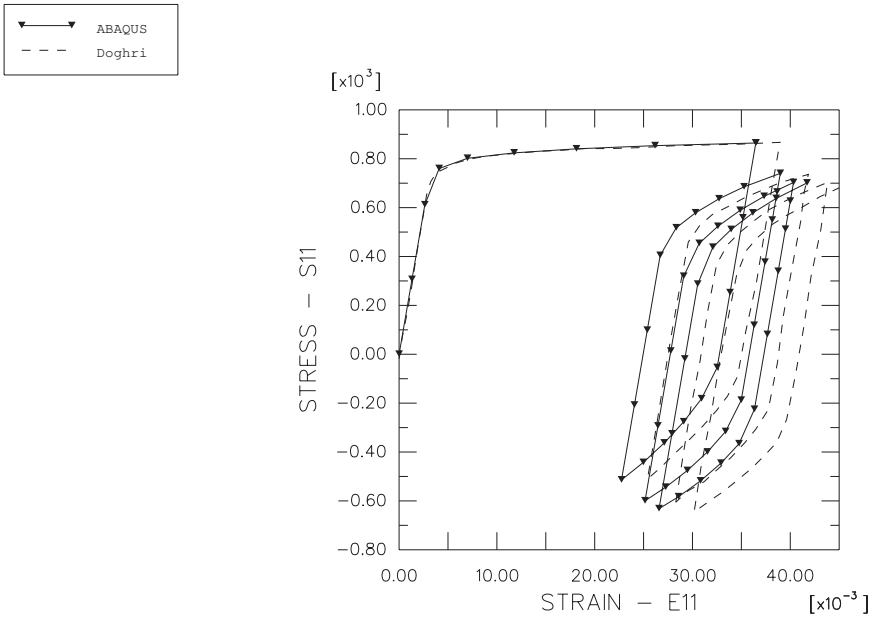


Figure 4: Evolution of stress versus strain in the vicinity of the root of the notch.

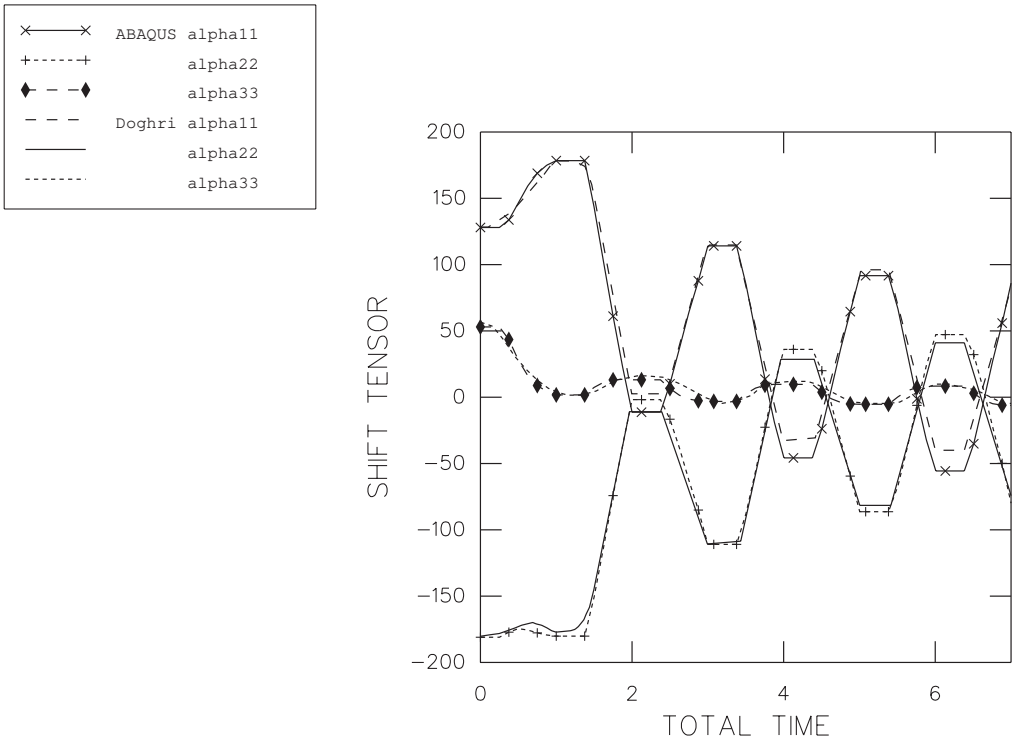


Figure 5: Evolution of the diagonal components of the deviatoric part of the backstress tensor.

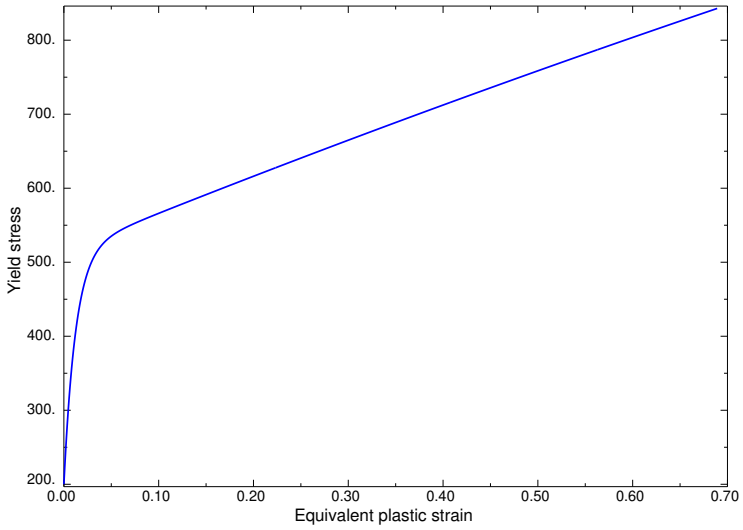


Figure 6: Mises yield surface.

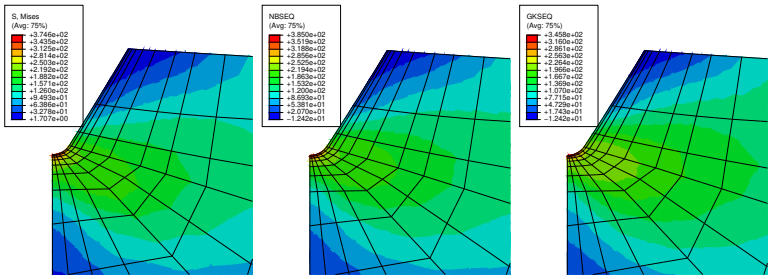


Figure 7: Stress distribution computed for the elastic-plastic material and Neuber and Glinka rules.

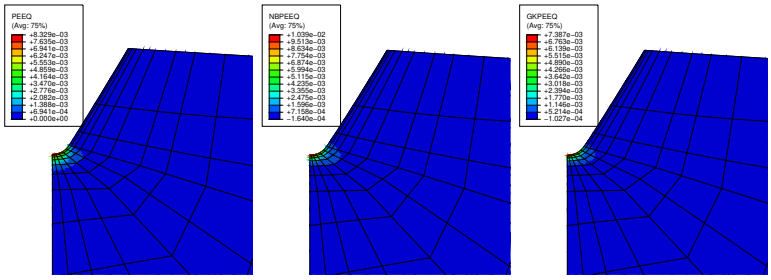


Figure 8: Equivalent plastic strain distribution computed for the elastic-plastic material and Neuber and Glinka rules.

Uniaxial ratcheting under tension and compression

Products: Abaqus/Standard

References:

- *Models for Metals Subjected to Cyclic Loading*
- **CYCLIC HARDENING*
- **PLASTIC*

Overview

This example shows that the combined isotropic/kinematic hardening model available in Abaqus can predict ratcheting and that the results obtained using this model correlate very well with experimental results.

The following Abaqus features are demonstrated:

- using the nonlinear isotropic/kinematic hardening model to predict deformation in a specimen subjected to monotonic and cyclic loading; and
- modeling the effect of ratcheting (accumulation of plastic strain under a cyclic load).

Application description

Preventing ratcheting is very important in the design of components subject to cyclic loading in the inelastic domain. The amount of plastic strain can accumulate continuously with an increasing number of cycles and may eventually cause material failure. Therefore, many cyclic plastic models have been developed with the goal of modeling ratcheting correctly.

This example considers two loading conditions: monotonic deformation and uniaxial cyclic tension and compression.

Geometry

The specimen studied is shown in *Figure 1*. All dimensions are specified in the figure. For the experiments (Portier et al., 2000) the specimens were obtained from a tube with an outer diameter of 130 mm and a wall thickness of 28 mm. The specimens were heat treated to ensure the initial isotropy of the material.

Materials

The specimen is made of austenitic type 316 stainless steel. The material mechanical properties are listed in *Table 1*. A detailed description of the calibration of parameters is given in *Material parameters determination* below.

Boundary conditions and loading

The specimen is constrained at the bottom surface in the longitudinal direction, and a load is applied to the top surface.

Abaqus modeling approaches and simulation techniques

In this example deformations of a specimen subject to monotonic and cyclic loads are studied. In both cases static analyses are performed. Taking advantage of the axial symmetry of the specimen, axisymmetric elements are used.

Summary of analysis cases

Case 1	Static analysis of a specimen subject to a monotonic load.
Case 2	Static analysis of a specimen subject to an unsymmetric cyclic load.

Case 1 Monotonic load

The experimental monotonic load data are used to calibrate the kinematic hardening model. The purpose of this case is to verify that the simulation results agree with the experimental results and to compare the accuracy of the results obtained using a model with one backstress and a model with two backstresses.

Analysis Types

A static stress analysis is performed.

Mesh design

The specimen is meshed with CAX4R and CAX3 elements. The mesh is shown in *Figure 2*.

Materials

The combined isotropic/kinematic hardening model is used to model the response of the material. This material model requires that the elastic parameters (Young's modulus and Poisson's ratio), the initial yield stress, the isotropic hardening parameters, and the kinematic hardening parameters are specified.

Material parameters determination

The elastic parameters, the initial yield stress, and the isotropic hardening parameters are assumed to be equal to those reported in Portier et al. (2000) for the Ohno and Wang model. The kinematic hardening component is defined by specifying half-cycle test data, where the data are obtained by digitizing the results reported by Portier et al. The values of all the parameters, including the kinematic hardening parameters obtained from the test data, are presented in *Table 1*.

Boundary conditions

The specimen is fixed in the longitudinal direction at the bottom surface.

Loads

A displacement of 0.45 mm is applied to the top surface.

Results and discussion

The simulation and experimental results are presented graphically in *Figure 3*. The strains and stresses are computed by averaging the values in the elements lying at the center of the specimen. The experimental curve shows three distinct regions: a linear elastic region, an elastic-plastic transition zone, and an almost linear response region at large strain values. The model with two backstresses captures this response very well. One of the

backstresses has a large value of the parameter γ, which captures the shape of the transition zone correctly, while the second backstress with a relatively small value of γ captures the nearly linear response at large strains correctly. The parameter γ in the model with one backstress has a relatively large value, which results in large discrepancies between the experimental and predicted responses at large strains.

Case 2 Uniaxial tension and compression cyclic analysis

The objective of this case is to show that the combined isotropic/kinematic hardening model can be used to predict the response of a material subject to a cyclic load accurately and, in particular, to predict the ratcheting effect. In addition, the results obtained using a model with one backstress are compared to those obtained using a model with two backstresses.

Analysis Types

A static stress analysis is performed.

Mesh design

The mesh is the same as in Case 1.

Materials

The material model is the same as in Case 1.

Boundary conditions

The specimen is fixed in the longitudinal direction at the bottom surface.

Loads

A cyclic load of $\sigma_{mean} = 11.0$ MPa and $\Delta\sigma = 15.4$ MPa is applied to the top surface of the specimen. This load produces an approximate cyclic load of $\sigma_{mean} = 100$ MPa and $\Delta\sigma = 140$ MPa at the center part of the specimen.

Results and discussion

The simulation results obtained for the model with one backstress and the model with two backstresses, together with the experimental results, are depicted in *Figure 4*. The strains were computed by averaging the strains in the elements lying at the center of the specimen. The figure shows that both simulation models are capable of predicting ratcheting. It also shows that the results obtained using the model with two backstresses correlate better with the experimental results.

Discussion of results and comparison of cases

The results of the analyses show that the combined isotropic/kinematic hardening model can be used to predict the ratcheting effect accurately. In addition, a substantial improvement in the agreement between simulation and experimental results can be achieved by using a model with multiple backstresses instead of a model with a single backstress. The former model predicts more accurately the shape of the stress-strain curve in the monotonic loading case and the ratcheting strain in a cyclic loading case. In this example increasing the number of backstresses from one to two produced a substantial improvement in the results. However, further increasing the number of backstresses does not significantly improve the results.

Files

Case 1 Monotonic

ratch_axi_monotonic_1.inp	Input file to analyze a specimen subjected to a monotonic load using the model with one backstress.
ratch_axi_monotonic_2.inp	Input file to analyze a specimen subjected to a monotonic load using the model with two backstresses.

Case 2 Cyclic

ratch_axi_unsymcyclic_1.inp	Input file to analyze a specimen subjected to a cyclic load using the model with one backstress.
ratch_axi_unsymcyclic_2.inp	Input file to analyze a specimen subjected to a cyclic load using the model with two backstresses.

References

- Portier, L., S. Calloch, D. Marquis, and P. Geyer, “Ratcheting Under Tension-Torsion Loadings: Experiments and Modelling,” International Journal of Plasticity, vol. 16, pp. 303–335, 2000.

Tables

Table 1: Mechanical properties for 316 steel.

Material properties:		
	Young's modulus	192.0 GPa
	Poisson's ratio	0.3
	Initial yield stress	120.0 MPa
Isotropic hardening parameters:		
	Q	120.0 MPa
	b	13.2
Kinematic hardening parameters:		
	Model with one backstress	
	C	218.5 GPa
		1956.6
	Model with two backstresses	
	C_1	2.067 GPa
	$_1$	44.7
	C_2	246.2 GPa
	$_2$	2551.4

Figures

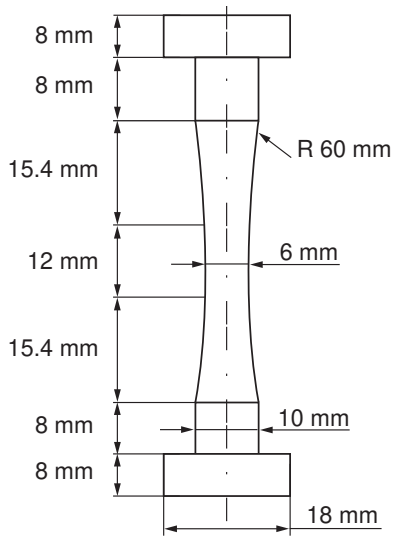


Figure 1: Geometry and size of the specimen.

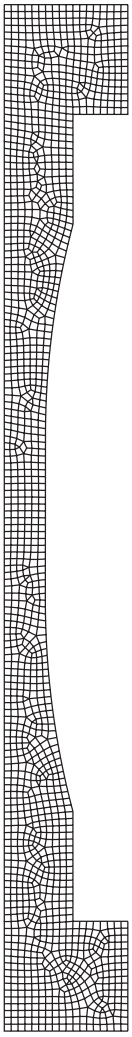


Figure 2: Finite element mesh of the specimen.

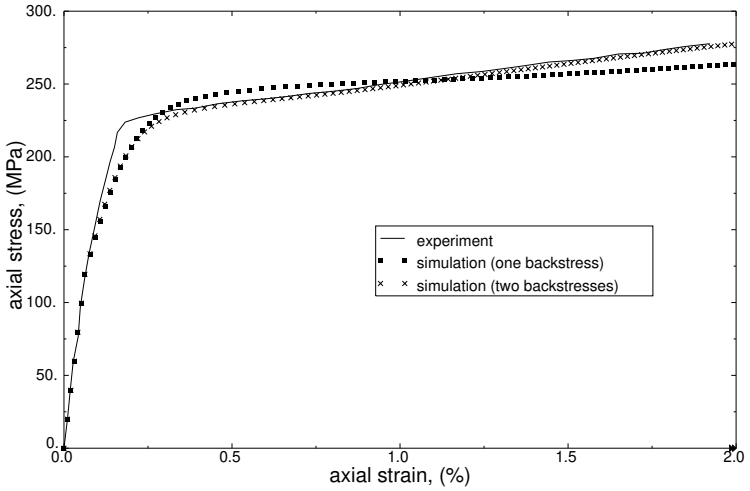


Figure 3: Stress-strain curves for monotonic tensile loading.

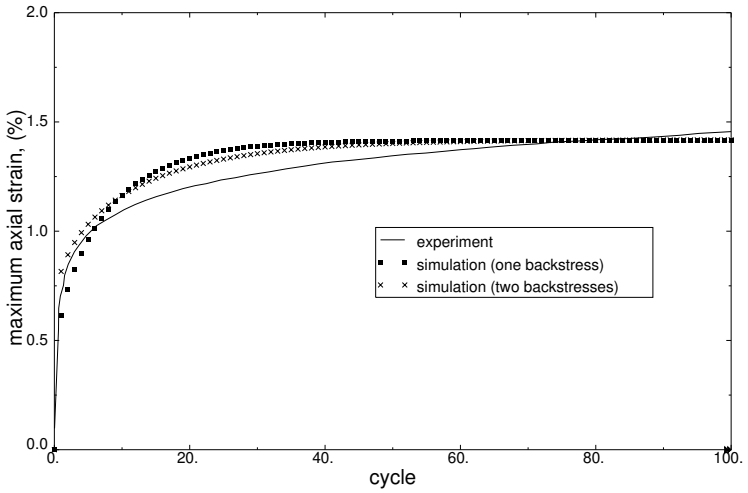


Figure 4: Maximum axial strain versus number of cycles.

Hydrostatic fluid elements: modeling an airspring

Products: Abaqus/Standard Abaqus/Explicit

Overview

This section shows two examples of the analysis of a cord-reinforced rubber airspring.
Static analyses are performed in Abaqus/Standard, and quasi-static analyses are performed in Abaqus/Explicit.

Airsprings are rubber or fabric actuators that support and contain a column of compressed air. They are used as pneumatic actuators and vibration isolators. Unlike conventional pneumatic cylinders, airsprings have no pistons, rods, or dynamic seals. This makes them better suited to handle off-center loading and shock. In addition, airsprings are considerably more flexible than other types of isolators: the airspring's inflation pressure can be changed to compensate for different loads or heights without compromising isolation efficiency. Dils (1992) provides a brief discussion of various practical uses of airsprings.

The first example is a three-dimensional, half-symmetry model that uses finite-strain shell elements to model the rubber spring and rebar to model the multi-ply steel reinforcements in the rubber membrane. In addition, a three-dimensional, element-based rigid surface is used to define the contact between the airspring and the lateral metal bead. The cord-reinforced rubber membrane is modeled using a hyperelastic material model with steel rebar.

The second example is a two-dimensional, axisymmetric version of the first model that uses composite axisymmetric, finite-strain shell elements to model the cord-reinforced rubber spring and an axisymmetric, element-based rigid surface in the contact definition. This model uses a composite shell section consisting of a thin orthotropic elastic layer sandwiched between two hyperelastic layers. The orthotropic layer captures the mechanical properties of the rebar definition used in the three-dimensional model.

The orthotropic material constants have been obtained by performing simple tests on a typical element of the three-dimensional model. The three-dimensional shell model uses rebar with material properties that are initially identical to the properties of the composite shell section in the axisymmetric shell model.

For comparison, Abaqus/Standard input files that use finite-strain membrane elements instead of finite-strain shell elements to model the cord-reinforced rubber spring are also included for both the axisymmetric and three-dimensional models.

In all analyses the airspring cavity is modeled using the surface-based fluid cavity capability (see *About Surface-Based Fluid Cavities*) and air inside the cavity is modeled as a compressible or “pneumatic” fluid satisfying the ideal gas law.

Geometry and model

The dimensions of the airspring have been inferred from the paper by Fursdon (1990). This airspring, shown in *Figure 1*, is fairly large and is used in secondary suspension systems on railway bogies. However, the shape of the airspring is typical of airsprings used in other applications. The airspring's cross-section is shown in *Figure 2*. The airspring is toroidal in shape, with an inner radius of 200 mm and an outer radius of 400 mm. The airspring has been idealized in the model as consisting of two circular, metal disks connected to each other via a rubber component. The lower disk has a radius of 200 mm, and the upper disk has a radius of 362.11 mm. The disks are initially coaxial and are 100 mm apart. The rubber component is doubly curved and toroidal in shape. The rubber is constrained in the radial direction by a circular bead 55 mm in radius that goes around the circumference of the upper disk.

The rubber “hose” in the half-symmetry, three-dimensional model is modeled with 550 S4R finite-strain shell elements. The mesh in the upper hemisphere of the hose is more refined than that in the lower hemisphere, because the rubber membrane undergoes a reversal in curvature in the upper region as it contours the circular bead attached to the upper disk. The circular bead is modeled using an axisymmetric, discrete rigid surface.

Contact with the rubber is enforced by defining a contact pair between this rigid surface and a surface defined on the (deformable) shell mesh in the contacting region. The metal disks are assumed to be rigid relative to the rubber component of the airspring. The lower metal disk is modeled using boundary conditions, while the upper disk is modeled as part of the rigid surface. The meshes of the rubber membrane and the rigid surface are shown in *Figure 3*.

The fluid cavity is modeled using the surface-based fluid cavity capability (see *About Surface-Based Fluid Cavities*). To define the cavity completely and to ensure proper calculation of its volume, surface elements are defined in the three-dimensional models along the bottom and top rigid disk boundaries of the cavity, even though no displacement elements exist along those surfaces. Since Abaqus does not provide two-dimensional surface elements, the rigid disk boundaries are modeled with structural elements instead of surface elements in the axisymmetric models. The cavity reference node 50000 has a single degree of freedom representing the pressure inside the cavity. Because of symmetry only half of the cavity boundary has been modeled. The cavity reference node has been placed on the model's symmetry plane, $y = 0$, to assure proper calculation of the cavity volume. *Figure 4* shows the mesh of the airspring's cavity.

To facilitate comparisons, the two-dimensional axisymmetric model uses the same cross-sectional mesh refinement as the 180° model. The rubber component is modeled with 25 SAX1 shell elements. The circular bead is modeled with an element-based rigid surface constructed of RAX2 rigid elements. Contact with the hose is enforced by defining a contact pair between this rigid surface and a surface defined on the (deformable) shell mesh in the contacting region. Once again, the lower rigid metal disk is modeled by boundary conditions, and the upper rigid metal disk is modeled as part of the rigid body. The mesh of the rubber membrane and the contact main surface is shown in *Figure 5*, and the mesh of the cavity is shown in *Figure 6*. For the membrane model the SAX1 elements are replaced with either MAX1 elements or MGAX1 elements.

Symmetry boundary conditions and initial shell curvature

Symmetry is exploited in the three-dimensional airspring model, and the plane $y = 0$ is a plane of symmetry. Accurate definition of the initial curvature of the surface being modeled is required, especially on the plane of symmetry. If the user does not provide this information by specifying the normal to the surface at the shell nodes, Abaqus estimates the normal direction based on the coordinates of the surrounding nodes on the shell. Normals computed in this fashion will be inaccurate on the symmetry plane: they will have out-of-plane components, which may lead to inaccurate results. To avoid these difficulties, direction cosines are specified for all shell nodes in the model.

Material properties

The walls of an airspring's rubber component are made from plies of symmetrically placed, positively and negatively oriented reinforcement cords. The walls of an actual component are made of several such layers. However, for the purposes of the three-dimensional example problem being considered, the airspring's wall is taken to be a rubber matrix with a single 6-mm-thick symmetric layer of positively and negatively oriented cords. The cords are modeled by uniformly spaced skew rebar in the shell elements. The rebar are assumed to be made of steel. The rubber is modeled as an incompressible Mooney-Rivlin (hyperelastic) material with $C_{10} = 3.2$ MPa and $C_{01} = 0.8$ MPa, and the steel is modeled as a linear elastic material with $E = 210.0$ GPa and $\nu = 0.3$.

Skew rebar orientations in shell elements are defined by giving the angle between the local 1-axis and the rebar. The default local 1-direction is the projection of the global x-axis onto the shell surface (see *Conventions*). It is for this reason, and to make the rebar definition uniform for all elements, that the axis of revolution of the airspring model has been chosen to be the global x-axis. Two rebar layers, `PLSBAR` and `MNSBAR`, have been defined with orientation angles of 18° and −18°, respectively. The cross-sectional area of the rebar is 1 mm^2, and they are spaced every 3.5 mm in the shell surface.

The above rebar specification is simplified and somewhat unrealistic. The reinforced plies used in the manufacture of the airspring are located in an initially cylindrical tube with uniform rebar angles. However, the transformation

of these layers from a cylindrical geometry to a toroidal one gives the airspring a variable rebar angle and rebar spacing that is dependent on the radius from the axis of revolution of the torus and on the initial rebar angle (see Fursdon, 1990). Hence, a more realistic simulation would require different rebar definitions in each ring of elements in the airspring model.

In the axisymmetric shell model the airspring walls are modeled by a three-layer composite shell section. The two outer layers are each 2.5 mm thick and made up of the same Mooney-Rivlin material that is used in the 180° model. The middle "rebar" layer is 1 mm thick and is made up of an orthotropic elastic material that captures the mechanical behavior of the positively and negatively oriented rebar definition used in the three-dimensional airspring model.

The plane stress orthotropic engineering constants are obtained by looking at the response of a typical element in the three-dimensional model (element 14) subjected to uniaxial extensions along the local 1- and 2-directions. Using a shell thickness of 1 mm, the in-plane states of stress and strain resulting from these two tests are

Test	ε_1	ε_2	σ_1 (MPa)	σ_2 (MPa)
1-direction	1.00×10^{-2}	-8.75×10^{-2}	2.48×10^{1}	-2.41×10^{-5}
2-direction	-1.05×10^{-3}	1.00×10^{-2}	-5.96×10^{-6}	2.86×10^{-1}

For a plane-stress orthotropic material the in-plane stress and strain components are related to each other as follows:

$$\begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \end{pmatrix} = \begin{pmatrix} 1/E_1 & -\nu_{12}/E_1 \\ -\nu_{21}/E_2 & 1/E_2 \end{pmatrix} \begin{pmatrix} \sigma_1 \\ \sigma_2 \end{pmatrix},$$

where E_1, E_2, ν_{12}, and ν_{21} are engineering constants. Solving for these constants using the above stress-strain relation and the results of the two uniaxial tests yields

$$\begin{aligned} E_1 &= 2.48 \times 10^3 \text{ MPa} \\ E_2 &= 2.86 \times 10^1 \text{ MPa} \\ \nu_{12} &= 9.1 \\ \nu_{21} &= 0.1 \end{aligned}$$

The remaining required engineering constants—G_{12}, G_{13}, and G_{23}—play no role in the rebar layer definition. Consequently, they have been arbitrarily set to be equal to the shear modulus of the rubber, which is given by $2\left(C_{10} + C_{01}\right)$.

For the axisymmetric membrane model the bulk material is chosen to have the same material properties (Mooney-Rivlin hyperelastic) as those used in the 180° model and the axisymmetric shell model. The rebar parameters and material properties are chosen such that they capture the initial material properties of the sandwiched steel layer in the axisymmetric shell model. The principal material directions do not rotate in the axisymmetric shell model (they are the default element basis directions—the meridional and the hoop directions, respectively). However, they do rotate with finite strain in the axisymmetric membrane model as a result of the use of rebar. Initial stresses are applied to the rebar in the axisymmetric membrane model.

In all analyses the air inside the airspring cavity has been modeled as an ideal gas with molecular weight of 0.044 kg and molar heat capacity of 30 J/kg °K.

Loading

In the Abaqus/Standard model the airspring is first pressurized to 506.6×10^3 kPa (5 atms) while holding the upper disk fixed. This pressure is applied by prescribing boundary degree of freedom 8 at the cavity reference node. In this case the air volume is adjusted automatically to fill the cavity.

In the next step the prescribed boundary condition on the pressure degree of freedom is removed, thus sealing the cavity with the current air volume. In addition, during this step the boundary condition on the vertical displacement degree of freedom of the rigid body reference node is removed, and in its place a downward load of 150 kN is applied.

The next step is a static linear perturbation procedure. In the axisymmetric model two load cases are considered: one tests the axial stiffness of the airspring with the cavity pressure allowed to vary (closed cavity conditions) and the other tests its axial stiffness with the cavity pressure fixed. The linear perturbation step in the three-dimensional analysis contains three load cases, all under variable cavity pressure (closed cavity) conditions: the first tests the axial stiffness of the airspring, the second tests its lateral stiffness, and the third tests its rotational stiffness for rocking motion in the symmetry plane.

The axisymmetric analysis concludes with a general step in which the airspring is compressed by increasing the downward load to 240.0 kN. The three-dimensional analysis concludes with a general step in which the airspring is subjected to a lateral displacement of 20 mm.

The loading for the Abaqus/Explicit model is similar to that for the Abaqus/Standard model, except for the linear perturbation steps. The airspring is first pressurized to 506.6×10^3 kPa (5 atms) while holding the upper disk fixed. In the next step the boundary condition on the pressure degree of freedom is removed, thereby sealing the cavity with the current air volume. In addition, for both the axisymmetric and three-dimensional models, the boundary condition on the vertical displacement degree of freedom of the rigid body reference node is modified, so that a downward displacement is applied. This is in contrast to the Abaqus/Standard axisymmetric analysis, where a force was applied. Since the airspring is pressurized, a sudden change in applied force would cause a sudden change in acceleration and induce a low frequency transient response. As a result, the simulation time required for the transient effects to diminish would be very long. Hence, we apply a displacement instead of a force. In the axisymmetric analysis the downward displacement is chosen to be 75 mm, so that the increase in pressure is close to the increase seen in Step 4 of the Abaqus/Standard axisymmetric analysis. The lateral displacement in the three-dimensional Abaqus/Explicit analysis is set to 20 mm so that the results from this step can be compared with those from Step 4 of the Abaqus/Standard three-dimensional analysis.

Results and discussion

Figure 7 and *Figure 8* show displaced shape plots of the axisymmetric shell model at the end of the pressurization step. It is of interest to compare the results from this model with those from the 180° model to validate the material model that was used for the rebar reinforcements in the axisymmetric model. A close look at the nodal displacements reveals that the deformation is practically identical for the axisymmetric and three-dimensional models. The results are also virtually identical between corresponding Abaqus/Standard and Abaqus/Explicit models. Moreover, the axial reaction force at the rigid body reference node is 156 kN for the axisymmetric model and 155 kN for the 180° model (after multiplication by a factor of 2). The cavity volume predicted by the axisymmetric model is 8.22×10^{-2} m^3 versus 8.34×10^{-2} m^3 for the 180° model (again, after multiplication by a factor of 2).

Linearized stiffnesses for the airspring are obtained from the Abaqus/Standard linear perturbation load cases. The stiffness is computed by dividing the relevant reaction force at the rigid body reference node by the appropriate displacement. For the axisymmetric model the airspring's axial stiffness under variable cavity pressure conditions is 826 kN/m; its axial stiffness under fixed cavity pressure conditions is 134 kN/m. The difference in axial stiffness between these two cases (a factor of 6) is the result of differences in cavity pressure experienced during axial compression. Under variable cavity pressure conditions, a fixed mass of fluid (air) is contained in a cavity whose volume is decreasing; thus, the cavity pressure increases. Under fixed cavity pressure conditions, the pressure is prescribed as a constant value for the load case. For the 180° model the predicted stiffnesses under variable cavity pressure are as follows: the axial stiffness is 821 kN/m, the lateral stiffness is 3.31 MN/m, and the rotational stiffness is 273 kN/m.

Figure 9 shows a series of displaced shape plots associated with the compression of the axisymmetric Abaqus/Standard airspring model during Step 4. For comparison, *Figure 10* shows a series of displaced shape plots associated with the compression of the axisymmetric Abaqus/Explicit model during Step 2. *Figure 11*

shows the corresponding load-deflection curves. Although the displacement of the rigid body in the Abaqus/Explicit analysis was applied over a short time period (which caused significant inertial effects in the model), there is still good agreement between the slope of the load-displacement curves from the two analyses. The response of the airspring is only slightly nonlinear; consequently, there is good agreement between the axial stiffness obtained with the linear perturbation load case and that obtained from the slope of the load-displacement curve. *Figure 12* shows a plot of cavity pressure versus the downward displacement of the rigid body in Step 4 of the Abaqus/Standard analysis and Step 2 of the Abaqus/Explicit analysis. The gauge pressure in the cavity increases by approximately 50% during this step. This pressure increase substantially affects the deformation of the airspring structure and cannot be specified as an externally applied load during the step since it is an unknown quantity. *Figure 13* shows a plot of cavity volume versus the downward displacement of the rigid body in Step 4 of the Abaqus/Standard analysis and Step 2 of the Abaqus/Explicit analysis. The cavity pressure and the cavity volume results from the static Abaqus/Standard analysis and the quasi-static Abaqus/Explicit analysis are virtually identical. The corresponding results from the axisymmetric membrane model (not shown) are also in good agreement with the above results.

Figure 14 shows the displaced shape of the 180° Abaqus/Standard model at the end of Step 4, in which a lateral displacement was applied to the airspring. *Figure 15* shows the corresponding displaced shape of the 180° model at the end of Step 2 of the Abaqus/Explicit analysis. *Figure 16* shows a plot of the load-displacement curves obtained from these steps. Although there is a certain amount of noise that results from the contact conditions and the coarseness of the mesh, the load-deflection curve shows good agreement between the analyses performed quasi-statically in Abaqus/Explicit and statically in Abaqus/Standard. The Abaqus/Explicit analysis was run in double precision to eliminate some of the noise in the load-displacement curve.

Input files

hydrofluidairspring_s4r.inp	Three-dimensional Abaqus/Standard model using shell elements.
hydrofluidairspring_s4r_surf.inp	Three-dimensional Abaqus/Standard model using shell elements with surface-to-surface contact.
hydrofluidairspring_sax1.inp	Axisymmetric Abaqus/Standard model using shell elements.
hydrofluidairspring_sax1_surf.inp	Axisymmetric Abaqus/Standard model using shell elements with surface-to-surface contact.
airspring_exp_s4r_surfcav.inp	Three-dimensional Abaqus/Explicit model using shell elements.
airspring_exp_sax1_surfcav.inp	Axisymmetric Abaqus/Explicit model using shell elements.
hydrofluidairspring_m3d4.inp	Three-dimensional Abaqus/Standard model using membrane elements.
hydrofluidairspring_m3d4_surf.inp	Three-dimensional Abaqus/Standard model using membrane elements with surface-to-surface contact.
hydrofluidairspring_max1.inp	Axisymmetric Abaqus/Standard analysis using rebar reinforced membrane elements.
hydrofluidairspring_mgax1.inp	Axisymmetric Abaqus/Standard analysis using rebar reinforced membrane elements with twist.
airspring_s4r_gcont_surfcav.inp	Three-dimensional Abaqus/Explicit analysis using shell elements and general contact.

References

- Dils, M., “Air Springs vs. Air Cylinders,” *Machine Design*, May 7, 1992.
- Fursdon, P. M. T., “Modelling a Cord Reinforced Component with Abaqus,” *6th UK Abaqus User Group Conference Proceedings*, 1990.

Figures

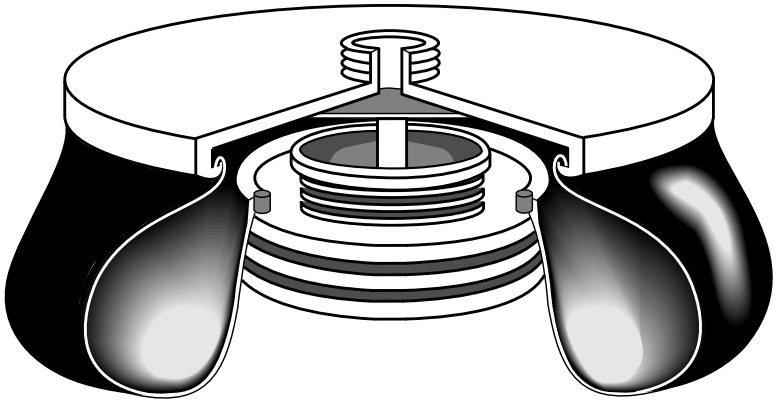


Figure 1: A cord-reinforced airspring.

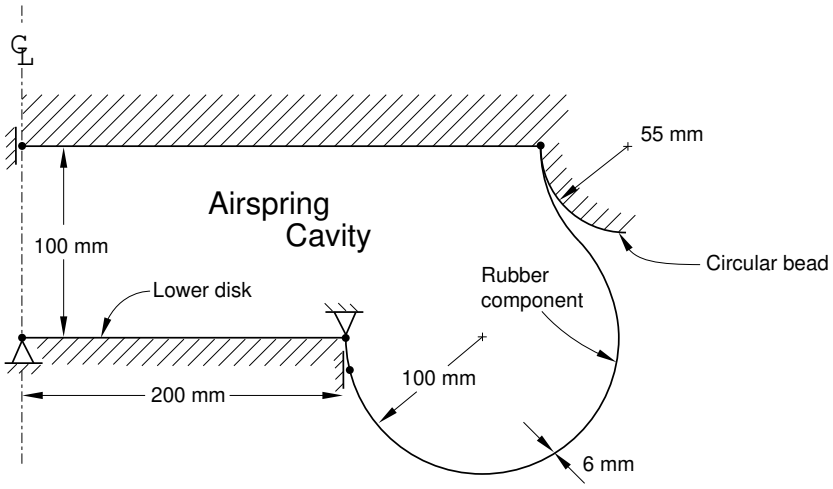


Figure 2: The airspring model cross-section.

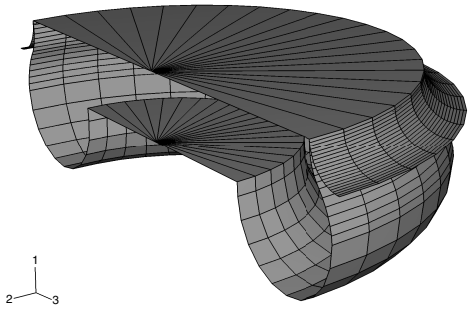


Figure 3: 180° model: mesh of the rubber membrane and the contact main surface.

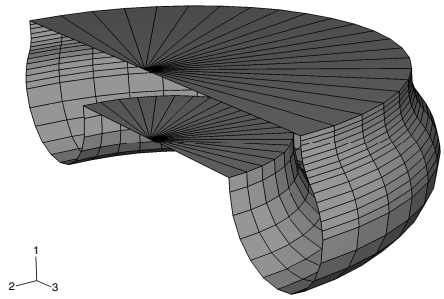


Figure 4: 180° model: mesh of the airspring cavity.

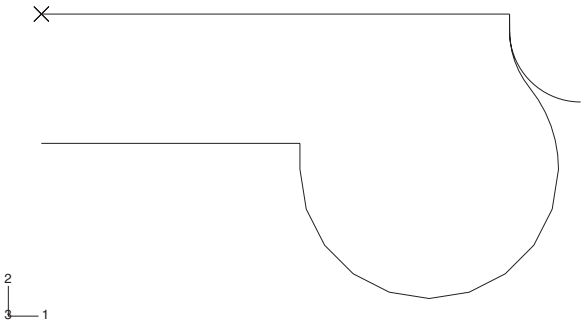


Figure 5: Axisymmetric model: mesh of the rubber membrane and the contact main surface.

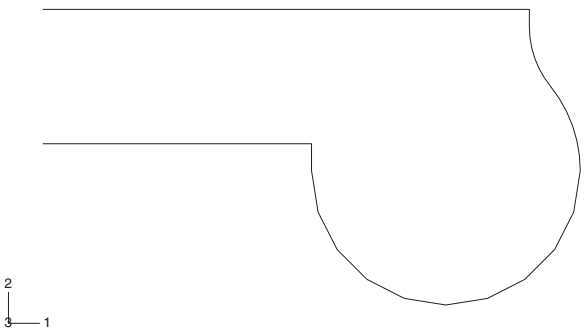


Figure 6: Axisymmetric model: mesh of the airspring cavity.

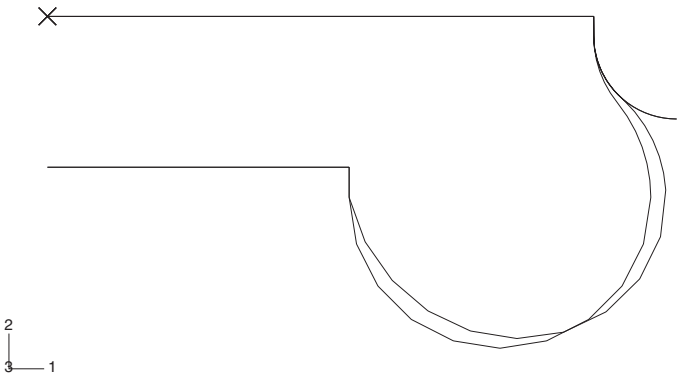


Figure 7: Axisymmetric Abaqus/Standard model: deformed configuration at the end of Step 1.

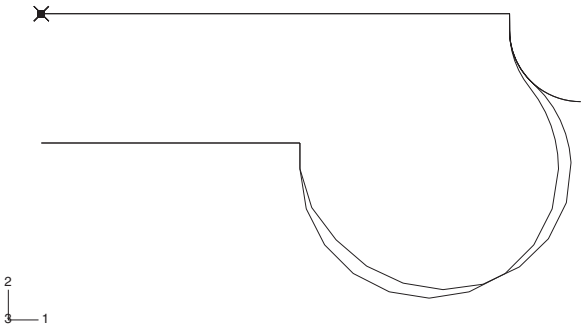


Figure 8: Axisymmetric Abaqus/Explicit model: deformed configuration at the end of Step 1.

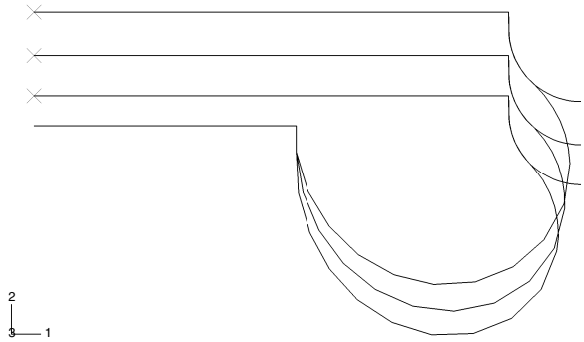


Figure 9: Axisymmetric Abaqus/Standard model: progressive deformed configurations during Step 4.

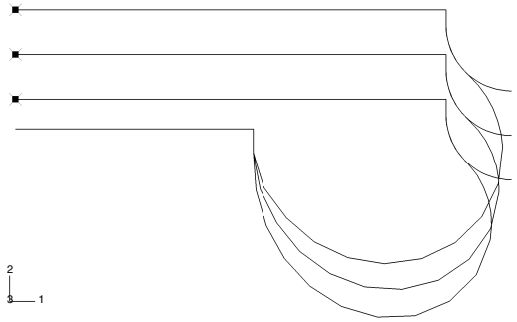


Figure 10: Axisymmetric Abaqus/Explicit model: progressive deformed configurations during Step 2.

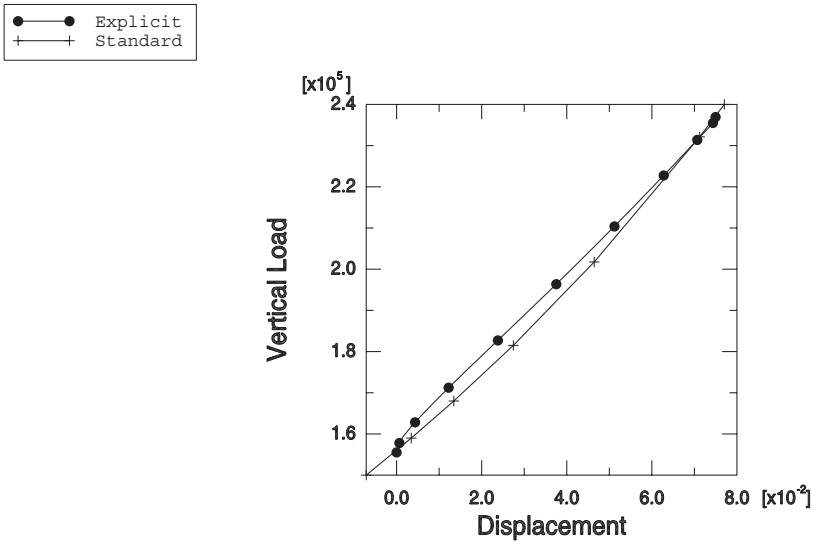


Figure 11: Load-displacement curves (axisymmetric models).

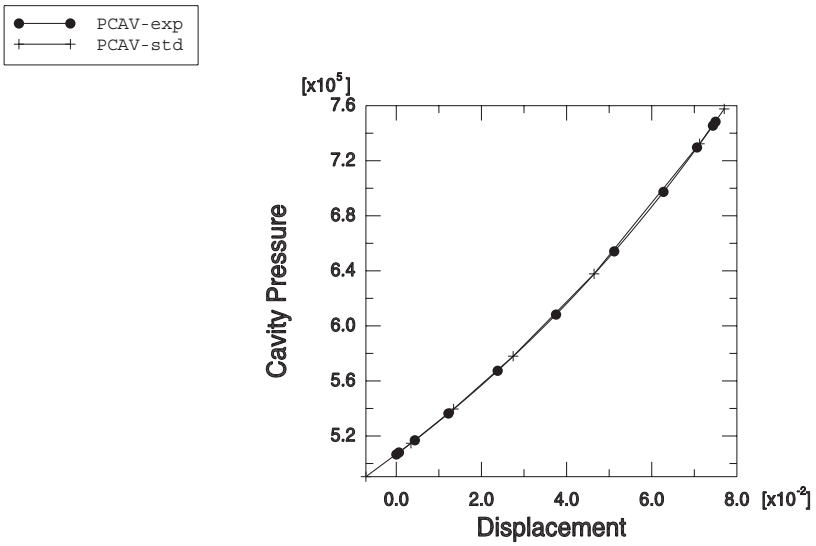


Figure 12: Cavity pressure versus downward displacement (axisymmetric models).

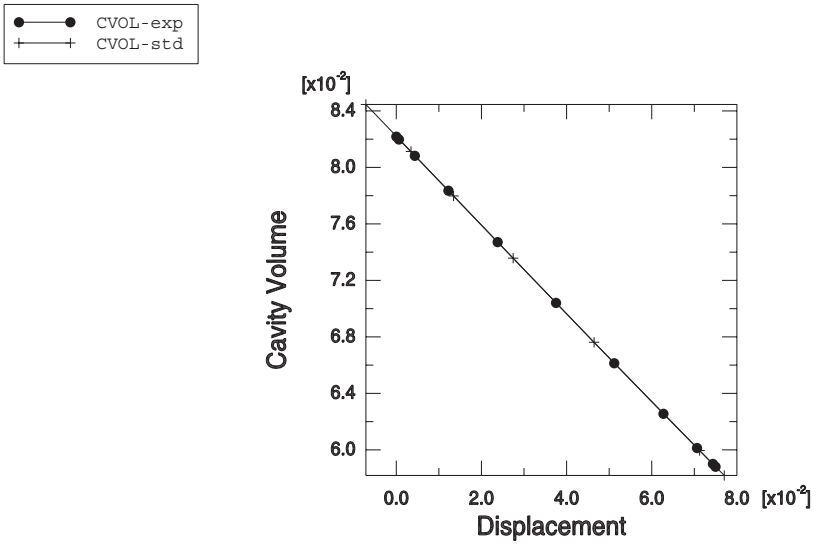


Figure 13: Cavity volume versus downward displacement (axisymmetric models).

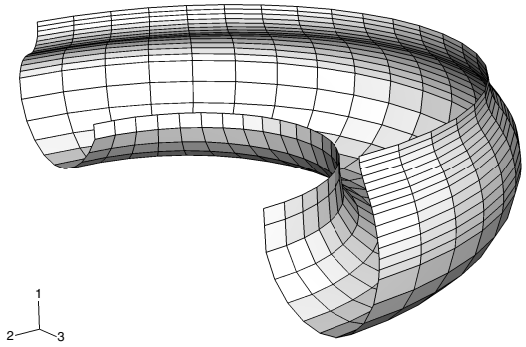


Figure 14: 180° Abaqus/Standard model: deformed configuration at the end of Step 4.

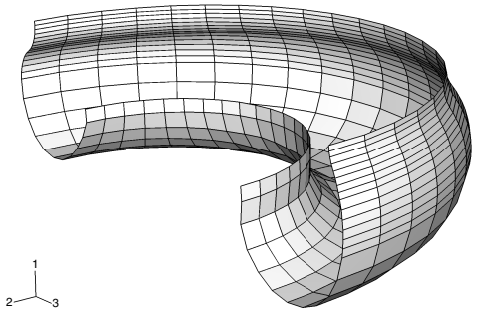


Figure 15: 180° Abaqus/Explicit model: deformed configuration at the end of Step 2.

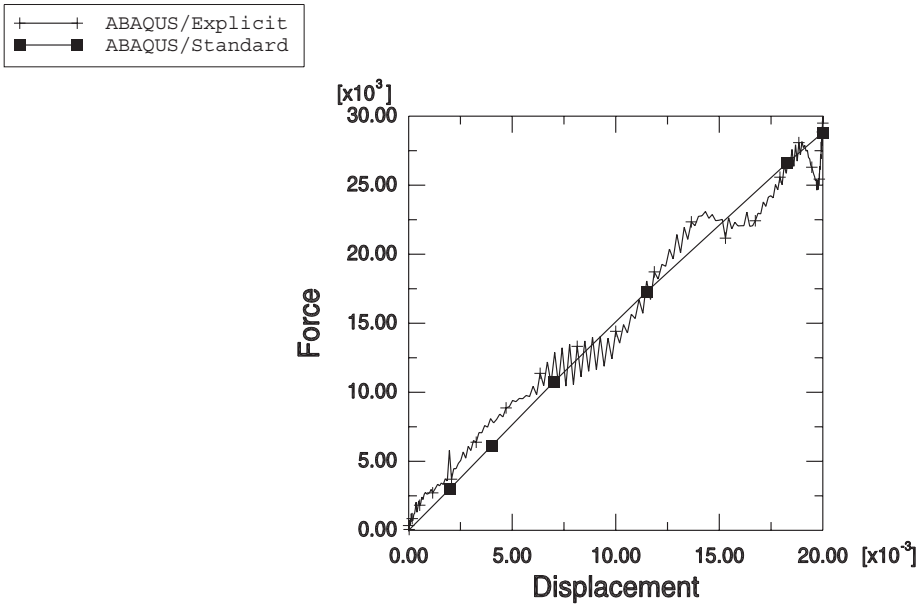


Figure 16: Load-displacement curves for the 180° analyses.

Shell-to-solid submodeling and shell-to-solid coupling of a pipe joint

Products: Abaqus/Standard Abaqus/Explicit Abaqus/CAE

Overview

This example demonstrates the use of the shell-to-solid submodeling and shell-to-solid coupling capabilities in Abaqus, which both provide cost-effective approaches to model enhancement.

Submodeling is the technique used in Abaqus for analyzing a local part of a model with a refined mesh, based on interpolation of the solution from an initial global model (usually with a coarser mesh) onto the nodes on the appropriate parts of the boundary of the submodel. Shell-to-solid submodeling models a region with solid elements, when the global model is made up of shell elements. This example uses the scaling parameter in the submodel boundary condition to scale the values of prescribed boundary conditions for driven variables without requiring the global model to be rerun.

Shell-to-solid coupling is a feature in Abaqus by which three-dimensional shell meshes can be coupled automatically to three-dimensional solid meshes. Unlike shell-to-solid submodeling, which first performs a global analysis on a shell model followed by a submodel analysis with a continuum model, the shell-to-solid coupling model uses a single analysis, with solid and shell elements used in different regions.

The analysis is tested as a static process in Abaqus/Standard and as a dynamic process in both Abaqus/Standard and Abaqus/Explicit. To demonstrate the shell-to-solid submodeling capability, the problem is solved quasi-statically in Abaqus/Explicit. The overall displacements are small, and to avoid noise-induced dynamic effects, the quasi-static Abaqus/Explicit analysis is run in double precision.

In addition, an Abaqus Scripting Interface script is included that creates a shell global model using Abaqus/CAE. The script then uses data from the output database created by the analysis of the global model to drive a solid submodel. The script ends by displaying an overlay plot of the global model and the submodel in the Visualization module.

Geometry and model

In this problem the joint between a pipe and a plate is analyzed. A pipe of radius 10 mm and thickness 0.75 mm is attached to a plate that is 10 mm long, 5 mm wide, and 1 mm thick. The pipe-plate intersection has a fillet radius of 1 mm. Taking advantage of the symmetry of the problem, only half the assembly is modeled. Both the pipe and the plate are assumed to be made of aluminum with $E = 69 \times 10^3$ MPa, $\nu = 0.3$, and $\rho = 2740$ kg/m^3.

The global model for the submodeling analysis is meshed with S4R elements as shown in *Figure 1*. The fillet radius is not taken into consideration in the shell model. The static submodel is meshed using three-dimensional C3D20R continuum elements (see *Figure 2*). A coarser mesh using C3D8R elements is chosen for the dynamic tests. The shell-to-solid coupling model is meshed with S4R shell elements and C3D20R continuum elements as shown in *Figure 3*. The continuum meshes used in the static submodeling and shell-to-solid coupling analyses are identical. The continuum meshes extend 10 mm along the pipe length, have a radius of 25 mm in the plane of the plate, and use four layers through the thickness. The continuum meshes accurately model the fillet radius at the joint. Hence, it is possible to calculate the stress concentration in the fillet. The problem could be expanded by adding a ring of welded material to simulate a welded joint (for this case the submodel would have to be meshed with new element layers representing the welded material at the joint). The example could also be expanded by including plastic material behavior in the submodel while using an elastic global model solution.

A reference static solution consisting entirely of C3D20R continuum elements is also included (see *Figure 4*). The mesh of the reference solution in the vicinity of the joint is very similar to that used in the submodeling and shell-to-solid coupling analyses.

The geometry and material properties of the Abaqus/Explicit shell-to-solid coupling model are identical to the Abaqus/Standard models. The Abaqus/Explicit shell model is meshed with S4R elements, and the continuum model is meshed with C3D10M elements.

Loading

The pipe is subjected to a concentrated load acting in the *x*-direction applied at the free end, representing a shear load on the pipe. An edge-based surface is defined at the free edge of the pipe. This surface is coupled to a reference node that is defined at the center of the pipe using a distributing coupling constraint. The concentrated load is applied to the reference point. For the submodeling approach the load magnitude is 10 N for the global analysis and 10 N and 20 N, respectively, in the two steps of the submodel analysis (a scale factor of 2.0 is applied to the displacements of the driven nodes in the second step of the submodel analysis). For the shell-to-solid coupling approach the load magnitude is 10 N and 20 N, respectively, in the two steps, as in the submodel analysis. For the dynamic cases the load is applied gradually over the entire step time by using a smooth-step amplitude curve.

Kinematic boundary conditions

The plate is clamped along all edges. In the solid submodel, kinematic conditions are interpolated from the global model at two surfaces of the submodel: one lying within the pipe and the other within the plate. The default center zone size, equal to 10% of the maximum shell thickness, is used. Thus, only one layer of driven nodes lies within the center zone, and only these nodes have all three displacement components driven by the global solution. For the remaining driven nodes only the displacement components parallel to the global model midsurface are driven from the global model. Thus, a single row of nodes transmits the transverse shear forces from the shell solution to the solid model.

Results and discussion

The loading and boundary conditions are such that the pipe is subjected to bending. The end of the pipe that is attached to the plate leads to deformation of the plate itself (see *Figure 5* and *Figure 6*). From a design viewpoint the area of interest is the pipe-plate joint where the pipe is bending the plate. Hence, this area is modeled with continuum elements to gain a better understanding of the deformation and stress state.

Figure 7 shows the contours of the out-of-plane displacement component in the plate for both the static submodel and the shell-to-solid coupling analyses. The submodel is in good agreement with the displacement of the global shell model around the joint. The out-of-plane displacement for the shell-to-solid coupling is slightly less than that for the submodel analysis but is in good agreement with the reference solution shown in *Figure 8*.

The stress concentration in the fillet radius is obtained for the solid models. The maximum von Mises stresses at the integration points and nodes for the reference solution, submodel, and shell-to-solid coupling analyses are shown in *Table 1*. As illustrated in *Table 1* and *Figure 9*, the von Mises stress computed in the shell-to-solid coupling analysis agrees very well with the reference solution. The continuity of displacements and the minimal distortion of the stress field at the shell-to-solid interface indicates that the shell-to-solid coupling has been modeled accurately. The difference in the maximum von Mises stress between the submodel analysis and the shell-to-solid coupling solution (see *Figure 10*) can be partially attributed to the fact that the global shell model is more flexible than the shell-to-solid coupling model. The *x*-displacement at the distributing coupling reference node for the global shell model due to the 10 N load is .605 mm compared to .513 mm for the shell-to-solid coupling analysis and .512 mm for the reference solution. Thus, the static submodel mesh is subjected to slightly higher deformation. If the global shell analysis is run with an *x*-displacement boundary condition of .513 mm on the reference node instead of a concentrated load of 10 N, the subsequent maximum nodal von Mises stress in the submodel analysis drops to 80.2 MPa, which is in better agreement with the reference solution. *Figure 11* and *Figure 12* show, respectively, the comparison of the out-of-plane displacement component in the continuum-mesh plate and the comparison of the von Mises stress for the submodel with scaled boundary

condition and the shell-to-solid coupling model with scaled load. The submodel results are in good agreement with the shell-to-solid coupling results.

The relatively large difference between the maximum von Mises stresses at the integration points and the nodes in the region of the fillet (as illustrated in *Table 1*) indicates that the mesh in the fillet region is probably too coarse and should be refined. No such refinement was performed in this example.

Overall, the Abaqus/Explicit shell-to-solid coupling analysis is in good agreement with the Abaqus/Standard shell-to-solid coupling results. The continuity of displacements and the minimal distortion of the stress field at the shell-to-solid interface indicate that the shell-to-solid coupling has been modeled accurately. The out-of-plane displacements of the plate predicted by Abaqus/Explicit are very close to the Abaqus/Standard values. The maximum nodal von Mises stress in the fillet region is 56 MPa, and the x-displacement at the coupling constraint reference node for the global shell model due to the 10 N load is .423 mm. The Abaqus/Explicit analysis is solved quasi-statically by assigning a nominal density of 500 kg/m^3 to the pipe-plate material and ramping up the load over 12,000 increments. Closer approximation to the static limit, achieved by reducing the density of the pipe-plate material to 50 kg/m^3, results in a maximum von Mises stress of 87 MPa in the fillet region, which is very close to the Abaqus/Standard result.

The results for the submodel dynamic cases agree well with the global results. Both Abaqus/Standard and Abaqus/Explicit submodels read the results of the same global Abaqus/Explicit analysis. Good agreement is also found between the Abaqus/Explicit and Abaqus/Standard submodel analyses.

Input files

Static and quasi-static input files

pipe_submodel_s4r_global.inp	S4R global model.
pipe_submodel_s4r_global_n.inp	Node definitions for the S4R global model.
pipe_submodel_s4r_global_e.inp	Element definitions for the S4R global model.
pipe_submodel_c3d20r_sub_s4r.inp	C3D20R submodel that uses the S4R global model. The scaling parameter is used in the second step.
pipe_submodel_c3d20r_sub_s4r_n.inp	Node definitions for the C3D20R submodel that uses the S4R global model.
pipe_submodel_c3d20r_sub_s4r_e.inp	Element definitions for the C3D20R submodel that uses the S4R global model.
pipe_submodel_s4_global.inp	S4 global model.
pipe_submodel_s4_global_n.inp	Node definitions for the S4 global model.
pipe_submodel_s4_global_e.inp	Element definitions for the S4 global model.
pipe_submodel_c3d20r_sub_s4.inp	C3D20R submodel that uses the S4 global model.
pipe_submodel_c3d20r_sub_s4_n.inp	Node definitions for the C3D20R submodel that uses the S4 global model.
pipe_submodel_c3d20r_sub_s4_e.inp	Element definitions for the C3D20R submodel that uses the S4 global model.
pipe_cae_c3d20rsub_s4.py	Python script that creates an S4 global model and a C3D20R submodel using Abaqus/CAE.
pipe_shell2solid_c3d20r_s4r.inp	Shell-to-solid coupling model with C3D20R and S4R elements. The load is scaled in the second step.
pipe_shell2solid_c3d20r_s4r_n1.inp	Node definitions for the shell-to-solid coupling model with C3D20R and S4R elements.
pipe_shell2solid_c3d20r_s4r_n2.inp	Node definitions for the shell-to-solid coupling model with C3D20R and S4R elements.

pipe_shell2solid_c3d20r_s4r_n3.inp	Node definitions for the shell-to-solid coupling model with C3D20R and S4R elements.
pipe_shell2solid_c3d20r_s4r_e1.inp	Element definitions for the shell-to-solid coupling model with C3D20R and S4R elements.
pipe_shell2solid_c3d20r_s4r_e2.inp	Element definitions for the shell-to-solid coupling model with C3D20R and S4R elements.
pipe_shell2solid_c3d20r_s4r_e3.inp	Element definitions for the shell-to-solid coupling model with C3D20R and S4R elements.
pipe_shell2solid_c3d10_s4r.inp	Shell-to-solid coupling model with C3D10 and S4R elements.
pipe_shell2solid_c3d10_s4r_n1.inp	Node definitions for the shell-to-solid coupling model with C3D10 and S4R elements.
pipe_shell2solid_c3d10_s4r_n2.inp	Node definitions for the shell-to-solid coupling model with C3D10 and S4R elements.
pipe_shell2solid_c3d10_s4r_n3.inp	Node definitions for the shell-to-solid coupling model with C3D10 and S4R elements.
pipe_shell2solid_c3d10_s4r_e1.inp	Element definitions for the shell-to-solid coupling model with C3D10 and S4R elements.
pipe_shell2solid_c3d10_s4r_e2.inp	Element definitions for the shell-to-solid coupling model with C3D10 and S4R elements.
pipe_shell2solid_c3d10_s4r_e3.inp	Element definitions for the shell-to-solid coupling model with C3D10 and S4R elements.
pipe_shell2solidx_c3d10m_s4r.inp	Abaqus/Explicit shell-to-solid coupling model with C3D10M and S4R elements.
pipe_shell2solidx_c3d10m_s4r_n1.inp	Node definitions for the Abaqus/Explicit shell-to-solid coupling model with C3D10M and S4R elements.
pipe_shell2solidx_c3d10m_s4r_n2.inp	Node definitions for the Abaqus/Explicit shell-to-solid coupling model with C3D10M and S4R elements.
pipe_shell2solidx_c3d10m_s4r_n3.inp	Node definitions for the Abaqus/Explicit shell-to-solid coupling model with C3D10M and S4R elements.
pipe_shell2solidx_c3d10m_s4r_e1.inp	Element definitions for the Abaqus/Explicit shell-to-solid coupling model with C3D10M and S4R elements.
pipe_shell2solidx_c3d10m_s4r_e2.inp	Element definitions for the Abaqus/Explicit shell-to-solid coupling model with C3D10M and S4R elements.
pipe_shell2solidx_c3d10m_s4r_e3.inp	Element definitions for the Abaqus/Explicit shell-to-solid coupling model with C3D10M and S4R elements.
pipe_c3d20r.inp	Reference model with C3D20R elements.
pipe_c3d20r_n.inp	Node definitions for the reference model with C3D20R elements.
pipe_c3d20r_e.inp	Element definitions for the reference model with C3D20R elements.

Dynamic input files

pipe_submodelx_s4r_global.inp	Abaqus/ExplicitS4R global model.
pipe_submodelx_s4r_global_n.inp	Node definitions for the Abaqus/ExplicitS4R global model.
pipe_submodelx_s4r_global_e.inp	Element definitions for the Abaqus/ExplicitS4R global model.
pipe_submodelx_c3d8r_sub_s4r.inp	Abaqus/ExplicitC3D8R submodel.
pipe_submodelx_c3d8r_sub_s4r_n.inp	Node definitions for the Abaqus/ExplicitC3D8R submodel.
pipe_submodelx_c3d8r_sub_s4r_e.inp	Element definitions for the Abaqus/ExplicitC3D8R submodel.

pipe_submodel_c3d8r_sub_s4r.inp	Abaqus/StandardC3D8R submodel.
pipe_submodel_c3d8r_sub_s4r_n.inp	Node definitions for the Abaqus/StandardC3D8R submodel.
pipe_submodel_c3d8r_sub_s4r_e.inp	Element definitions for the Abaqus/StandardC3D8R submodel.

Tables

Table 1: Von Mises stress comparison for static analyses.

	Maximum integration point von Mises stress (MPa)	Maximum nodal von Mises stress (MPa)
Shell-to-solid submodeling	80.1	97.5
Shell-to-solid coupling	59.8	72.6
Reference	59.9	73.6

Figures

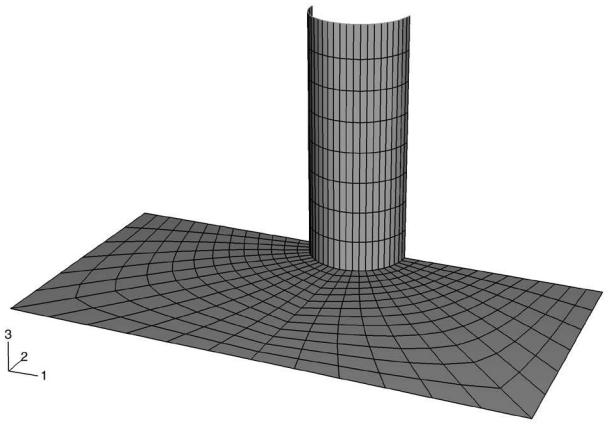


Figure 1: Global shell model of pipe-plate structure.

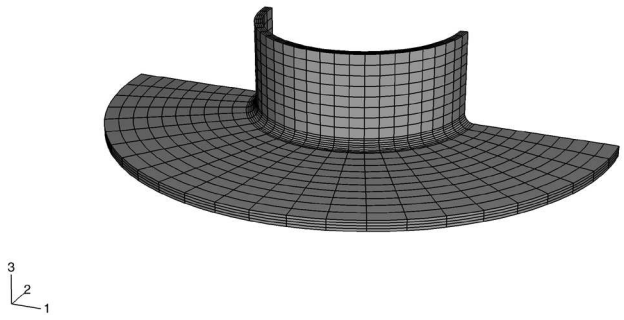


Figure 2: Magnified solid submodel of the pipe-plate joint.

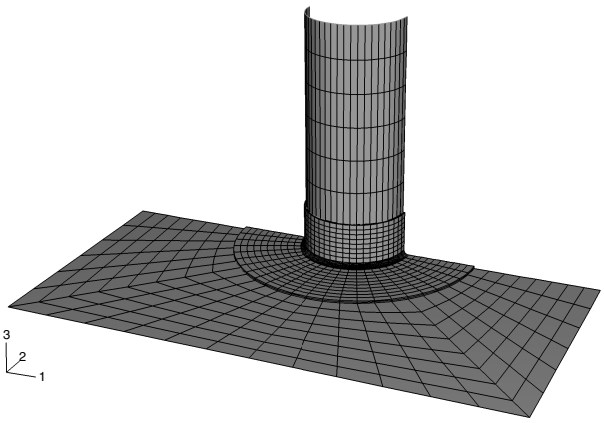


Figure 3: Shell-to-solid coupling model of the pipe-plate joint.

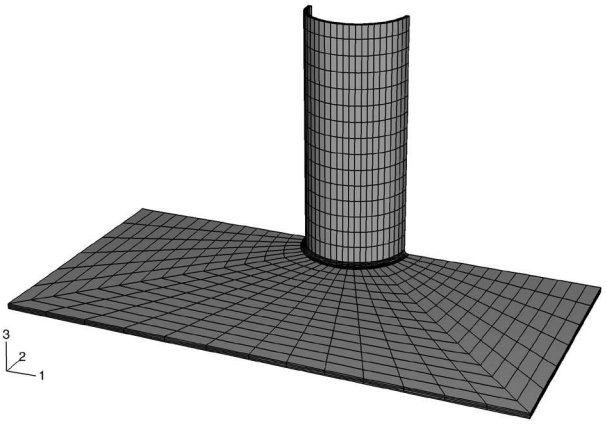


Figure 4: Solid reference model of the pipe-plate joint.

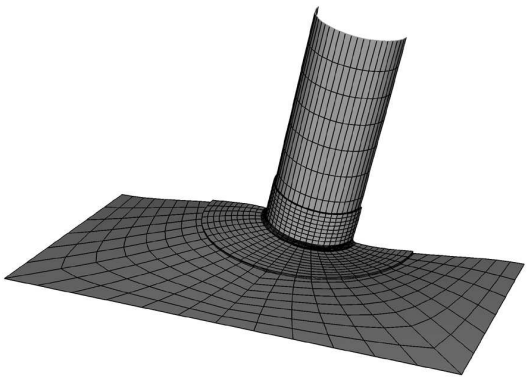


Figure 5: Solid submodel overlaid on the shell model in the deformed state, using a magnification factor of 20.

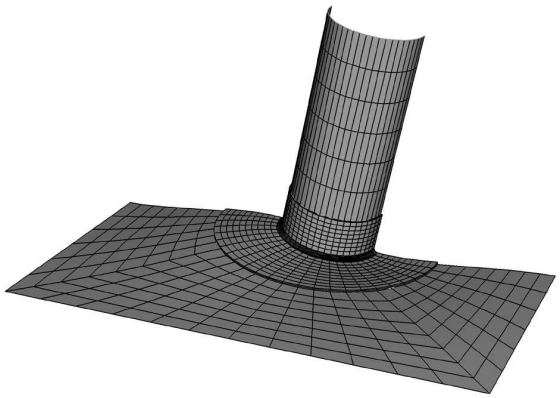


Figure 6: Shell-to-solid coupling model in the deformed state, using a magnification factor of 20.

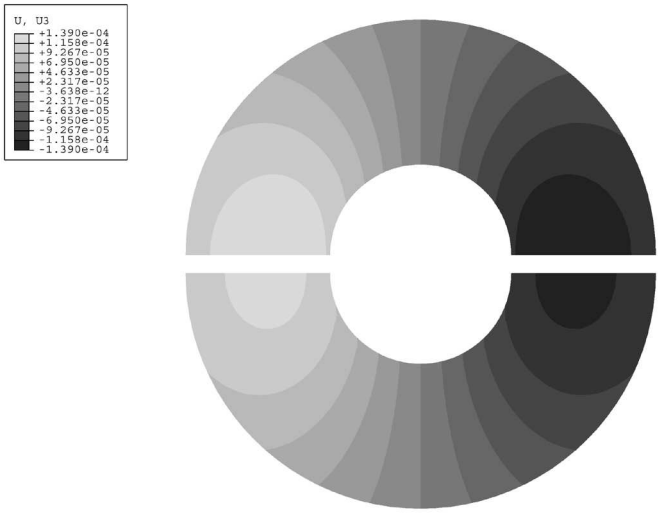


Figure 7: Comparison of out-of-plane displacement in the continuum mesh plate for the submodel (top) and the shell-to-solid coupling analysis (bottom).

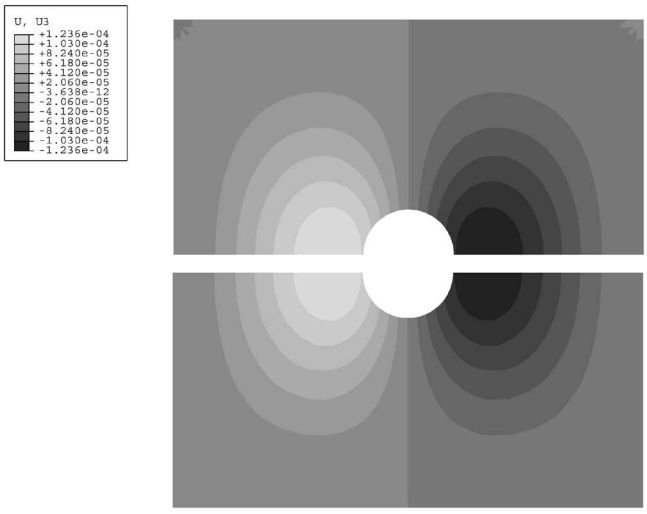


Figure 8: Comparison of out-of-plane displacement in the plate for the reference solution (top) and the shell-to-solid coupling analysis (bottom).

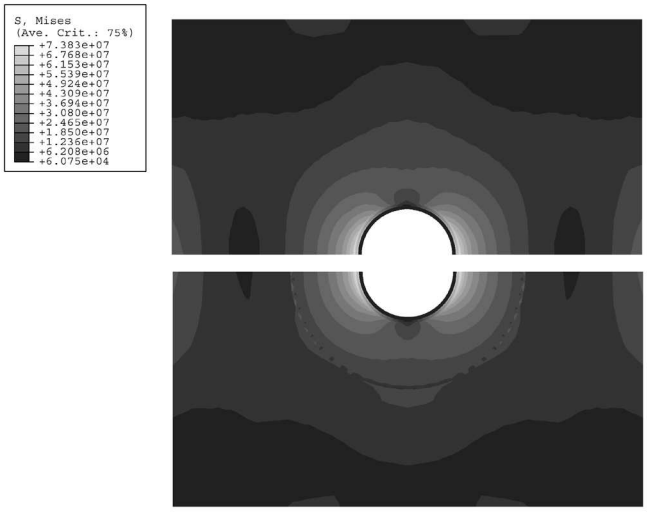


Figure 9: Comparison of the von Mises stress in the plate for the reference solution (top) and the shell-to-solid coupling analysis (bottom).

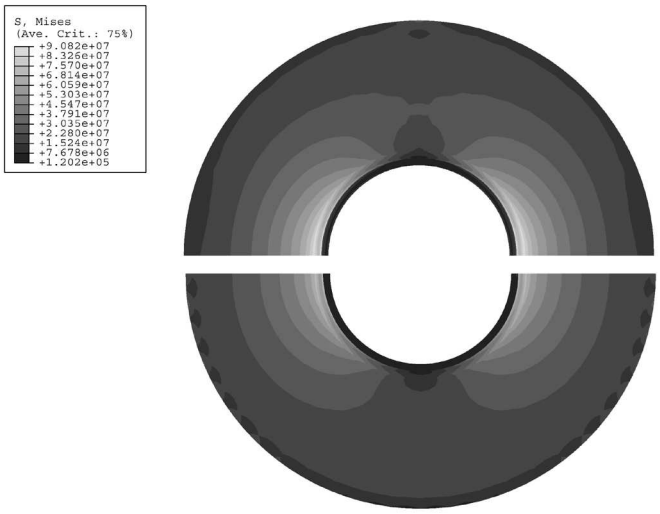


Figure 10: Comparison of the von Mises stress in the continuum mesh plate for the submodel (top) and the shell-to-solid coupling analysis (bottom).

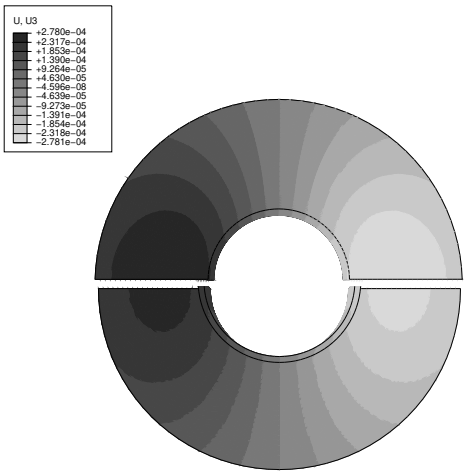


Figure 11: Comparison of out-of-plane displacement in the continuum mesh plate for the submodel with scaled boundary (top) and the shell-to-solid coupling analysis with scaled load (bottom).

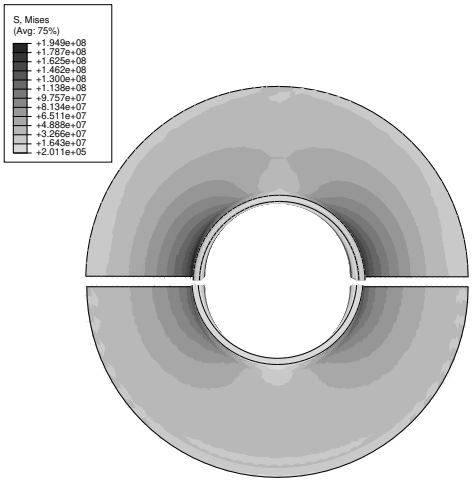


Figure 12: Comparison of the von Mises stress in the continuum mesh plate for the submodel with scaled boundary condition (top) and the shell-to-solid coupling analysis with scaled load (bottom).

Stress-free element reactivation

Products: Abaqus/Standard

Overview

This example demonstrates element reactivation for problems where new elements are to be added in a stress-free state.
Typical examples of stress-free element reactivation include the construction of a gravity dam, in which unstressed layers of material are added to a mesh that has already deformed under geostatic load, or a tunnel in which a concrete or steel support liner is installed.

Element pair reactivation during a step (*Element and Contact Pair Removal and Reactivation*) provides for this type of application directly because the strain in newly added elements corresponds to the deformation of the mesh since the reactivation.

Verification of the element pair reactivation capability is provided in *Model change*.

Problem description

The example considers the installation of a concrete liner to support a circular tunnel. Practical geotechnical problems usually involve a complex sequence of construction steps. The construction details determine the appropriate analysis method to represent these steps accurately. Such details have been avoided here for the sake of simplifying the illustration.

The tunnel is assumed to be excavated in clay, with a Young's modulus of 200 MPa and a Poisson's ratio of 0.2 (see *Figure 1*). The diameter of the tunnel is 8 m, and the tunnel is excavated 20 m below ground surface. The material surrounding the excavation is discretized with first-order 4-node plane strain elements (element type CPE4). The infinite extent of the soil is represented by a 30-m-wide mesh that extends from the surface to a depth of 50 m below the surface. The left-hand boundary represents a vertical symmetry axis. Far-field conditions on the bottom and right-hand-side boundaries are modeled by infinite elements (element type CINPE4). No mesh convergence studies have been performed to establish if these boundary conditions are placed far enough away from the excavation.

An initial stress field due to gravitational and tectonic forces exists through the depth of the soil. It is assumed that this stress varies linearly with depth and that the ratio between the horizontal and vertical stress components is 0.5. The self weight of the clay is 20.0 kN/m^3.

The excavation of the tunnel material is accomplished by applying the forces that are required to maintain equilibrium with the initial stress state in the surrounding material as loads on the perimeter of the tunnel. These loads are then reduced to zero to simulate the excavation. The three-dimensional effect of face advancement during excavation is taken into account by relaxing the forces gradually over several steps. The liner is installed after 40% relaxation of the loads. Further deformation continues to occur as the face of the excavation advances. This ongoing deformation loads the liner.

In the first input file the 150-mm-thick liner is discretized with one layer of incompatible mode elements (element type CPE4I). These elements are recommended in regions where bending response must be modeled accurately. In the second input file beam elements are used to discretize the liner. The liner is attached rigidly to the tunnel. The concrete is assumed to have cured to a strength represented by the elastic properties shown in *Figure 1* by the time the liner is loaded. The liner is not shown in this diagram.

It is expected that an overburden load representing the weight of traffic and buildings exists after the liner is installed.

Analysis method

The excavation and installation of the liner is modeled in four analysis steps. In the first step the initial stress state is applied and the liner elements are removed. Concentrated loads that are in equilibrium with the initial stress field are applied on the perimeter of the tunnel. These forces were obtained from an independent analysis where the displacements on the tunnel perimeter were constrained. The reaction forces at the constrained nodes are the loads applied here. The second step begins the tunnel excavation by reducing the concentrated loads on the tunnel surface. The loads are reduced by 40% in this step before the liner is installed in the third step. No deformation takes place in the soil or liner during the third step. In the fourth step the surface load is applied, and the excavation is completed by removing the remainder of the load on the tunnel perimeter.

In problems involving geometric nonlinearities with finite deformation, it is important to recognize that element reactivation occurs in the configuration at the start of the reactivation step. If the NLGEOM parameter were used in this problem, the thickness of the liner, when modeled with the continuum elements, would have a value at reactivation that would be different from its original value. This result would happen because the outside nodes (the nodes on the tunnel/liner interface) displace with the mesh, whereas the inside nodes remain at their current locations since liner elements are inactive initially. This effect is not relevant in this problem because geometric nonlinearities are not included. However, it may be significant for problems involving finite deformation, and it may lead to convergence problems in cases where elements are severely distorted upon reactivation. This problem would not occur in the model with beam elements because they have only one node through the thickness. In the model where the liner is modeled with continuum elements, the problem can be eliminated if the inner nodes are allowed to follow the outer nodes prior to reactivation, which can be accomplished by applying displacement boundary conditions on the inner nodes. Alternatively, the liner can be overlaid with (elastic) elements of very low stiffness. These elements use the same nodes as the liner but are so compliant that their effect on the analysis is negligible when the liner is present. They remain active throughout the analysis and ensure that the inner nodes follow the outer nodes, thereby preserving the liner thickness.

Results and discussion

Figure 2 shows the stress state at a material point in the liner. The figure clearly indicates that the liner remains unstressed until reactivated.

Figure 3 compares the axial stress obtained from the CPE4I and beam elements at the top and bottom of the liner section. A local cylindrical coordinate system (*Orientations*) is used to orient the liner stresses in the continuum element model along the beam axis so that these stresses can be compared directly with the results of the beam element model. The small difference between the results can be attributed to the element type used in the discretization of the liner: the beam element model uses a plane stress condition, and the continuum element model uses a plane strain condition.

Input files

modelchangedemo_continuum.inp	**MODEL CHANGE* with continuum elements.
modelchangedemo_beam.inp	**MODEL CHANGE* with beam elements.
modelchangedemo_node.inp	Nodal coordinates for the soil.
modelchangedemo_element.inp	Element definitions for the soil.

Figures

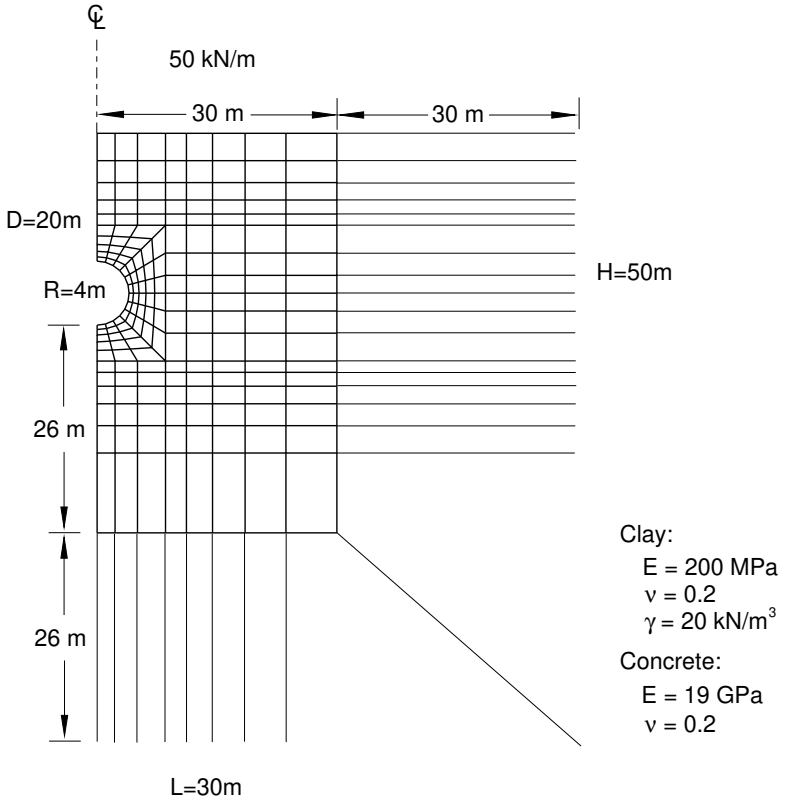


Figure 1: Geometry and finite element discretization.

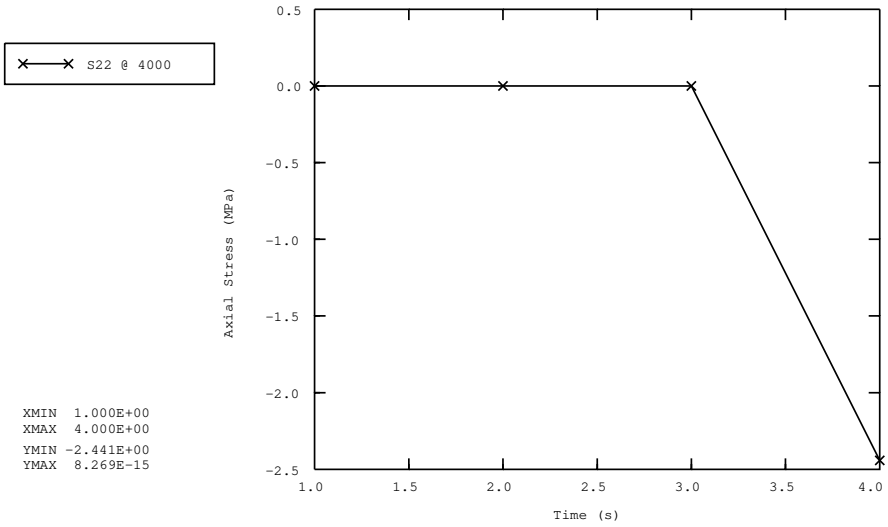


Figure 2: Liner stress during analysis history.

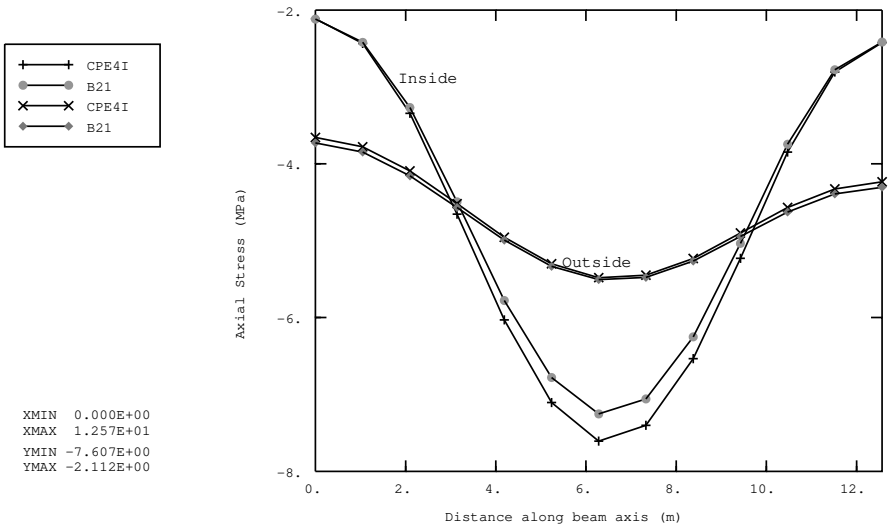


Figure 3: Axial stress along beam inside and outside.

Transient loading of a viscoelastic bushing

Products: Abaqus/Standard

Overview

This example demonstrates the automatic incrementation capability provided for integration of time-dependent material models and the use of the viscoelastic material model in conjunction with large-strain hyperelasticity in a typical design application.

The structure is a bushing, modeled as a hollow, viscoelastic cylinder. The bushing is glued to a rigid, fixed body on the outside and to a rigid shaft on the inside, to which the loading is applied. A static preload is applied to the shaft, which moves the inner shaft off center. This load is held for sufficient time for steady-state response to be obtained. Then a torque is applied instantaneously and held for a long enough period of time to reach steady-state response. We compute the bushing's transient response to these events.

Geometry and model

The viscoelastic bushing has an inner radius of 12.7 mm (0.5 in) and an outer radius of 25.4 mm (1.0 in). We assume that the bushing is long enough for plane strain deformation to occur. The problem is modeled with first-order reduced-integration elements (CPE4R). The mesh is regular, consisting of 6 elements radially, repeated 56 times to cover the 360° span in the hoop direction. The mesh is shown in *Figure 1*. No mesh convergence studies have been performed.

The fixed outer body is modeled by fixing both displacement components at all the outside nodes. The nodes in the inner boundary of the bushing are connected, using a kinematic coupling constraint, to a node located in the center of the model. This node, thus, defines the inner shaft as a rigid body.

Material

The material model is not defined from any particular physical material.

The instantaneous behavior of the viscoelastic material is defined by hyperelastic properties. A polynomial model with $N = 1$ (a Mooney-Rivlin model) is used for this, with the constants $C_{10} =$ 27.56 MPa (4000 psi), $C_{01} =$ 6.89 MPa (1000 psi), and $D_1 =$ 0.0029 MPa^{-1} (0.00002 psi^{-1}).

The viscous behavior is modeled by a time-dependent shear modulus, $G_R(t)$, and a time-dependent bulk modulus, $K_R(t)$, each of which is expanded in a Prony series in terms of the corresponding instantaneous modulus,

$$G_R(t)/G_0 = 1 - \sum_{i=1}^{2} \bar{g}_i^P \left(1 - \exp\left(-\frac{t}{\tau_i}\right)\right)$$

$$K_R(t)/K_0 = 1 - \sum_{i=1}^{2} \bar{k}_i^P \left(1 - \exp\left(-\frac{t}{\tau_i}\right)\right).$$

The relative moduli $\bar{g}_i^P$ and $\bar{k}_i^P$ and time constants τ_i are

i	$\bar{g}_i^P$	$\bar{k}_i^P$	τ_i, sec
1	0.2	0.5	0.1

i	$\overline{g}_i^P$	$\overline{k}_i^P$	τ_i, sec
2	0.1	0.2	0.2

This model results in an initial instantaneous Young's modulus of 206.7 MPa (30000 psi) and Poisson's ratio of 0.45. It relaxes pressures faster than shear stresses.

Analysis

The analysis is done in four steps. The first step is a preload of 222.4 kN (50000 lbs) applied in the x-direction to the node in the center of the model in 0.001 sec with a static procedure (*Static Stress Analysis*). The static procedure does not allow viscous material behavior, so this response is purely elastic. During the second step the load stays constant and the material is allowed to creep for 1 sec by using the quasi-static procedure (*Quasi-Static Analysis*). Since 1 sec is a long time compared with the material time constants, the solution at that time should be close to steady state. The accuracy of the automatic time incrementation during creep response can be controlled. This accuracy tolerance is an upper bound on the allowable error in the creep strain increment in each time increment. It is chosen as 5×10^{-4}, which is small compared to the elastic strains. The third step is another static step. Here the loading is a torque of 1129.8 N-m (10000 lb-in) applied in 0.001 sec. The fourth step is another quasi-static step with a time period of 1 sec.

Results and discussion

Figure 2 through *Figure 5* depict the deformed shape of the bushing at the end of each step. Each of the static loads produces finite amounts of deformation, which are considerably expanded during the holding periods. *Figure 6* shows the displacement of the center of the bushing in the x-direction and its rotation as functions of time.

Input files

viscobushing.inp Input data for the analysis.

Figures

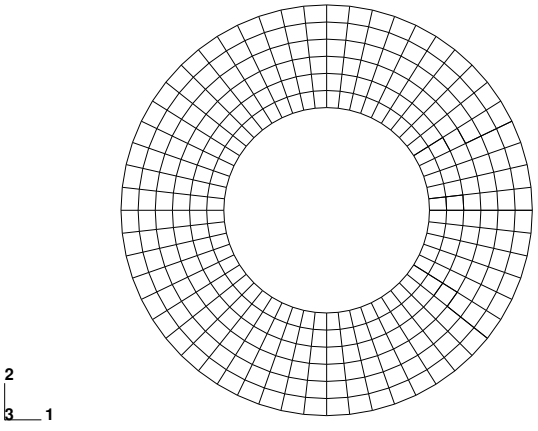


Figure 1: Finite element model of viscoelastic bushing.

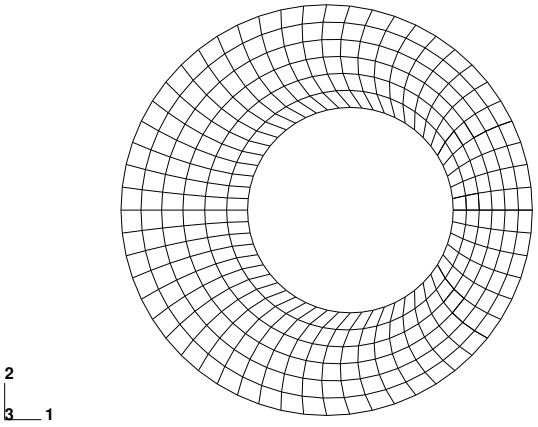


Figure 2: Deformed model after horizontal static loading.

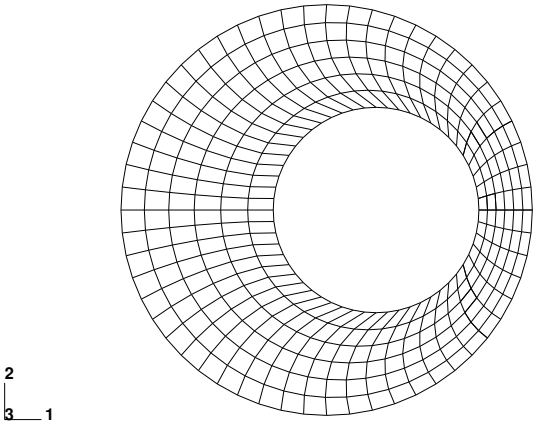


Figure 3: Deformed model after first holding period.

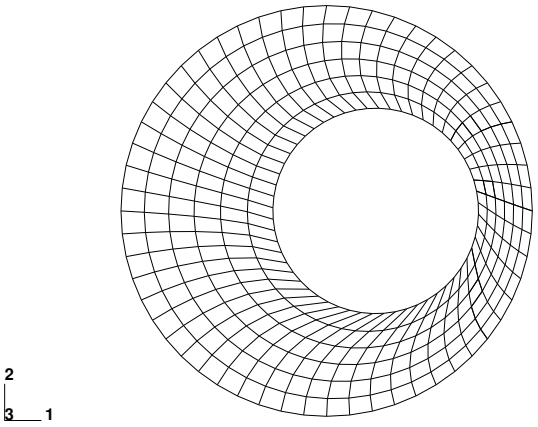


Figure 4: Deformed model after static moment loading.

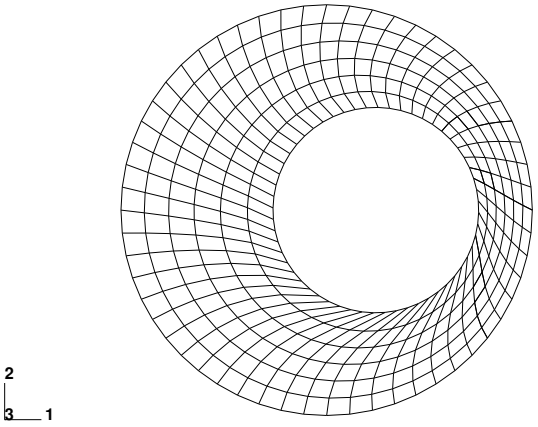


Figure 5: Deformed model after second holding period.

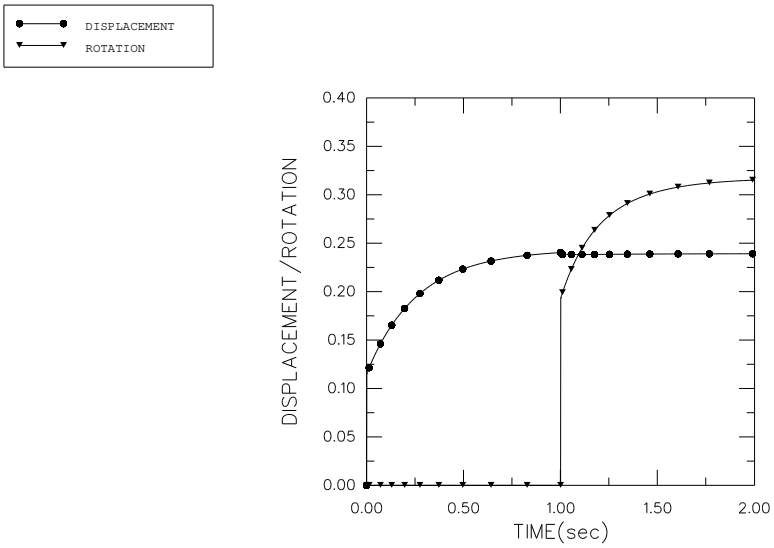


Figure 6: Displacement and rotation of center of bushing.

Indentation of a thick plate

Products: Abaqus/Explicit

Overview

This example illustrates the use of adaptive meshing and distortion control in deep indentation problems.

Problem description

A deep indentation problem is solved for both axisymmetric and three-dimensional geometries, as shown in *Figure 1*. Each model consists of a rigid punch and a deformable blank. The punch has a semicircular nose section and a radius of 100 mm. The blank is modeled as a crushable foam with the elastic response given as follows (see Schluppkotten, 1999):

$E =$ 7.5 MPa (Young's modulus) and

$\nu =$ 0.0 (elastic Poisson's ratio).

The material parameters for the isotropic hardening are given as

$k = \sigma_c^0/p_c^0 =$ 1.0 (yield strength ratio) and

$\nu_p =$ 0.0 (plastic Poisson's ratio),

and the density is

$\rho =$ 60 kg/m^3.

In both cases the punch is fully constrained except in the vertical direction. A deep indentation is made by moving the punch into the blank to a depth of 250 mm when adaptive meshing is used and to a depth of 285 mm when distortion control is used. The displacement of the punch is prescribed using a smooth-step amplitude so that a quasi-static response is generated.

Case 1: Axisymmetric model

The blank is meshed with CAX4R elements and measures 300 × 300 mm. The punch is modeled as an analytical rigid surface using a planar analytical surface in conjunction with a rigid body constraint. The bottom of the blank is constrained in the *x*- and *z*-directions, and symmetry boundary conditions are prescribed at *r*=0.

Case 2: Three-dimensional models

Two models are analyzed. For one model the blank is meshed uniformly, while for the other a graded mesh is used. For both models the blank is meshed with C3D8R elements and measures 600 × 300 × 600 mm. The punch is modeled as an analytical rigid surface using a three-dimensional surface of revolution in conjunction with a rigid body constraint. The bottom of the blank is fully constrained.

Adaptive meshing

A single adaptive mesh domain that incorporates the entire blank is used for each model. A Lagrangian boundary region type (the default) is used to define the constraints along the bottom of the plate for both models and along the axis of symmetry in two dimensions. A sliding boundary region (the default) is used to define the contact

surface on the plate. To obtain a good mesh throughout the simulation, the number of mesh sweeps is increased to 3 as part of the specification of the adaptive mesh domain. For the graded three-dimensional model a graded smoothing objective is specified to preserve the gradation of the mesh while adaptive meshing is performed.

Distortion control

In contrast to the adaptive meshing technique, distortion control does not attempt to maintain a high-quality mesh throughout an analysis but instead tries to prevent negative element volumes or other excessive distortion from occurring during an analysis. By using distortion control, it is possible to prevent an analysis from failing prematurely when the mesh is coarse relative to the strain gradients and the amount of compression. The distortion control capability is tested for axisymmetric and three-dimensional models with a uniformly meshed blank.

Results and discussion

Figure 2 to *Figure 4* show the initial configurations for the axisymmetric model, the three-dimensional uniform mesh model, and the three-dimensional graded mesh model. Although the punch is not shown in these figures, it is initially in contact with the plate. *Figure 5* shows the final deformed mesh for the axisymmetric indentation. The meshing algorithm attempts to minimize element distortion both near and away from the contact surface with the punch. *Figure 6* and *Figure 7* show the deformed mesh of the entire blank and a quarter-symmetry, cutaway view, respectively, for the three-dimensional model with an initially uniform mesh. Even under this depth of indentation, elements appear to be nicely shaped both on the surface and throughout the cross-section of the plate.

Figure 8 and *Figure 9* show the deformed mesh of the entire plate and a quarter-symmetry, cutaway view, respectively, for the three-dimensional case with an initially graded mesh. Adaptive meshing with the graded smoothing objective preserves the mesh gradation throughout the indentation process while simultaneously minimizing element distortion. Preserving mesh gradation in adaptivity problems is a powerful capability that allows mesh refinement to be concentrated in the areas of highest strain gradients. A contour plot of equivalent plastic strain for the graded mesh case is shown in *Figure 10*.

Figure 11 shows the final deformed mesh for the axisymmetric indentation using distortion control without adaptive meshing. *Figure 12* and *Figure 13* show the deformed mesh of the entire blank and a quarter-symmetry, cutaway view, respectively, for the three-dimensional model with an initially uniform mesh using distortion control without adaptive meshing. The distortion control simply prevents element distortion near the contact surface with the punch. Without distortion control both of the analyses fail prematurely under this depth of indentation.

Input files

ale_indent_axi.inp	Case 1 using adaptive meshing.
ale_indent_sph.inp	Case 2 with a uniform mesh using adaptive meshing.
ale_indent_gradedsph.inp	Case 2 with a graded mesh using adaptive meshing.
ale_indent_sphelset.inp	External file referenced by Case 2.
dis_indent_axi.inp	Case 1 using distortion control.
dis_indent_sph.inp	Case 2 with a uniform mesh using distortion control.

References

- Schluppkotten, J., Investigation of the Abaqus Crushable Foam Plasticity Model, Internal report of BMW AG, 1999.

Figures

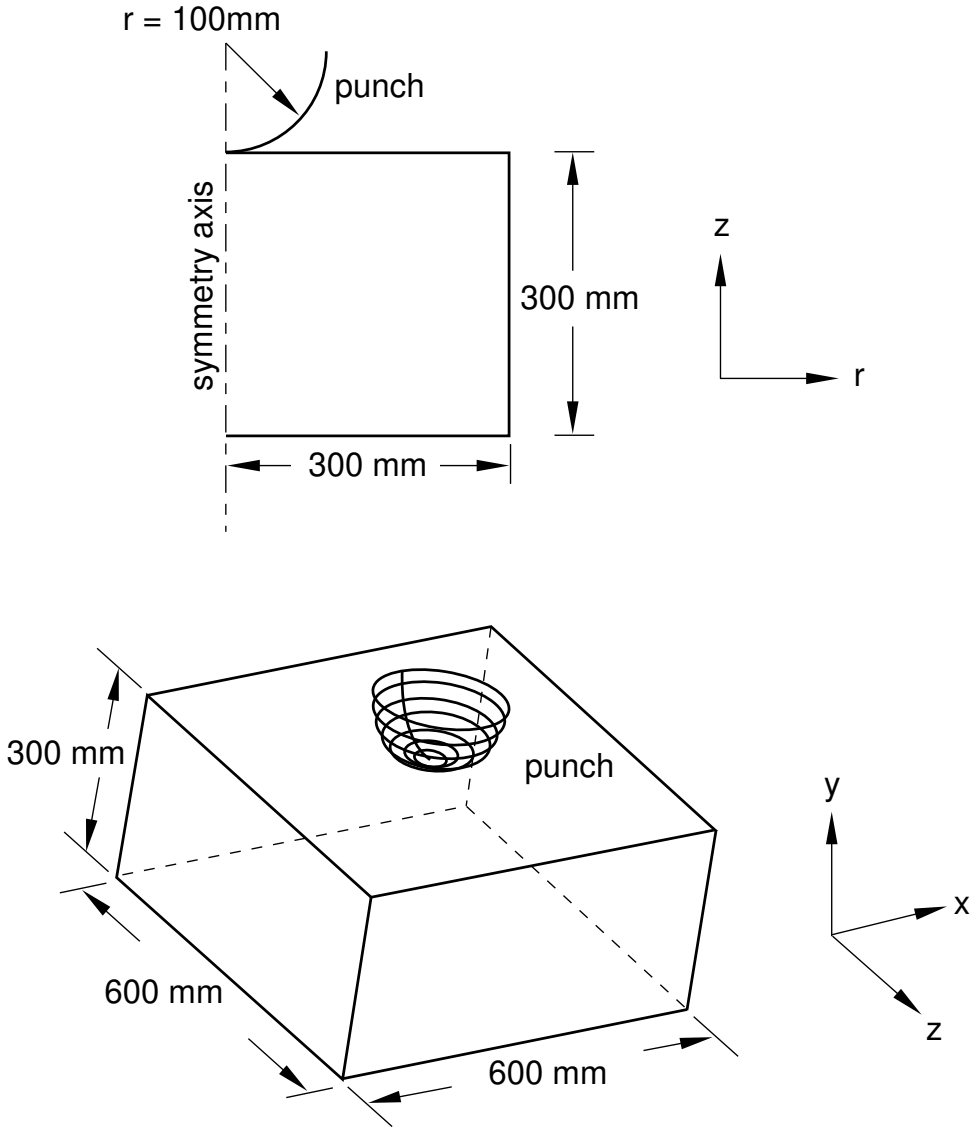


Figure 1: Axisymmetric and three-dimensional model geometries.

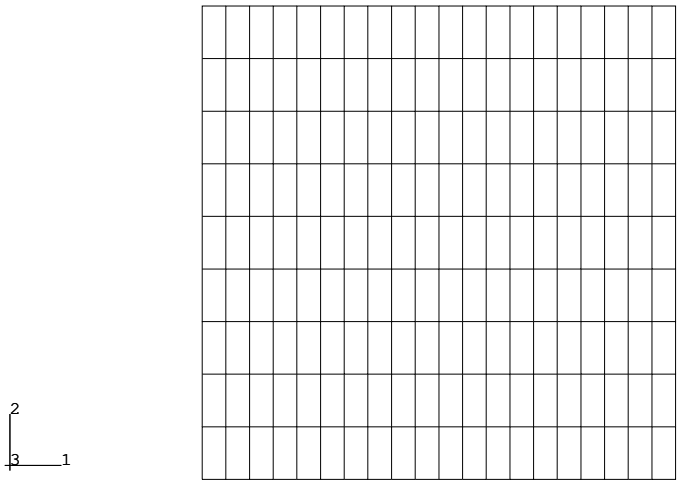


Figure 2: Initial configuration for the axisymmetric model.

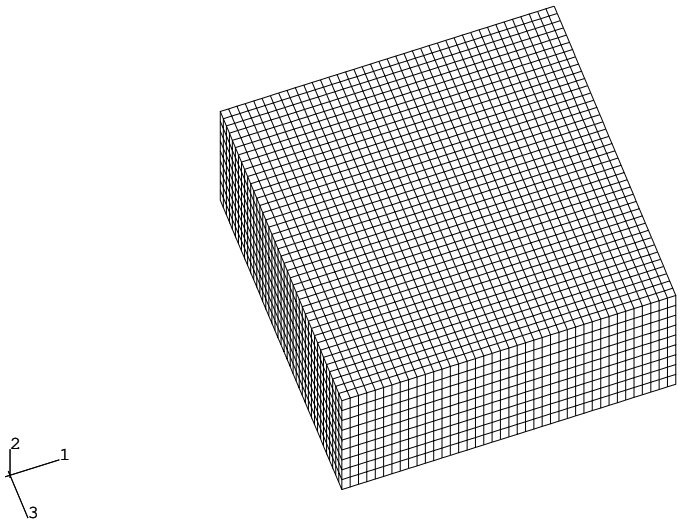


Figure 3: Initial configuration for the three-dimensional model with a uniform mesh.

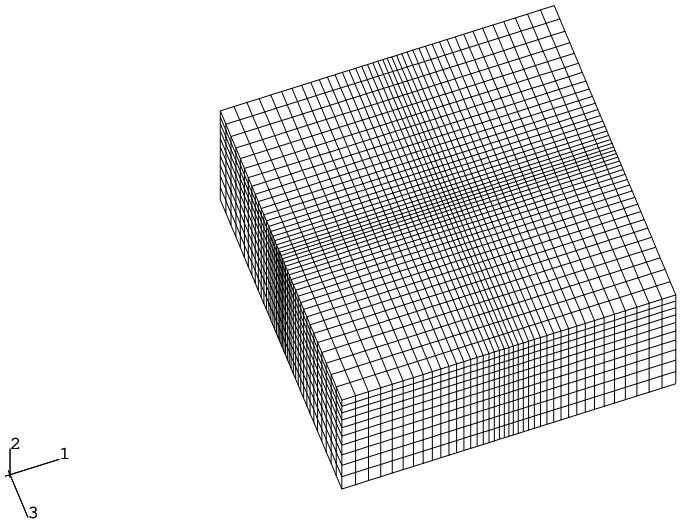


Figure 4: Initial configuration for the three-dimensional model with a graded mesh.

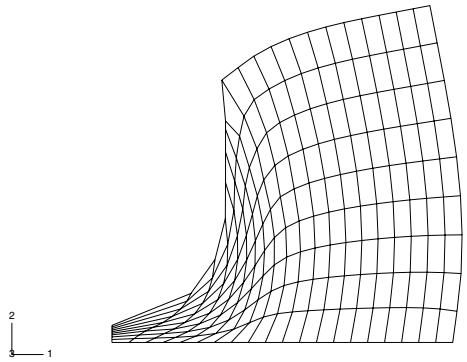


Figure 5: Deformed configuration for the axisymmetric model.

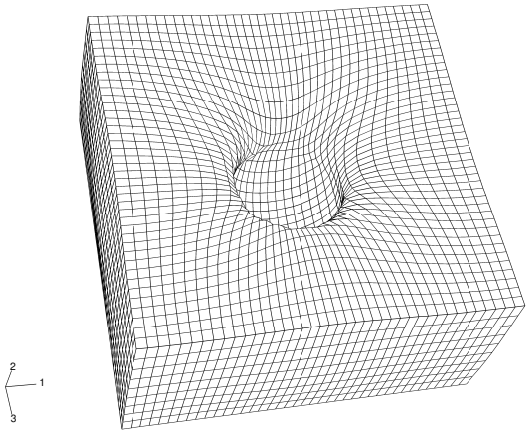


Figure 6: Deformed configuration for the three-dimensional model with an initially uniform mesh.

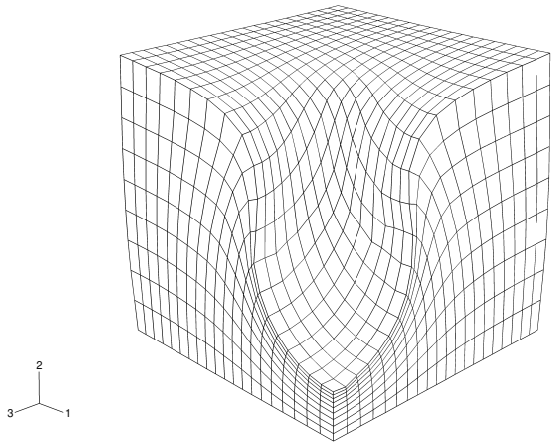


Figure 7: Quarter-symmetry, cutaway view of the deformed configuration for the three-dimensional model with an initially uniform mesh.

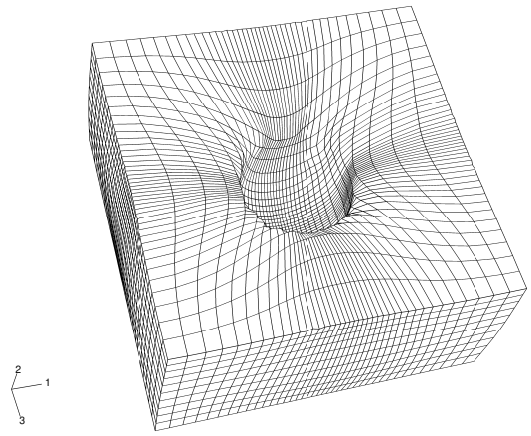


Figure 8: Deformed configuration for the three-dimensional model with an initially graded mesh.

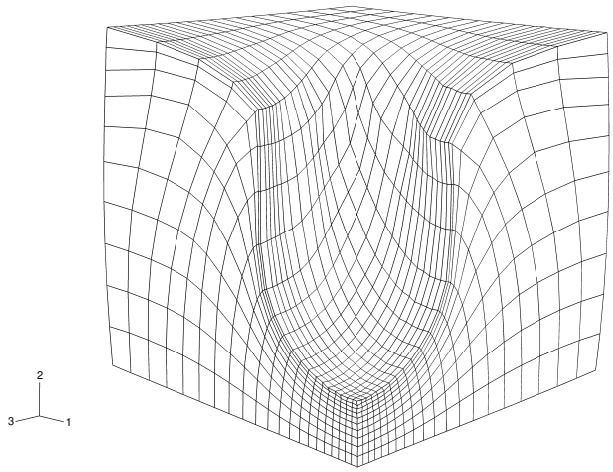


Figure 9: Quarter-symmetry, cutaway view of the deformed configuration for the three-dimensional model with an initially graded mesh.

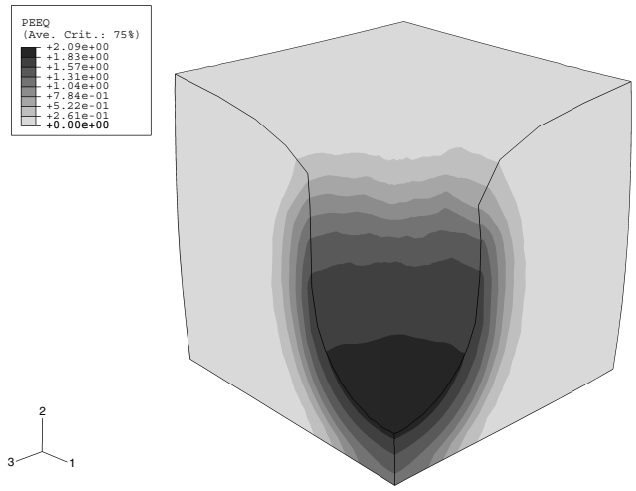


Figure 10: Contours of equivalent plastic strain for the three-dimensional model with an initially graded mesh.

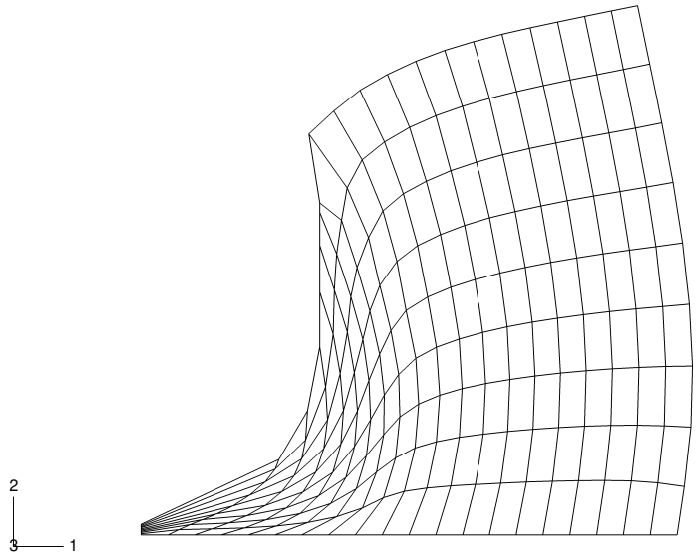


Figure 11: Deformed configuration for the axisymmetric model using distortion control.

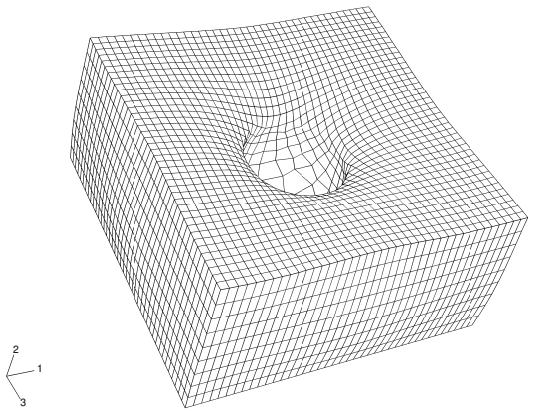


Figure 12: Deformed configuration for the three-dimensional model with an initially uniform mesh using distortion control.

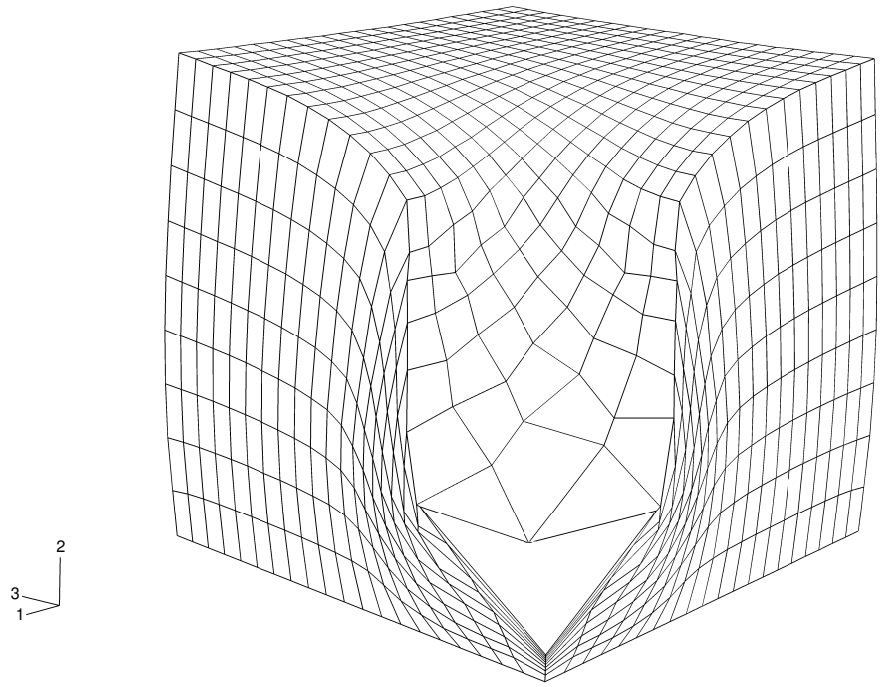


Figure 13: Quarter-symmetry, cutaway view of the deformed configuration for the three-dimensional model with an initially uniform mesh using distortion control.

Damage and failure of a laminated composite plate

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example demonstrates how the nonlinear material behavior of a composite laminate can be specified as a function of solution-dependent variables.
The user subroutines *USDFLD* in Abaqus/Standard and *VUSDFLD* in Abaqus/Explicit can be used to modify the standard linear elastic material behavior (for example, to include the effects of damage) or to change the behavior of the nonlinear material models in Abaqus. The material model in this example includes damage, resulting in nonlinear behavior. It also includes various modes of failure, resulting in abrupt loss of stress carrying capacity (Chang and Lessard, 1989). The analysis results are compared with experimental results.

Problem description and material behavior

A composite plate with a hole in the center is subjected to in-plane compression. The plate is made of 24 plies of T300/976 graphite-epoxy in a $[(-45/+45)_6]_s$ layup. Each ply has a thickness of 0.1429 mm (0.005625 in); thus, the total plate thickness is 3.429 mm (0.135 in). The plate has a length of 101.6 mm (4.0 in) and a width of 25.4 mm (1.0 in), and the diameter of the hole is 6.35 mm (0.25 in). The plate is loaded in compression in the length direction. The thickness of the plate is sufficient that out-of-plane displacements of the plate can be ignored. The compressive load is measured, as well as the length change between two points, originally a distance of 25.4 mm (1.0 in) apart, above and below the hole. The plate geometry is shown in *Figure 1*.

The material behavior of each ply is described in detail by Chang and Lessard. The initial elastic ply properties are longitudinal modulus E_x=156512 MPa (22700 ksi), transverse modulus E_y=12962 MPa (1880 ksi), shear modulus G_{xy}=6964 MPa (1010 ksi), and Poisson's ratio ν_{xy}=0.23. The material accumulates damage in shear, leading to a nonlinear stress-strain relation of the form

$$\gamma_{xy} = G_{xy}^{-1}\sigma_{xy} + \alpha\sigma_{xy}^{3},$$

where G_{xy} is the (initial) ply shear modulus and the nonlinearity is characterized by the factor α=2.44×10^{-8} MPa^{-3} (0.8×1^{-5} ksi^{-3}).

Failure modes in laminated composites are strongly dependent on geometry, loading direction, and ply orientation. Typically, one distinguishes in-plane failure modes and transverse failure modes (associated with interlaminar shear or peel stress). Since this composite is loaded in-plane, only in-plane failure modes need to be considered, which can be done for each ply individually. For a unidirectional ply as used here, five failure modes can be considered: matrix tensile cracking, matrix compression, fiber breakage, fiber matrix shearing, and fiber buckling. All the mechanisms, with the exception of fiber breakage, can cause compression failure in laminated composites.

The failure strength in laminates also depends on the ply layup. The effective failure strength of the layup is at a maximum if neighboring plies are orthogonal to each other. The effective strength decreases as the angle between plies decreases and is at a minimum if plies have the same direction. (This is called a ply cluster.) Chang and Lessard have obtained some empirical formulas for the effective transverse tensile strength; however, in this model we ignore such effects. Instead, we use the following strength properties for the T300/976 laminate: transverse tensile strength Y_t=102.4 MPa (14.86 ksi), ply shear strength S_c=106.9 MPa (15.5 ksi), matrix compressive strength Y_c=253.0 MPa (36.7 ksi), and fiber buckling strength X_c=2707.6 MPa (392.7 ksi).

The strength parameters can be combined into failure criteria for multiaxial loading. Four different failure modes are considered in the model analyzed here.

- *Matrix tensile cracking* can result from a combination of transverse tensile stress, σ_y, and shear stress, σ_{xy}.

  The failure index, e_m, can be defined in terms of these stresses and the strength parameters, Y_t and S_c. When the index exceeds 1.0, failure is assumed to occur. Without nonlinear material behavior, the failure index has the simple form,

  $$e_m^2 = \left(\frac{\sigma_y}{Y_t}\right)^2 + \left(\frac{\sigma_{xy}}{S_c}\right)^2.$$

  With nonlinear shear behavior taken into consideration, the failure index takes the more complex form,

  $$e_m^2 = \left(\frac{\sigma_y}{Y_t}\right)^2 + \frac{2\sigma_{xy}^2/G_{xy}+3\alpha\sigma_{xy}^4}{2S_c^2/G_{xy}+3\alpha S_c^4}.$$

- *Matrix compressive failure* results from a combination of transverse compressive stress and shear stress. The failure criterion has the same form as that for matrix tensile cracking:

  $$e_m^2 = \left(\frac{\sigma_y}{Y_c}\right)^2 + \frac{2\sigma_{xy}^2/G_{xy}+3\alpha\sigma_{xy}^4}{2S_c^2/G_{xy}+3\alpha S_c^4}.$$

  The same failure index is used since the previous two failure mechanisms cannot occur simultaneously at the same point. After the failure index exceeds 1.0, both the transverse stiffness and Poisson's ratio of the ply drop to zero.

- *Fiber-matrix shearing failure* results from a combination of fiber compression and matrix shearing. The failure criterion has essentially the same form as the other two criteria:

  $$e_{fs}^2 = \left(\frac{\sigma_x}{X_c}\right)^2 + \frac{2\sigma_{xy}^2/G_{xy}+3\alpha\sigma_{xy}^4}{2S_c^2/G_{xy}+3\alpha S_c^4}.$$

  This mechanism can occur simultaneously with the other two criteria; hence, a separate failure index is used. Shear stresses are no longer supported after the failure index exceeds 1.0, but direct stresses in the fiber and transverse directions continue to be supported.

- *Fiber buckling failure* occurs when the maximum compressive stress in the fiber direction ($-\sigma_x$) exceeds the fiber buckling strength, X_c, independent of the other stress components:

  $$e_b = -\frac{\sigma_x}{X_c}.$$

  It is obvious that, unless the shear stress vanishes exactly, fiber-matrix shearing failure occurs prior to fiber buckling. However, fiber buckling may follow subsequent to fiber shearing because only the shear stiffness degrades after fiber-matrix shearing failure. Fiber buckling in a layer is a catastrophic mode of failure. Hence, after this failure index exceeds 1.0, it is assumed that the material at this point can no longer support any loads.

In this example the primary loading mode is shear. Therefore, failure of the plate occurs well before the fiber stresses can develop to a level where fiber buckling takes place, and this failure mode need not be taken into consideration.

Chang and Lessard assume that after failure occurs, the stresses in the failed directions drop to zero immediately, which corresponds to brittle failure with no energy absorption. This kind of failure model usually leads to immediate, unstable failure of the composite. This assumption is not very realistic: in reality, the stress-carrying capacity degrades gradually with increasing strain after failure occurs. Hence, the behavior of the composite after onset of failure is not likely to be captured well by this model. Moreover, the instantaneous loss of stress-carrying capacity also makes the postfailure analysis results strongly dependent on the refinement of the finite element mesh and the finite element type used.

Material model implementation

To simulate the shear nonlinearity and the failure modes (matrix failure in tension or compression and fiber-matrix shear failure), the elastic properties are made linearly dependent on three field variables. The first field variable represents the matrix failure index, the second represents the fiber-matrix shear failure index, and the third represents the shear nonlinearity (damage) prior to failure. The dependence of the elastic material properties on the field variables is shown in *Table 1*.

To account for the nonlinearity, the nonlinear stress-strain relation must be expressed in a different form: the stress at the end of the increment must be given as a linear function of the strain. The most obvious way to do this is to linearize the nonlinear term, leading to the relation

$$\gamma_{xy}^{(i+1)} = \left(G_{xy}^{-1} + \alpha \left(\sigma_{xy}^{(i)} \right)^2 \right) \sigma_{xy}^{(i+1)},$$

where i represents the increment number. This relation can be written in inverted form as

$$\sigma_{xy}^{(i+1)} = \frac{G_{xy}}{1+\alpha G_{xy} \left(\sigma_{xy}^{(i)} \right)^2} \gamma_{xy}^{(i+1)},$$

thus providing an algorithm to define the effective shear modulus.

However, this algorithm is not very suitable because it is unstable at higher strain levels, which is readily demonstrated by stability analysis. Consider an increment where the strain does not change; i.e., $\gamma_{xy}^{(i+1)} = \gamma_{xy}^{(i)} = \gamma_{xy}$. Let the stress at increment i have a small perturbation from $\sigma_{xy}^{e\,(i)}$, the exact solution at that increment: $\sigma_{xy}^{(i)} = \sigma_{xy}^{e\,(i)} + \Delta\sigma_{xy}^{(i)}$. Similarly, at increment i+1, $\sigma_{xy}^{(i+1)} = \sigma_{xy}^{e\,(i+1)} + \Delta\sigma_{xy}^{(i+1)}$. For the algorithm to be stable, $\Delta\sigma_{xy}^{(i+1)}$ should not be larger than $\Delta\sigma_{xy}^{(i)}$. The perturbation in increment i+1 is calculated by substituting $\sigma_{xy}^{(i)}$ in the effective shear modulus equation and linearizing it about $\sigma_{xy}^{e\,(i)}$:

$$\Delta\sigma_{xy}^{(i+1)} = \frac{-2\alpha G_{xy}^2 \sigma_{xy} \gamma_{xy}}{\left(1+\alpha G_{xy} \sigma_{xy}^2\right)^2} \Delta\sigma_{xy}^{(i)},$$

where $\sigma_{xy} = \sigma_{xy}^{e\,(i)}$. The perturbation in increment i+1 is larger than the perturbation in increment i if

$$2\alpha G_{xy}^2 \sigma_{xy} \gamma_{xy} > \left(1 + \alpha G_{xy} \sigma_{xy}^2\right)^2,$$

which, after elimination of γ_{xy}, reduces to the expression

$$\alpha \sigma_{xy}^3 > G_{xy}^{-1} \sigma_{xy}.$$

Hence, instability occurs when the “nonlinear” part of the shear strain is larger than the “linear” part of the shear strain.

To obtain a more stable algorithm, we write the nonlinear stress-strain law in the form

$$\gamma_{xy} + \beta \sigma_{xy}^3 = G_{xy}^{-1} \sigma_{xy} + (\alpha + \beta) \sigma_{xy}^3,$$

where β is an as yet unknown coefficient. In linearized form this leads to the update algorithm

$$\left(1 + \beta \left(\sigma_{xy}^{(i)} \right)^3 / \gamma_{xy}^{(i)} \right) \gamma_{xy}^{(i+1)} = \left(G_{xy}^{-1} + (\alpha + \beta) \left(\sigma_{xy}^{(i)} \right)^2 \right) \sigma_{xy}^{(i+1)},$$

or, in inverted form,

$$\sigma_{xy}^{(i+1)} = \frac{1+\beta\left(\sigma_{xy}^{(i)}\right)^3/\gamma_{xy}^{(i)}}{1+(\alpha+\beta)G_{xy}\left(\sigma_{xy}^{(i)}\right)^2} G_{xy}\gamma_{xy}^{(i+1)}.$$

Following the same procedure as that for the original update algorithm, it is readily derived that a small perturbation, $\Delta\sigma_{xy}^{(i)}$, in increment i reduces to zero in increment i+1 if $\beta = 2\alpha$. Hence, the optimal algorithm appears to be

$$\sigma_{xy}^{(i+1)} = \frac{1+2\alpha\left(\sigma_{xy}^{(i)}\right)^3/\gamma_{xy}^{(i)}}{1+3\alpha G_{xy}\left(\sigma_{xy}^{(i)}\right)^2} G_{xy}\gamma_{xy}^{(i+1)}.$$

Finally, this relation is written in terms of the damage parameter d:

$$\sigma_{xy}^{(i+1)} = (1-d)\, G_{xy}\gamma_{xy}^{(i+1)},$$

where

$$d = \frac{3\alpha G_{xy}\left(\sigma_{xy}^{(i)}\right)^2 - 2\alpha\left(\sigma_{xy}^{(i)}\right)^3/\gamma_{xy}^{(i)}}{1+3\alpha G_{xy}\left(\sigma_{xy}^{(i)}\right)^2}.$$

This relation is implemented in user subroutines `USDFLD` and `VUSDFLD`, and the value of the damage parameter is assigned directly to the third field variable used for definition of the elastic properties.

The failure indices are calculated with the expressions discussed earlier, based on the stresses at the start of the increment:

$$e_m^2 = \left(\frac{\sigma_y^{(i)}}{Y_t}\right)^2 + \frac{2\left(\sigma_{xy}^{(i)}\right)^2/G_{xy} + 3\alpha\left(\sigma_{xy}^{(i)}\right)^4}{2S_c^2/G_{xy} + 3\alpha S_c^4} \quad \text{if} \quad \sigma_y^{(i)} > 0,$$

$$e_m^2 = \left(\frac{\sigma_y^{(i)}}{Y_c}\right)^2 + \frac{2\left(\sigma_{xy}^{(i)}\right)^2/G_{xy} + 3\alpha\left(\sigma_{xy}^{(i)}\right)^4}{2S_c^2/G_{xy} + 3\alpha S_c^4} \quad \text{if} \quad \sigma_y^{(i)} < 0,$$

$$e_{fs}^2 = \left(\frac{\sigma_x^{(i)}}{X_c}\right)^2 + \frac{2\left(\sigma_{xy}^{(i)}\right)^2/G_{xy} + 3\alpha\left(\sigma_{xy}^{(i)}\right)^4}{2S_c^2/G_{xy} + 3\alpha S_c^4}.$$

The values of the failure indices are not assigned directly to the field variables: instead, they are stored as solution-dependent state variables. Only if the value of a failure index exceeds 1.0 is the corresponding user-defined field variable set equal to 1.0. After the failure index has exceeded 1.0, the associated user-defined field variable continues to have the value 1.0 even though the stresses may reduce significantly, which ensures that the material does not "heal" after it has become damaged.

Finite element model

The plate consists of 24 plies of T300/976 graphite-epoxy in a $[(-45/+45)_6]_s$ layup. Instead of modeling each ply individually, we combine all plies in the −45° direction and all plies in the +45° direction. Consequently, only two layers need to be modeled separately:

1. A layer in the −45° direction with a thickness of 1.715 mm (0.0675 in).
2. A layer in the +45° direction with a thickness of 1.715 mm (0.0675 in).

The corresponding finite element model consists of two layers of CPS4 plane stress elements, with thicknesses and properties as previously discussed. The quarter-symmetry finite element model is shown in *Figure 1*.

The implementation of nonlinear material behavior with user-defined field variables is *explicit*: the nonlinearity is based on the state at the start of the increment. Hence, in Abaqus/Standard analyses the user must ensure that the time increments are sufficiently small, which is particularly important because the automatic time increment control in Abaqus/Standard is ineffective with the explicit nonlinearity implemented in *USDFLD*. If automatic time incrementation is used, the maximum time increment can be controlled from within subroutine *USDFLD* with the variable `PNEWDT`. This capability is useful if there are other nonlinearities that require automatic time incrementation. In this example the only significant nonlinearity is the result of the material behavior. Hence, fixed time incrementation can be used effectively. In Abaqus/Explicit analyses the stable time increment is usually sufficiently small to ensure good accuracy.

Results and discussion

For this problem experimental load-displacement results were obtained by Chang and Lessard. The experimental results, together with the numerical results obtained with Abaqus/Standard, are shown in *Figure 2*. The agreement between the experimental and numerical results is excellent up to the point where the load maximum is reached. After that, the numerical load-displacement curve drops off sharply, whereas the experimental data indicate that the load remains more or less constant. Chang and Lessard also show numerical results: their results agree with the results obtained by Abaqus but do not extend to the region where the load drops off. The dominant failure mode in this plate is fiber/matrix shear: failure occurs first at a load of approximately 12.15 kN (2700 lbs) and continues to grow in a stable manner until a load of approximately 13.5 kN (3000 lbs) is reached. *Figure 3* shows the extent of the damage in the Abaqus/Standard finite element model at the point of maximum load. In this figure an element is shaded if fiber/matrix shear failure has occurred at at least three integration points. These results also show excellent agreement with the results obtained by Chang and Lessard.

As discussed earlier, the sharp load drop-off in the numerical results is the result of the lack of residual stress carrying capacity after the failure criterion is exceeded. Better agreement could be reached only if postfailure material data were available. Without postfailure data the results are very sensitive to the mesh and element type, which is clearly demonstrated by changing the element type from CPS4 (full integration) to CPS4R (reduced integration). The results are virtually identical up to the point where first failure occurs. After that point the damage in the CPS4R model spreads more rapidly than in the CPS4 model until a maximum load of about 12.6 kN (2800 lbs) is reached. The load then drops off rapidly.

The problem is also analyzed with Abaqus/Standard models consisting of S4R and S4 elements. The elements have a composite section with two layers, with each layer thickness equal to the thickness of the plane stress elements in the CPS4 and CPS4R models. The results that were obtained with the S4R and S4 element models are indistinguishable from those obtained with the CPS4R element model.

The numerical results obtained with Abaqus/Explicit using the CPS4R element model (not shown) are consistent with those obtained with Abaqus/Standard.

Input files

Abaqus/Standard input files

damagefailcomplate_cps4.inp	CPS4 elements.
damagefailcomplate_cps4.f	User subroutine *USDFLD* used in damagefailcomplate_cps4.inp.
damagefailcomplate_node.inp	Node definitions.
damagefailcomplate_element.inp	Element definitions.
damagefailcomplate_cps4r.inp	CPS4R elements.
damagefailcomplate_cps4r.f	User subroutine *USDFLD* used in damagefailcomplate_cps4r.inp.
damagefailcomplate_s4.inp	S4 elements.

damagefailcomplate_s4.f	User subroutine *USDFLD* used in damagefailcomplate_s4.inp.
damagefailcomplate_s4r.inp	S4R elements.
damagefailcomplate_s4r.f	User subroutine *USDFLD* used in damagefailcomplate_s4r.inp.

Abaqus/Explicit input files

damagefailcomplate_cps4r_xpl.inp	CPS4R elements.
damagefailcomplate_cps4r_xpl.f	User subroutine *VUSDFLD* used in damagefailcomplate_cps4r_xpl.inp.
damagefailcomplate_node.inp	Node definitions.
damagefailcomplate_element.inp	Element definitions.

References

- Chang, F-K., and L. B. Lessard, "Damage Tolerance of Laminated Composites Containing an Open Hole and Subjected to Compressive Loadings: Part I—Analysis," *Journal of Composite Materials*, vol. 25, pp. 2–43, 1991.

Tables

Table 1: Dependence of the elastic material properties on the field variables.

Material State	Elastic Properties				FV1	FV2	FV3
No failure	E_x	E_y	ν_{xy}	G_{xy}	0	0	0
Matrix failure	E_x	0	0	G_{xy}	1	0	0
Fiber/matrix shear	E_x	E_y	0	0	0	1	0
Shear damage	E_x	E_y	ν_{xy}	0	0	0	1
Matrix failure and fiber/matrix shear	E_x	0	0	0	1	1	0
Matrix failure and shear damage	E_x	0	0	0	1	0	1
Fiber/matrix shear and shear damage	E_x	E_y	0	0	0	1	1
All failure modes	E_x	0	0	0	1	1	1

Figures

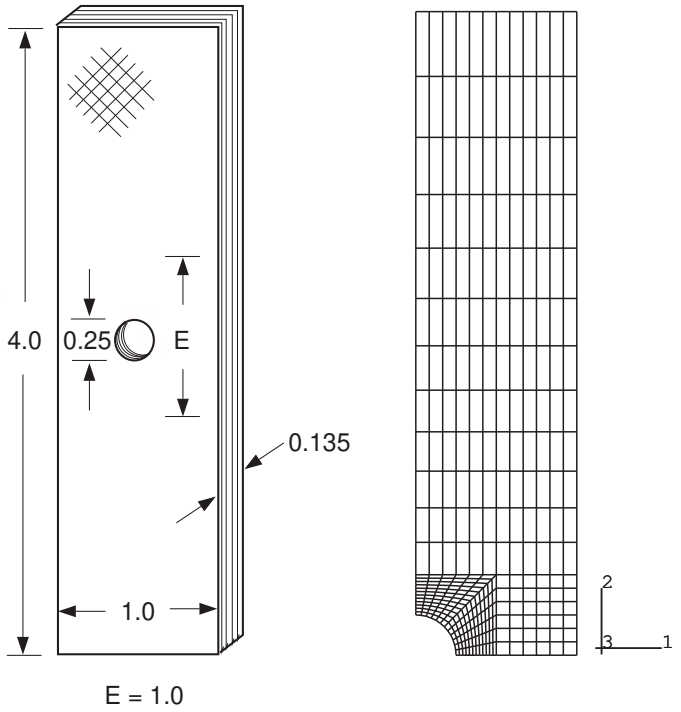


Figure 1: Plate geometry.

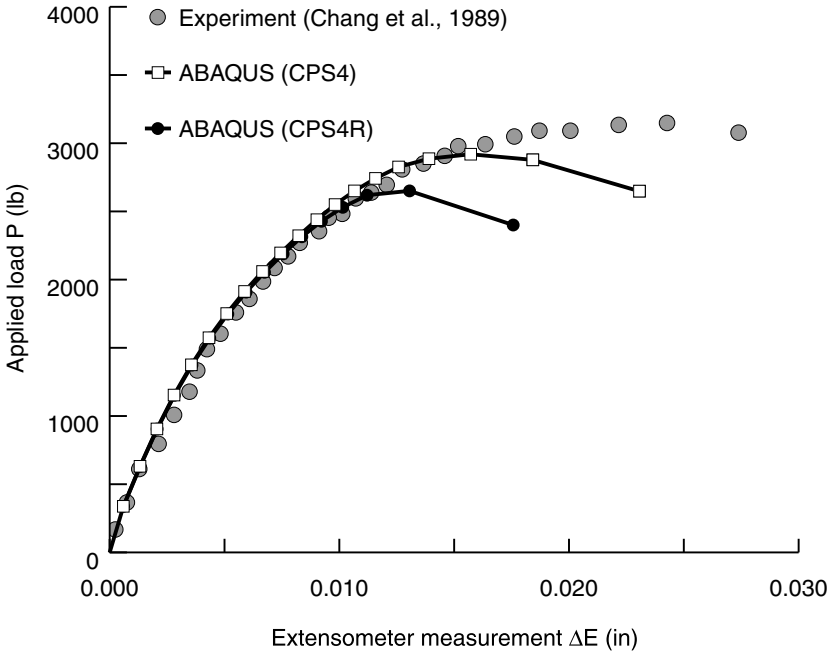


Figure 2: Experimental and numerical (Abaqus/Standard) load displacement curves.

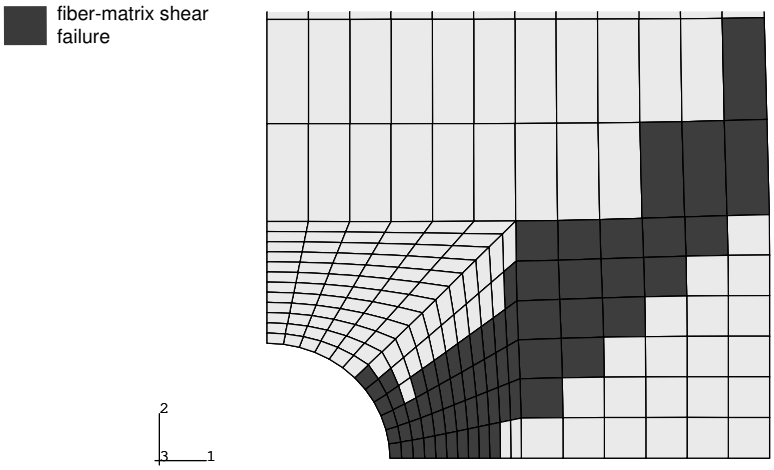


Figure 3: Distribution of material damage at maximum load obtained with Abaqus/Standard.

Analysis of an automotive boot seal

Products: Abaqus/Standard

Overview

This example shows the deformation of a boot seal caused by a typical angular movement of the shaft. It provides a demonstration and verification of the finite-sliding capability in three-dimensional deformable-to-deformable contact and self-contact in Abaqus. This problem also demonstrates how to model a hyperelastic material using the *UMAT* user subroutine.

Boot seals are used to protect constant velocity joints and steering mechanisms in automobiles. These flexible components must accommodate the motions associated with angulation of the steering mechanism. Some regions of the boot seal are always in contact with an internal metal shaft, while other areas come into contact with the metal shaft during angulation. In addition, the boot seal might also come into contact with itself, both internally and externally. The contacting regions affect the performance and longevity of the boot seal.

Geometry and model

The boot seal with the internal shaft is shown in *Figure 1*. The corrugated shape of the boot seal tightly grips the steering shaft at one end, while the other end is fixed. The rubber seal is modeled with first-order, hybrid brick elements with two elements through the thickness using symmetric model generation. The seal has a nonuniform thickness varying from a minimum of 3.0 mm to a maximum of 4.75 mm at the fixed end. The internal shaft is considered to be rigid and is modeled as an analytical rigid surface; the radius of the shaft is 14 mm. The rigid body reference node is located precisely in the center of the constant velocity joint.

The rubber is modeled as a slightly compressible neo-Hookean (hyperelastic) material with C_{10}=0.752 MPa and D_1=0.026 MPa^{-1}. For illustrative purposes, input files using the Marlow and the Valanis-Landel models are included; the models are defined using uniaxial test data generated by running a uniaxial test with the neo-Hookean model.

Contact is specified between the rigid shaft and the inner surface of the seal. Self-contact is specified on the inner and outer surfaces of the seal.

Loading

The mounting of the boot seal and the angulation of the shaft are carried out in a three-step analysis. The inner radius at the neck of the boot seal is smaller than the radius of the shaft so as to provide a tight fit between the seal and the shaft. In the first step the initial interference fit is resolved, corresponding to the assembly process of mounting the boot seal onto the shaft. The automatic “shrink” fit method is utilized. The second step simulates the angulation of the shaft by specifying a finite rotation of 20° at the rigid body reference node of the shaft. During the third step, the angulated shaft travels around the entire circumference to demonstrate the robustness of the algorithm.

User subroutine for neo-Hookean hyperelasticity

In Abaqus/Standard user subroutine *UHYPER* is used to define a hyperelastic material. However, in this problem we illustrate the use of user subroutine *UMAT* as an alternative method of defining a hyperelastic material. In particular, we consider the neo-Hookean hyperelastic material model. The form of the neo-Hookean strain energy density function is given by

$$U\left(\overline{I}_1, \overline{I}_2, J\right) = C_{10}\left(\overline{I}_1 - 3\right) + \frac{1}{D_1}(J-1)^2.$$

Here, $\overline{I}_1$, $\overline{I}_2$, and J are the strain invariants of the deviatoric left Cauchy-Green deformation tensor $\overline{\mathbf{B}}$. This tensor is defined as $\overline{\mathbf{B}} \stackrel{\text{def}}{=} \overline{\mathbf{F}} \cdot \overline{\mathbf{F}}^T$, where $\overline{\mathbf{F}} = J^{-\frac{1}{3}}\mathbf{F}$ is the distortion gradient. *Hyperelastic material behavior* contains detailed explanations of these quantities.

The constitutive equation for a neo-Hookean material is

$$\boldsymbol{\sigma} = \frac{2}{J} C_{10}\left(\overline{\mathbf{B}} - \frac{1}{3}\,\text{trace}\,\left(\overline{\mathbf{B}}\right)\mathbf{I}\right) + \frac{2}{D_1}\left(J-1\right)\mathbf{I},$$

where $\boldsymbol{\sigma}$ is the Cauchy stress. The material Jacobian, $\mathbf{C}$, is defined by the variation of the Kirchhoff stress

$$\delta\left(J\boldsymbol{\sigma}\right) = J\mathbf{C}\cdot\delta\mathbf{D},$$

where $\delta\mathbf{D}$ is the virtual rate of deformation and is defined as

$$\delta\mathbf{D} \stackrel{\text{def}}{=} \text{sym}\,\left(\delta\mathbf{F}\cdot\mathbf{F}^{-1}\right).$$

For a neo-Hookean material the components of $\mathbf{C}$ are given by

$$C_{ijkl} = \frac{2}{J}C_{10}\left(\frac{1}{2}\left(\delta_{ik}\overline{B}_{jl} + \overline{B}_{ik}\delta_{jl} + \delta_{il}\overline{B}_{jk} + \overline{B}_{il}\delta_{jk}\right) - \frac{2}{3}\delta_{ij}\overline{B}_{kl} - \frac{2}{3}\overline{B}_{ij}\delta_{kl} + \frac{2}{9}\delta_{ij}\delta_{kl}\overline{B}_{mm}\right) + \frac{2}{D_1}\left(2J-1\right)\delta_{ij}\delta_{kl}.$$

Results and discussion

Figure 2 shows the deformed configuration of the model. The rotation of the shaft causes the stretching of one side and compression on the other side of the boot seal. The surfaces have come into self-contact on the compressed side. *Figure 3* shows the contours of maximum principal stresses in the boot seal.

Comparison of the analysis times when using fixed and automated contact patches shows that both analyses complete in approximately the same amount of time. This can be expected for this type of problem since the fixed contact patches are limited in size to a few elements. For the case with fixed contact patches the wavefront is somewhat larger, requiring more memory and solution time per iteration. However, this is offset by the time required to form new contact patches and to reorder the equations for the case with automatic contact patches. The results obtained with the model that uses user subroutine *UMAT* are identical to those obtained using the built-in Abaqus material model.

Input files

bootseal.inp	Analysis with node-to-surface contact.
bootseal_surf.inp	Analysis with surface-to-surface contact.
bootseal_2d.inp	Two-dimensional model for symmetric model generation in bootseal.inp.
bootseal_2d_surf.inp	Two-dimensional model for symmetric model generation in bootseal.inp using surface-to-surface contact.
bootseal_umat.inp	Analysis with user subroutine *UMAT*.
bootseal_2d_umat.inp	Two-dimensional model for symmetric model generation in bootseal_umat.inp.
bootseal_umat.f	*UMAT* for the neo-Hookean hyperelasticity model.

bootseal_marlow.inp	Analysis with Marlow hyperelasticity model.
bootseal_2d_marlow.inp	Two-dimensional model for symmetric model generation in bootseal_marlow.inp.
bootseal_valanis_landel.inp	Analysis with Valanis-Landel hyperelasticity model.
bootseal_2d_valanis_landel.inp	Two-dimensional model for symmetric model generation in bootseal_valanis_landel.inp.

Figures

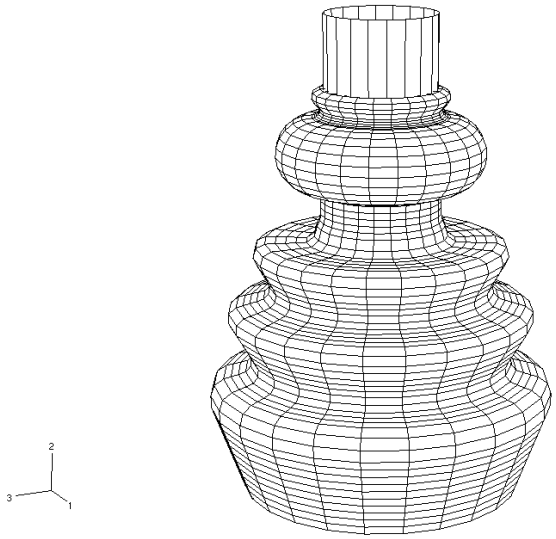


Figure 1: Undeformed model.

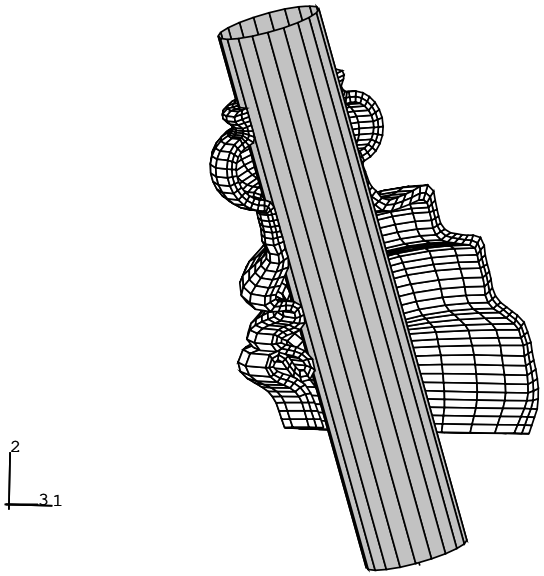


Figure 2: Deformed configuration of half the model.

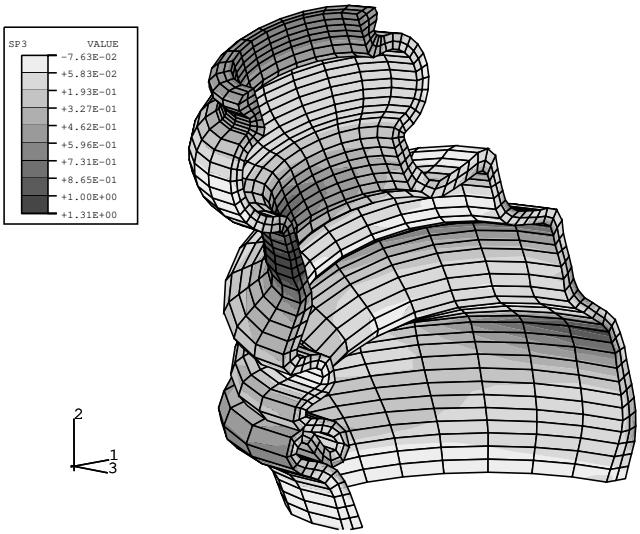


Figure 3: Contours of maximum principal stress in the seal.

Pressure penetration analysis of an air duct kiss seal

Products: Abaqus/Standard

Overview

This example shows how nonlinear finite element analysis of an air duct kiss seal can be used to determine the seal performance.
Seals are common structural components that often require design analyses. Information such as a load-deflection curve, seal deformation and stresses, and contact pressure distribution is readily obtained from these analyses. Abaqus allows for pressure penetration effects between the seal and the contacting surfaces to be considered in these analyses, making routine analyses of seals more realistic and accurate. Analyses of clutch seals, threaded connectors, car door seals, and air duct kiss seals are some applications where pressure penetration effects are important.

The surface-based pressure penetration capability is used to simulate pressure penetration between contacting surfaces. It is invoked by using the pressure penetration option, which is described in *Fluid Pressure Penetration Loads*. This capability is provided for simulating cases where a joint between two deforming bodies (for example, between two components threaded onto each other) or between a deforming body and a rigid surface (such as a soft gasket used in a joint) is exposed at one or multiple ends to a fluid or air pressure. This air pressure will penetrate into the joint and load the surfaces forming the joint until some area of the surfaces is reached where the contact pressure between the abutting surfaces exceeds the critical value specified on the pressure penetration option, cutting off further penetration.

Geometry and model

The major consideration in an air duct kiss seal design is to provide sealing while avoiding excessive closure force. A poorly designed air duct seal that minimizes the amount of effort to close the fan cowl door may fail to prevent leakage and reduce wind noise. The model used in this example is a simplified version of an air duct kiss seal. It illustrates how pressure penetration effects can be modeled using Abaqus.

The seal modeled is a rolled shape seal. An axisymmetric model of the seal is developed first, as shown in *Figure 1*. A three-dimensional model of the seal is also developed with only a 5-degree section of the seal discretized, as shown in *Figure 2*. The top horizontal rigid surface represents the air fan cowl door, and the bottom horizontal rigid surface represents the seal groove. The rolled seal is 2.54 mm (0.1 in) thick and 74.66 mm (2.9 in) high; and its inner diameters at the top and bottom surfaces are 508.5 mm (20 in) and 528.3 mm (20.8 in), respectively. A folded metal clip is partially bonded to the top surface of the seal. The thickness of the metal clip is 0.48 mm (0.019 in).

The material of the seal is taken to be an incompressible rubberlike material. To obtain the material constants, the Ogden form of the strain energy function with $N = 4$ is used to fit the uniaxial test data. The metal clip is made of steel, with a Young's modulus of 206.8 GPa (3.0×10^7 lb/in^2) and a Poisson's ratio of 0.3. CAX4H elements are used to model the seal and the metal clip in the axisymmetric model, and C3D8H elements are used in the three-dimensional model. The contact pair approach is used to model the contact between the top surface of the metal clip and the top rigid surface representing the fan cowl door, where the pressure penetration is likely to occur. The contact pair approach is also used to model the contact between the seal and the bottom rigid surface, the contact between the seal and the unbonded portion of the metal clip, and the self-contact of the seal. The mechanical interaction between the contact surfaces is assumed to be frictional contact. Therefore, the friction option is used to specify friction coefficients. To increase computational efficiency, the slip tolerance (the ratio of allowable maximum elastic slip to characteristic contact surface face dimension) on the friction option is specified for the contact surfaces between the seal and the metal clip because the dimensions of these elements vary greatly. Fixed boundary conditions are applied initially to the reference node of the top rigid surface, 5001, and the reference node of the bottom rigid surface, 5002. The vertical edge at the bottom of the seal is constrained such that it cannot be moved in the 1-direction. The bottom node of the vertical edge, 1,

touches the bottom rigid surface and is held fixed in the 2-direction. The top rigid surface is located initially 1.27 mm (0.05 in) above the top surface of the metal clip.

The seal and the unbonded portion of the clip are loaded by air pressure on all of their inner surfaces and by contact pressure generated by closing the air fan cowl door. Two nonlinear static steps, all of which include large-displacement effects, are used to simulate these loading conditions.

In the first step the top rigid surface moves 35.56 mm (1.4 in) downward in the *y*-direction, simulating the closing of the fan cowl door.

In the second step the inner surface of the seal is subjected to a uniform air pressure load of 206.8 KPa (30.0 lb/in^2) since some gaps between the seal and the top rigid surface have been closed. The pressure penetration is simulated between the top surface of the metal clip (`PPRES`), which includes 31 elements, and the top rigid surface (`CFACE`). Air pressure penetration does not need to be modeled between the metal clip and the seal because they are well bonded.

The pressure penetration option is invoked to define the node exposed to the air pressure, the magnitude of the air pressure, and the critical contact pressure. The surface `PPRES` is exposed to the air pressure at node 597, with a pressure magnitude of 206.8 KPa (30.0 lb/in^2). A default value of zero for the critical contact pressure is used, indicating that the pressure penetration occurs only when contact at a secondary node is lost.

Results and discussion

The deformed configuration and the contours of the contact pressures on the seal at the end of Step 1 are shown in *Figure 3* and *Figure 4* for the axisymmetric model and in *Figure 5* for the three-dimensional model. A nonuniform contact pressure is observed along the surface of the seal. The contact pressure at the first five secondary nodes is zero.

The penetrating pressure loads are applied during Step 2. The air pressure is applied immediately to elements associated with the first five secondary nodes since the contact pressure there is zero and the pressure penetration criterion is satisfied. For the axisymmetric model the spread of the penetration is captured in *Figure 6* through *Figure 14*, which show the deformed seal, the contact pressure profile, and the air pressure profiles corresponding to load increments 2, 10, and 16 of Step 2. The pressures applied to the surface corresponding to these three increments are 1.296 KPa (0.188 lb/in^2), 13.96 KPa (2.03 lb/in^2), and 70.88 KPa (10.28 lb/in^2), respectively. For the three-dimensional model the spread of the penetration is captured in *Figure 15* through *Figure 17*, which show the contact pressure profiles corresponding to load increments 2, 6, and 14 of Step 2.

Increased penetrating pressure loads applied in Step 2 further reduce the contact pressure, eventually causing complete air penetration through the seal. The seal was lifted off from the air fan cowl door except at the last secondary node, 663, where the contact pressure is well maintained due to imposed boundary conditions and the air pressures. For the axisymmetric model the development of the weakening of the sealing is captured in *Figure 18* through *Figure 21*, which show the deformed seal and the contact pressure profile corresponding to load increment 20 and at the end of Step 2. The pressures applied to the surface corresponding to these two increments are 112.3 KPa (16.28 lb/in^2) and 206.8 KPa (30.0 lb/in^2), respectively. For the three-dimensional model the development of the weakening of the sealing is captured in *Figure 22* through *Figure 23*, which show the contact pressure profile corresponding to load increment 19 and at the end of Step 2.

The behavior of the seal throughout the loading histories can be best described by plotting the air penetration distance as a function of the air pressure, as shown in *Figure 24* for both axisymmetric and three-dimensional models. The air penetration distance in *Figure 24* is measured from the pressure front at the beginning of Step 2, which is the seventh tick mark point from the left side in *Figure 4*. It is clear that air penetration into the seal accelerates only when the pressure is on the order of 51.7 KPa (7.5 lb/in^2). The air completely penetrates through the seal when the pressure is 82.7 KPa (12.0 lb/in^2), which is approximately equal to 80% of the sea level atmospheric pressure.

In addition, the same model is analyzed with the adaptive automatic stabilization scheme, which improves the robustness by automatically adjusting the damping factor based on the convergence history while having very

little effect on the results. The dissipated stabilization energy is found to be small when the adaptive stabilization scheme is used.

Input files

presspenairductseal.inp	Pressure penetration simulation of an air duct kiss seal.
presspenairductseal_stabil_adap.inp	Same as presspenairductseal.inp with adaptive automatic stabilization.
presspenairductseal_node.inp	Node definitions for the seal model.
presspenairductseal_elem_metal.inp	Element definitions for the metal part of the seal model.
presspenairductseal_elem_rub.inp	Element definitions for the rubber part of the seal model.
presspenairductseal_c3d8h.inp	Pressure penetration simulation of an air duct kiss seal in three dimensions.

Figures

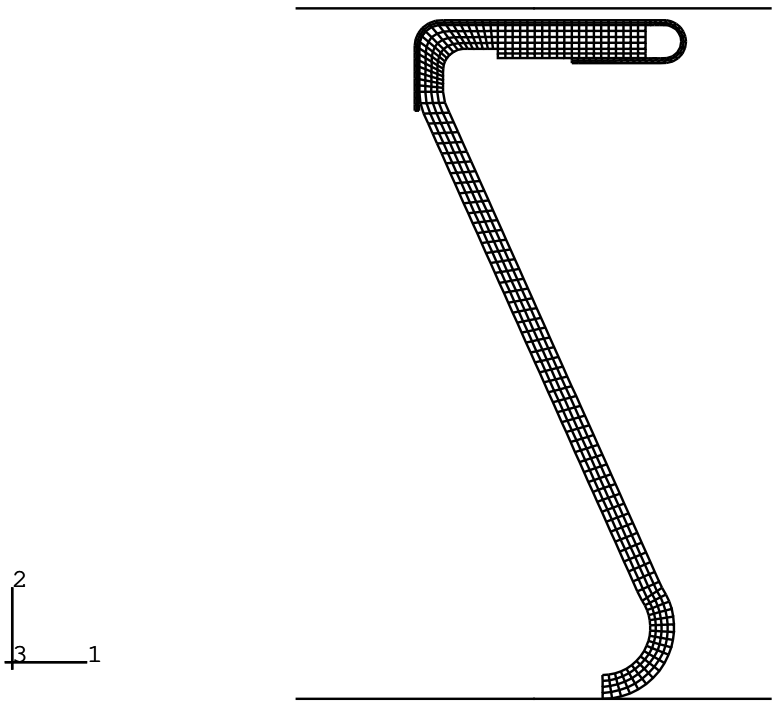


Figure 1: Axisymmetric model of air duct kiss seal.

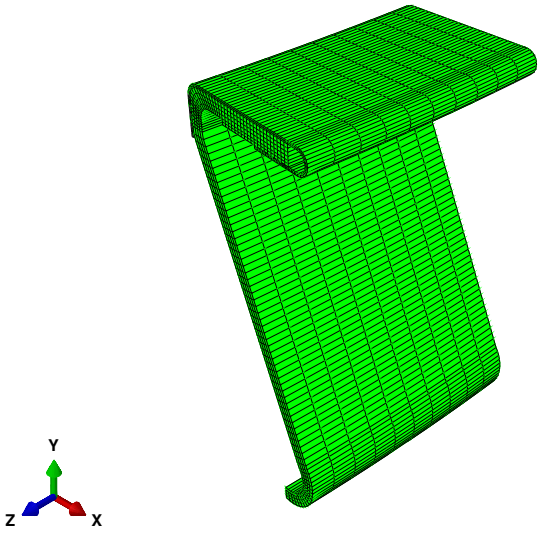


Figure 2: Three-dimensional model of air duct kiss seal.

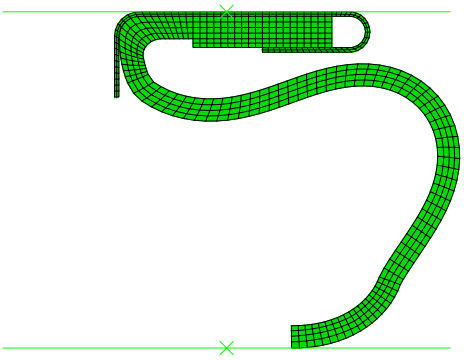


Figure 3: For the axisymmetric model, deformed configuration of the seal at the end of Step 1.

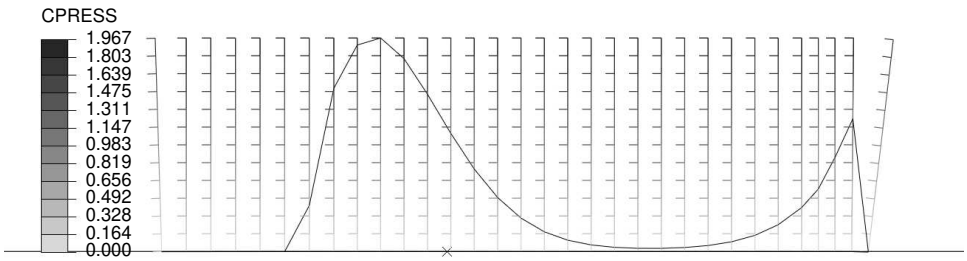


Figure 4: For the axisymmetric model, contact stress contours in the seal at the end of Step 1.

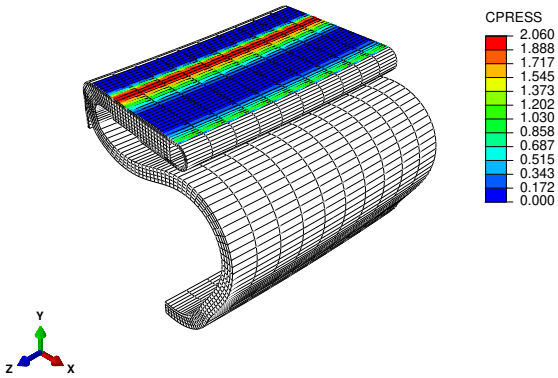


Figure 5: For the three-dimensional model, deformed configuration and contact stress contours of the seal at the end of Step 1.

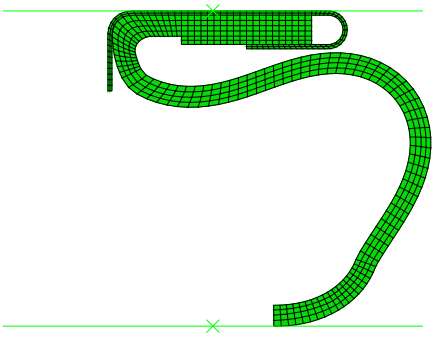


Figure 6: For the axisymmetric model, deformed configuration of the seal at Step 2, increment 2.

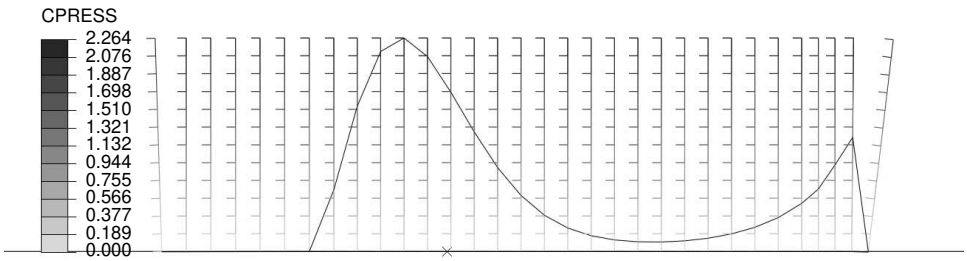


Figure 7: For the axisymmetric model, contact stress contours in the seal at Step 2, increment 2.

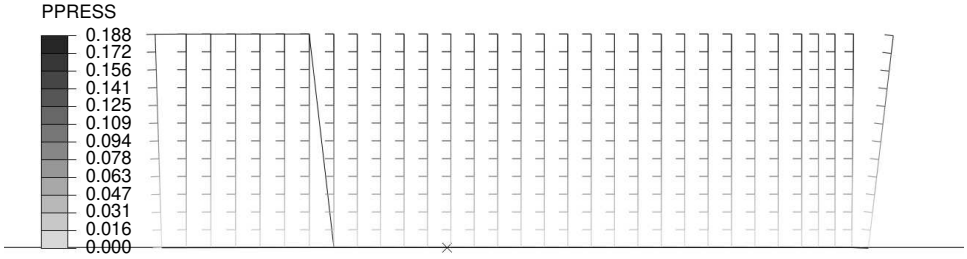


Figure 8: For the axisymmetric model, air pressure contours in the seal at Step 2, increment 2.

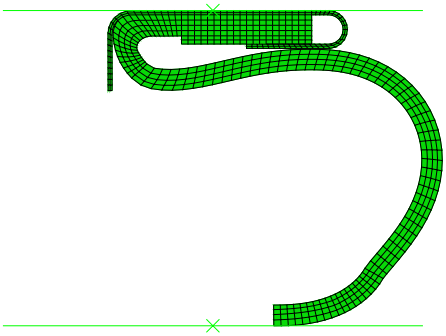


Figure 9: For the axisymmetric model, deformed configuration of the seal at Step 2, increment 10.

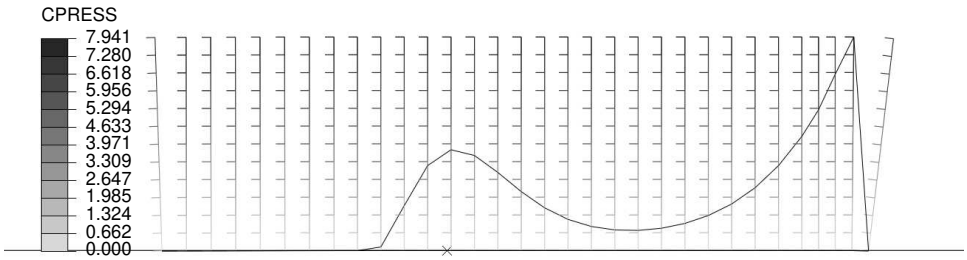


Figure 10: For the axisymmetric model, contact stress contours in the seal at Step 2, increment 10.

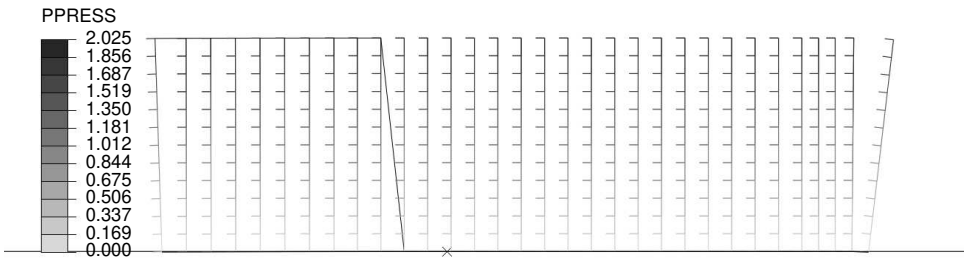


Figure 11: For the axisymmetric model, air pressure contours in the seal at Step 2, increment 10.

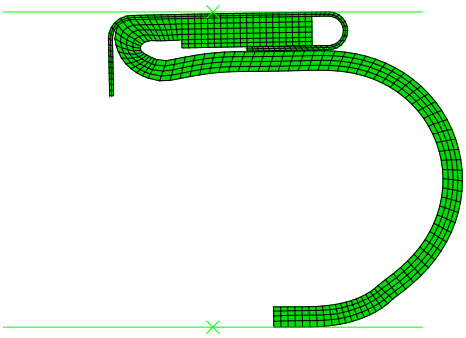


Figure 12: For the axisymmetric model, deformed configuration of the seal at Step 2, increment 16.

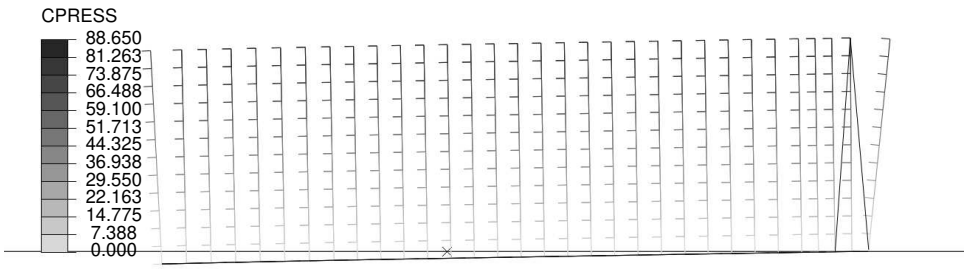


Figure 13: For the axisymmetric model, contact stress contours in the seal at Step 2, increment 16.

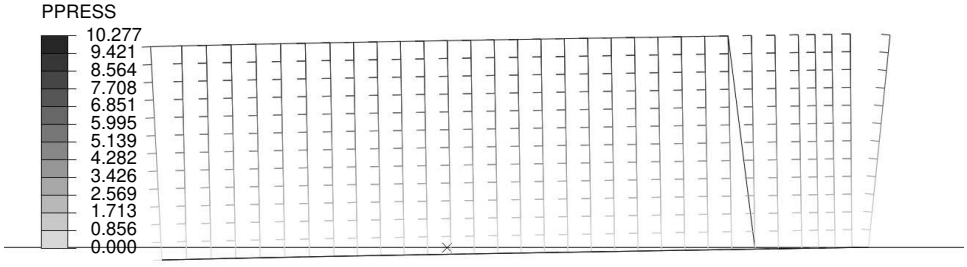


Figure 14: For the axisymmetric model, air pressure contours in the seal at Step 2, increment 16.

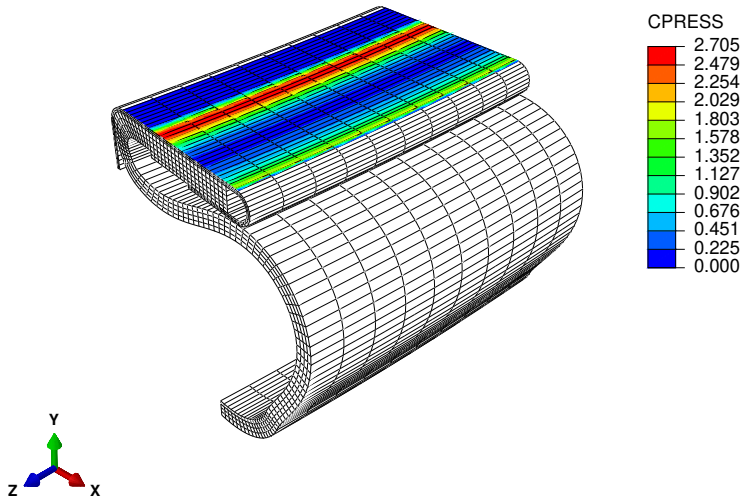


Figure 15: For the three-dimensional model, deformed configuration and contact stress contours of the seal at Step 2, increment 2.

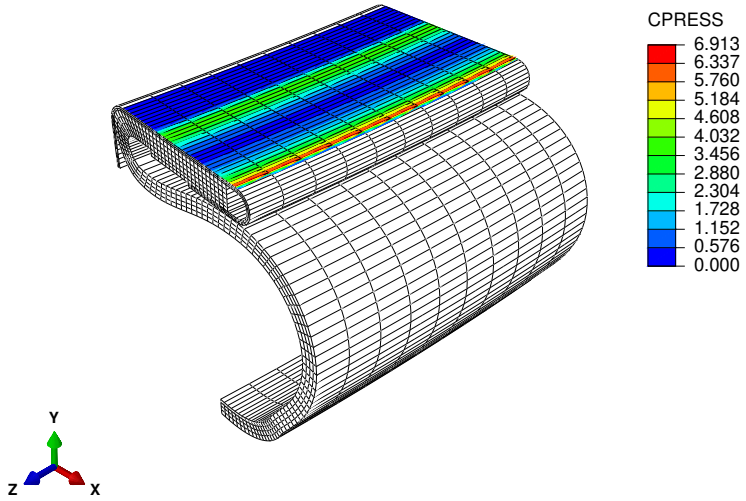


Figure 16: For the three-dimensional model, deformed configuration and contact stress contours of the seal at Step 2, increment 6.

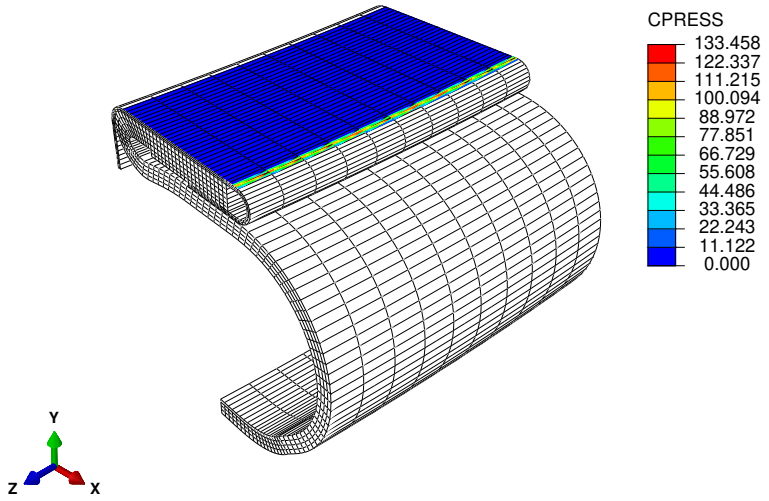


Figure 17: For the three-dimensional model, deformed configuration and contact stress contours of the seal at Step 2, increment 14.

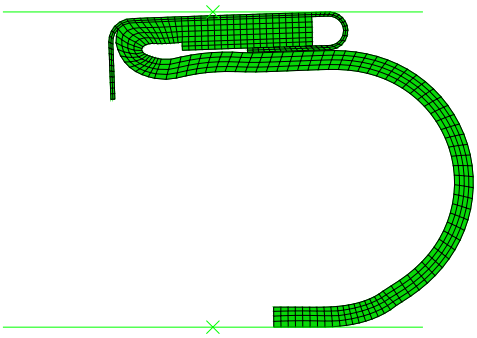


Figure 18: For the axisymmetric model, deformed configuration of the seal at Step 2, increment 20.

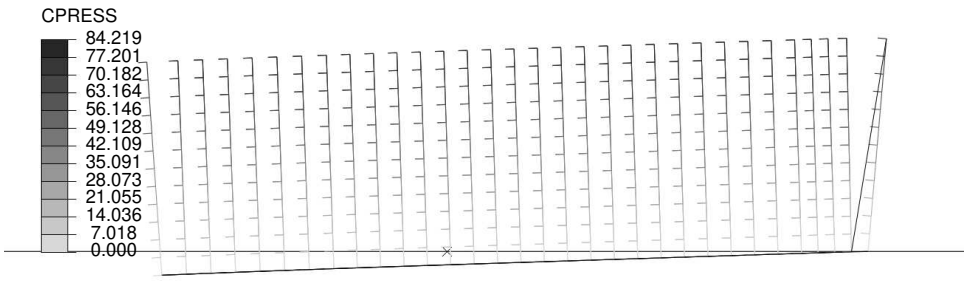


Figure 19: For the axisymmetric model, contact stress contours in the seal at Step 2, increment 20.

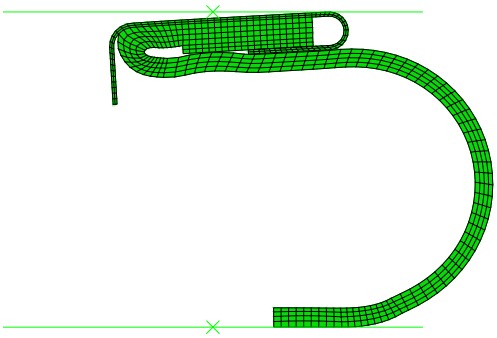


Figure 20: For the axisymmetric model, deformed configuration of the seal at the end of Step 2.

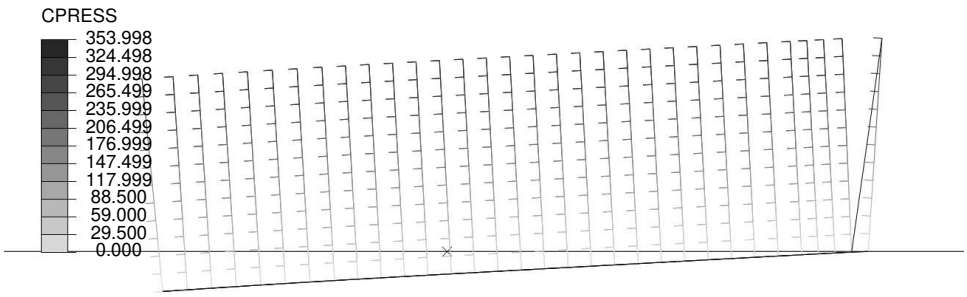


Figure 21: For the axisymmetric model, contact stress contours in the seal at the end of Step 2.

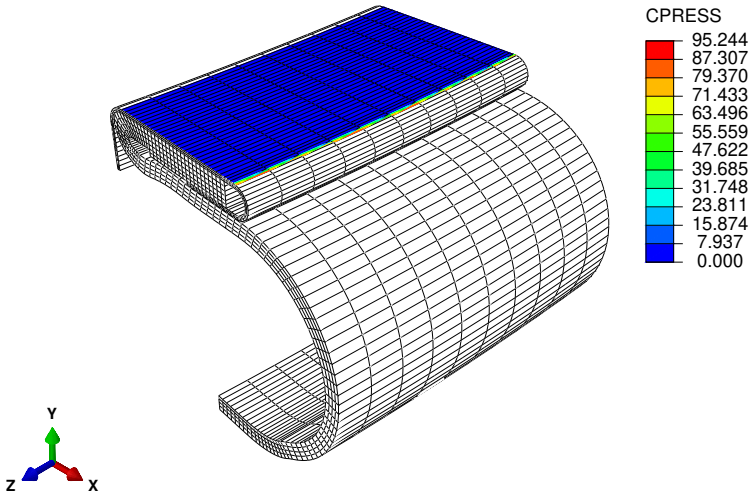


Figure 22: For the three-dimensional model, deformed configuration and contact stress contours of the seal at Step 2, increment 19.

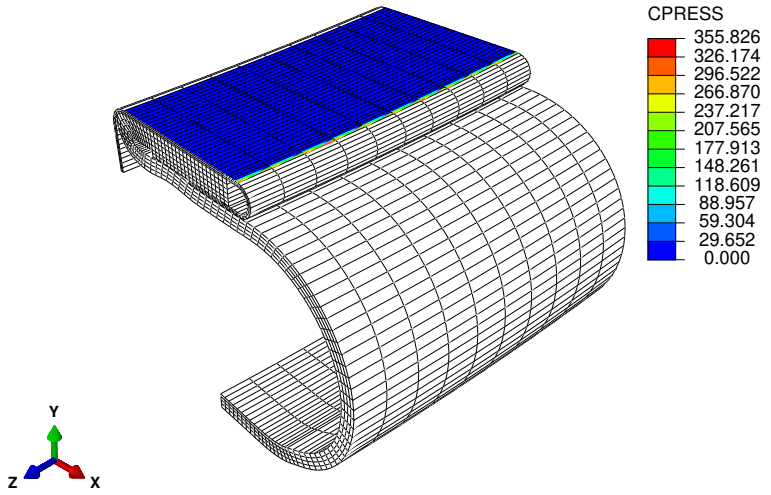


Figure 23: For the three-dimensional model, deformed configuration and contact stress contours of the seal at the end of Step 2.

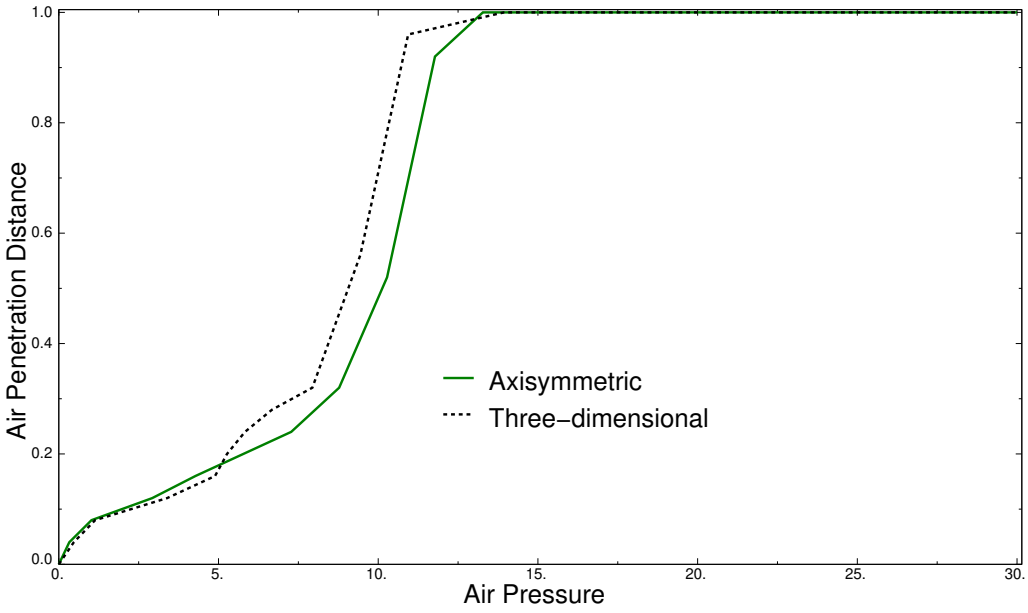


Figure 24: Air penetration distance as a function of air pressure in the seal.

Self-contact in rubber/foam components: jounce bumper

Products: Abaqus/Standard Abaqus/Explicit

References:

- *Self-contact in rubber/foam components: rubber gasket*

Overview

This example of an automotive jounce bumper demonstrates the use of the single-surface contact capability available for large-sliding analysis in Abaqus.

Components that deform and change their shape substantially can fold and have different parts of the surface come into contact with each other. In such cases it can be difficult to predict at the outset of the analysis where such contact may occur and, therefore, it can be difficult to define two independent surfaces to make up a contact pair.

A jounce bumper, sometimes referred to as a “helper spring,” is a highly compressible component that is used as part of the shock isolation system in a vehicle. It is typically located above the coil spring that connects the wheels to the frame. Microcellular material is used because of its high compressibility and low Poisson's ratio value at all but fully compressed configurations.

The bumper is mounted on a mandrel with a diameter larger than the bumper's inner diameter (*Figure 1*). The first step of the analysis solves this interference fit problem. The bumper initially sits against a fixed flat rigid surface on one end; on the other end, another flat rigid surface is used to model the compression of the bumper. The geometry of the bumper is such that it folds in three different locations. Separate surfaces are defined at the locations where self-contact is expected. This modeling technique produces an economical analysis because the scope of contact searches is limited.

Geometry and model

The bumper is 76.0 mm (3.0 in) long and has an inside diameter of 20.0 mm (.8 in). The mandrel, which is modeled as a rigid surface, has a diameter of 22.0 mm (.9 in). The bumper is modeled with the hyperfoam material model. The compressible, nonlinear elastic behavior is described by a strain energy function. A mesh of either CAX3 or CAX4R elements is produced using an automatic mesh generator. *Figure 1* and *Figure 2* show the initial mesh with CAX3 elements and CAX4R elements, respectively. In addition to the portions of the bumper's surface used to define self-contact, additional regions are defined to model contact between the bumper and the fixed surface, the bumper and the mandrel, and the bumper and the moving rigid surface. A small amount of friction (a Coulomb coefficient of 0.05) is applied to all of the surfaces.

Results and discussion

The bumper analysis is a two-step process. In the first step the interference between the bumper's inner diameter and the mandrel is resolved. In the Abaqus/Standard analysis the automatic “shrink” fit method (see *Common Difficulties Associated with Contact Modeling in Abaqus/Standard*) is used: the calculated initial penetration is allowed at the beginning of the step and scaled linearly to zero at the end of the step. In the Abaqus/Explicit analysis the interference resolution step is performed in one of three ways. In the first approach the initial penetration is solved using the "shrink” fit method (the same as in the Abaqus/Standard analysis). In the second approach the mandrel is positioned so that no contact or overclosure exists between the bumper and the mandrel at the outset of the analysis. The rigid surface representing the mandrel is then moved in the radial direction to simulate the compression of the bumper due to the interference fit. In the third approach the shrink fit solution from Abaqus/Standard is imported into Abaqus/Explicit. A comparison of the Mises stresses predicted by

Abaqus/Standard and Abaqus/Explicit at the end of the interference step shows that the results are very similar (see *Figure 3* and *Figure 4*).

In the second step the bottom surface compresses the bumper 42.0 mm (1.7 in) as a result of the application of displacement boundary conditions to the reference node of the surface. *Figure 5*, *Figure 6*, and *Figure 7* show the final deformed shape of the bumper; the high compressibility of the material is apparent, as well as the folding of the surface onto itself. Although a general knowledge of where the folding would occur was used in the definition of the self-contacting surfaces, it is not necessary to know exactly where the kinks in the surface will form.

The final deformed shapes predicted by Abaqus/Standard and Abaqus/Explicit for the CAX3 element mesh are the same (see *Figure 5* and *Figure 6*, respectively). A similar shape is predicted by Abaqus/Explicit when CAX4R elements are used (see *Figure 7*). However, the solution obtained with CAX4R elements in Abaqus/Explicit reveals that local buckling occurs in the upper-left folding radius. This makes a similar analysis using CAX4R elements in Abaqus/Standard very difficult. The local buckling is not captured in the CAX3 analysis due to the stiffer nature of these elements.

The load versus displacement curves of the bottom surface are shown in *Figure 8*. The results obtained with CAX3 elements in Abaqus/Standard and with CAX3 and CAX4R elements in Abaqus/Explicit are very similar. The energy absorption capacity of the bumper is seen through these curves.

Input files

selfcontact_bump_std_cax3.inp	Jounce bumper model for Abaqus/Standard using CAX3 elements.
selfcontact_bump_surf.inp	Jounce bumper model for Abaqus/Standard using CAX3 elements and surface-to-surface contact.
selfcontact_bump_std_imp_surf.inp	Jounce bumper model for Abaqus/Standard using CAX3 elements and surface-to-surface contact. This input file depends on selfcontact_bump_surf.inp.
selfcontact_bump_xpl_cax3.inp	Jounce bumper model for Abaqus/Explicit using CAX3 elements.
selfcontact_bump_xpl_cax4r.inp	Jounce bumper model for Abaqus/Explicit using CAX4R elements.
selfcontact_bump_xpl_cax4r_shrink.inp	Jounce bumper model for Abaqus/Explicit using CAX4R elements; resolution of the initial interference using the "shrink" fit method.
selfcontact_bump_std_resinter_cax4r.inp	Jounce bumper model for Abaqus/Standard using CAX4R elements to resolve the interference fit.
selfcontact_bump_std_imp_cax3.inp	Jounce bumper model for Abaqus/Standard using CAX3 elements; interference fit solution imported from Abaqus/Standard.
selfcontact_bump_xpl_imp_cax3.inp	Jounce bumper model for Abaqus/Explicit using CAX3 elements; interference fit solution imported from Abaqus/Standard.
selfcontact_bump_xpl_imp_cax4r.inp	Jounce bumper model for Abaqus/Explicit using CAX4R elements; interference fit solution imported from Abaqus/Standard.
selfcontact_bump_node_cax3.inp	Node definitions for the bumper model with CAX3 elements.
selfcontact_bump_node_cax4r.inp	Node definitions for the bumper model with CAX4R elements.
selfcontact_bump_element_cax3.inp	Element definitions for the bumper model with CAX3 elements.
selfcontact_bump_element_cax4r.inp	Element definitions for the bumper model with CAX4R elements.
selfcontact_bump_surfdef_cax3.inp	Surface definitions for the bumper model with CAX3 elements.
selfcontact_bump_surfdef_cax4r.inp	Surface definitions for the bumper model with CAX4R elements.

Figures

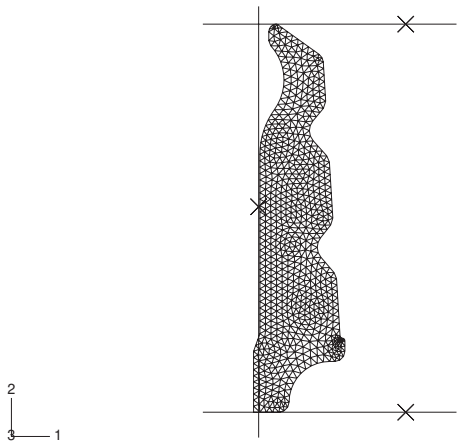


Figure 1: Jounce bumper initial mesh with CAX3 elements (Abaqus/Standard).

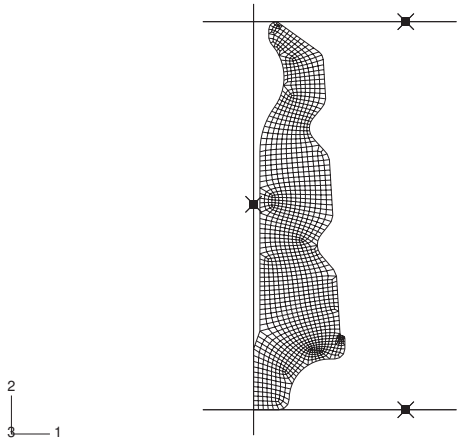


Figure 2: Jounce bumper initial mesh with CAX4R elements (Abaqus/Explicit).

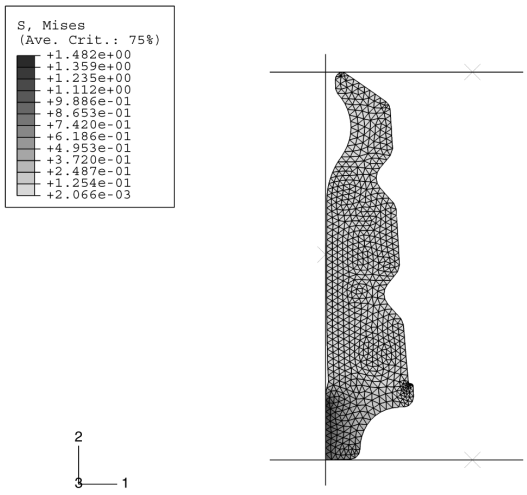


Figure 3: Mises stresses in bumper after interference is resolved with Abaqus/Standard automatic shrink fit option.

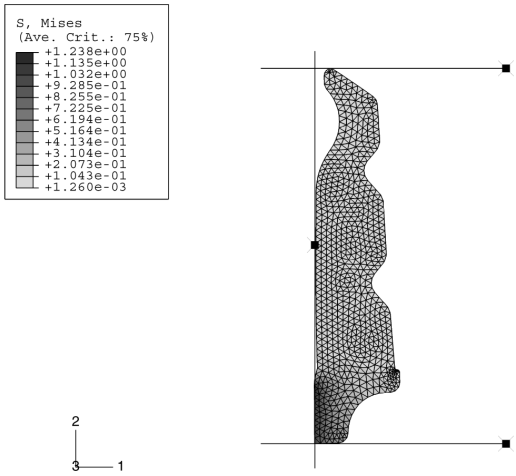


Figure 4: Mises stresses in bumper after interference is resolved with Abaqus/Explicit.

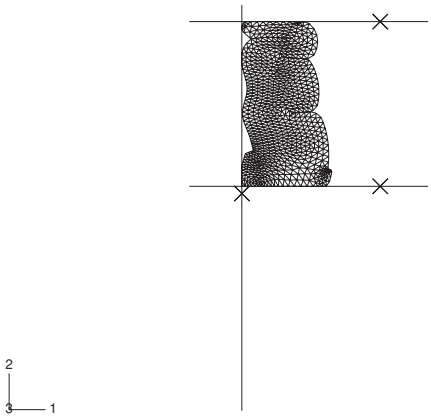


Figure 5: Bumper mesh after crushing (Abaqus/Standard; CAX3 elements).

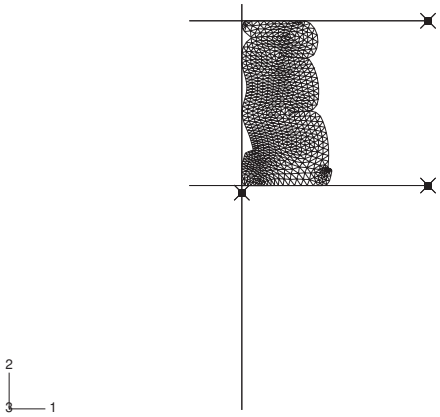


Figure 6: Bumper mesh after crushing (Abaqus/Explicit; CAX3 elements).

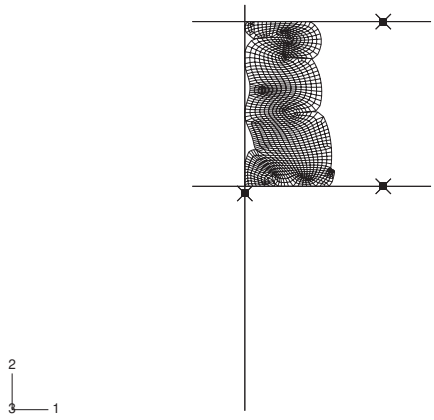


Figure 7: Bumper mesh after crushing (Abaqus/Explicit; CAX4R elements).

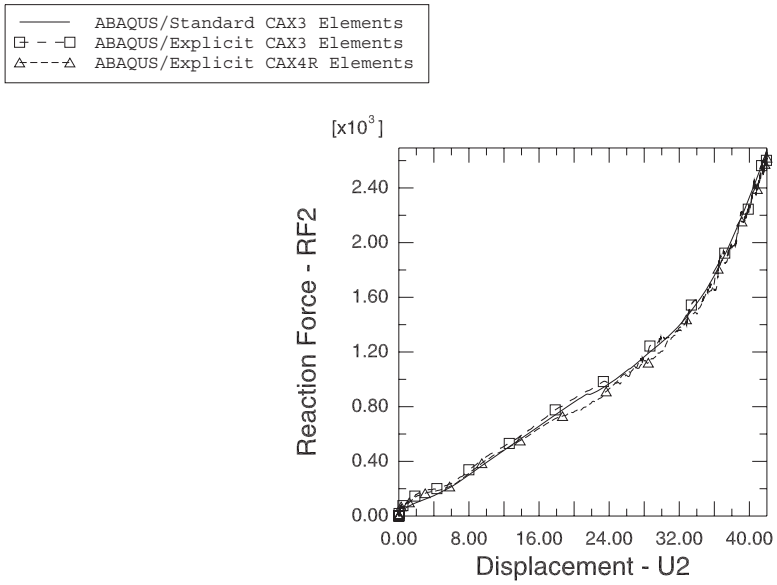


Figure 8: Bumper load-displacement curve.

Self-contact in rubber/foam components: rubber gasket

Products: Abaqus/Standard Abaqus/Explicit

References:

- *Self-contact in rubber/foam components: jounce bumper*

Overview

This example of an automotive rubber gasket demonstrates the use of the single-surface contact capability available for large-sliding analysis in Abaqus.

Components that deform and change their shape substantially can fold and have different parts of the surface come into contact with each other. In such cases it can be difficult to predict at the outset of the analysis where such contact may occur and, therefore, it can be difficult to define two independent surfaces to make up a contact pair.

This model is used to analyze an oil pan gasket, which enhances the sealing of the oil pan against the engine block. A primary objective of gasket designers is to reach or exceed a threshold value of contact pressure at the gasket bead/cover/engine block interfaces. Experience shows that, above such a threshold, oil will not leak. Another item of interest is the load-deflection curve obtained when compressing the gasket cross-section since it is indicative of the bolt load required to attain a certain gap between the oil pan and the engine block. Finally, the analysis provides details to ensure that stresses and strains are within acceptable bounds.

The rubber gasket is embedded in a plastic backbone. It has two planes of symmetry and a bead that, when compressed, provides the sealing effect (*Figure 1*). A flat rigid surface, parallel to one of the symmetry planes, pushes the gasket into the backbone. The geometry of the gasket is such that it folds in two different locations. In this model the entire free surface of the gasket and of the backbone is declared as a single surface allowed to contact itself. This modeling technique, although very simple, is more expensive because of the extensive contact searches required, as well as a larger wavefront of the equation system when using Abaqus/Standard.

The analysis is performed using both Abaqus/Standard and Abaqus/Explicit.

Geometry and model

The rubber gasket is modeled as a quarter of a plane strain section, initially in contact with a flat rigid surface. The clearance between the plastic backbone and the surface is 0.612 mm (.024 in). The height of the bead in the gasket is 1.097 mm (.043 in). The backbone is modeled with a linear elastic material with a Young's modulus of 8000.0 MPa (1160 ksi) and a Poisson's ratio of 0.4. In Abaqus/Standard the gasket is modeled as a fully incompressible hyperelastic material, which is much softer than the backbone material at all strain levels. In Abaqus/Explicit a small amount of compressibility is assumed for the gasket material. The nonlinear elastic behavior of the gasket is described by a strain energy function that is a first-order polynomial in the strain invariants. The model is discretized with first-order quadrilaterals. Standard elements are used for the backbone. In Abaqus/Standard full-integration hybrid elements are used for the gasket, while reduced-integration elements are used to model the gasket in Abaqus/Explicit. The interface between the gasket and the backbone is assumed to be glued with no special treatment required. A single surface definition covers all of the free surface of the gasket and the backbone. Through the definition of contact pairs, this surface is allowed to contact both the rigid surface and itself. A small amount of friction (Coulomb coefficient of 0.05) is applied to the interface with the rigid surface, which is assumed to be lubricated. Sticking surface behavior, through the specification of rough friction (*Frictional Behavior*), is applied when the gasket contacts itself, denoting a clean surface.

Results and discussion

The gasket analysis is a single-step procedure in which the rigid surface moves down almost all of the backbone clearance (0.61 mm or .024 in). The relative rigidity of the backbone forces the rubber gasket to fit inside the cavity provided by the backbone, folding in two regions (*Figure 2* and *Figure 3*). Although the general vicinity of the location of the folds can be estimated from the initial configuration, their exact locations are difficult to predict.

The deformed shape of the gasket and the locations of the folds predicted by Abaqus/Standard and Abaqus/Explicit agree well. The rigid surface load-displacement curve is also in good agreement, as shown in *Figure 4*.

Acknowledgements

SIMULIA would like to thank Mr. DeHerrera of Freudenberg-NOK General Partnership for providing these examples.

Input files

selfcontact_gask.inp	Gasket model for Abaqus/Standard.
selfcontact_gask_xpl.inp	Gasket model for Abaqus/Explicit.
selfcontact_gask_node.inp	Node definitions for the gasket model.
selfcontact_gask_element1.inp	Element definitions for the rubber part of the gasket model.
selfcontact_gask_element2.inp	Element definitions for the backbone part of the gasket model.
selfcontact_gask_c3d8h.inp	Three-dimensional gasket model for Abaqus/Standard.

Figures

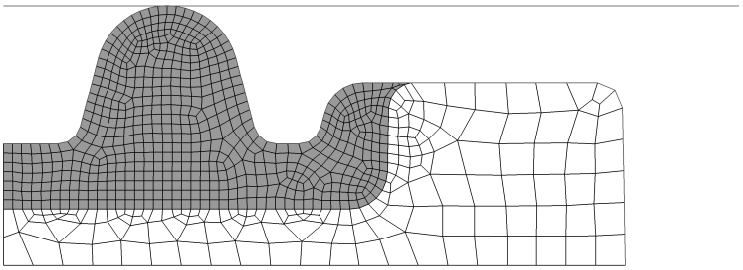


Figure 1: Gasket initial mesh.

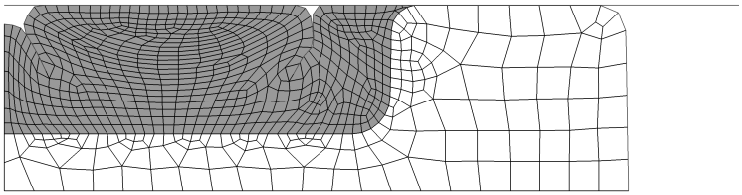


Figure 2: Gasket mesh after loading as predicted by Abaqus/Standard.

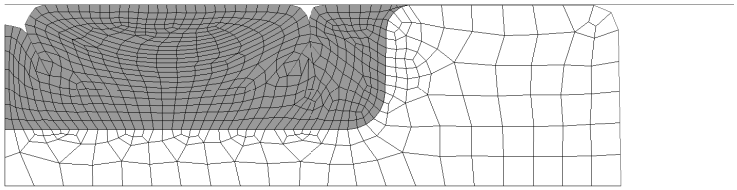


Figure 3: Gasket mesh after loading as predicted by Abaqus/Explicit.

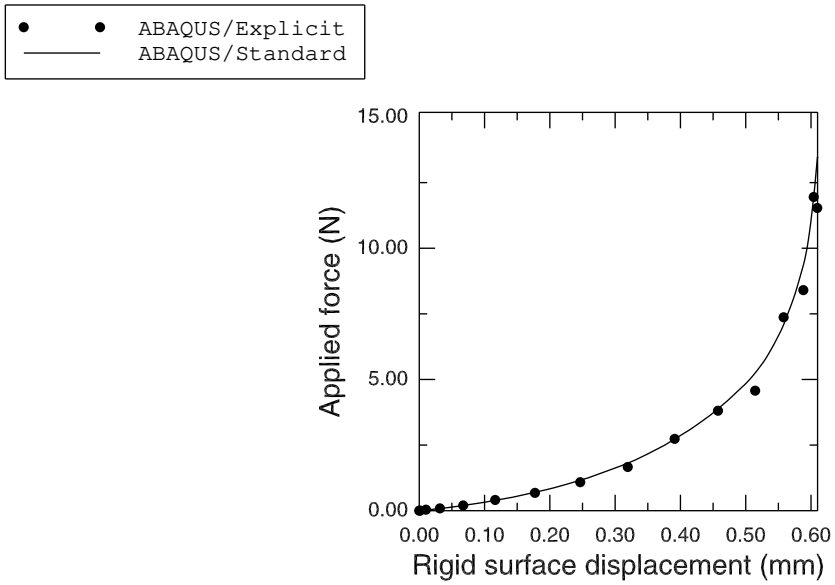


Figure 4: Rigid surface load-displacement curve.

Submodeling of a stacked sheet metal assembly

Products: Abaqus/Standard

Overview

This example shows how the submodeling methodology in Abaqus provides accurate modeling that is more economical than using a globally refined mesh in a single analysis.

Sheet metal stampings stacked and fitted on top of each other and secured together via mechanical fasteners such as bolts or rivets are commonly used in the automotive industry. Examples include seat belt anchors and seating track assemblies. The submodeling capability in Abaqus facilitates economical, yet detailed, prediction of the ultimate strength and integrity of such jointed assemblies. A global model analysis of an assembly is first performed to capture the overall deformation of the system. Subsequently, the displacement results of this global analysis are used to drive the boundaries of a submodeled region of critical concern.

In a finite element analysis of such a structure, shell elements are commonly used to represent the sheet metal stampings. The nodes of each shell typically lie along the mid-plane of the shell thickness. The thickness of the shells is used in the structural calculations but is not taken into account in the contact calculations. Hence, a structure composed of a stack-up of several sheet stampings may have the nodes of each sheet all lying in the same spatial plane. This close proximity creates uncertainty in a submodel analysis since Abaqus will not be able to determine the correct correspondence between the sheets in the submodel and the global model. Therefore, Abaqus provides a capability that allows the user to specify particular elements of the global model that are used to drive a particular set of nodes in a submodel, which eliminates the uncertainty. This capability is demonstrated in this example problem.

Geometry and model

The global model consists of five separate metal stampings meshed with S4R and S3R shell elements. An exploded view of the global finite element model is shown in *Figure 1*. The stampings are stacked one upon the other by collapsing the configuration in the 3-direction. All the shell elements are 0.5 mm thick, with all nodes positioned at the mid-surface of each shell. The separate meshes are connected together with BEAM-type MPCs between corresponding perimeter nodes on the large bolt holes through each layer. The nodes on the edges of the two small holes at the bottom of Layer 1 are constrained in all six degrees of freedom, representing the attachment point to ground. The translational degrees of freedom of the nodes around the perimeter of Layer 2 are also constrained, representing the far-field boundary condition in that plate.

Several surface definitions are used to model the contact between the various adjacent layers. The contact definitions prevent unwanted penetration between shell element layers. The small-sliding contact formulation is employed. Most of the contact in this problem is between adjacent layers, but there is also direct contact between Layer 2 and Layer 4. To avoid overconstraints, it is important that no point on Layer 4 simultaneously contact Layer 3 and Layer 2; therefore, node-based surfaces are used for the secondary surfaces. This precludes accurate calculation of contact stresses, but that is not important in this case since more accurate contact stresses are obtained in the submodel.

All five stampings are made of steel and are modeled as an elastic-plastic material. The elastic modulus is 207,000 MPa, Poisson's ratio is 0.3, and the yield stress is 250 MPa. The metal plasticity definition includes moderate strain hardening.

The submodel stampings are truncated versions of the global model, located in the same physical position as the global model. In this case these are the regions of concern for high stresses and potential failure of the joint. The submodel is discretized with a finer mesh than the global model to provide a higher level of accuracy. *Figure 2* shows an exploded view of the submodel. Because the stampings in the submodel contain the large bolt holes, the submodel contains BEAM-type MPCs in a manner analogous to that in the global model.

The submodel has several surface definitions and contact pairs to avoid penetration of one stamping into another. The submodel contains no node-based surfaces, however. The contact is modeled as element-based surface-to-surface in each layer.

The material definition and shell thicknesses in the submodel are the same as those in the global model.

Results and discussion

The global model is loaded by enforcing prescribed boundary conditions on the protruding edge of Layer 3. This edge is displaced −5.0 mm in the 1-direction and −12.5 mm in the 3-direction. *Figure 3* shows the deformed shape of the global model. The displacements at the nodes are saved to the results file for later use by the submodel analysis.

The submodel driven nodes are loaded using a submodel boundary condition. The perimeter nodes of each layer of the submodel that correspond to a "cut" out of the global geometry are driven by the interpolated nodal displacement results in the global results file. Each driven node set is in a separate shell layer. Therefore, the submodel analysis contains multiple submodels, which designate the global model element sets to be searched for the responses that drive the submodel driven node sets. For example, the driven nodes in submodel Layer 1 (node set `L1BC`) are driven by the results for the global element set which contains the elements of (global) Layer 1 (element set `LAYER1`). The driven nodes for Layers 2–4 are specified in a similar way. Because submodel Layer 5 has no driven nodes, only four submodels are required.

Figure 4 shows the deformed shape of the submodel. *Figure 5* and *Figure 6* show contour plots of the out-of-plane displacements in Layer 2 for the global model and submodel, respectively. In both cases the displacement patterns are similar; however, the maximum displacement predicted by the global model is about 7.8% larger than that predicted by the submodel.

Input files

stackedassembly_s4r_global.inp	S4R global model.
stackedassembly_s4r_global_mesh.inp	Key input data for the S4R global model.
stackedassembly_s4r_sub.inp	S4R submodel.
stackedassembly_s4r_sub_mesh.inp	Key input data for the S4R submodel.

Figures

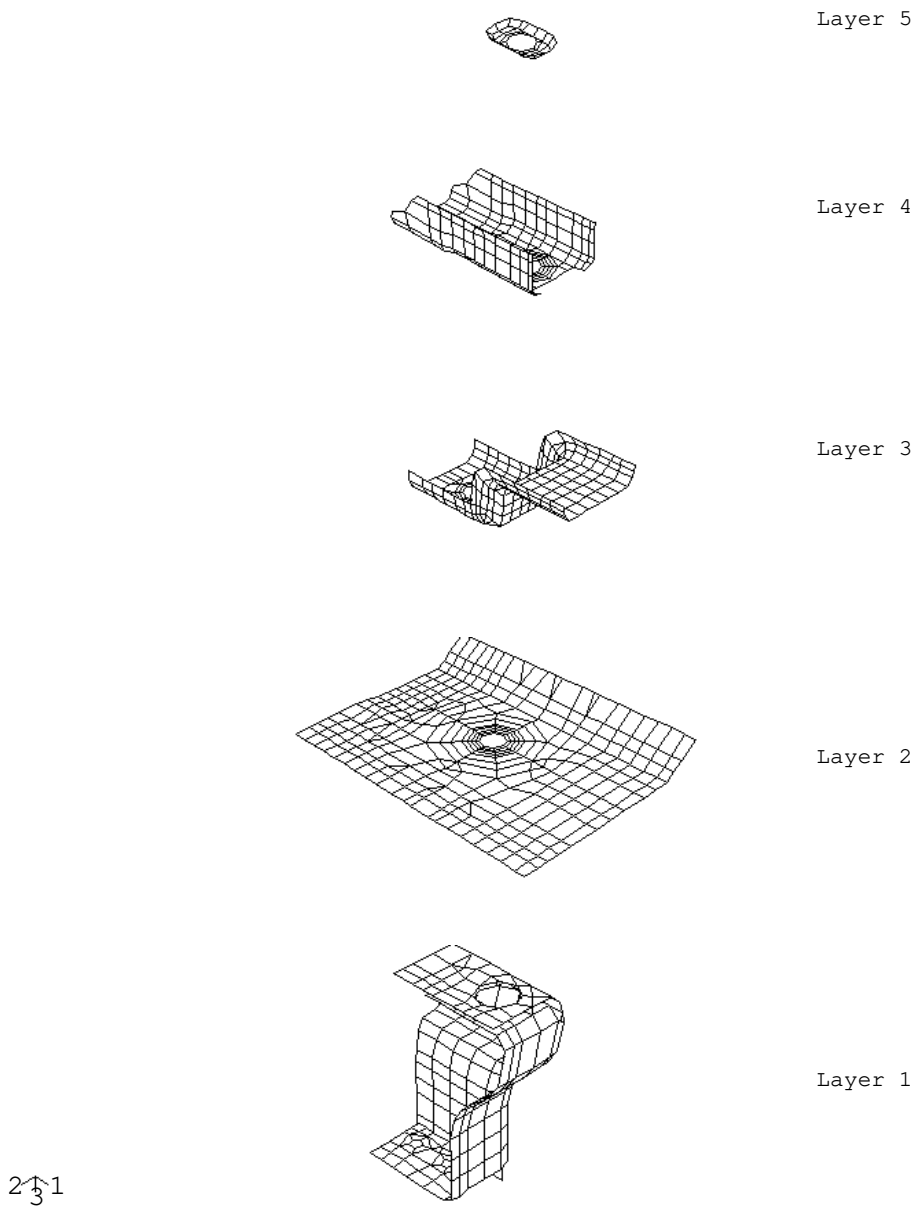


Figure 1: Exploded view of global model.

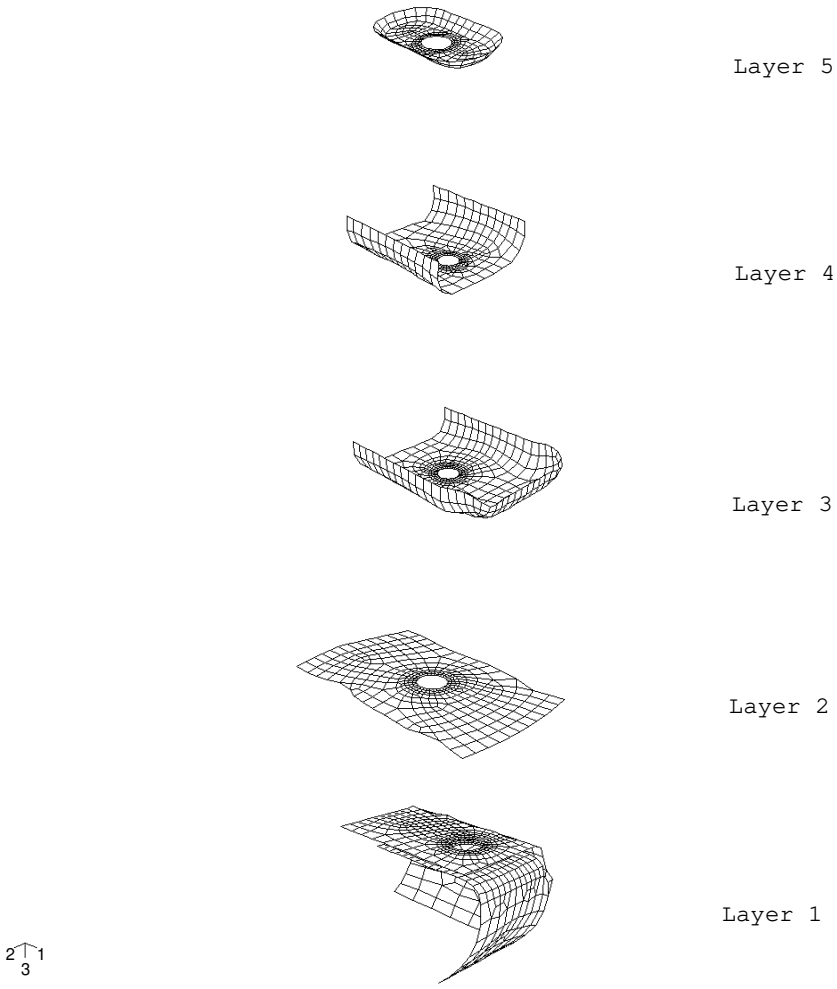


Figure 2: Exploded view of submodel.

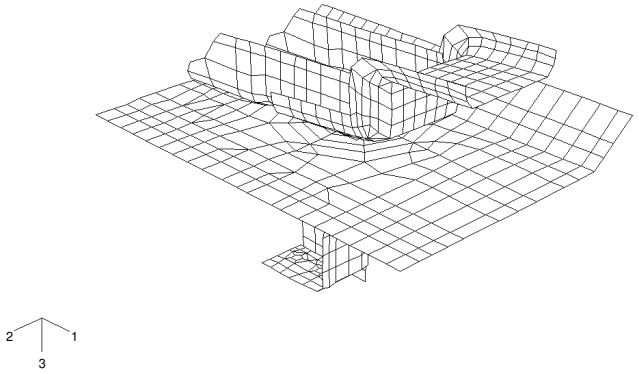


Figure 3: Deformed shape of global model.

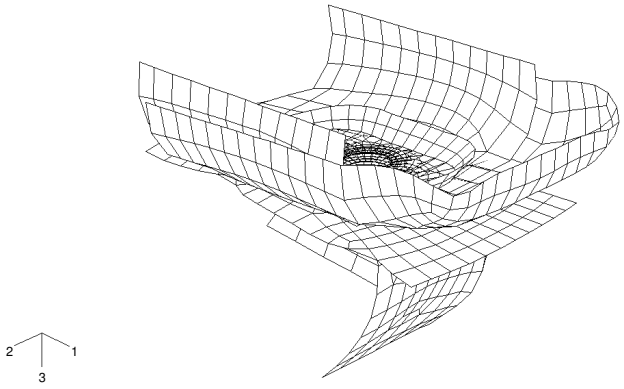


Figure 4: Deformed shape of submodel.

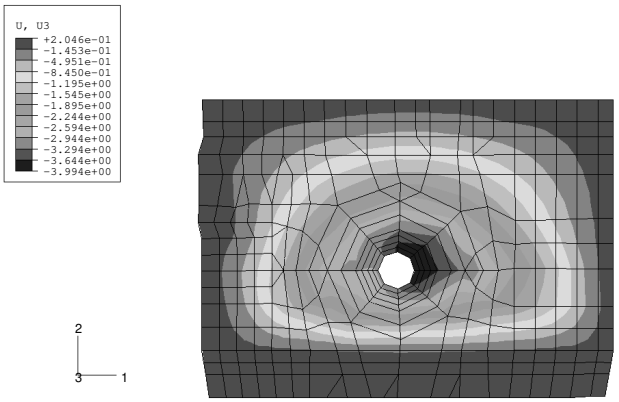


Figure 5: Out-of-plane displacement in Layer 2, global model.

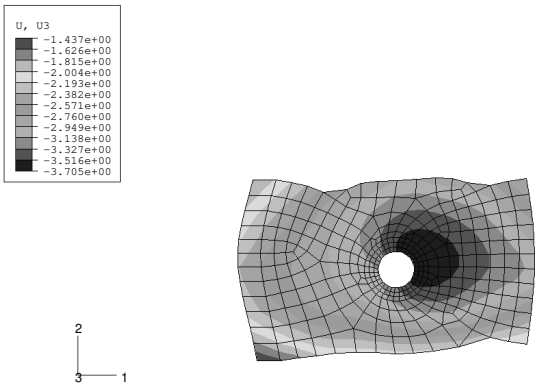


Figure 6: Out-of-plane displacement in Layer 2, submodel.

Axisymmetric analysis of a threaded connection

Products: Abaqus/Standard

Overview

This example shows the wide range of modeling, analysis, and output capabilities available in Abaqus to assess the design of a connector under loading conditions.

Threaded connectors are commonly used components in the piping and offshore industry. They must withstand a variety of loading conditions: thread engagement, torque, bending, axial pullout, internal pressure under operating and overload conditions, and potential fluid leakage through threaded connections.

Two Abaqus methods that are particularly useful for analyzing threaded connectors are the specification of an allowable contact interference and of pressure penetration loads.

- The automatic “shrink” fit method can be used to automatically resolve the overclosure of two contacting surfaces. This method is applicable only during the first step of an analysis, and it cannot be used with self-contact. See *Modeling Contact Interference Fits in Abaqus/Standard* for details.
- The surface-based pressure penetration capability described in *Fluid Pressure Penetration Loads* is used to simulate pressure penetration between contacting surfaces. This capability is provided for simulating cases where a joint between two deforming bodies (for example, between two components threaded onto each other) or between a deforming body and a rigid surface (such as a soft gasket used in a joint) is exposed at one or multiple ends to a fluid pressure. This pressure will penetrate into the joint and load the surfaces forming the joint until some area of the surfaces is reached where the contact pressure between the abutting surfaces exceeds the critical value specified in the pressure penetration load, cutting off further penetration.

The contact output variables in Abaqus can provide the designer a wealth of information about the performance of a connector during all steps of an analysis. When modeling surface-based contact with axisymmetric elements (CAX- and CGAX-type elements) an output quantity of particular use is the maximum torque that can be transmitted about the z-axis by a specified contact pair. The maximum torque, T, is a scalar value defined as

$$T = \int\int r^2 p \, ds \, d\theta,$$

where p is the pressure transmitted across the interface, r is the radius to a point on the interface, and s is the current distance along the interface in the r–z plane. T is not a real torque; it is a computed limit of torque that a contact pair may transmit about the z-axis assuming that all the secondary nodes on the contact surface are slipping and that the friction coefficient is set to 1. The actual maximum torque that can be transmitted about the z-axis by a specified contact pair can be estimated by scaling T by the friction coefficient specified for the contact pair. The value of T can be output by requesting the contact output variable CTRQ.

This example demonstrates the usefulness of the specification of allowable contact interference and of pressure penetration loads as well as the Abaqus contact output variables in an axisymmetric analysis of a particular threaded connector.

Geometry and model

A three-dimensional cut-away view of the threaded connection assembly analyzed in this example is shown in *Figure 1*. Although the actual threads are helical, they are represented with an axisymmetric geometry. Previous experience has shown this simplification to be appropriate for these types of problems. Both the “pin” and the “box” are made from steel with a Young's modulus of 207 GPa (30×10^6 psi) and a Poisson's ratio of 0.3, which is characterized by a von Mises plasticity model. The unthreaded section of the pin has inner and outer radii of 48.6 mm (1.913 in) and 57.2 mm (2.25 in), respectively. The major diameter of the threads on the pin (diameter

measured at the crest of the threads) is slightly larger than the major diameter of the threads on the box (diameter measured at the roots of the threads); thus, there is an initial interference between the threads on the pin and on the box.

The deformed axisymmetric mesh (after the initial interference has been resolved) in the vicinity of the threads is illustrated in *Figure 2*. Contact is modeled by the interaction of contact surfaces defined by grouping specific faces of the elements in the contacting regions.

Loading and boundary conditions

Two analyses of the threaded connection are performed: an axisymmetric analysis using CAX4 elements and an axisymmetric analysis with twist using CGAX4 elements. The first four steps for the two analyses are identical. The CGAX4 model has an additional fifth step.

The initial interference fit of the threads on the pin and box is resolved in the first step using the automatic "shrink" fit method with a friction coefficient of 0. In the second step the assembly is held fixed while the friction coefficient is changed from 0 to 0.1 using changes to friction properties. An internal gauge pressure of 0.689 MPa (100 psi) is applied to the connector in the third step. The pressure on the contact surfaces is applied using pressure penetration loading. In the first three steps the displacements in the 2-direction are constrained to be zero at both ends of the assembly. To simulate an axial load in the fourth step, a displacement boundary condition of –0.254 cm (–0.1 in) is applied to the end of the box in the 2-direction. In the fifth step for the CGAX4 model the end of the pin is held fixed while the end of the box is rotated 0.1 radians about the 2-axis, simulating a torque being applied to the connector. The actual torques generated about the 2-axis by the frictional stresses in the fifth step are given by the output variable CMS2. This value is compared to the estimated value given by CTRQ for the fourth step.

Results and discussion

All analyses are performed as large-displacement analyses. The results from the first four steps for both models are identical. *Figure 3* and *Figure 4*, respectively, show the von Mises stress distributions in the threaded assembly after the overclosure has been resolved in Step 1 and after the displacement boundary condition has been applied in Step 4. As is illustrated in *Figure 4*, some of the threads on the pin are beginning to pull out at the end of Step 4. However, plots of the pressure penetration on the contact surface of the box in *Figure 5* and *Figure 6* show that the seal of the threads is maintained; thus, no leakage is indicated. If the seal had failed, the penetration pressure on the box surface in contact with the pin would be 0.689 MPa (100 psi) instead of 0. Other contact output variables such as CPRESS and COPEN provide additional information about the contact state throughout the analysis.

The scaled values of CTRQ (scaled by the friction coefficient of 0.1) and the values of CMS2 for all five steps in the CGAX4 analysis are illustrated in *Figure 7*. The value of CTRQ at the end of Step 4 is 1.22×10^6 lb-in, which translates into an estimated maximum torque of 1.22×10^5 lb-in for a friction coefficient of 0.1. The value of CMS2 computed during Step 5 for the CGAX4 model is 1.18×10^5 lb-in. The 3.7% difference in this example between the predicted and actual torque values can be attributed to a slight change in the normal pressure distribution between the contact surfaces that occurs when the box is rotated. The value of CMS2 is zero for the first four steps since no frictional stresses are generated between the contact surfaces until the fifth step. The value of CTRQ increases in the first step as the overclosure is resolved and dips in the fourth step due to the change in the contact pressure as the box is pulled away from the pin (see *Figure 7*).

Acknowledgements

SIMULIA gratefully acknowledges the ExxonMobil Upstream Research Corporation for their cooperation in implementing the CTRQ output variable and for supplying the geometry, mesh, and material properties used in this example.

Input files

threadedconnector_cax4.inp	Axisymmetric analysis of the threaded connector using CAX4 elements.
threadedconnector_cax4_n.inp	Node definitions for the axisymmetric analysis of the threaded connector using CAX4 elements.
threadedconnector_cax4_e.inp	Element definitions for the axisymmetric analysis of the threaded connector using CAX4 elements.
threadedconnector_cgax4.inp	Axisymmetric analysis of the threaded connector using CGAX4 elements.
threadedconnector_cgax4_n.inp	Node definitions for the axisymmetric analysis of the threaded connector using CGAX4 elements.
threadedconnector_cgax4_e.inp	Element definitions for the axisymmetric analysis of the threaded connector using CGAX4 elements.

Figures

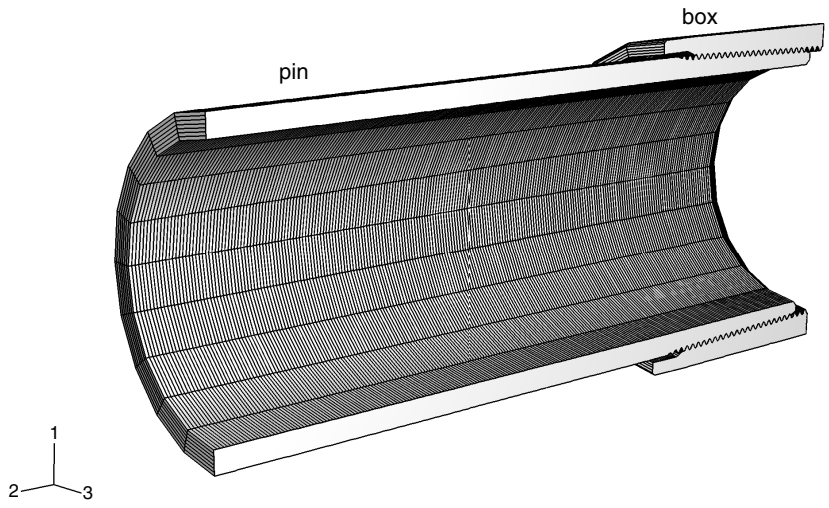


Figure 1: Three-dimensional cut-away view of the threaded connection.

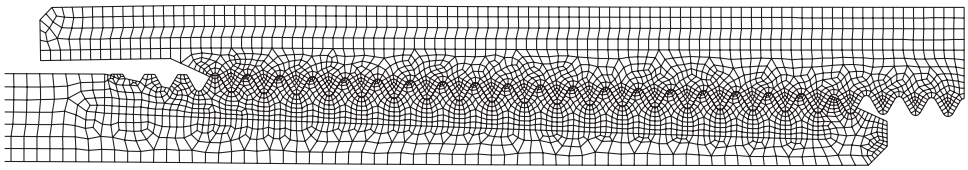


Figure 2: Axisymmetric mesh in the vicinity of the threads after the initial interference has been resolved.

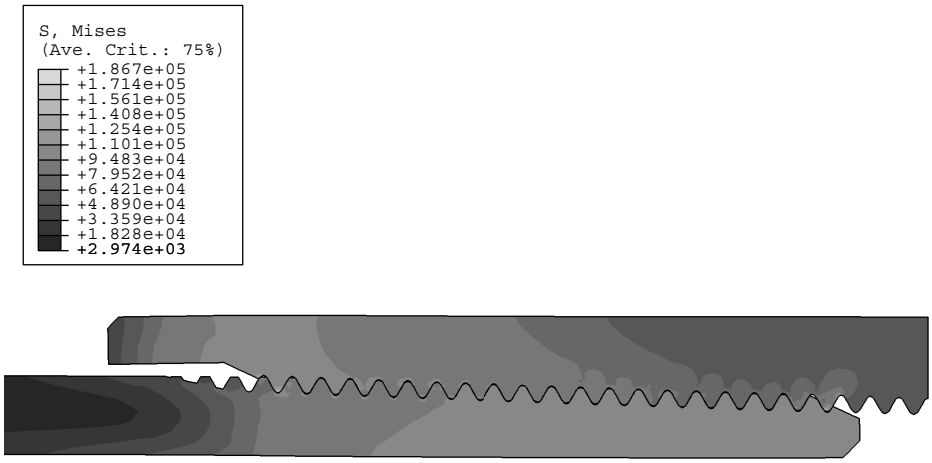


Figure 3: Stress distribution in the threads after the initial overclosure has been resolved.

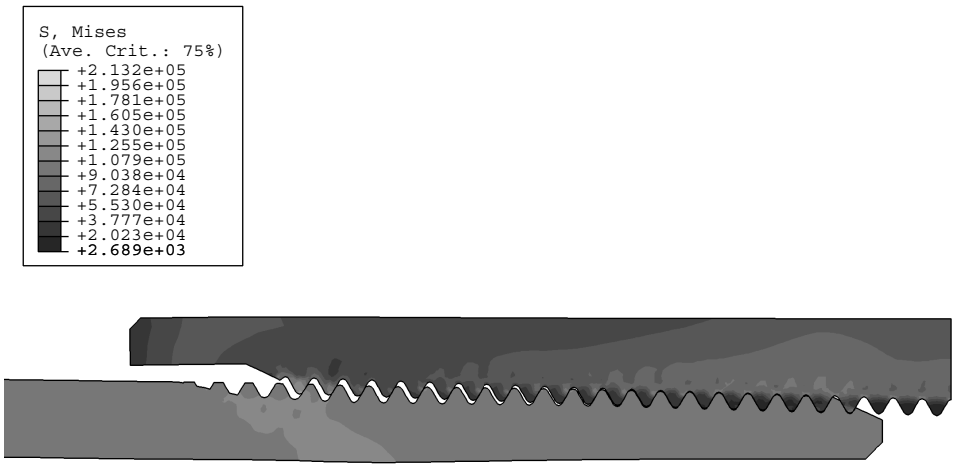


Figure 4: Stress distribution in the threads after axial loading.

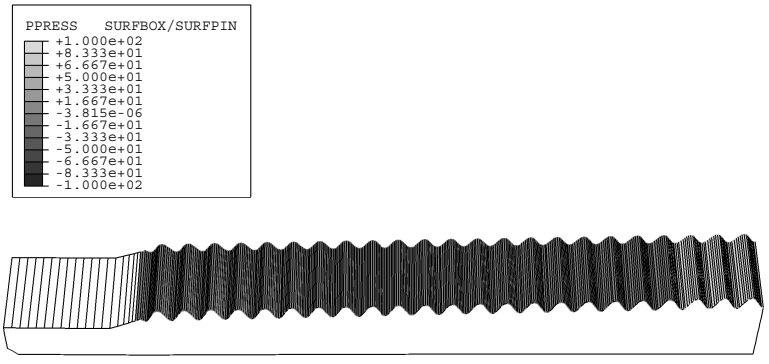


Figure 5: Pressure penetration on box contact surface after axial loading.

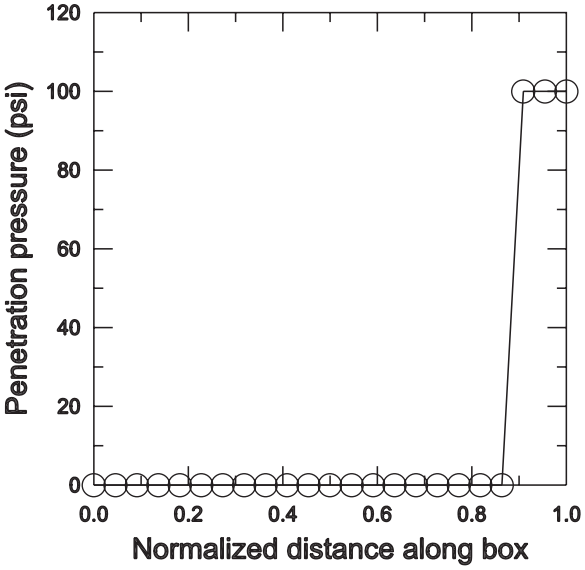


Figure 6: Plot of pressure penetration on box contact surface after axial loading.

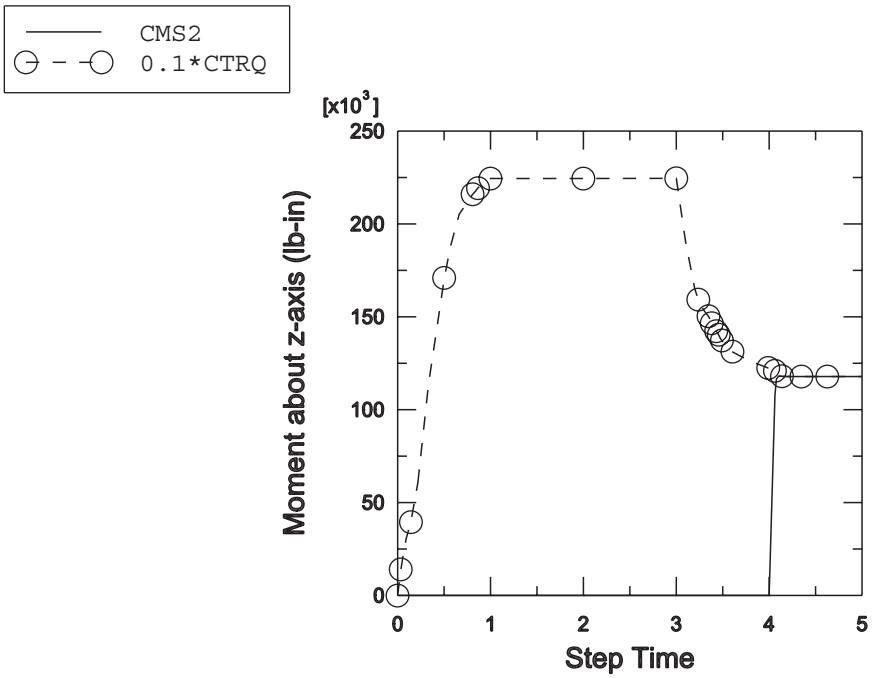


Figure 7: Comparison of 0.1*CTRQ to CMS2 for the CGAX4 model.

Direct cyclic analysis of a cylinder head under cyclic thermomechanical loadings

Products: Abaqus/Standard

Overview

This example demonstrates the use of the direct cyclic analysis procedure to obtain results that can be used for fatigue life calculations, which are fundamental in assessing product performance.

It is well known that a highly loaded structure, such as a cylinder head in an engine subjected to large temperature fluctuations and clamping loads, can undergo plastic deformations. After a number of repetitive loading cycles there will be one of three possibilities: elastic shakedown, in which case there is no danger of low-cycle fatigue; plastic shakedown, leading to a stabilized plastic strain cycle, in which case energy dissipation criteria will be used to estimate the number of cycles to failure; and plastic ratcheting, in which case the design is rejected. The classical approach to obtaining the response of such a structure is to apply the periodic loading repetitively to the structure until a stabilized state is obtained or plastic ratcheting occurs. This approach can be quite expensive, since it may require application of many loading cycles to obtain the steady response. To avoid the considerable numerical expense associated with such a transient analysis, the direct cyclic analysis procedure, described in *Direct Cyclic Analysis*, can be used to calculate the cyclic response of the structure directly.

Geometry and model

The cylinder head analyzed in this example is depicted in *Figure 1*. The cylinder head (which is a single cylinder) has three valve ports, each with an embedded valve seat; two valve guides; and four bolt holes used to secure the cylinder head to the engine block.

The body of the cylinder head is made from aluminum with a Young's modulus of 70 GPa, a yield stress of 62 MPa, a Poisson's ratio of 0.33, and a coefficient of thermal expansion of 22.6×10^{-6} per °C at room temperature. In this example the region in the vicinity of the valve ports, where the hot exhaust gases converge, is subjected to cyclic temperature fluctuations ranging from a minimum value of 35°C to a maximum value of 300°C. The temperature distribution when the cylinder head is heated to its peak value is shown in *Figure 2*. Under such operating conditions plastic deformation, as well as creep deformation, is observed. The two-layer viscoelastic-elastoplastic model, which is best suited for modeling the response of materials with significant time-dependent behavior as well as plasticity at elevated temperatures, is used to model the aluminum cylinder head (see *Two-Layer Viscoplasticity*). This material model consists of an elastic-plastic network that is in parallel with an elastic-viscous network. The Mises metal plasticity model with kinematic hardening is used in the elastic-plastic network, and the power-law creep model with strain hardening is used in the elastic-viscous network. Since the elastic-viscoplastic response of aluminum varies greatly over this range of temperatures, temperature-dependent material properties are specified.

The two valve guides are made of steel, with a Young's modulus of 106 GPa and a Poisson's ratio of 0.35. The valve guides fit tightly into two of the cylinder head valve ports and are assumed to behave elastically. The interface between the two components is modeled by using matched meshes that share nodes along the interface.

The three valve seats are made of steel, with a Young's modulus of 200 GPa and a Poisson's ratio of 0.3. The valve seats are press-fit into the cylinder head valve ports. This is accomplished by defining radial constraint equations of the form $u^r = u^p - u^s$ between the nodes on the valve seat surface and the nodes on the valve port surface, where u^p is the radial displacement on the valve port, u^s is the radial displacement on the valve seat, and u^r is a reference node. During the first step of the analysis a prescribed displacement is applied to the reference node, resulting in normal pressures developing between the two components. The valve seats are assumed to behave elastically.

All of the structural components (the cylinder head, the valve guides, and the valve seats) are modeled with three-dimensional continuum elements. The model consists of 19394 first-order brick elements (C3D8) and 1334 first-order prism elements (C3D6), resulting in a total of about 80,000 degrees of freedom. The C3D6 elements are used only where the complex geometry precludes the use of C3D8 elements.

Loading and boundary constraints

The loads are applied to the assembly in two analysis steps. In the first step the three valve seats are press-fit into the corresponding cylinder head valve port using linear multi-point equation constraints and prescribed displacement loadings as described above. A static analysis procedure is used for this purpose. The cyclic thermal loads are applied in the second analysis step. It is assumed that the cylinder head is securely fixed to the engine block through the four bolt holes, so the nodes along the base of the four bolt holes are secured in all directions during the entire simulation.

For an engine cylinder head, the valve seat press-fit occurs in the engine assembly process and the cyclic thermal loading occurs under engine operating conditions. Taking this into consideration, you can request the long-term response for two-layer viscoplasticity in the static analysis procedure. The choice of the instantaneous or long-term elastic solution affects only the results of the first static step, but the effect on the direct cyclic thermomechanical analysis is negligible.

The cyclic thermal loads are obtained by performing an independent thermal analysis. In this analysis three thermal cycles are applied to obtain a steady-state thermal cycle. Each thermal cycle involves two steps: heating the cylinder head to the maximum operating temperature and cooling it to the minimum operating temperature using concentrated flux and film conditions. The nodal temperatures for the last two steps (one thermal cycle) are assumed to be a steady-state solution and are stored in a results (`.fil`) file for use in the subsequent thermomechanical analysis. The maximum value of the temperature occurs in the vicinity of the valve ports where the hot exhaust gases converge. The temperature in this region (node 50417) is shown in *Figure 3* as a function of time for a steady-state cycle.

In the second step of the mechanical analysis cyclic nodal temperatures generated from the previous heat transfer analysis are applied. The direct cyclic procedure with a fixed time incrementation of 0.25 and a load cycle period of 30 is specified in this step, resulting in a total number of 120 increments for one iteration. The number of terms in the Fourier series and the maximum number of iterations are 40 and 100, respectively.

For comparison purposes the same model is also analyzed using the classical transient analysis, which requires 20 repetitive steps before the solution is stabilized. A cyclic temperature loading with a constant time incrementation of 0.25 and a load cycle period of 30 is applied in each step.

Results and discussion

One of the considerations in the design of a cylinder head is the stress distribution and deformation in the vicinity of the valve ports. *Figure 4* shows the von Mises stress distribution in the cylinder head at the end of a loading cycle (iteration 75, increment 120) in the direct cyclic analysis. The total strain distribution at the same time in the direct cyclic analysis is shown in *Figure 5*. The deformation and stress are most severe in the vicinity of the valve ports, making this region critical in the design. The results shown in *Figure 6* through *Figure 16* are measured in this region (element 50152, integration point 1). *Figure 6*, *Figure 7*, and *Figure 8* show the evolution of the stress component, plastic strain component, and viscous strain component, respectively, in the global 1-direction throughout a complete load cycle during iterations 50, 75, and 100 in the direct cyclic analysis. The time evolution of the stress versus the plastic strain, shown in *Figure 9*, is obtained by combining *Figure 6* with *Figure 7*. Similarly, the time evolution of the stress versus the viscous strain, shown in *Figure 10*, is obtained by combining *Figure 6* with *Figure 8*. The shapes of the stress-strain curves remain unchanged after iteration 75, as do the peak and mean values of the stress over a cycle. However, the mean value of the plastic strain and the mean value of the viscous strain over a cycle continue to grow from one iteration to another iteration, indicating that the plastic ratcheting occurs in the vicinity of the valve ports.

Similar results for the evolution of stress versus plastic strain and the evolution of stress versus viscous strain during cycles 5, 10, and 20 obtained using the classical transient approach are shown in *Figure 11* and *Figure 12*, respectively. The plastic ratcheting is observed to be consistent with that predicted using the direct cyclic approach. A comparison of the evolution of stress versus plastic strain obtained during iteration 100 in the direct cyclic analysis with that obtained during cycle 20 in the transient approach is shown in *Figure 13*. A similar comparison of the evolution of stress versus viscous strain obtained using both approaches is shown in *Figure 14*. The shapes of the stress-strain curves are similar in both cases.

One advantage of using the direct cyclic procedure, in which the global stiffness matrix is inverted only once, instead of the classical approach in Abaqus/Standard is the cost savings achieved. In this example the total computational time leading to the first occurrence of plastic ratcheting in the direct cyclic analysis (75 iterations) is approximately 70% of the computational time spent in the transient analysis (20 steps). The savings will be more significant as the problem size increases.

Additional cost savings for the solution can often be obtained by using a smaller number of terms in the Fourier series and/or a smaller number of increments in an iteration. In this example, if 20 rather than 40 Fourier terms are chosen, the total computational time leading to the first occurrence of plastic ratcheting in the direct cyclic analysis (75 iterations) is approximately 65% of the computational time spent in the transient analysis (20 steps). Furthermore, if a fixed time incrementation of 0.735 rather than 0.25 is specified, leading to a total number of 41 increments for one iteration, the total computational time in the direct cyclic analysis is reduced by a factor of three without compromising the accuracy of the results. A comparison of the evolution of stress versus plastic strain obtained using fewer Fourier terms during iteration 75 is shown in *Figure 15*. A similar comparison of the evolution of stress versus viscous strain obtained using fewer Fourier terms is shown in *Figure 16*. The shapes of the stress-strain curves and the amount of energy dissipated during the cycle are similar in both cases, although the case with fewer Fourier terms provides less accurate stress results.

Another advantage of using the direct cyclic approach instead of the classical approach is that the likelihood of plastic ratcheting or stabilized cyclic response can be predicted automatically by comparing the displacement and residual coefficients with some internal control variables. There is no need to visualize the detailed results for the whole model throughout the loading history, which leads to a further reduction of the data storage and computational time associated with output. For this example examination of the displacement and the residual coefficients written to the message (`.msg`) file makes it clear that the constant term in the Fourier series does not stabilize and, thus, plastic ratcheting occurs.

Acknowledgements

SIMULIA gratefully acknowledges PSA Peugeot Citroën and the Laboratory of Solid Mechanics of the Ecole Polytechnique (France) for their cooperation in developing the direct cyclic analysis capability and for supplying the geometry and material properties used in this example.

Input files

dircyccylinderhead_heat.inp	Input data for the heat transfer analysis.
dircyccylinderhead_heat_mesh.inp	Node and element definitions for the heat transfer analysis.
dircyccylinderhead_heat_sets.inp	Node set, element set, and surface definitions for the heat transfer analysis.
dircyccylinderhead_heat_load1.inp	Loading definitions during the heating process for the heat transfer analysis.
dircyccylinderhead_heat_load2.inp	Loading definitions during the cooling process for the heat transfer analysis.
dircyccylinderhead_dcm.inp	Input data for the direct cyclic analysis.
dircyccylinderhead_dcm_mesh.inp	Node and element definitions for the direct cyclic analysis.

dircyccylinderhead_dcm_sets.inp	Node set and element set definitions for the direct cyclic analysis.
dircyccylinderhead_dcm_eqc.inp	Kinematic constraint definitions for the direct cyclic analysis.
dircyccylinderhead_dcm_ps.inp	Post output for the direct cyclic analysis.

References

- Maitournam, H., B. Pommier, and J. J. Thomas, “Détermination de la réponse asymptotique d'une structure anélastique sous chargement thermomécanique cyclique,” *C. R. Mécanique*, vol. 330, pp. 703–708, 2002.
- Maouche, N., H. Maitournam, and K. Dang Van, “On a new method of evaluation of the inelastic state due to moving contacts,” *Wear*, pp. 139–147, 1997.
- Nguyen-Tajan, T. M. L., B. Pommier, H. Maitournam, M. Houari, L. Verger, Z. Z. Du, and M. Snyman, “Determination of the stabilized response of a structure undergoing cyclic thermomechanical loads by a direct cyclic method,” *Abaqus Users' Conference Proceedings*, 2003.

Figures

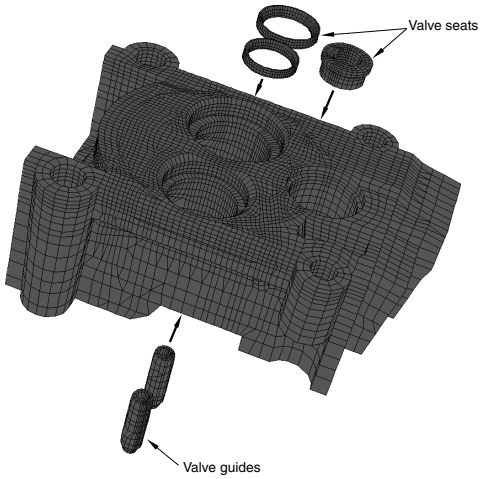


Figure 1: A cylinder head model.

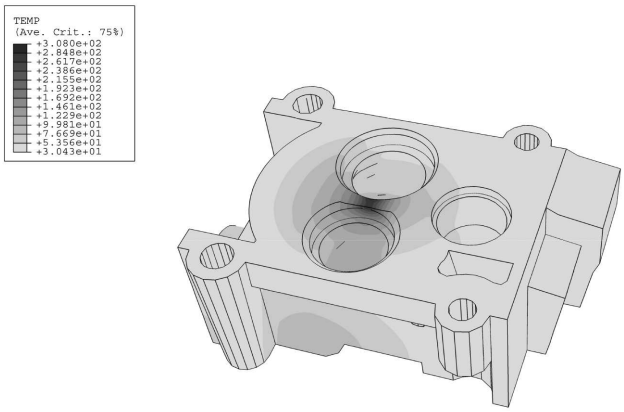


Figure 2: Temperature distribution when the cylinder head is heated to its peak value.

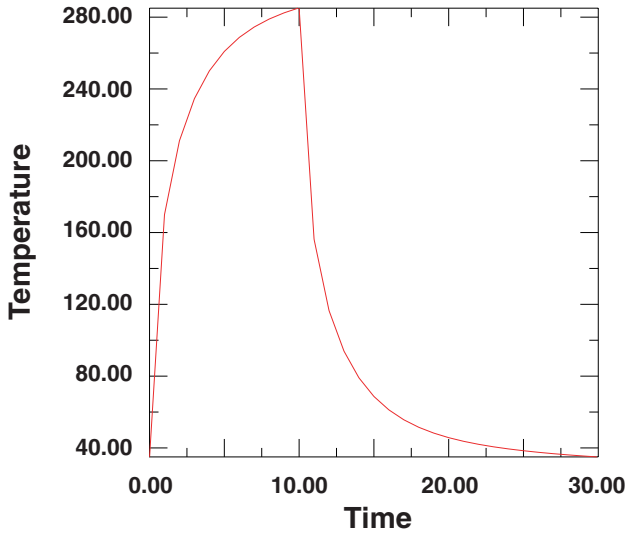


Figure 3: Temperature at node 50417 as a function of time for a steady-state cycle.

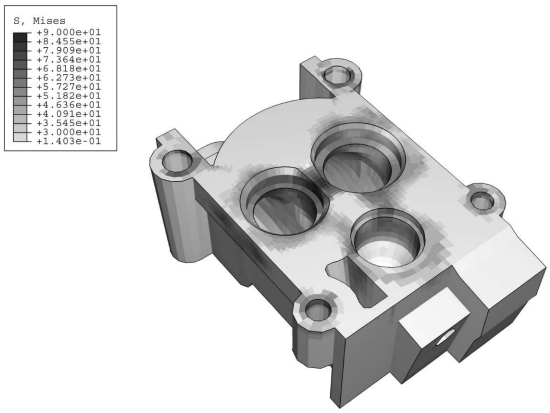


Figure 4: Von Mises stress distribution in the cylinder head at the end of a loading cycle (iteration 75, increment 120) in the direct cyclic analysis.

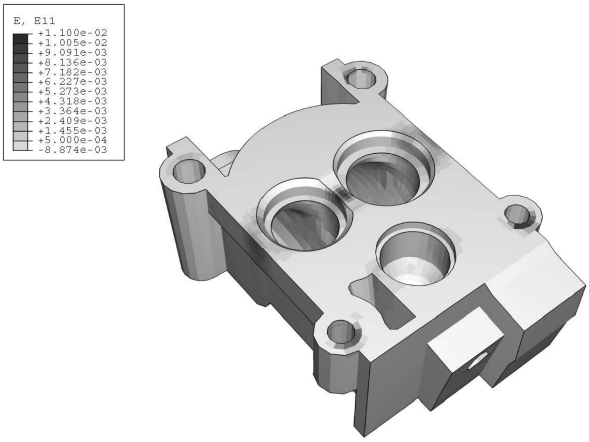


Figure 5: Total strain distribution in the cylinder head at the end of a loading cycle (iteration 75, increment 120) in the direct cyclic analysis.

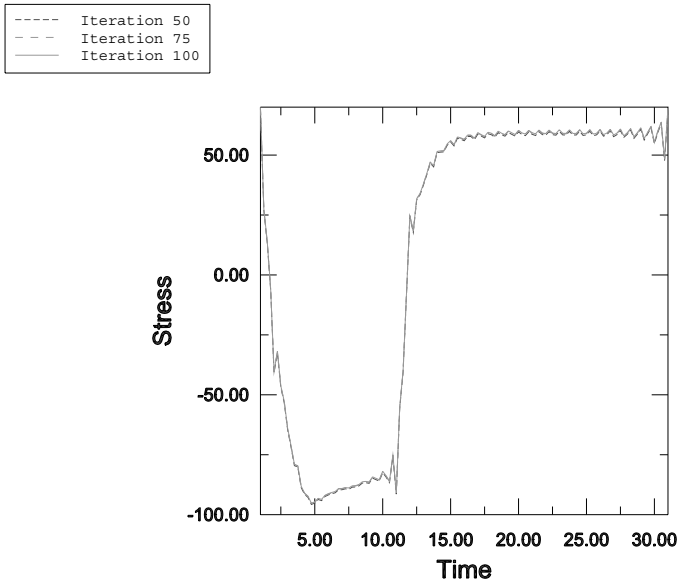


Figure 6: Evolution of the stress component in the global 1-direction during iterations 50, 75, and 100 in the direct cyclic analysis.

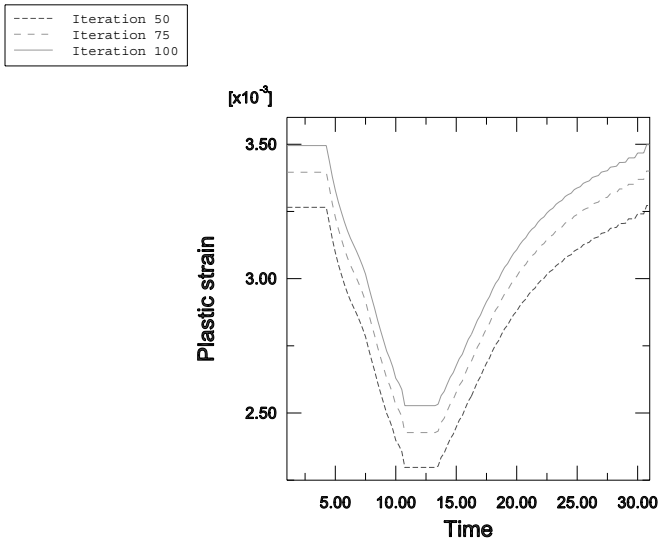


Figure 7: Evolution of the plastic strain component in the global 1-direction during iterations 50, 75, and 100 in the direct cyclic analysis.

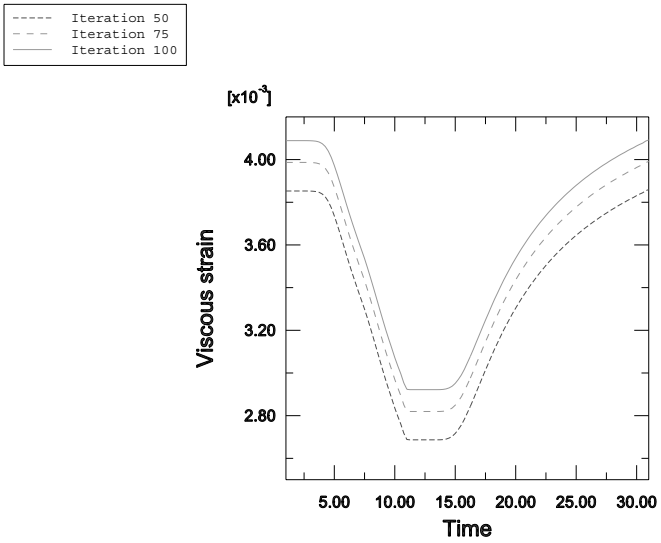


Figure 8: Evolution of the viscous strain component in the global 1-direction during iterations 50, 75, and 100 in the direct cyclic analysis.

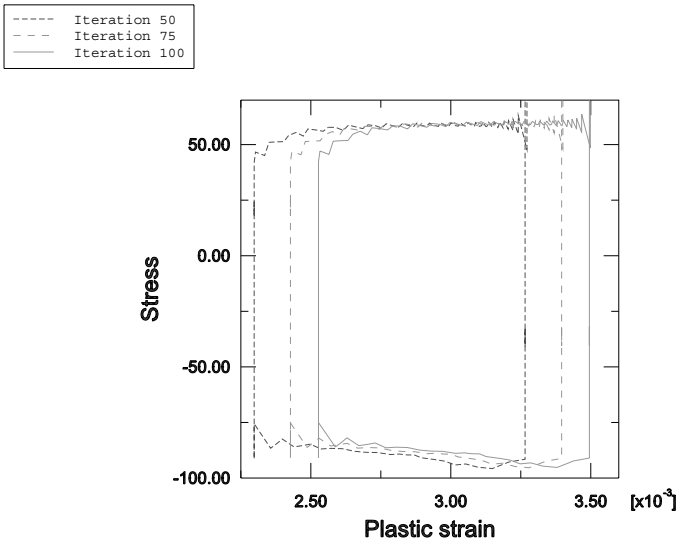


Figure 9: Evolution of the stress versus plastic strain during iterations 50, 75, and 100 in the direct cyclic analysis.

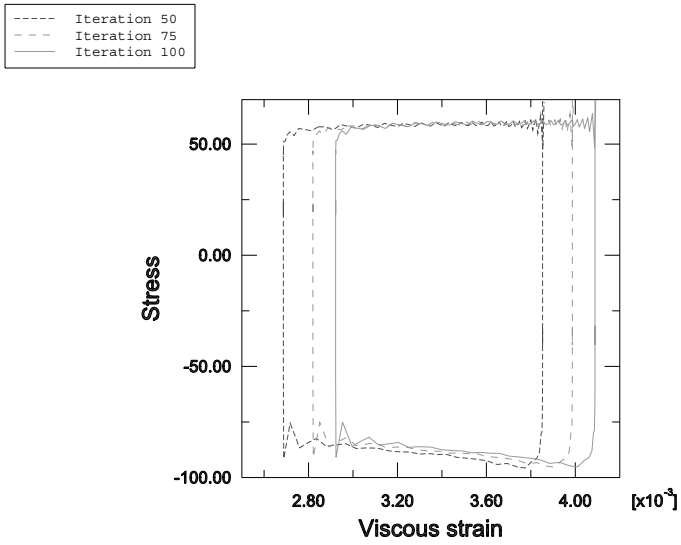


Figure 10: Evolution of the stress versus viscous strain during iterations 50, 75, and 100 in the direct cyclic analysis.

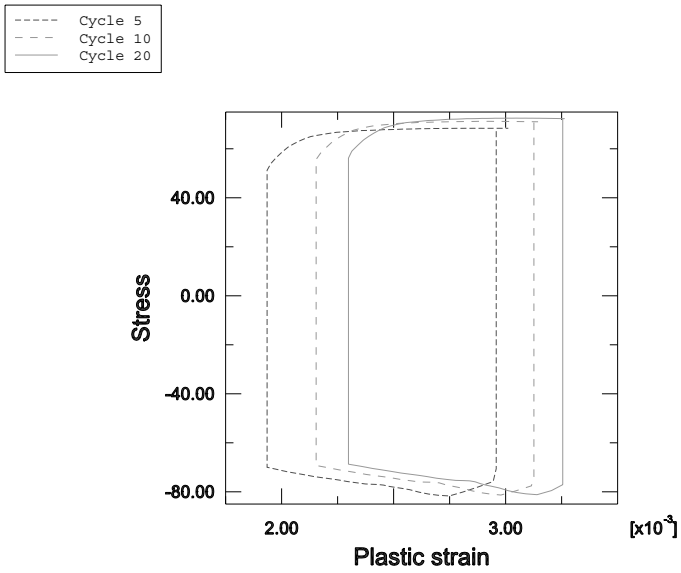


Figure 11: Evolution of the stress versus plastic strain during steps 5, 10, and 20 in the transient analysis.

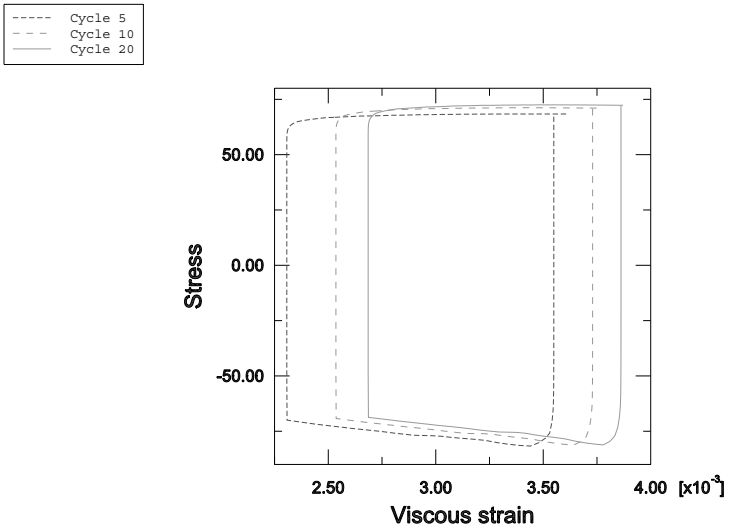


Figure 12: Evolution of the stress versus viscous strain during steps 5, 10, and 20 in the transient analysis.

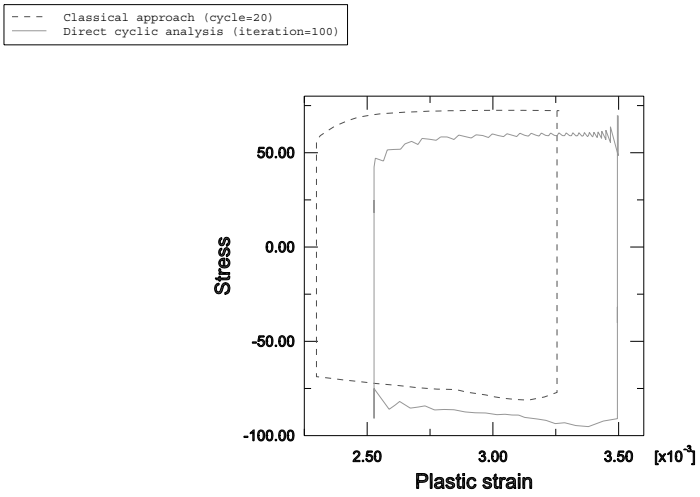


Figure 13: Comparison of the evolution of stress versus plastic strain obtained with the direct cyclic analysis and transient analysis approaches.

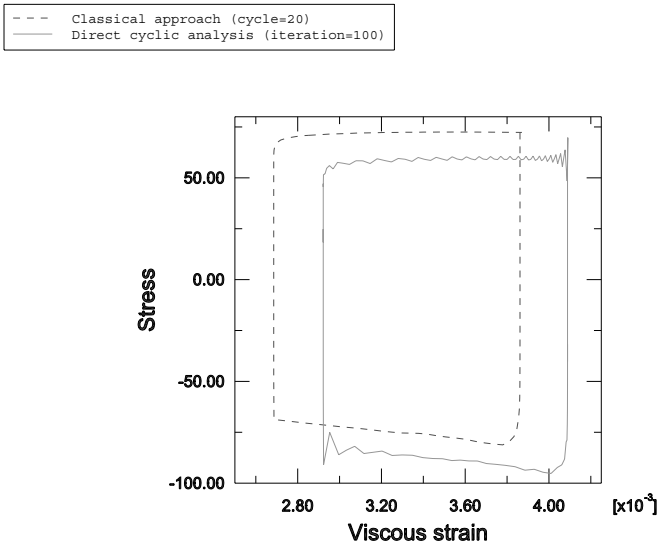


Figure 14: Comparison of the evolution of stress versus viscous strain obtained with the direct cyclic analysis and transient analysis approaches.

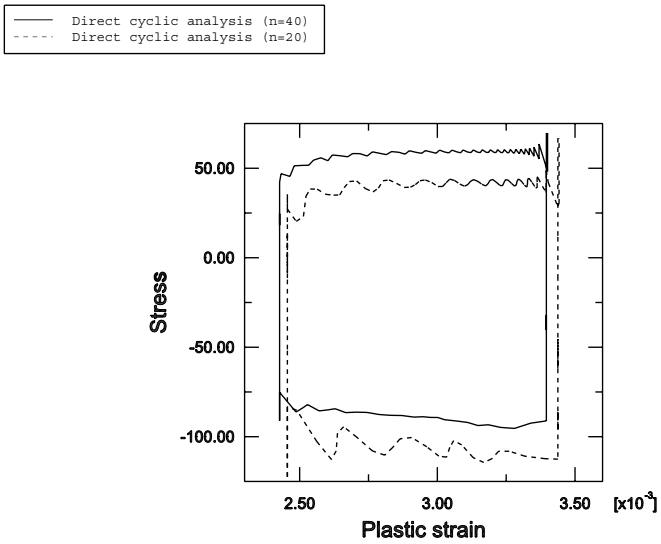


Figure 15: Comparison of the evolution of stress versus plastic strain obtained using different numbers of Fourier terms during iteration 75 in a direct cyclic analysis.

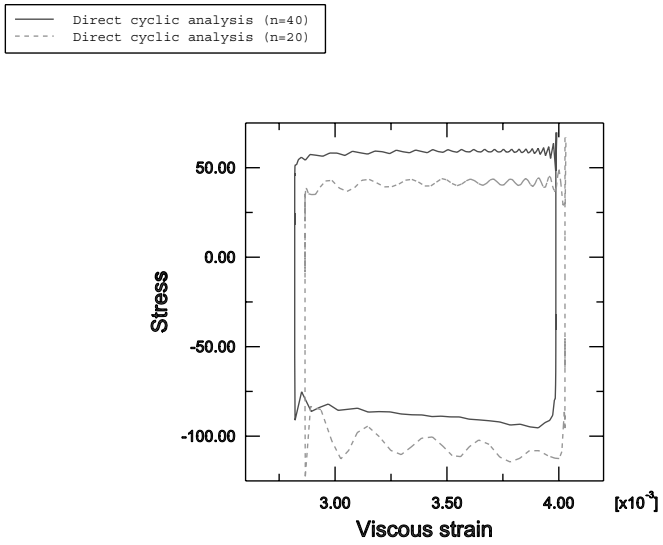


Figure 16: Comparison of the evolution of stress versus viscous strain obtained using different numbers of Fourier terms during iteration 75 in a direct cyclic analysis.

Erosion of material (sand production) in an oil wellbore

Products: Abaqus/Standard

Overview

This example demonstrates the use of adaptive meshing and adaptive mesh constraints in Abaqus/Standard to model the large-scale erosion of material such as sand production in an oil well as oil is extracted.

In Abaqus/Standard the erosion of material at the external surface is modeled by declaring the surface to be part of an adaptive mesh domain and by prescribing surface mesh motions that recede into the material. Abaqus/Standard will then remesh the adaptive mesh domain using the same mesh topology but accounting for the new location of the surface. All the material point and node point quantities will be advected to their new locations. This example also demonstrates the use of mesh-to-mesh solution mapping in a case where a new mesh topology is desired to continue the analysis beyond a certain stage.

Problem description

The process of optimizing the production value of an oil well is complex but can be simplified as a balance between the oil recovery rate (as measured by the volumetric flow rate of oil), the sustainability of the recovery (as measured by the amount of oil ultimately recovered), and the cost of operating the well. In practice, achieving this balance requires consideration of the erosion of rock in the wellbore, a phenomenon that occurs when oil is extracted under a sufficiently high pressure gradient. This erosion phenomenon is generally referred to as “sand production.” Depending on the flow velocities the sand may accumulate in the well, affecting the sustainability of recovery, or it may get carried to the surface along with the oil. The sand in the oil causes erosion in the piping system and its components such as chokes and pipe bends, increasing the costs of operating the well. Excessive sand production is, therefore, undesirable and will limit oil recovery rates. A typical measure of these limits on recovery rates is the so-called “sandfree rate.” A sandfree rate might be based on direct damage caused by sand production or might be based on the cost of sand management systems that limit the damage of the sand. The former measure is called the maximum sandfree rate, or MSR. The latter measure is called the acceptable sandfree rate, or ASR. The ASR concept has become possible with the availability of many commercial sand management systems as well as new designs of piping components. The ASR concept has also engendered a need for predicting the sand production rates to properly choose and size the sand management systems and piping components. In this example we focus on measures that can be obtained from Abaqus to predict these rates.

Geometry and model

The geometry of an oil well has two main components. The first is the wellbore drilled through the rock. The second component is a series of perforation tunnels that project perpendicular to the wellbore axis. These tunnels, which are formed by explosive shape charges, effectively increase the surface area of the wellbore for oil extraction. The perforation tunnels are typically arranged to fan out in a helical fashion around the wellbore, uniformly offset in both vertical spacing and azimuthal angle.

Three-dimensional model

The domain of the problem considered in the three-dimensional example is a 203 mm (8 in) thick circular slice of oil-bearing rock with both the wellbore and perforation tunnels modeled. The domain has a diameter of 10.2 m (400 in). The perforation tunnels emanate radially from the wellbore and are spaced 90° apart. Each perforation tunnel is 43.2 mm (1.7 in) diameter and 508 mm (20 in) long. The wellbore has a radius of 158.8 mm (6.25 in). Due to symmetry only a quarter of the domain that contains one perforation tunnel is modeled. *Figure 1* shows

the finite element model. The rock is modeled with C3D8P elements, and the well's casing is modeled with M3D4 elements.

Planar model

The planar model is a simplified version of the three-dimensional model, where the perforation tunnels and wellbore casing are neglected. The rock is modeled with CPE4P elements. The model domain consists of a quarter-symmetry square domain of length 10.2 m (400 in), with a single wellbore with a radius of 158.8 mm (6.25 in). *Figure 5* shows the finite element model. *Figure 6* details the region near the wellbore.

Material

A linear Drucker-Prager model with hardening is chosen for the rock, and the casing in the three-dimensional model is linear elastic.

Loading

The loading sequence for a wellbore analysis generally includes

- establishing geostatic equilibrium, based on the overburden loading;
- simulation of material removal operations, including drilling the wellbore and forming the perforation tunnel; and
- applying a drawdown pressure in the wellbore to simulate pumping.

This sequence is modeled slightly differently in the three-dimensional and planar models.

Three-dimensional model

The analysis consists of five steps. First, a geostatic step is performed where equilibrium is achieved after applying the initial pore pressure, the initial stress, and the distributed load representing the soil above the perforation tunnel. The second step represents the drilling operation where the elements representing the wellbore and the perforation tunnel are removed using the element removal capability in Abaqus (see *Element and Contact Pair Removal and Reactivation*). In the third step the boundary conditions are changed to apply the pore pressure on the face of the perforation tunnel. In the fourth step a steady-state soils analysis is carried out in which the pore pressure on the perforation tunnel surface is reduced to the desired drawdown pressure of interest. The fifth step consists of a soils consolidation analysis for four days during which the erosion occurs.

Planar model

The analysis consists of three steps. First, a geostatic step is performed where equilibrium is achieved after applying the initial pore pressure and the initial stress, representing an underbalanced state on the wellbore. The second step represents the drilling operation where the elements representing the wellbore are removed using the element removal capability, and the boundary conditions are changed to apply the pore pressure on the face of the perforation tunnel. The third step consists of a soils consolidation analysis for 32 hours during which the erosion occurs. As discussed below in *Rezoning the planar model*, this third step is interrupted in order to rezone the model.

Erosion criterion

There are two main sources of eroded material in a wellbore. One of the sources is volumetric and is due to the material that is broken up due to high stresses and transported by the fluid through the pores. The other source is surface based and is due to the material that is broken up by the hydrodynamic action of the flow on the surface. Depending on the properties of the oil-bearing strata and the flow velocities, one or the other may be the dominant

source of eroded material. Development of equations describing the erosion in a wellbore is an active research field. In this example we consider surface erosion only but choose a form for the erosion equation that has dependencies that are similar to those used by Papamichos and Stavropoulou for volumetric erosion. This approximation is reasonable because high stresses exist only in a very thin layer surrounding the wellbore. The erosion equation is

$$\mathbf{V}_e = \lambda (1 - n) c \, v_w,$$

where $\mathbf{V}_e$ is the erosion velocity, v_w is the pore fluid velocity, c is the transport concentration, n is the porosity, and λ is the so-called sand production coefficient. λ depends on the equivalent plastic strain ($\bar{\varepsilon}^{pl}$). It is zero below a cutoff equivalent plastic strain ($\bar{\varepsilon}^{pl}_{\text{cutoff}}$), is equal to $\lambda_1 \left(\bar{\varepsilon}^{pl} - \bar{\varepsilon}^{pl}_{\text{cutoff}} \right)$, and is limited by a constant λ_2.

Both λ_1 and λ_2 must be determined experimentally. In this example $\bar{\varepsilon}^{pl}_{\text{cutoff}} = 0.028$, $\lambda_1 = 4$, and $\lambda_2 = 0.01$. We choose $c = 0.001$ as recommended by Papamichos and Stavropoulo. These values are chosen to show visible erosion in a reasonably short analysis time.

Adaptive mesh domain

Erosion is modeled during the final step of each analysis. The erosion equation describes the velocity of material recession as a local function of solution quantities. Abaqus/Standard provides functionality through adaptive meshing for imposing this surface velocity, maintaining its progression normal to the surface as the surface moves, and adjusting subsurface nodes to account for large amounts of erosive material loss.

Erosion itself is described through a spatial adaptive mesh constraint, which is applied to all the nodes on the surface of the perforation tunnel. Adaptive mesh constraints can be applied only on adaptive mesh domains; in this example a sufficiently large extent of the finite element mesh near the wellbore and perforation tunnel surfaces is declared as the adaptive mesh domain. A cut section of the adaptive mesh domain for the three-dimensional model, including the perforation tunnel, is shown in *Figure 2*. The adaptive mesh domain for the planar model is the regular mesh near the wellbore (refer to *Figure 6* and *Figure 8*). Identification of the adaptive mesh domain will result in smoothing of the near-surface mesh that is necessary to enable erosion to progress to arbitrary depths. All the nodes on the boundary of the adaptive mesh domain where it meets the regular mesh must be considered as Lagrangian to respect the adjacent nonadaptive elements.

A velocity adaptive mesh constraint is defined. The generality and solution dependence of the erosion equation are handled by describing the erosion equation in user subroutine *UMESHMOTION* (see *Defining ALE Adaptive Mesh Domains in Abaqus/Standard*). User subroutine *UMESHMOTION* is called at a given node for every mesh smoothing sweep. Mesh velocities computed by the Abaqus/Standard meshing algorithm for that node are passed into *UMESHMOTION*, which modifies them to account for the erosion velocities computed at that node. The modified velocities are determined according to the equation for $\mathbf{V}_e$, where local results are needed for $\bar{\varepsilon}^{pl}$, n, and v_w. To obtain these values, we request results for output variables PEEQ, VOIDR, and FLVEL respectively, noting that void ratio is related to porosity by $(1 - n) = 1/(1 + \text{VOIDR})$. Since these output variables are all available at element material points, the utility routine `GETVRMAVGATNODE` is used to obtain results extrapolated to the surface nodes (see *Obtaining Material Point Information Averaged at a Node*).

Rezoning the planar model

Rezoning, the process of creating a new mesh in the model's deformed configuration, is a useful technique in addressing element distortion in erosion problems.

The erosion model in this example is solution dependent in the sense that it defines an erosion velocity as a function of local values of equivalent plastic strain, PEEQ, and fluid velocity, FLVEL. Since these node-located

results values are extrapolated from adjacent elements' material point results, using utility routine GETVRMAVGATNODE, the quality of the results used in the erosion model is dependent on local element quality. In practice, as elements near the surface deform, there is a tendency for instability in the erosion model.

This instability is mitigated by rezoning the model and providing a more regular mesh to continue the analysis. The rezoned model is created in Abaqus/CAE, and transfer of state variables occurs using mesh-to-mesh solution mapping in Abaqus/Standard. Rezoning occurs at 55,000 seconds (15.3 hours) into the 32-hour erosion analysis of the planar model.

Extracting two-dimensional profiles and remeshing using Abaqus/CAE

The rezoned model is created by extracting the two-dimensional profile of the deformed rock region from the output database for the original analysis. You perform this operation in Abaqus/CAE by entering commands into the command line interface at the bottom of the Abaqus/CAE main window. To extract the deformed geometry from the output database as an orphan mesh part, use the command PartFromOdb, which takes the following arguments:

name

The name of the orphan mesh part to be created.

odb

The output database object returned from the command `openOdb`.

instance

The name of the part instance in the initial model in capital letters.

shape

Determines whether to import the part in its UNDEFORMED or DEFORMED shape.

The command PartFromOdb returns a Part object that is passed to the command Part2DGeomFrom2DMesh. This command creates a geometric Part object from the orphan mesh imported earlier. It takes the following arguments:

name

The name of the part to be created.

part

The part object returned from the command PartFromOdb.

featureAngle

A float specifying the angle (in degrees) between line segments that triggers a break in the geometry.

Once the profile of the deformed part has been created, you will prepare an input file for the subsequent period in the erosion analysis as follows:

- Reestablish all attributes that relate to the geometry of the deformed part. These attributes include load and boundary condition definitions, and set and surface definitions.
- Remesh the part.

- Create a single consolidation step that completes the duration of your intended erosion period.
- Write out the new input file.

Mesh-to-mesh solution mapping in Abaqus/Standard

The interpolation technique used in solution mapping is a two-step process. First, values of all solution variables are obtained at the nodes of the old mesh by extrapolating the values from the integration points to the nodes of each element and averaging those values over all elements abutting each node. The second step is to locate each integration point in the new mesh with respect to the old mesh. The variables are then interpolated from the nodes of the element in the old mesh to the location in the new mesh. All solution variables are interpolated automatically in this way so that the solution can proceed on the new mesh. Whenever a model is mapped, it can be expected that there will be some discontinuity in the solution because of the change in the mesh. To address this discontinuity the rezone analysis includes a geostatic step, which reestablishes equilibrium before the erosion process continues in a following step.

Results and discussion

The analysis scenarios and results differ between the three-dimensional and planar analyses.

Three-dimensional analysis

The consolidation analysis in which erosion takes place is run for a time period of four days to observe the initiation of sand production and predict its initial rate. *Figure 3* shows the perforation tunnel at the end of four days where it is seen that the largest amount of material is eroded near the junction of the wellbore and perforation tunnel. Further away from the wellbore boundary the amount of erosion progressively decreases. This behavior is expected because there are high strains near the junction of the wellbore and perforation tunnel, and the erosion criterion is active only for values of the equivalent plastic strain above a threshold value.

The amount of the volume change due to erosion in an adaptive domain is available using the history output variable VOLC. The actual amount of the solid material eroded depends on the porosity of the rock and is

obtained by multiplying VOLC by $(1 - n)$. *Figure 4* shows the volume change of the sand produced in cubic inches over the time period of the consolidation step. As the material consolidates, the erosion rate slows down. The stresses in the perforation hole reduce and stabilize over the time period. From *Figure 4* it can be concluded that the stresses generated by the drawdown pressure, the fluid velocities, the wellbore and casing geometry, and the initial perforation tunnel geometry are such that this perforation tunnel will produce sand at a more stable rate as oil recovery continues; however, this rate could vary with further changes to the perforation tunnel caused by the erosion. At a design stage any of these parameters could be modified to limit the sand production rate. Many perforation tunnels emanate from a wellbore, and the total sand production from the wellbore will be the sum total of all the perforation tunnels.

Planar analysis

The underbalanced consolidation analysis in which erosion takes place is run for a time period of 32 hours to observe the initiation of sand production and predict its initial rate. *Figure 7* shows the wellbore at 15 hours, the end of the first analysis job. Based on this configuration a new mesh is created, as shown in *Figure 8*. The final configuration of this model, representing 32 hours of erosion, is shown in *Figure 9*. As expected, the results show that erosion continues outward from the location of the maximum equivalent plastic straining. *Figure 10* shows the total amount of sand produced, in cubic inches per inch of depth, over the time period of the consolidation step.

Input files

exa_erosion.inp	Three-dimensional model of the oil wellbore perforation tunnel.

exa_erosion_planar.inp	Planar model of the oil wellbore.
exa_erosion_planar_rezone.inp	Rezone analysis of the planar model.
exa_erosion.f	*UMESHMOTION* user subroutine.

References

- Papamichos, E., and M. Stavropoulou, "An Erosion-Mechanical Model for Sand Production Rate Prediction," *International Journal of Rock Mechanics and Mining Sciences*, no. 35, pp. 4–5, Paper No 090, 1998.

Figures

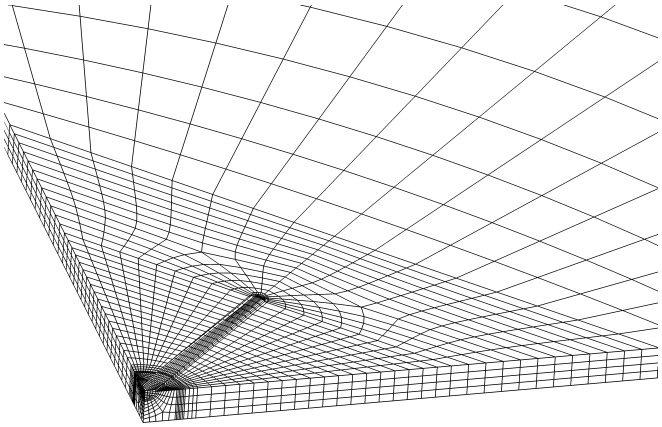


Figure 1: A cut section of the model showing the wellbore and perforation tunnel.

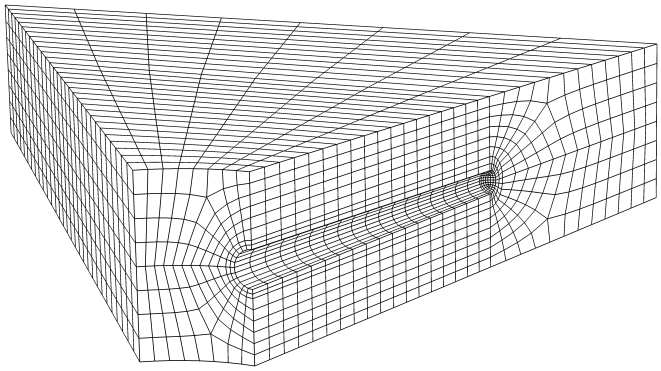


Figure 2: Half-section of the adaptive mesh domain showing the wellbore face and the perforation tunnel.

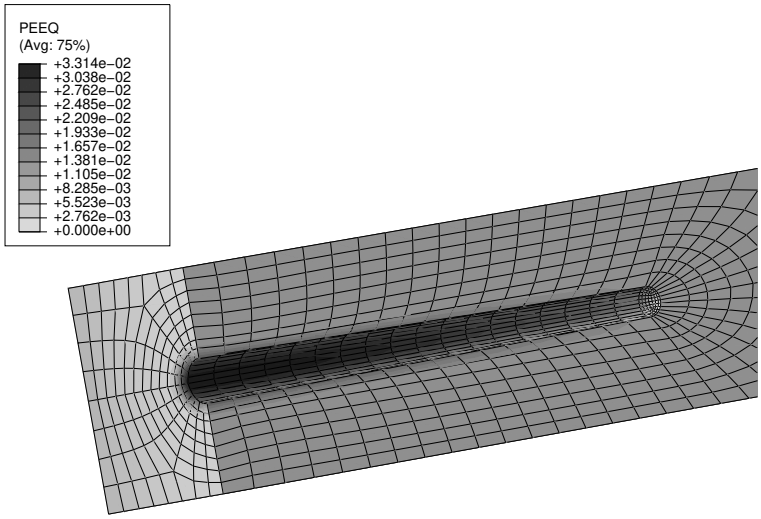


Figure 3: Shape of the perforation tunnel after four days of erosion.

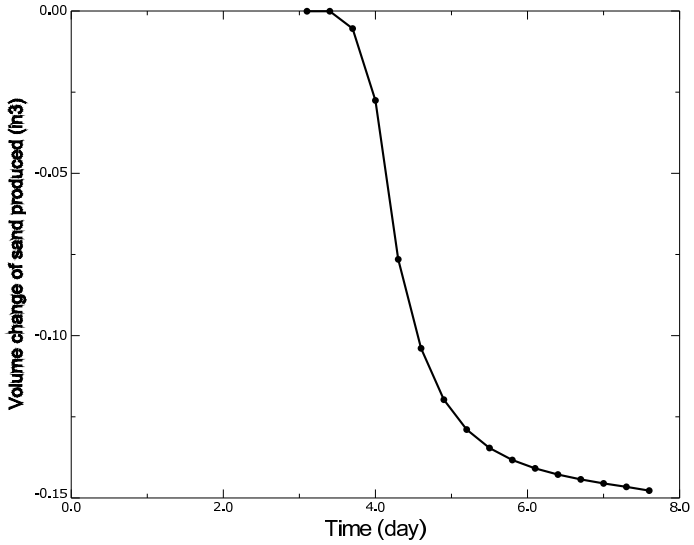


Figure 4: Total sand production volume change in a single perforation tunnel indicating a stabilized rate as the consolidation continues.

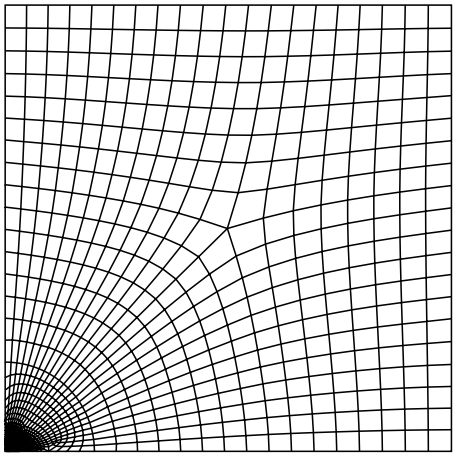


Figure 5: Planar model mesh.

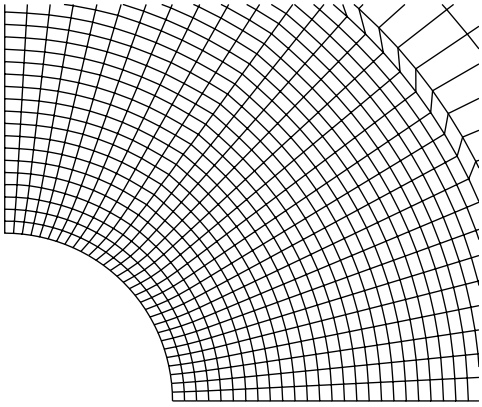


Figure 6: Planar model mesh: wellbore region detail.

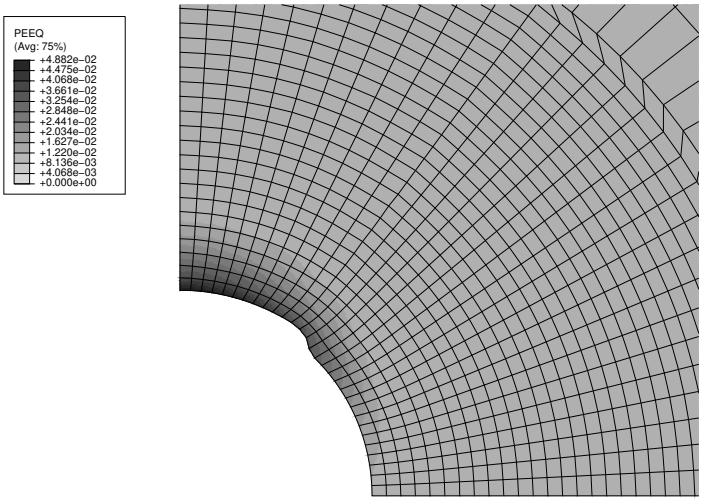


Figure 7: Equivalent plastic strain distribution and eroded shape of the wellbore.

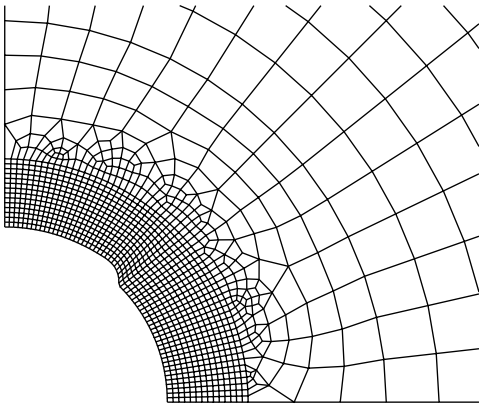


Figure 8: Planar model mesh: wellbore region detail: rezone analysis.

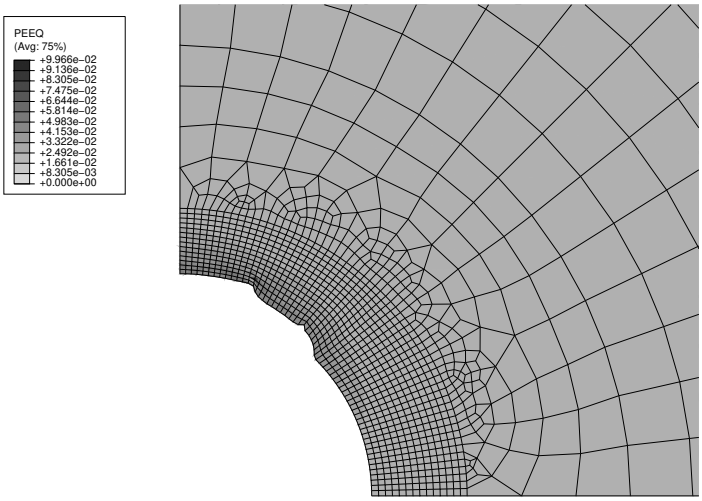


Figure 9: Equivalent plastic strain distribution and eroded shape of the wellbore: rezone analysis.

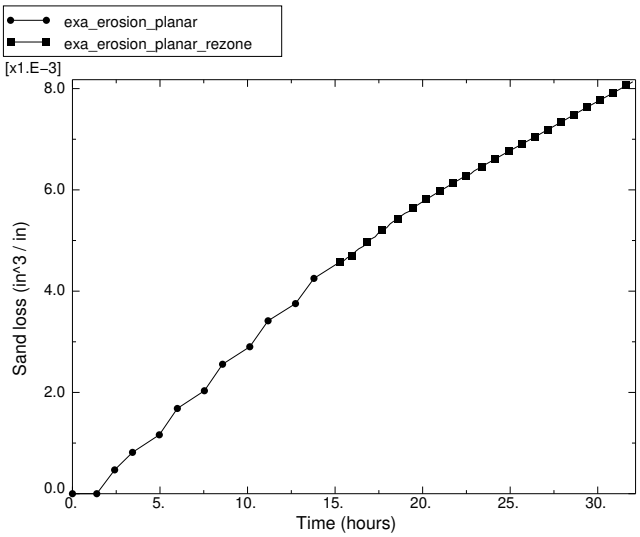


Figure 10: Total sand production volume per unit of depth.

Submodel stress analysis of pressure vessel closure hardware

Products: Abaqus/Standard

References:

- *About Submodeling*
- *Node-Based Submodeling*
- *Surface-Based Submodeling*
- **BOUNDARY*
- **DSLOAD*
- **SUBMODEL*

Overview

This example demonstrates the use of surface-based submodeling as a technique to obtain more accurate solutions than those obtained using node-based submodeling.

The particular scenarios studied include cases where:

- the submodel displacement field is expected to differ from the global model displacement field by a rigid translation and
- the geometry of the submodel differs from the global model in a region whose response is primarily load controlled.

Application description

This example examines the stress behavior of closure head standpipe structures in a nuclear reactor vessel closure assembly. The vessel assembly forms the pressure boundary surrounding the fuel core. This example considers the following loading conditions:

- pre-tension load in the stud bolts,
- constant internal pressure, and
- loading due to the control rod drive mechanism (CDM) plug.

The loading conditions cover the most basic structural operation of a reactor vessel. The International System of units (SI) is used in the following sections to describe the model. The analysis itself is performed in English units. The model and analysis are derived from details of the Shippingport pressurized water reactor (1958).

Geometry

The problem domain comprises a cylindrical vessel shell, a hemispherical bottom head, a dome-shaped closure head, and the closure and seal assembly, as shown in *Figure 1*. The overall height of the vessel shell including the bottom head is 7650 mm (301 in). The bottom head has an inner radius of 1410 mm (55.5 in) and a thickness of 157 mm (6.18 in). The inlet nozzles on the bottom head are not considered in this example. The inner radius of the vessel shell is 1380 mm (54.5 in), and the thickness is 213 mm (8.40 in). The closure head has a height of 2330 mm (91.8 in), an inner radius of 1310 mm (51.5 in), and a thickness of 210 mm (8.25 in).

The closure head includes eight standpipes with CDM plugs inserted in each. The standpipes have an inside diameter of 472 mm (12 in) and an outside diameter of 630 mm (16 in) and extend roughly 1000 mm (25 in) above the closure head. The CDM plugs have an outside diameter of 465 mm (11.8 in) and a flange diameter of

630 mm (16 in) and are 748 mm (19 in) tall. Each CDM plug sits in a closure head standpipe on a 404 mm (10.25 in) diameter ledge.

The closure head is attached to the vessel shell by a seal and closure assembly. The assembly includes 40 stud bolts passing through the bolting flanges of the closure head and the vessel shell, each of which is restrained by two cap nuts (one on each end). Each stud bolt is 2290 mm (90 in) in length and has a diameter of 146 mm (5.75 in). The closure nuts are 304 mm (12 in) long with a thickness of 28.6 mm (1.13 in). To complete the closure assembly, an Omega seal is welded to the under surface of the closure head and top surface of the vessel shell.

Materials

All components are constructed of high-strength steel.

Boundary conditions and loading

A pre-tension load of 2200 kN (5×10^6 lbf) is applied to each bolt. The inner surfaces of the head and the vessel shell are subject to a constant pressure of 1.38×10^7 Pa (2000 psi) from the water.

Interactions

Contact occurs between

- the reactor vessel and closure head,
- the lower nuts and the reactor vessel bolting flange,
- the upper nuts and the closure head bolting flange, and
- the closure head standpipe and CDM plug.

Model terminology

This example illustrates the use of the submodeling technique in ways that are generalizations of the Abaqus user interface concept of a global model driving the response of a submodel. Specifically, some submodel analyses described in this section represent material domains that are adjacent to, rather than lying within, the domain considered in the initial analysis. To more clearly describe the models in this example, the term “source model” is used instead of “global model.” A source model is a model that provides solution results to a subsequent submodel analysis.

Abaqus modeling approaches and simulation techniques

The objective of the analyses in this example is an understanding of stresses in the region of the closure head standpipe.

Summary of analysis cases

Case 1 Reactor closure analysis: reference solution	The stress distribution in the closure head is determined from a single, finely meshed model that includes closure head standpipe and CDM plug details.
Case 2 Submodeling of the closure head standpipe region	A defeatured source model of the vessel assembly is analyzed first. Features excluded are the closure head standpipes and the CDM plugs. This model then drives a submodel with a more detailed representation of the closure head standpipe region.
Case 3 Submodel application of CDM hardware loading	The CDM plug is analyzed separately as a source model. Boundary conditions are introduced on the CDM plug where it interacts with the closure head standpipe seating ledge to determine the surface traction characteristics at this interface. This model is then used to drive a submodel of the remaining vessel assembly.

In the two submodel analysis cases the node-based submodeling technique, in which the submodel is driven with displacements, is compared to the surface-based submodeling technique, in which the submodel is driven with stresses. The three cases are discussed in more detail below.

Case 1 Reactor closure analysis: reference solution

This reference case determines the stress response of the reactor vessel assembly when subjected to boltup and pressure loading using a single analysis. By comparison, the other modeling cases make use of submodeling. The reactor vessel assembly is cyclic-symmetric with respect to the axis of the cylindrical vessel body and only one-quarter of the whole assembly is modeled. The global geometry is shown in *Figure 2*.

Analysis types

A static stress analysis is performed.

Mesh design

The vessel is meshed with C3D20R elements, and the closure head is meshed with C3D10M elements. All other parts including the head, bolts, and Omega seal are meshed with C3D8R elements. The mesh is shown in *Figure 3*.

Materials

The elastic model is used in all models with a Young's modulus of 2.07×10^{11} N/m^2 (3.0×10^7 lbf/in^2) and a Poisson's ratio of 0.29.

Boundary conditions

Symmetric boundary conditions are applied to the two side surfaces of the vessel quarter. The nodes on the centerline are constrained separately and are free to move only along the vessel central axis. The node located at the center of the vessel bottom surface is pinned to give the model a statically determinant condition.

Loads

A pre-tension load of 2200 kN (5×10^6 lbf) is applied to each stud bolt in the model. The inner surfaces of the head, the vessel body, and the nozzle are subject to a constant pressure of 1.38×10^7 Pa (2000 psi) from the water.

Bolting of the CDM plug to the closure head standpipe, defined as the CDM assembly, is simulated in this analysis by the application of a pair of concentrated loads acting through distributing coupling constraints. Refer to *Figure 4* for identification of the loaded regions of each CDM assembly. For each CDM plug one coupling constraint acts on the top surface of the plug (region B). For each standpipe one coupling constraint acts across the top surface (region C). The reference node for each coupling constraint is positioned along the center axis of the CDM assembly so that vertical concentrated loads can be applied without generating overturning moments. A bolting force of 106 kN (2.4×10^5 lbf) for each CDM assembly is chosen as adequate to overcome the liftoff force due to the vessel internal pressure. This force is applied vertically in the up direction to the standpipe coupling constraint. A downward force is applied to the accompanying coupling constraint on the CDM plug, but this force is lessened by the amount of the pressure-generating liftoff force due to the operating pressure acting on region A (shown in *Figure 4*), since the capping of this region is not considered explicitly in the model. Based on the diameter of region A, this pressure liftoff force equals 99 kN (2.26×10^5 lbf).

Constraints

The Omega seal is tied to the flange surfaces of the vessel head and the vessel body. As mentioned above, distributing coupling constraints are applied to the CDM plugs and the closure head standpipes.

Interactions

Small-sliding contact definitions are prescribed between

- the reactor vessel and closure head,
- the lower nuts and the reactor vessel bolting flange,
- the upper nuts and the closure head bolting flange, and
- the CDM plug and the accompanying seating ledge in the closure head.

Analysis steps

The analysis is performed in a single static step with automatic stabilization to help establish the contact between the stud bolts, head, seal, and vessel. Results show that the static dissipation energy is minimal compared to the strain energy; therefore, its effect on the response can be neglected.

Output requests

Default field and history output requests are specified in the step.

Results and discussion

This case is provided as a reference. Submodel analysis results are compared to these results in the discussion of Case 2 and Case 3 below.

Case 2 Submodel analysis of the closure head standpipe region

This case is representative of the most common submodel analysis approach: a global analysis of a coarse source model followed by a detailed submodel analysis representing a smaller region of the source model. Here, the coarse source model excludes details of the CDM plug and closure head standpipe as an illustration of a source model with significant defeaturing—a common motivation for subsequent submodel analysis. The submodel comprises a portion of the closure head and two CDM plugs, using a finer mesh and with feature details included. The relation between the source model and submodel is shown in *Figure 5*.

Analysis types

A static stress analysis is performed.

Mesh design

In the source model the vessel and closure head are meshed with C3D20R elements; all the other parts including the head, bolts, and Omega seal are meshed with C3D8R elements. The source model mesh is shown in *Figure 6*.

For the submodel the CDM plug is meshed with C3D8R elements and the closure head is meshed with C3D10M elements. The submodel mesh is shown in *Figure 7*.

Materials

The material model is the same as in Case 1.

Boundary conditions

The source model boundary conditions reflect those applied in Case 1. Similarly, the submodel has symmetric boundary conditions applied to the two side surfaces of the closure head. In the node-based submodel analysis,

submodel boundary conditions are applied to the submodel boundary. In the surface-based submodel analysis, a boundary condition is applied in the 2-direction on the coupling constraint for each CDM plug in the closure head to suppress the rigid body mode.

Loads

The source model loads reflect those applied in Case 1 except for the bolting of the CDM plugs to the closure head, which is introduced in the submodel.

In the surface-based submodel analysis, submodel distributed loads are applied to the submodel boundary surface.

Constraints

Distributing coupling constraints are applied to the CDM plugs and the closure head standpipes, as in Case 1.

Interactions

Contact interactions are the same as in Case 1.

Analysis steps

The global analysis of the source model is performed in a single static step with automatic stabilization to help establish the contact between the stud bolts, head, seal, and vessel. The submodel analysis is performed in a single static step.

Output requests

Default field and history output requests are specified in the step.

Results and discussion

Von Mises stress results are compared along paths in two regions in the closure head:

- the first comparison is made along a ligament through the closure head shell, as shown in *Figure 8*; and
- the second comparison is made along a circular path in the vicinity of the CDM hardware seating ledge, as shown in *Figure 9*.

By reviewing the stress distribution comparisons (discussed below), you can see that in this case, surface-based submodeling is superior to node-based submodeling for results lying within the main closure head shell. In the upper region of the standpipe, neither method provides adequate results indicating that the level of defeaturing in the source model is too great for an accurate submodel analysis of the standpipe region.

Closure head shell ligament

Figure 10 compares the von Mises stress distribution on the path shown in *Figure 8* for the reference model, the node-based submodel solution, and the surface-based submodel solution. These results show that the surface-based submodel solution provides a more accurate stress distribution than the node-based submodel technique in this region. This result is consistent with the guidelines documented in *Surface-Based Submodeling*, namely that a surface-based solution is more accurate in cases where the environment is load controlled—the vessel pressurization dominates the closure head response in the shell region—and the submodel geometry differs from the source model geometry—the source model does not include the standpipe detail.

In practice, the classification of an analysis according to these guidelines, particularly the classification of load-controlled vs. displacement-controlled, is often not obvious nor is the reference solution available for comparison. Therefore, you should always compare measures of interest between the source model and the submodel on or near the submodel driven boundary and confirm that they show reasonable agreement. In this

case the von Mises stress is compared along a path, shown in *Figure 7*, cutting across the submodel driven boundary. In *Figure 11* the results comparison shows that the surface-based submodel solution provides a stress distribution on the submodel boundary that more closely matches that for the global solution of the source model. This plot also shows the reference solution, which shows better agreement with the surface-based submodel solution at the outer edge of the shell and better agreement with the node-based submodel solution at the inner edge of the shell. Hence, the agreement between global model and submodel stress distributions, while necessary, is not sufficient to confirm an adequate submodel solution at all locations. You must also use judgment as to whether geometric differences are too great between the source model and submodel.

Standpipe seating ledge

Figure 12 compares the von Mises stress distribution on the path shown in *Figure 9* for the reference model, the node-based submodel solution, and the surface-based submodel solution. The stress results near the seating ledge show that neither submodeling technique is clearly superior or provides adequate accuracy. This follows from the fact that the standpipe and seating ledge region did not appear at all in the source model analysis; the defeaturing was too severe in this case for an adequate submodel solution in this region.

This seating ledge stress comparison makes it clear that although a favorable comparison of results on the submodel boundary, as was done in *Figure 11*, is necessary, it is not sufficient to ensure adequate submodel results in all locations in the model. In this case the seating ledge region was absent entirely from the defeatured source model, and you should not expect accurate results in this region.

Case 3 Submodel application of CDM plug loading

This case represents an atypical use of submodeling in which the source model is associated with a small part of the structure and the submodel comprises most of the overall structure. Here, the source model focuses on the CDM plugs to predict how each of the CDM plugs loads the closure head.

The subsequent submodel analysis uses results from the source analysis for loading the remainder of the structure. The regions considered for the source model and submodel are shown in *Figure 13*.

Mesh design

In the source model the CDM plug hardware is meshed with C3D8R elements. The source model mesh is shown in *Figure 13*. The geometry for the remaining structure is also shown in this figure to illustrate the plug positioning relative to the overall reactor assembly.

The submodel mesh is nearly identical to that shown in *Figure 3* for Case 1. The only difference is that the CDM plugs are excluded from the model in Case 3.

Boundary conditions

The submodel analysis of the CDM plug source model simulates contact with the seating ledge with a boundary condition constraint on the plug seating surface.

Loads

The loading follows that for Case 1 with the application of the loads split between the source model and submodel.

Source model analysis

The bolt load applied to the CDM plug is simulated through a downward concentrated force applied to a distributing coupling constraint reference node in each of the CDM plugs. The magnitude of this force is the bolting force of 106 kN (2.4×10^5 lbf) less the pressure generating liftoff force of 99 kN (2.26×10^5 lbf), for the reasons detailed in the Case 1 description.

Submodel analysis

A pre-tension load of 2200 kN (5×10^6 lbf) is applied to each stud bolt in the model. The inner surfaces of the head, the vessel body, and the nozzle are subject to a constant pressure of 1.38×10^7 Pa (2000 psi) from the water.

The bolt load applied to the standpipe is simulated through an upward concentrated force applied to a distributing coupling constraint reference node in each of the standpipes. The magnitude of this force is the bolting force of 106 kN (2.4×10^5 lbf) less the pressure generating liftoff force of 99 kN (2.26×10^5 lbf).

Constraints

All constraint definitions are the same as in Case 1.

Interactions

All interaction definitions are the same as in Case 1, except that the contact interaction between the CDM plug and the standpipe seating ledge is effected through submodel loads and boundary conditions.

Run procedure

Run the analyses with the input files listed for Case 3 below.

Results and discussion

Stress results are considered on the same paths defined for comparison of reference and submodel results in Case 2. The von Mises stress distribution on these paths is compared in *Figure 14* and *Figure 15* for the two forms of submodeling and the reference solution.

These results show that in both the high-stressed region, shown in the ligament stress plot, and in the vicinity of the seating ledge, the surface-based submodeling approach provides a more accurate solution. The poor results for node-based submodeling follow from the fact that the assembly model—the submodel in this case—elongates along the vessel main axis. The CDM assembly region experiences this elongation as a rigid body translation. The standpipe seating ledge, however, is constrained in its movement by the submodel boundary conditions. These boundary conditions follow from the separate analysis of the CDM plug source model that does not consider the solution-dependent elongation of the vessel assembly.

Discussion of results and comparison of cases

Case 2 and Case 3 illustrate situations where you may see improved accuracy when using the surface-based submodeling approach.

The effect of stiffness change on submodel analysis

In cases where the submodel stiffness matches that of the source model, you can expect, using reasonable modeling practices, that the submodel analysis will provide adequate results. In cases where the submodel stiffness differs, such as in Case 2, you must exercise caution in evaluating your submodel solution. Comparison of stress contours on the common boundary of the source model and submodel can aid you in determining if your solution is adequate. In the case of significant defeaturing, you should not rely on the submodeling analysis technique in any form for detailed stress response in areas absent from the source model, such as the closure head standpipe.

The effect of displacement discrepancies on submodel analysis

In cases where you expect that the submodel displacement solution will differ from the corresponding source model solution by only a rigid body motion, such as in Case 3, you can expect that a node-based submodeling

approach will give incorrect results. In this case you can use the alternative surface-based submodeling of stresses and obtain improved solution accuracy.

Files

Case 1 Reactor closure analysis: reference solution

ReactorHead_reference.inp	Input file to analyze the reactor vessel closure assembly.

Case 2 Submodeling of the closure head standpipe region

ReactorHead_global.inp	Global analysis of the reactor vessel closure assembly source model with a defeatured closure head.
ReactorHead_submodel_node.inp	Closure head submodel analysis using node-based submodeling.
ReactorHead_submodel_surface.inp	Closure head submodel analysis using surface-based submodeling.

Case 3 Submodel application of CDM plug loading

ReactorHead_CDMdetail.inp	Global analysis of the CDM plug source model.
ReactorHead_assembly_node.inp	Reactor vessel closure assembly submodel analysis with CDM loading effected through node-based submodeling.
ReactorHead_assembly_surface.inp	Reactor vessel closure assembly submodel analysis with CDM loading effected through surface-based submodeling.

References

- Naval Reactors Branch, Division of Reactor Development, United States Atomic Energy Commission,*The Shippingport Pressurized Water Reactor,* Reading, Massachusetts: Addison Wesley Publishing Company, 1958.

Figures

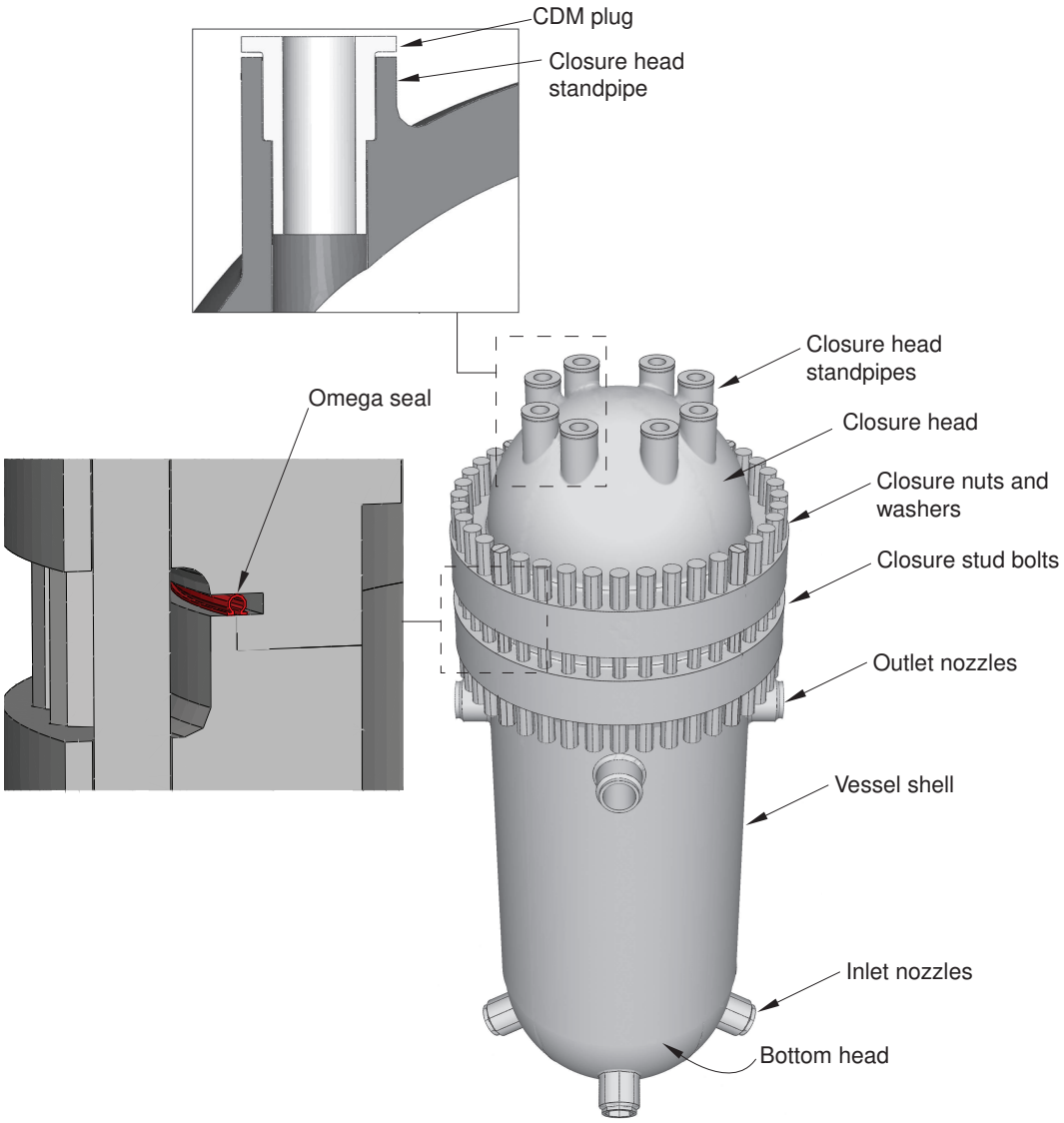


Figure 1: Reactor vessel assembly.

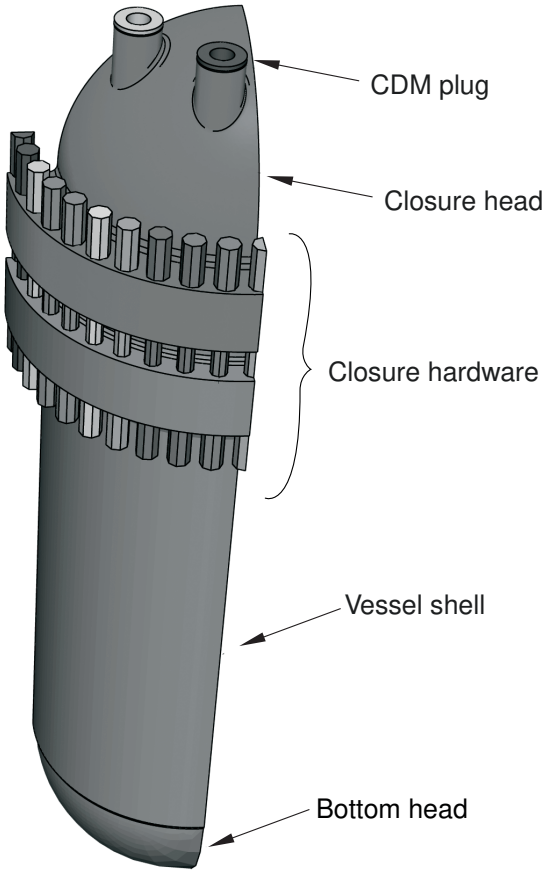


Figure 2: Reactor vessel assembly model.

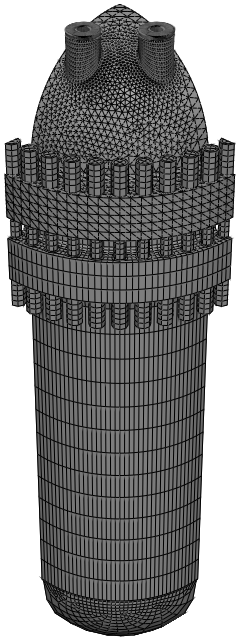


Figure 3: Reference analysis mesh.

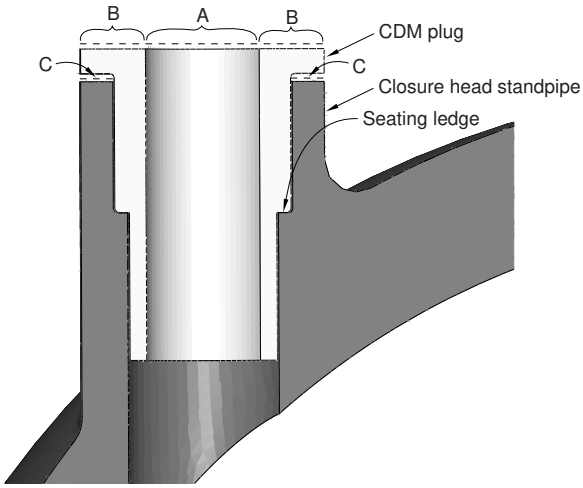


Figure 4: Load application areas on the CDM plug and closure head standpipe.

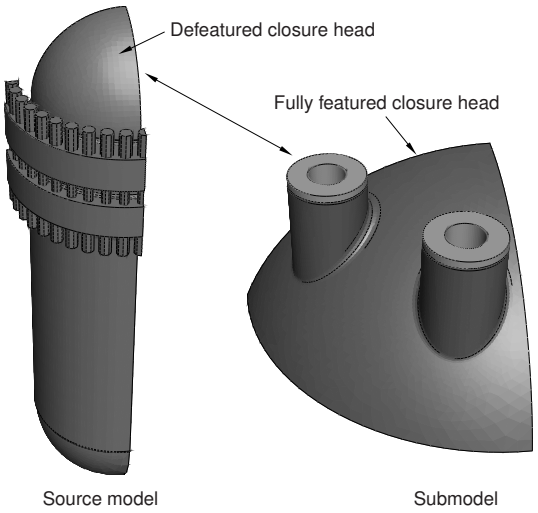


Figure 5: Case 2 closure head submodel relation to the source model.

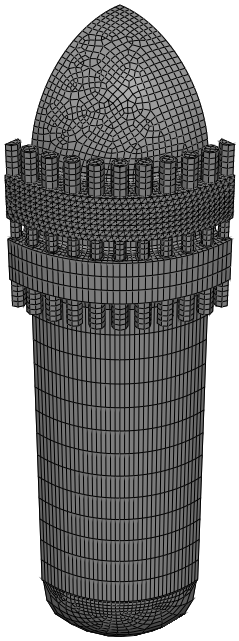


Figure 6: Case 2 global analysis mesh with defeatured closure head.

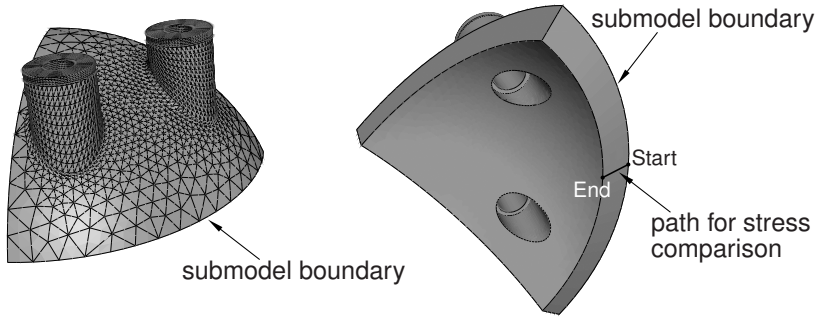


Figure 7: Case 2 submodel analysis showing the mesh (left) and the path definition for stress comparison to the source model (right).

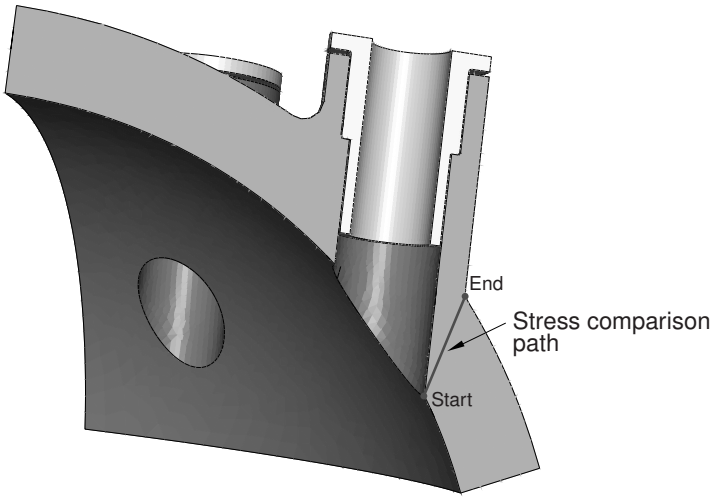


Figure 8: Through-ligament path definition.

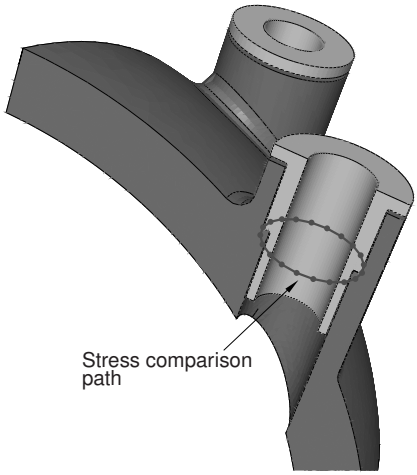


Figure 9: Seating ledge path definition.

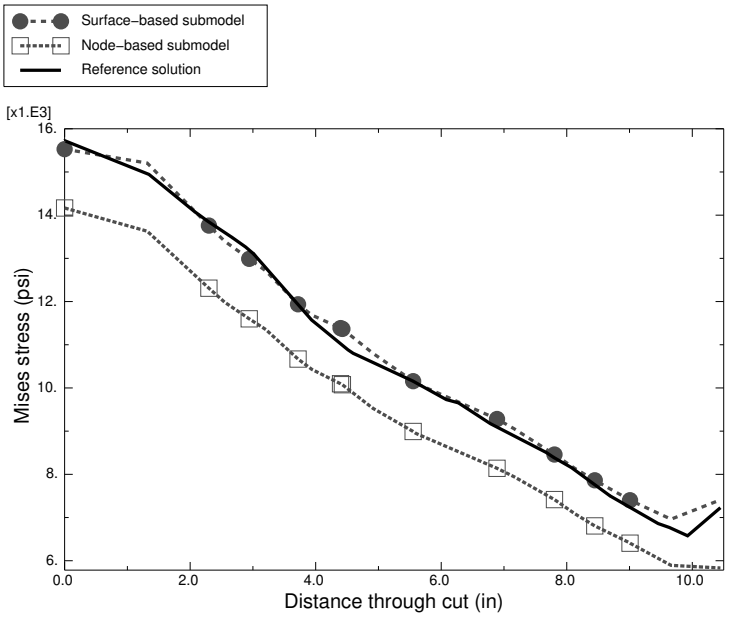


Figure 10: Case 2 stress distribution comparison through the closure head ligament.

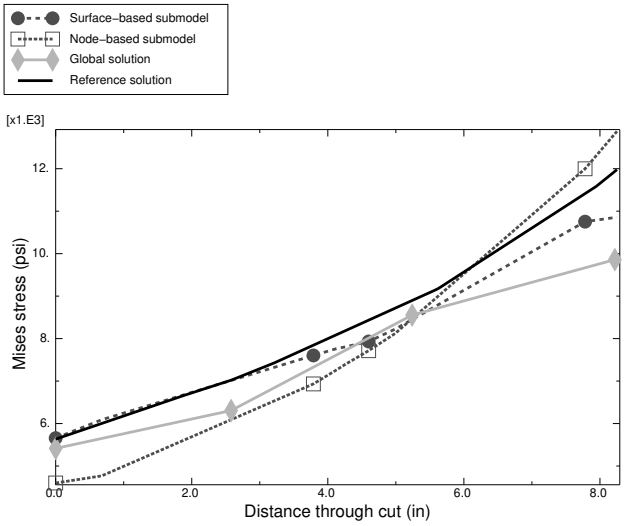


Figure 11: Comparison of the closure head through-thickness von Mises stress distribution at the location of the submodel boundary.

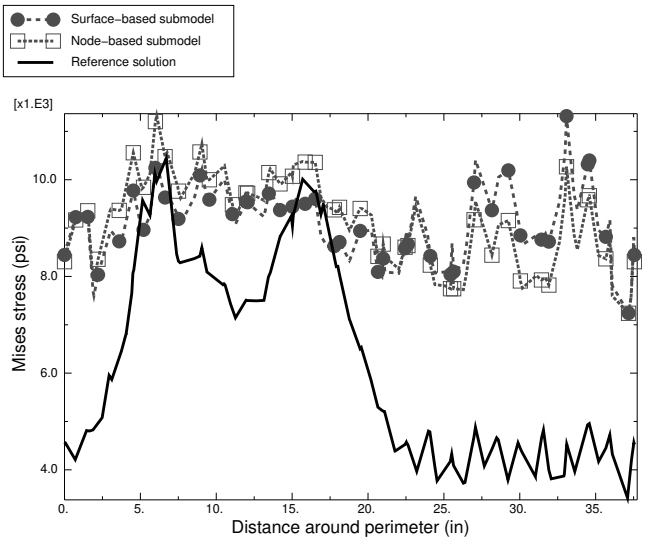


Figure 12: Case 2 stress distribution comparison around the seating ledge.

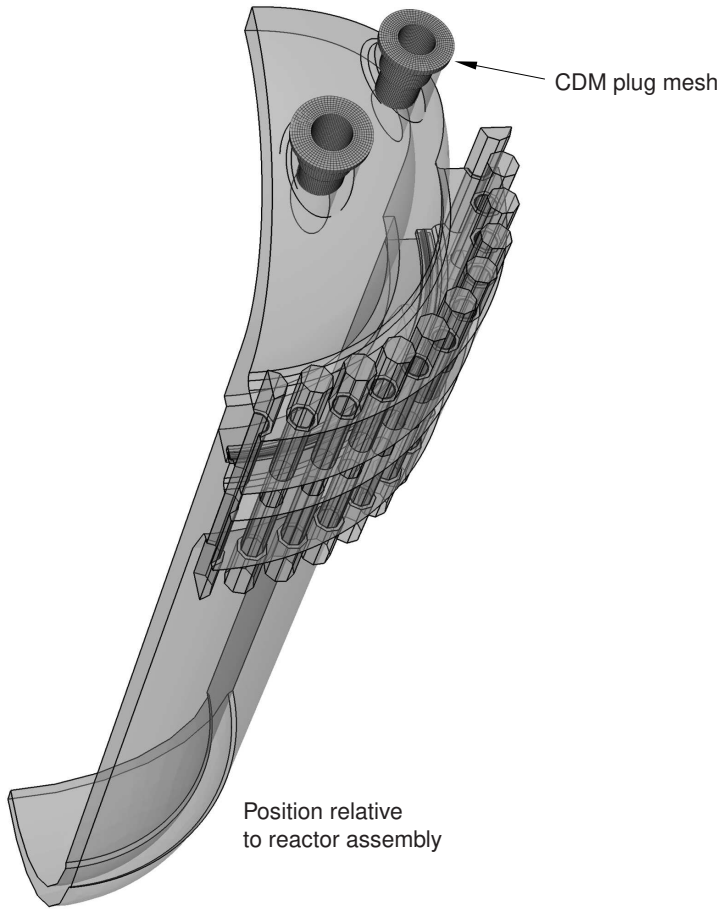


Figure 13: Case 3 CDM plug analysis mesh.

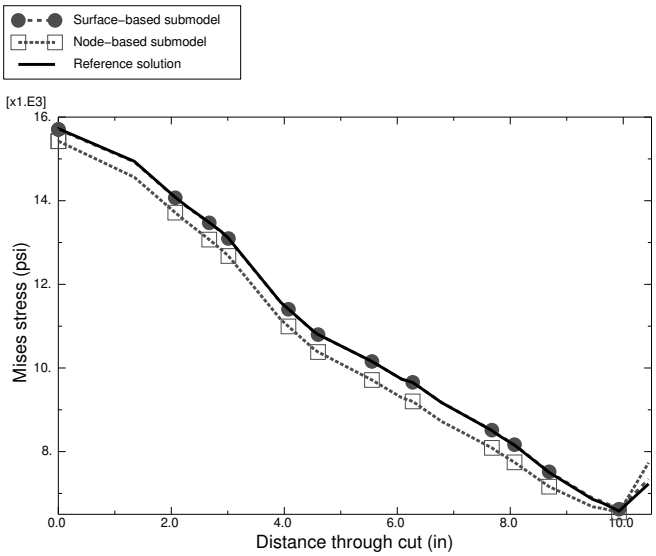


Figure 14: Case 3 stress distribution comparison through the closure head ligament.

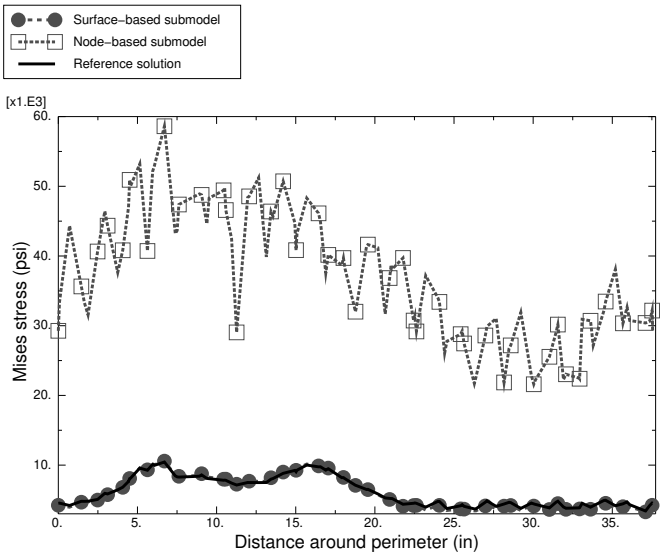


Figure 15: Case 3 stress distribution comparison around the seating ledge.

Using a composite layup to model a yacht hull

Products: Abaqus/Standard Abaqus/CAE Abaqus/Viewer

References:

- *Shell Elements*
- *Creating composite layups*
- *Composite layups*

Overview

This example shows how to create a composite layup to model a yacht hull.
The following Abaqus features are demonstrated:

- importing the shell geometry of a yacht hull from an ACIS (`.sat`) file,
- creating a composite layup using Abaqus/CAE,
- applying plies in the layup to regions of the model,
- viewing a ply stack plot from a region of the model,
- viewing an envelope plot that shows the critical plies in each region of the model, and
- viewing an *X*–*Y* plot through the thickness of an element.

Application description

Composite hulls are used routinely in the yacht industry. Composite materials allow manufacturers to create high-performance marine vessels that incorporate the complex hull shapes that engineers have derived from computational fluid dynamics analyses and from experimental testing. Composites also provide the strength, rigidity, and low mass that high-performance yachts require. However, incorporating many layers of material with varying orientations in a complex three-dimensional finite element model can be time consuming. The addition of local reinforcements complicates the process. These issues are described by Bosauder et al. (2006).

The composite layup capability in Abaqus/CAE simplifies the process of composites modeling by mirroring the procedure that manufacturers follow on the shop floor—stacking sheets of composite material in a region of a mold and aligning the material in a specified direction. The Abaqus/CAE composite layup editor allows you to easily add a ply, choose the region to which it is applied, specify its material properties, and define its orientation. You can also read the definition of the plies in a layup from data in a text file, which is convenient when the data are stored in a spreadsheet or are generated by a third-party tool.

Geometry

Figure 1 shows the hull, mast, rigging, and keel of the yacht model. The geometry of the model is imported as a single part from an ACIS (`.sat`) file, as shown in *Figure 2*. The part models one half of the hull, and symmetric boundary conditions are applied. The hull represents a high-performance 20-meter yacht with reinforced bulkheads that stiffen the structure. The infrastructure above the deck does not play a role in modeling the performance of the hull and is not included in the model.

Sets are created that correspond to the regions of the composite layup to which plies are applied.

Materials

The model is partitioned into 27 regions. Each region contains plies of glass-epoxy cloth surrounding a Nomex core. Most regions contain nine plies—four glass-epoxy plies on either side of the Nomex core. However, additional plies are added to reinforce regions of high strain. Some bulkheads are reinforced with stringers made of glass-epoxy cloth with an effective Young's modulus of 128000 N/mm^2. *Table 1* shows the material properties of the glass-epoxy cloth, and *Table 2* shows the material properties of the Nomex core.

Figure 3 shows several rows of the composite layup table and illustrates how plies and material orientations are assigned to a region of the model. *Figure 4* shows a ply stack plot of the same region.

Boundary conditions and loading

The center of the model is constrained to be symmetric about the *y*-axis, as shown in *Figure 2*. The following loads are applied:

- A hydrostatic pressure is applied to the hull. The pressure is modeled with an analytical field that increases the pressure linearly along the *z*-axis.
- Concentrated forces that model the tension from the sail rigging are applied to the front, rear, and side of the deck. The forces are applied along the *x*-axis of a datum coordinate system. Each coordinate system has an origin at the location of the load, and the *x*-axis orients the load toward the location of the top of the mast. The concentrated forces are transferred to the deck through distributing couplings.
- The load from the mast is applied at the base of the hull in the *z*-direction.
- The keel is modeled with a lumped mass attached to the hull through a kinematic coupling.
- An inertia relief load is applied at the center of the hull to bring the model into equilibrium after the loads are applied.

Abaqus modeling approaches and simulation techniques

A single loading case is considered that uses a static analysis to study the effect of the loading on the composite layup.

Mesh design

The model is meshed by Abaqus/CAE using the free meshing technique and quadrilateral-dominated elements.

Loads

- The tension load from the sail rigging is 5500 N at the front of the hull, 4000 N at the rear of the hull, and 7500 N at the side of the hull.
- The load from the mast is 17500 N.
- A lumped mass of 10 metric tons models the keel.

Constraints

A kinematic coupling transfers the weight of the keel to the base of the hull, and three distributing couplings transfer the load from the rigging to the hull.

Analysis steps

A single static load step is defined for the analysis; nonlinear effects are not included.

Output requests

By default, Abaqus/CAE writes field output data from only the top and bottom section points of a composite layup, and no data are generated from the other plies. In this model, output is requested for all section points in all plies. This allows you to create an envelope plot of the entire model that indicates which plies in each region are carrying the highest strain.

Results and discussion

Figure 5 shows an envelope plot of the in-plane shear strain (E12) in the middle of the hull. *Figure 6* shows the through-thickness variation of this strain component.

Files

You can use the Abaqus/CAE Python scripts to create the model and to run the analysis. You can also use the Abaqus/Standard input file to run the analysis.

compositehull_model.py	Script to create the model using the geometry from compositehull_geometry.sat and the composite layup from compositehull_layup.txt.
compositehull_geometry.sat	ACIS file containing the geometry of the model.
compositehull_layup.txt	A comma-separated text file defining the plies in the composite layup.
compositehull_job.py	Script to analyze the model.
compositehull_job.inp	Input file to analyze the model.

References

- Bosauder, P., D. Campbell, and B. Jones, “Improvements in the Commercial Viability of Finite Element Analysis (FEA) for Accurate Engineering of Marine Structures,” JEC conference, Paris, March 2006.

Tables

Table 1: Material properties of the glass-epoxy cloth.

Variable	Value
E_1	35000 N/mm^2
E_2	7500 N/mm^2
ν_{12}	0.3
G_{12}	3600 N/mm^2
G_{13}	3000 N/mm^2
G_{23}	3000 N/mm^2
ρ	1.5×10^{-9} metric tons/mm^3

Table 2: Material properties of the Nomex core.

Variable	Value
E_1	10 N/mm^2
E_2	10 N/mm^2
ν_{12}	0.3
G_{12}	1 N/mm^2
G_{13}	30 N/mm^2
G_{23}	30 N/mm^2
ρ	8.0×10^{-11} metric tons/mm^3

Figures

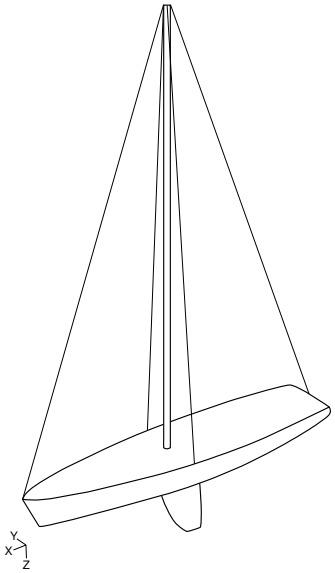


Figure 1: The yacht model.

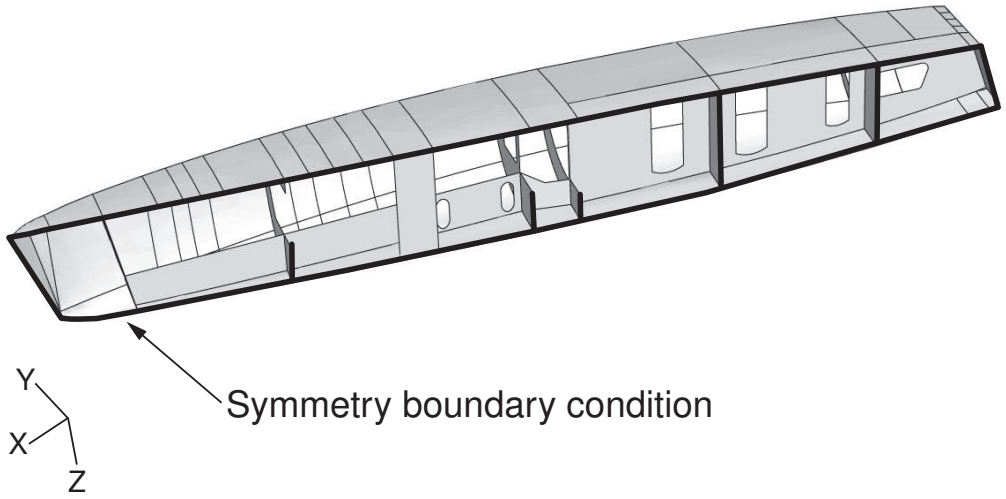
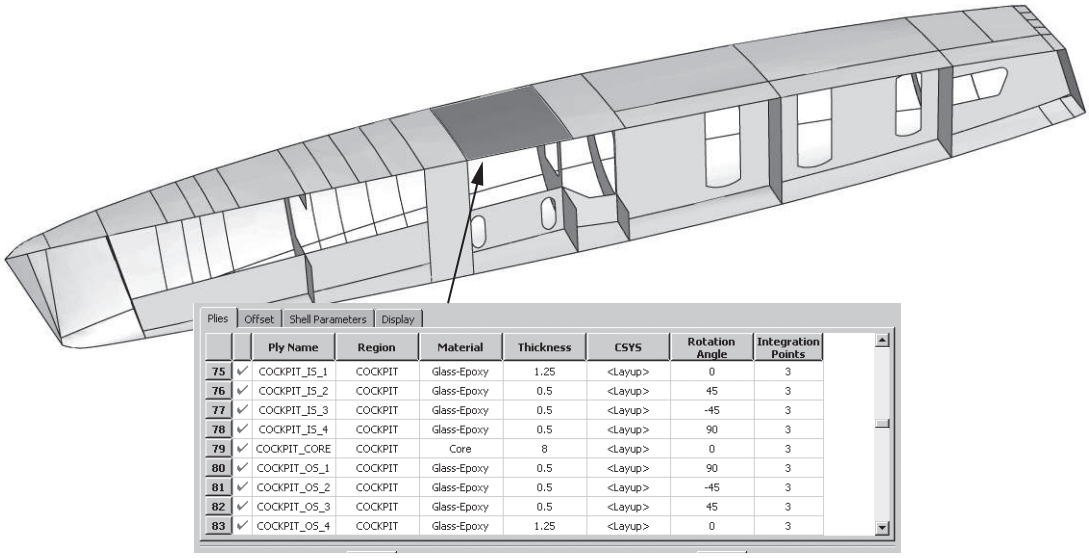


Figure 2: The symmetric model.



		Ply Name	Region	Material	Thickness	CSYS	Rotation Angle	Integration Points
75	✓	COCKPIT_IS_1	COCKPIT	Glass-Epoxy	1.25	<Layup>	0	3
76	✓	COCKPIT_IS_2	COCKPIT	Glass-Epoxy	0.5	<Layup>	45	3
77	✓	COCKPIT_IS_3	COCKPIT	Glass-Epoxy	0.5	<Layup>	-45	3
78	✓	COCKPIT_IS_4	COCKPIT	Glass-Epoxy	0.5	<Layup>	90	3
79	✓	COCKPIT_CORE	COCKPIT	Core	8	<Layup>	0	3
80	✓	COCKPIT_OS_1	COCKPIT	Glass-Epoxy	0.5	<Layup>	90	3
81	✓	COCKPIT_OS_2	COCKPIT	Glass-Epoxy	0.5	<Layup>	-45	3
82	✓	COCKPIT_OS_3	COCKPIT	Glass-Epoxy	0.5	<Layup>	45	3
83	✓	COCKPIT_OS_4	COCKPIT	Glass-Epoxy	1.25	<Layup>	0	3

Figure 3: Assigning plies in the layup table to a region of the model.

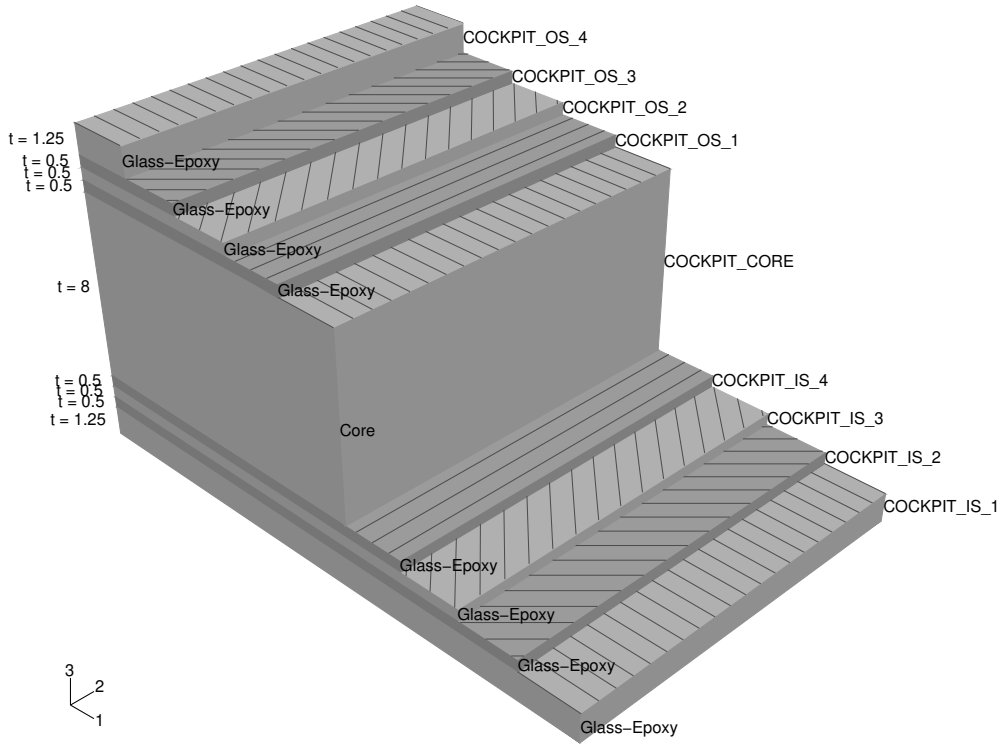


Figure 4: A ply stack plot from the cockpit.

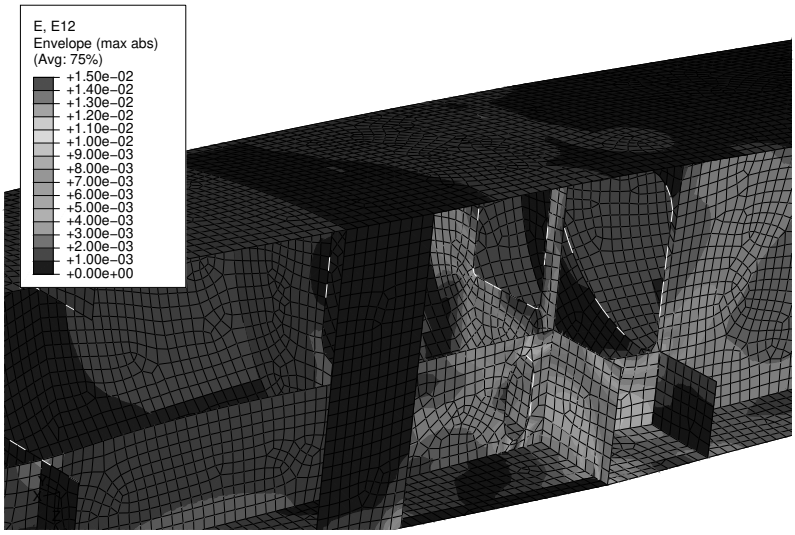


Figure 5: Envelope plot of strain in the critical plies in the center of the hull.

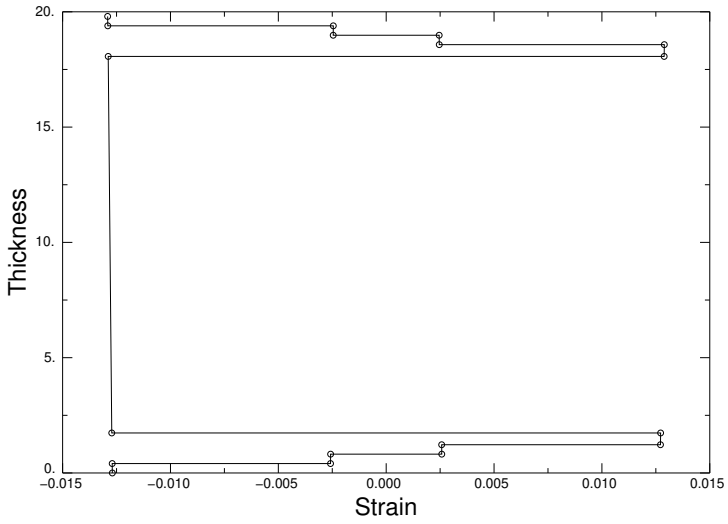


Figure 6: Strain (E12) across the thickness of an element.

Energy computations in a contact analysis

Products: Abaqus/Standard

References:

- *About General Contact in Abaqus/Standard*

Overview

This example shows how to use contact-related energy output variables and other meaningful energy output variables in a static nonlinear elastic, finite-sliding, frictional contact analysis.
The following energy output variables are introduced and explained:

- frictional dissipation,
- elastic contact energy, which represents energies stored in penalty springs of contact constraints and "softened" contact constraints; and
- the remaining work done by contact forces that is not accounted for by other energy output variables.

Application description

Energy output can provide valuable insight into model behavior and increase confidence in numerical accuracy. Large relative displacements on a contact interface within a time increment can lead to significant energy contributions that are not always intuitive.

Problem setup

The model represents a clamped snapping arrowhead that passes through an opening under a wall when pushed by a prescribed x-displacement. The geometry and mesh of C3D8R elements are shown in *Figure 1*. The perimeter of the right wall is fixed. *Figure 2* shows an intermediate position with the structure visibly bent when the arrowhead is in contact while passing through the opening. *Figure 3* shows that some stress remains at the end of simulation, which is related to some inaccuracy in the integration of the constitutive behavior. Ideally, the final position should be totally stress and strain free.

Abaqus modeling approaches and simulation techniques

General contact uses penalty enforcement of contact constraints in the normal contact direction by default. Frictional constraints in the tangential directions are also enforced by default with the penalty method. This penalty method can be thought of as inserting normal and tangential springs in the model while contact is active. Such springs store elastic energy that is released when contact opens up. The contact constraint elastic energy in the normal direction (ALLCCEN) and the contact constraint elastic energy in tangential directions (ALLCCET) represent nonnegative sums of the energies in all such active "contact springs" (or "penalty springs") in the model. The energy output variable ALLCCE is their sum: ALLCCE = ALLCCEN + ALLCCET. The use of "softened" contact relationships (see *Contact Pressure-Overclosure Relationships*) can also contribute energy to ALLCCEN. There is no contact constraint elastic energy associated with directly enforced hard contact and Lagrange friction.

The incremental frictional dissipation (ALLFD) together with incremental ALLCCET represents the physical incremental work of the frictional contact forces. Both ALLFD and ALLCCET remain zero in frictionless problems.

The elastic contact energy output variables (ALLCCEN, ALLCCET, and ALLCCE) are computed in total form every time increment. Other energies are computed incrementally, with the incremental contribution added to the running totals each increment.

To understand what the contact constraint discontinuity work (ALLCCDW) represents, first consider the full incremental work of contact forces on incremental displacements, which may produce incremental contributions to the following three contact energy variables:

- Frictional dissipation (ALLFD)
- Recoverable energy stored in contact penalty springs and "softened" contact constraints (ALLCCE)
- Stabilization dissipation associated with contact (ALLCCSD), not present in this example

Output variable ALLCCDW accounts for the remaining work done by contact forces; i.e., "full contact work" is equal to ALLFD + ALLCCE + ALLCCSD – ALLCCDW. ALLCCDW can be positive or negative, and it may be counterintuitive. For example, frictionless hard contact cannot physically store energy on the interface because normal contact forces are always orthogonal to incremental tangential displacements. Numerically, such contact constraints may have energy associated with them as demonstrated by nonzero ALLCCDW in this example, as discussed below. The total energy balance ETOTAL = ALLIE + ALLFD + ALLCCE – ALLWK – ALLCCDW.

Results and discussion

Figure 4 shows a complete set of nonzero energy output variables for this example. The most significant energies are external work (ALLWK) and contact constraint discontinuity work (ALLCCDW). In this example these output variables have different signs. If the external work were modified to be ALLWK + ALLCCDW, the major energies would be as shown in *Figure 5*. This combination of ALLWK and ALLCCDW is approximately in balance with the combination of frictional dissipation and elastic energy throughout the simulation.

The modified external work (ALLWK + ALLCCDW) is often representative of the physical external work in contact problems in terms of being equal to the sum of the stored and dissipated energies. Consider a particular contact constraint having a gap distance, g_i, in one increment and becoming closed with contact force, f_{i+1}, in the next increment (see *Figure 6*). A trapezoidal rule for integrating the work done by the contact force multiplies the average force by the relative incremental motion. In this case the resulting contribution to ALLCCDW is negative $1/2 * g_i * f_{i+1}$. This energy contribution is nonphysical and would disappear in the numerics as the time increment tends to zero. When contact opens up, similar behavior happens with sign reversals. Numerical integration for ALLWK is also limited with respect to accounting accurately for sudden changes in external forces. Summing ALLWK and ALLCCDW often cancels the respective nonphysical energy contributions, and the net effect on the total energy balance (ETOTAL) is zero.

A nonzero incremental ALLCCDW contribution can also occur if the normal direction changes during a time increment that also has relative motion between contact surfaces. The first significant drop in ALLCCDW for this example (see *Figure 4*) occurs from increment 4 to increment 5, when the leading wall edge slips from the inclined arrowhead surface to the top surface, as illustrated in *Figure 7*, causing a significant change in the contact constraint normal direction.

With finer time incrementation, ALLCCDW generally becomes smaller, as demonstrated in *Figure 8*. The maximum time increment allowed with so-called large time increments is five times larger than with small time increments. Convergence of global Newton iterations is more difficult to reach with small time increments because of the model behavior near the last stage of the analysis when the arrowhead snaps to the stress-free state. The "back of the head" touches the rear edge of the wall along the way, whereas with large time increments, this touching is avoided completely. The difference is reflected by the reaction force X becoming negative for small time increments (see *Figure 9*).

Elastic contact energies are shown in *Figure 10*. They are small compared to the major energies in the model. If ALLCCEN is not small, contact penalty stiffness may need to be increased to improve solution accuracy. If

ALLCCET is not small, the frictional elastic or slip tolerance may need to be decreased to increase the frictional penalty stiffness.

Ideally, ETOTAL should be exactly constant (in this example zero), but actually it is constant only approximately. The approximation is due to the fact that the recoverable strain energy (ALLIE) is computed by a modified trapezoidal rule, rather than an exact trapezoidal rule (which is not implemented due to dramatic computational performance implications). Such behavior is common in geometrically nonlinear problems, and one can only assess that the variation in ETOTAL is small compared to meaningful energies.

Nonzero initial ETOTAL will (appropriately) occur, for example, when initial stress is specified, such that the initial value of ALLIE is positive while other energies are initially zero. Nonzero initial ETOTAL also occurs when initial velocities are defined for a dynamic analysis; in this case initial kinetic energy is positive.

Convergence tolerances can also influence energy results because the energy balance depends on the balance of internal and external forces and moments in the model.

Input files

snaparrow_gc.inp	Input data for the analysis with large time increments.
snaparrow_gc2.inp	Input data for the analysis with small time increments.

Figures

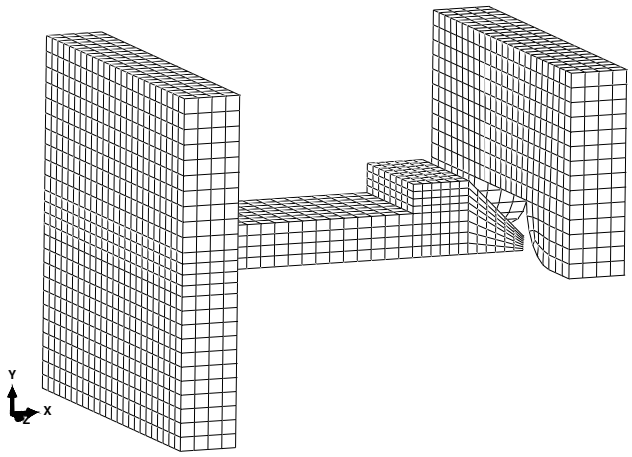


Figure 1: Initial position.

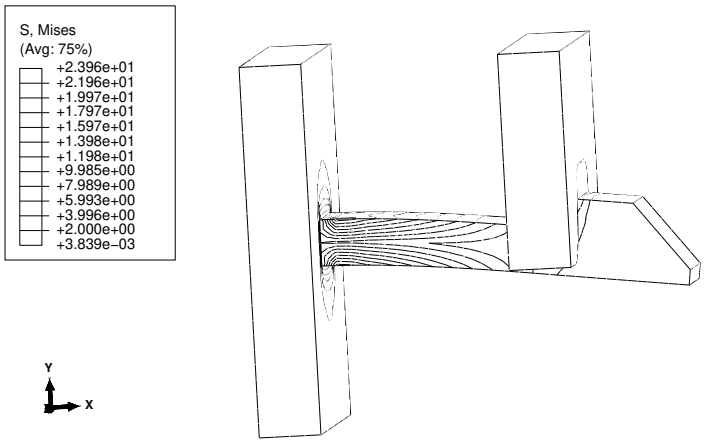


Figure 2: Intermediate position.

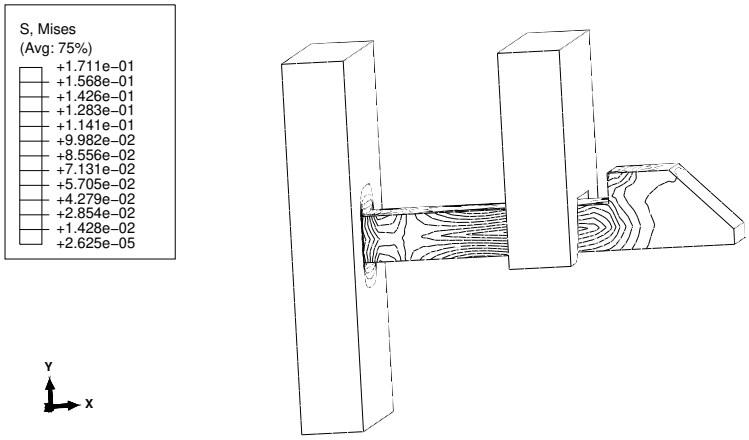


Figure 3: Final position.

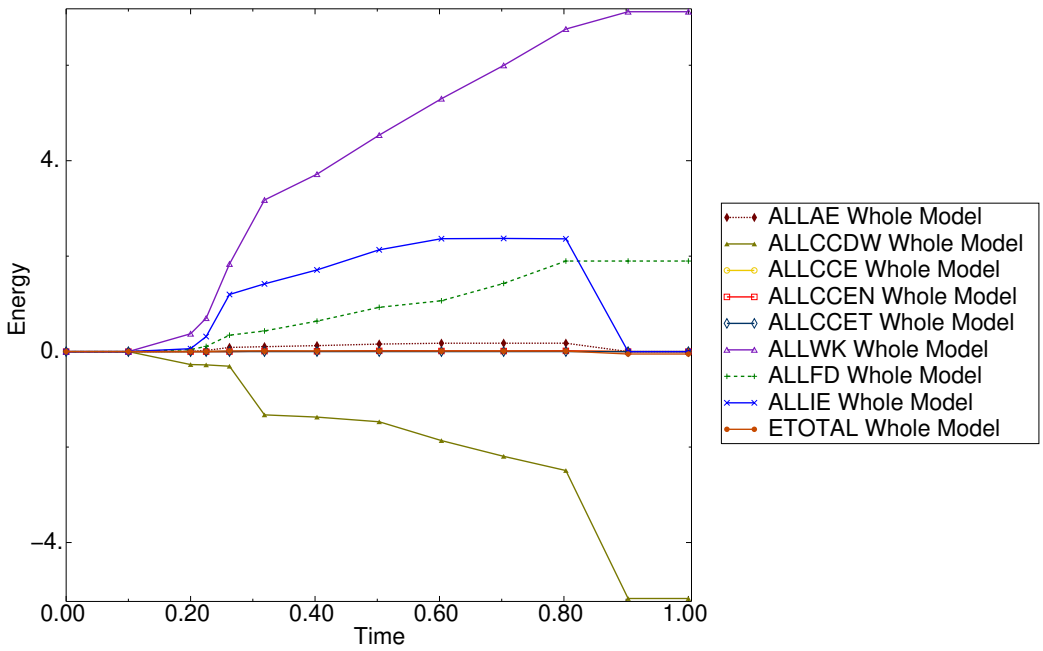


Figure 4: Nonzero energies in the model.

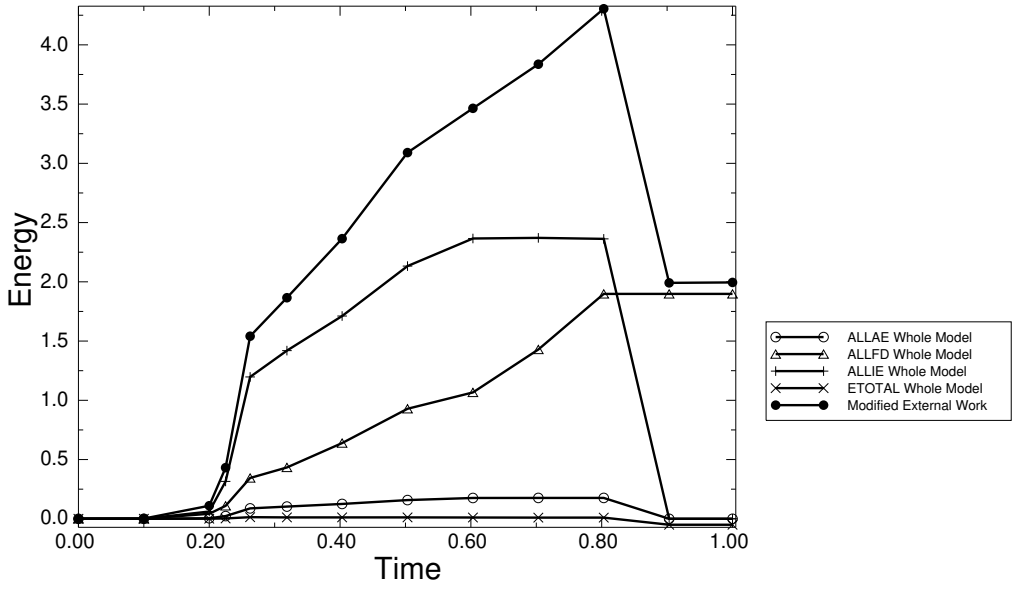


Figure 5: Major energies with modified external work.

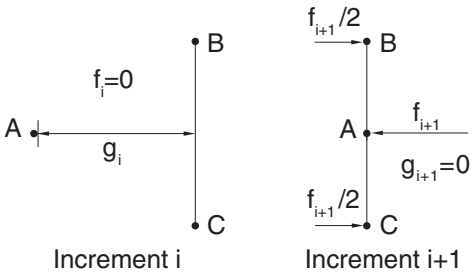


Figure 6: One contact point example to illustrate contributions to ALLCCDW.

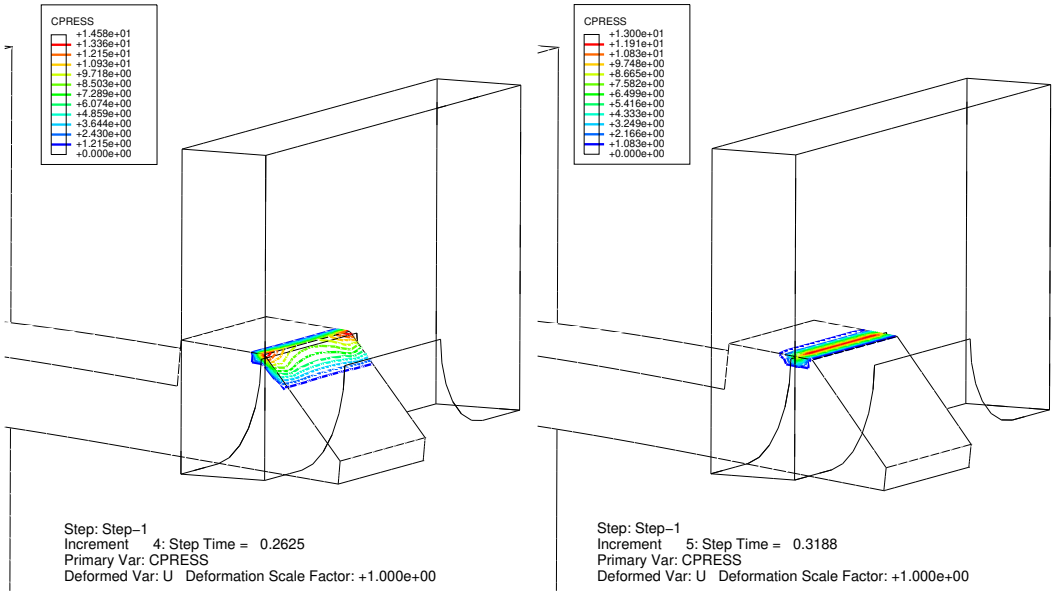


Figure 7: Contact normal direction change causes significant incremental change in ALLCCDW; large time increment.

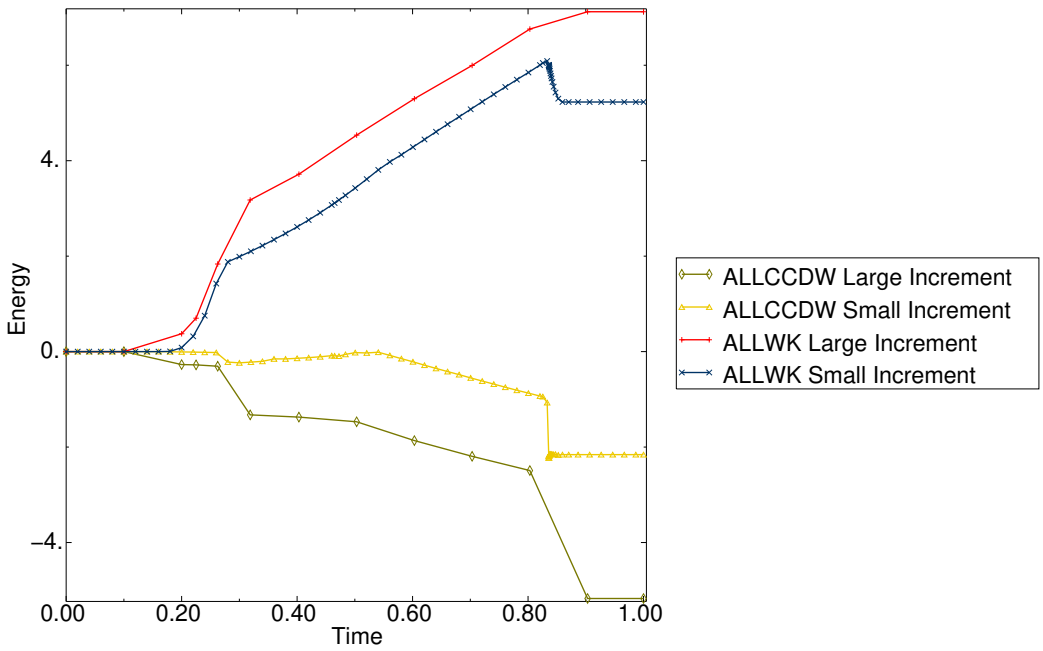


Figure 8: ALLCCDW and ALLWK with large and small time increments.

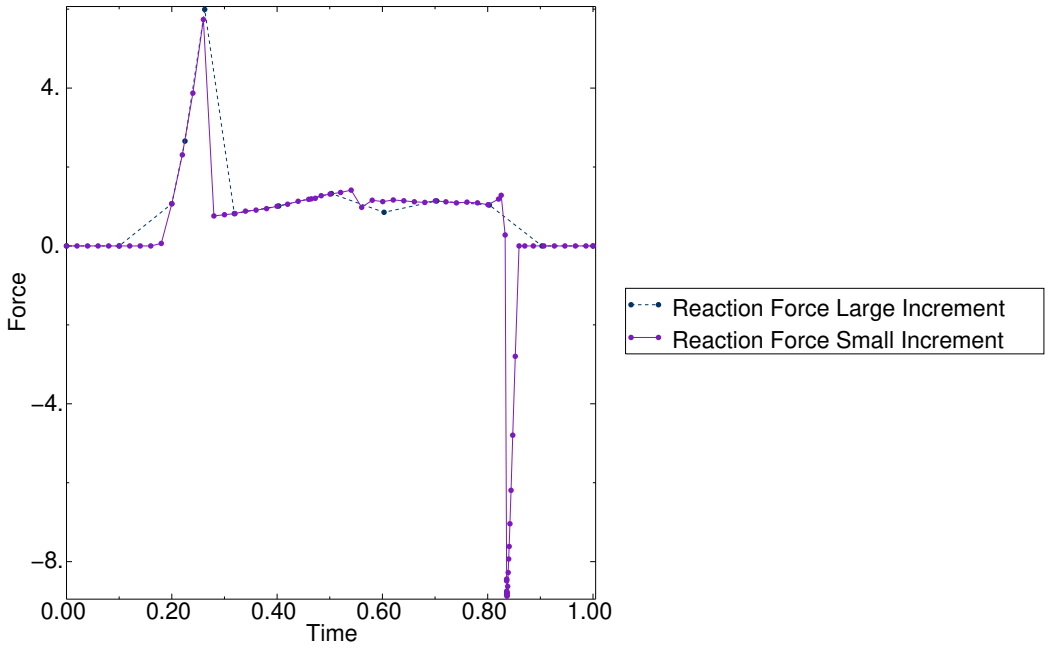


Figure 9: Reaction force X with large and small time increments.

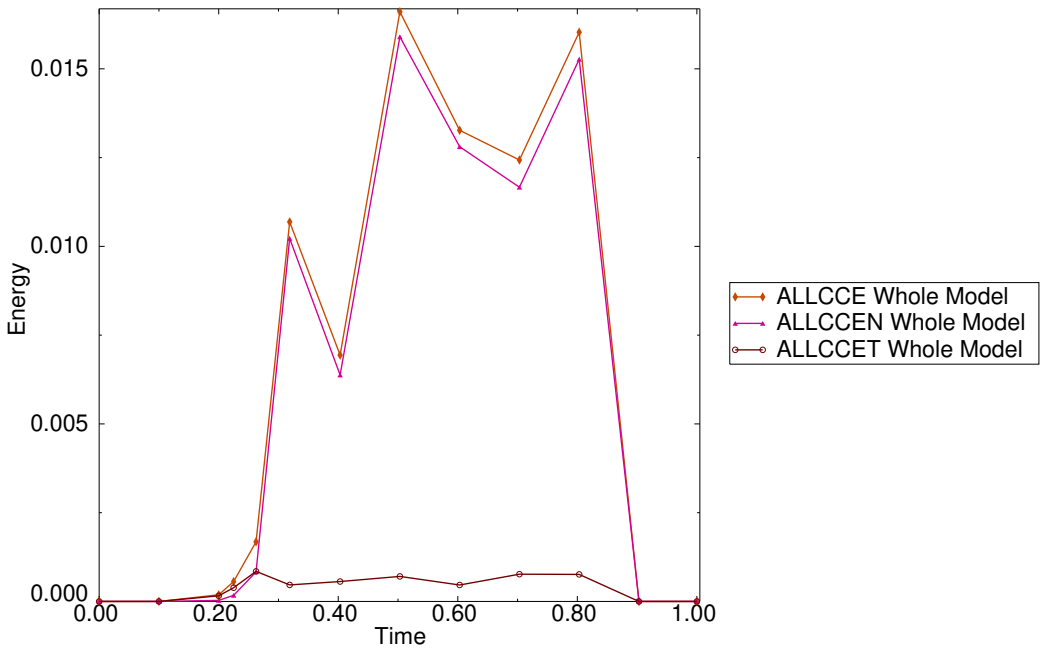


Figure 10: Contact elastic energies; large time increments.

Bending of a beam in a sleeve

Products: Abaqus/Standard

References:

- *About General Contact in Abaqus/Standard*

Overview

This example demonstrates the use of the edge-to-surface contact formulation that is applicable when a beam enclosing or enclosed by a surface of a shell or a solid comes in contact with that surface.

Application description

Four contact formulations are available for general contact in Abaqus/Standard: surface-to-surface, edge-to-surface, edge-to-edge, and vertex-to-surface. For the edge-to-surface contact formulation involving line elements, such as beams and trusses, the default secondary or main role assignment for the line element is that the line element acts as the secondary unless the line element radius exceeds twice the typical facet dimension of the other surface. More accurate contact resolution occurs with a relatively large diameter line element acting as the main surface to a more-refined faceted surface.

The edge-to-surface contact formulation with a main role assignment for beam elements is illustrated with this simple example of bending a beam in contact with a cylindrical sleeve.

Geometry

A beam of radius 10 mm and length 150 mm passes through a cylindrical sleeve of radius 15 mm, length 20 mm, and thickness 1 mm. The beam and cylinder are coaxial (*Figure 1*). Both are made of steel with a Young's modulus of 200 GPa and a Poisson's ratio of 0.3. The contact between the beam and cylindrical sleeve is considered frictionless with a "hard" pressure-overclosure relationship. The general contact formulation with self-contact of an all-encompassing surface is used.

Boundary conditions and loading

One end of the beam is held fixed while the other end is subjected to a bending load of 20 N along the global *Y*-axis. The sleeve is held fixed rigidly.

Abaqus modeling approaches and simulation techniques

The edge-to-surface contact formulation for beam elements in Abaqus/Standard general contact is invoked automatically. Since self-contact within the beam or the shell surface is not applicable in this example, an all-encompassing surface with frictionless contact and a "hard" pressure-overclosure relationship is used.

Mesh design

The shell mesh is offset to the interior surface of the sleeve.

Results and discussion

Since the beam edge is treated as the main while the sleeve is treated as the secondary, the contact between the beam and the sleeve surface is tracked well, leading to a good result and completion of the analysis (*Figure 2*).

If the beam edge acts as a secondary to the main surface of the sleeve, the beam bends but penetrates the cylindrical sleeve significantly as contact is not enforced properly.

Input files

beam_bending_sleeve.inp	Bending of a beam in a sleeve.

Figures

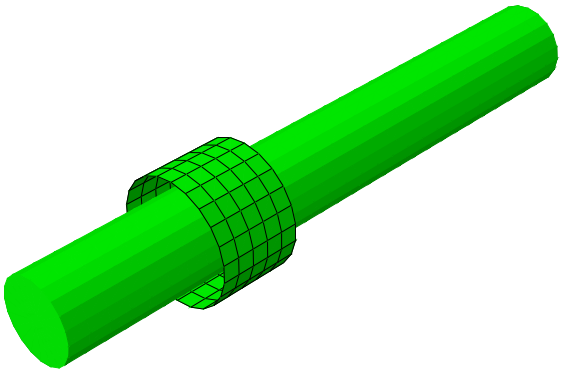


Figure 1: A beam passing through a cylindrical sleeve.

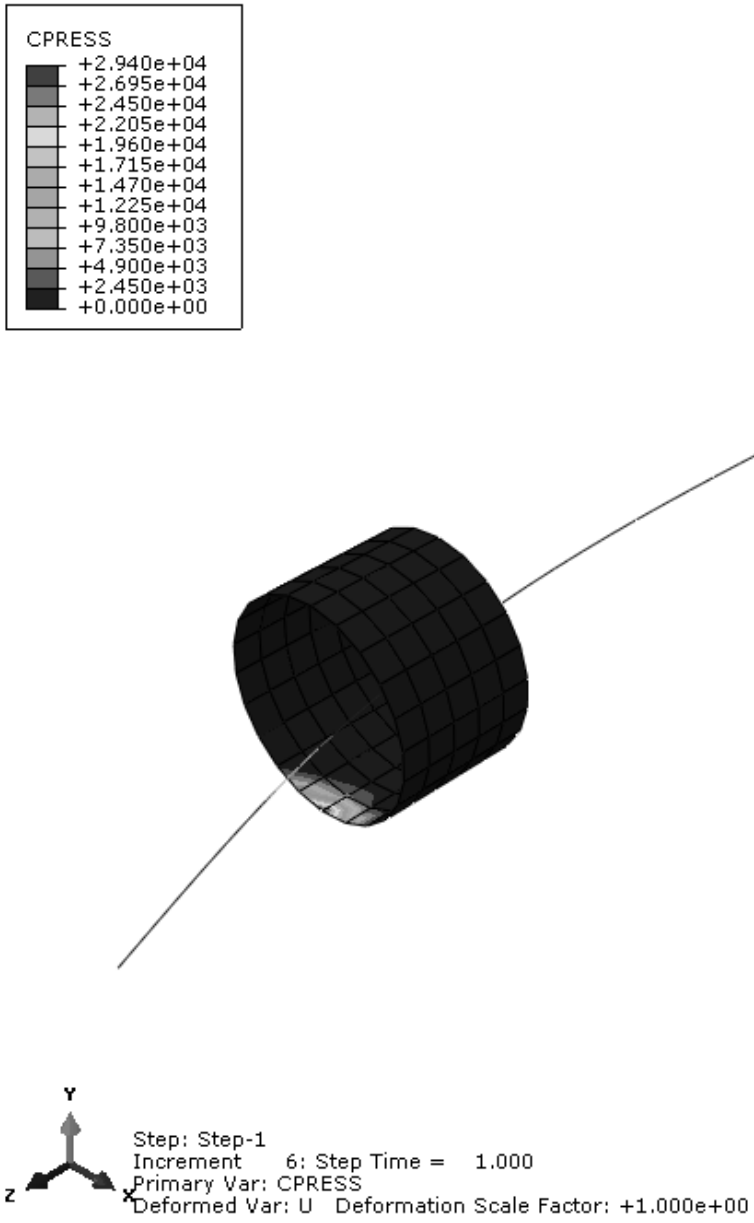


Figure 2: Contact pressure (CPRESS) contour plot for the beam treated as the main.

Fluid pressure penetration analysis of an O-ring seal of a pipe connection

Products: Abaqus/Standard Abaqus/Explicit

References:

- *Fluid Pressure Penetration Loads*
- **DSLOAD*
- **PRESSURE PENETRATION*

Overview

This example demonstrates the use of fluid pressure penetration loads simulated as distributed surface loads or pairwise pressure loads. The demonstration is through an O-ring seal of a pipe connection.

The following Abaqus features are demonstrated:

- fluid pressure loads acting on a wetted surface region, with the wetted surface region evolving based on consideration of evolving contact conditions;
- self-contact of rubber after significant deformation; and
- use of explicit dynamic and implicit dynamic procedures to approximate quasi-static conditions (but with some dynamic effects remaining that require engineering judgment).

Application description

This example demonstrates fluid pressure-penetration loading in Abaqus for an O-ring seal simulation. O-rings are widely used in machine design to seal connections that require the transmission of fluid and gas. It is crucial for the design to ensure that the seal contains the fluid under operating conditions. The simulation tracks the evolution of the region exposed to fluid and applies surface pressure approximating the effects of fluid to that region without directly modeling the fluid. The purpose of this example is to provide perspective on modeling methods. Subsequent design iterations (not discussed in this example) might lead to significant design improvement.

Geometry

Figure 1 shows a schematic diagram for a scenario in which an O-ring helps seal a pipe connection. During installation, the O-ring is compressed into the seal groove in the lower flange to create a tight and leak-proof seal, as shown in the cut view of *Figure 2*.

A simplified model is employed that focuses on the interactions between the O-ring and its surrounding surfaces where contact and fluid pressure loads develop. The O-ring is relatively soft; therefore, the surrounding parts are modeled as rigid surfaces for the preliminary study summarized in this example. *Figure 3* shows an axisymmetric model in swept view, including the deformable O-ring and rigid surfaces representing the flat upper flange and the bottom flange with a channel. *Figure 4* shows the relevant dimensions. The O-ring has an inner diameter of 4 mm and an outer diameter of 8 mm. The seal groove has a depth of 6.1 mm and top and bottom widths of 8.2 mm and 6.6 mm, respectively. The center of the O-ring is 20 mm from the axis of revolution.

Abaqus modeling approaches and simulation techniques

Mesh design

The finite element model is created with first-order axisymmetric elements with reduced integration (CAX4R). *Figure 3* shows the finite element mesh in swept view.

Materials

The O-ring material is modeled as neo-Hookean hyperelastic with $C_{10} = 1000$ MPa and $D_1 = 1 \times 10^{-5}$ MPa^{-1}; such that the initial shear modulus, $\mu_0 = 2\,C_{10}$, is 2000 MPa, and the initial bulk modulus, $K_0 = \frac{2}{D_1}$, is 2×10^5 MPa. The ratio $\frac{K_0}{\mu_0} = 100$ corresponds to the suggested upper limit for Abaqus/Explicit, as discussed in *Compressibility in Abaqus/Explicit*.

The actual density of rubber is approximately 2.24×10^{-9} tonne/mm^3; however, due to the modeling strategy discussed in *Analysis steps*, the density assigned to the O-ring material for this example is 0.01 tonne/mm^3.

Interactions

All of the models considered have general contact specified globally to resist penetrations, and they all enforce a friction coefficient of 0.1. Contact occurs between the O-ring and adjacent surfaces, and the interior surface of the O-ring experiences self-contact.

The extent and evolution of the active fluid pressure–penetration loading region depends on the evolution of the contact state.

- Test case variations that use the surface-based fluid pressure–penetration loading capability use the contact state from general contact.
- Test case variations using the pairwise fluid pressure–penetration loading capability can only use the contact state from contact pairs. Therefore, these test variations have contact pairs defined between the O-ring and the adjacent rigid surfaces in addition to general contact that is specified globally.
- General contact creates contact exclusions for interactions associated with contact pairs automatically (as discussed in *Defining the General Contact Domain* for Abaqus/Explicit and *Defining the General Contact Domain* for Abaqus/Standard), such that those interactions are processed only by contact pairs. However, self-contact involving the interior surface of the O-ring is processed with general contact.

Analysis steps

The surface representing the bottom flange is held fixed throughout the simulation. The upper flange is moved downward by 3.9 mm to push the O-ring into the seal groove in Step 1. Step 2 simulates the effects of pressurized fluid entering the modeled region from the top left, which applies pressure to the exposed portion of the O-ring. It is assumed that the fluid slowly leaks into the region of the O-ring such that quasi-static conditions are a reasonable approximation (unless the seal is about to fail). However, the fluid pressure load magnitude is suddenly set to 750 MPa at the beginning of the second step.

Dynamic procedure types are used for this example to maximize robustness, due to the inherent stabilization effects of inertia. Material densities and step times are tuned such that the kinetic energy remains small compared to the strain energy for both steps. The duration of Step 1 is 0.8 seconds, and the duration of Step 2 is 0.1 seconds. The specified material density is much higher than that of the actual material, as discussed in *Materials*.

Application of fluid pressure loading on the O-ring in Step 2 causes dynamic overshoot in the displacement response, which tends to predict a larger penetration distance of fluid than would occur without dynamic overshoot.

Simulations with the following procedure sequences are considered:

- Procedure sequence 1:
  - Step 1: Explicit dynamic
  - Step 2: Explicit dynamic
- Procedure sequence 2:
  - Step 1: Implicit dynamic
  - Step 2: Implicit dynamic
- Procedure sequence 3:
  - Step 1: Implicit dynamic
  - Step 2: Explicit dynamic (after import from Abaqus/Standard to Abaqus/Explicit; as discussed in *Transferring Results between Abaqus/Explicit and Abaqus/Standard*)

Procedure sequence 3 is included because the most efficient modeling strategy for some fluid pressure-penetration workflows is to simulate preloading with Abaqus/Standard followed by import to Abaqus/Explicit for fluid pressure–penetration loading. Procedure sequence 3 does not result in the most efficient approach for this example among the cases considered.

Simulations with the following fluid pressure loading options are considered (see *Fluid Pressure Penetration Loads* for a discussion of these options):

- Pressure penetration load type 1: Pressure penetration loads applied as a distributed surface load, and the local algorithm controls the evolution of the region exposed to fluid pressure (applicable to Abaqus/Standard and Abaqus/Explicit).
- Pressure penetration load type 2: Pressure penetration loads applied as a distributed surface load, and the wetting advance algorithm controls the evolution of the region exposed to fluid pressure (applicable to Abaqus/Explicit).
- Pressure penetration load type 3: Pressure penetration loads applied by the pairwise fluid pressure penetration loading capability (applicable to Abaqus/Standard).

For the local algorithm, fluid pressure (PPRESS) is applied on the load surface where the contact stress (CPRESS) is below a critical contact pressure, which requires judicious selection of the load surface.

For the wetting advance algorithm and the algorithm used by the pairwise fluid pressure–penetration loading capability, you must specify a node or node set on the surface initially exposed to fluid pressure. The wetted surface region expands from these nodes when conditions are met for the fluid to penetrate into the contact interface.

Solution controls

In addition to using an artificially high material density, variable mass scaling is specified for Abaqus/Explicit steps to avoid having the stable time increment drop below a specified value. The first step causes significant distortion of some elements near the inner surface of the O-ring. Variable mass scaling adds a small amount of mass at these isolated locations to maintain the specified time increment size.

Discussion of results and comparison of cases

Simulation results for each model considered predict that the O-ring prevents leakage of fluid past the seal for the conditions described. A contour plot of the output variable PPRESS at a particular output frame shows the magnitude of the fluid pressure loading as a function of surface position. Animations and contour plots of PPRESS plots in *Figure 5* through *Figure 10* show the evolution of the fluid pressure loading on the O-ring and the deformation response for simulations associated with different model variations. To view the animations, see

the HTML documentation. Wetting is reversible for pressure-penetration load type 1 but irreversible for pressure-penetration load types 2 and 3. This distinction is apparent in the respective animations: the extent of the nonzero PPRESS region only grows for simulations with pressure-penetration load types 2 and 3, but it shrinks near the end of simulations with pressure-penetration load type 1.

History plots of kinetic energy (ALLKE) and internal energy (ALLIE) for Abaqus/Explicit and Abaqus/Standard simulations are shown in *Figure 11* and *Figure 12*, respectively. The fluid pressure–penetration loading is applied in Step 2, which starts at simulation time 0.8. ALLKE remains a small fraction of ALLIE after ALLIE becomes significant; however, significant dynamic overshoot (and rebound) effects are apparent in the evolution of configurations in animations of *Figure 5* through *Figure 8*. The maximum penetration distance of the fluid pressure into the contact interface for a truly static response would be less than what occurs for the simulations shown.

Performance

The model variations include three ways to simulate the first step (prior to fluid pressure–penetration loading) and six ways to simulate the second step. Abaqus/Explicit is fastest for the first step. Run times for the other cases and other steps relative to the run time of the first step in Abaqus/Explicit are as follows:

- The run time for the first step in Abaqus/Standard with general contact is 2.3 times longer than the run time in Abaqus/Explicit for the first step.
- The run time for the first step in Abaqus/Standard with general contact plus contact pairs is 2.1 times longer than the simulation in Abaqus/Explicit for the first step.
- The run time for the second step of non-import simulations in Abaqus/Explicit with either the local or the wetting advance algorithm is 15% percent of the run time in Abaqus/Explicit for the first step.
- The run time for the second step of import simulations in Abaqus/Explicit with either the local or the wetting advance algorithm is 17% percent of the run time in Abaqus/Explicit for the first step.
- The run time for the second step in Abaqus/Standard with the local algorithm is 5.4 times longer than the run time of the first step in Abaqus/Explicit.
- The run time for the second step in Abaqus/Standard with pairwise pressure penetration loading is 5.2 times longer than the run time of the first step in Abaqus/Explicit.

Import

Abaqus/Explicit is faster in both steps of this analysis. Results obtained with simulation sequence 3 (involving import to Abaqus/Explicit) are in good agreement with other results. Refer to *Figure 13* through *Figure 14* for a comparison of the fluid pressure (PPRESS) between simulation sequence 2 and 3.

Other considerations

The pressure penetration load applied at any increment is based on the contact status at the beginning of that increment. This can lead to dependence on the time increment size for Abaqus/Standard simulations with fluid pressure penetration. Small time increments are recommended to obtain accurate results. Abaqus/Explicit is well suited to treating complex evolutions like this due to its inherently small time increment sizes. For more information, see *Fluid Pressure Penetration Loads*. Based on the consistency of the Abaqus/Standard and Abaqus/Explicit results for this example, it appears that the time increment size Abaqus/Standard uses is adequately small in these simulations.

The use of nonphysical density in this example is based on the assumption that fluid slowly leaks into the region of the O-ring. A dynamic study with realistic density is required to investigate the effects of the sudden exposure of the O-ring to high pressure fluid.

Input files

O-ring_Thick_local_xpl.inp	Input file for procedure sequence 1 and pressure penetration load type 1.
O-ring_Thick_wetadv_xpl.inp	Input file for procedure sequence 1 and pressure penetration load type 2.
O-ring_Thick_local_std.inp	Input file for procedure sequence 2 and pressure penetration load type 1.
O-ring_Thick_cp_std.inp	Input file for procedure sequence 2 and pressure penetration load type 3.
O-ring_Thick_local_xpl_import.inp	Input file for Step 2 of procedure sequence 3 with pressure penetration load type 1 (importing from the end of Step 1 of O-ring_Thick_local_std.inp).
O-ring_Thick_wetadv_xpl_import.inp	Input file for Step 2 of procedure sequence 3 with pressure penetration load type 2 (importing from the end of Step 1 of O-ring_Thick_local_std.inp).

Figures

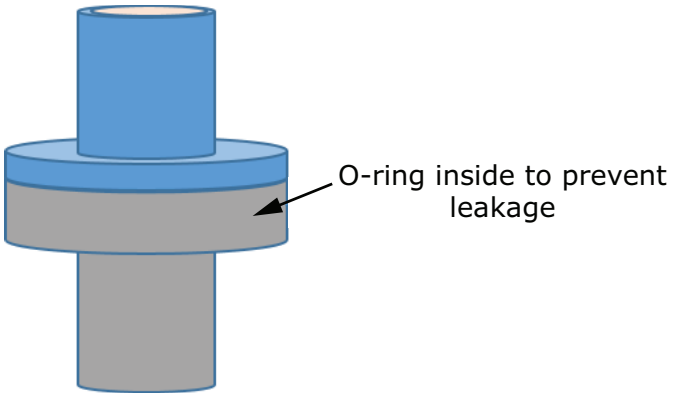


Figure 1: Schematic diagram for a pipe connection.

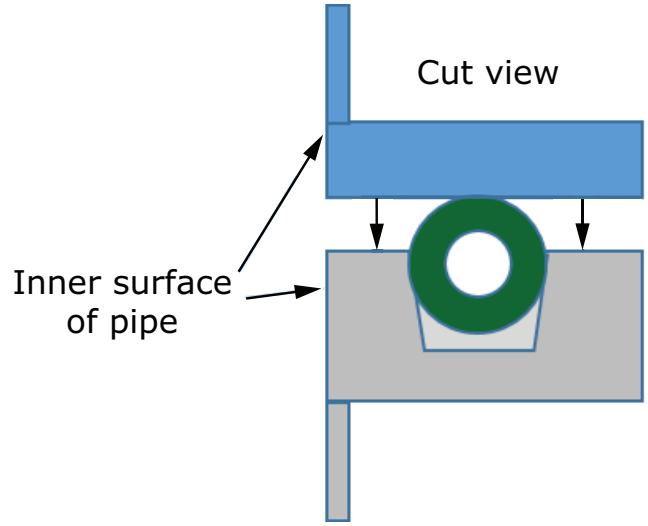


Figure 2: Cut view of the pipe connection.

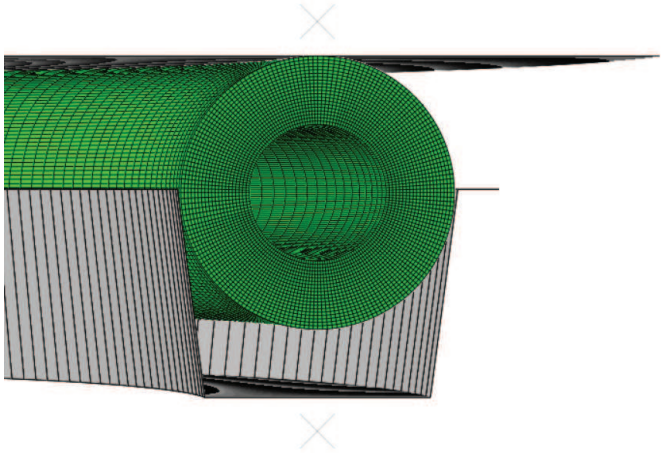


Figure 3: Axisymmetric model of the O-ring seal shown in swept view.

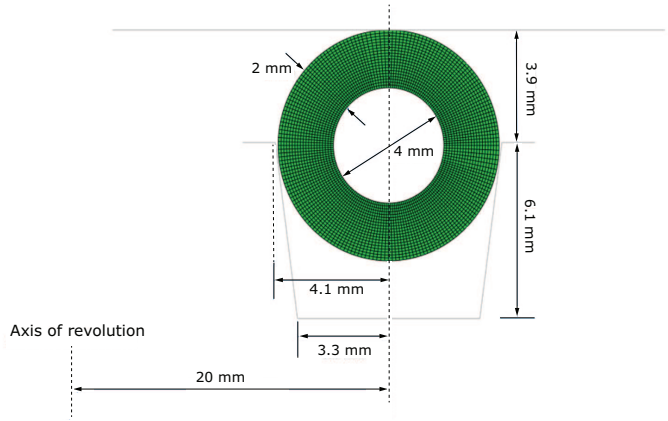


Figure 4: Dimensions of the O-ring seal.

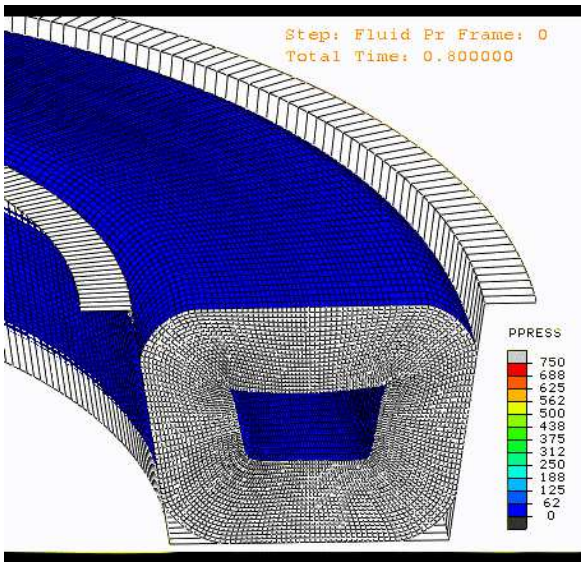


Figure 5: Evolution of fluid pressure PPRESS for procedure sequence 1 with pressure penetration load type 1.

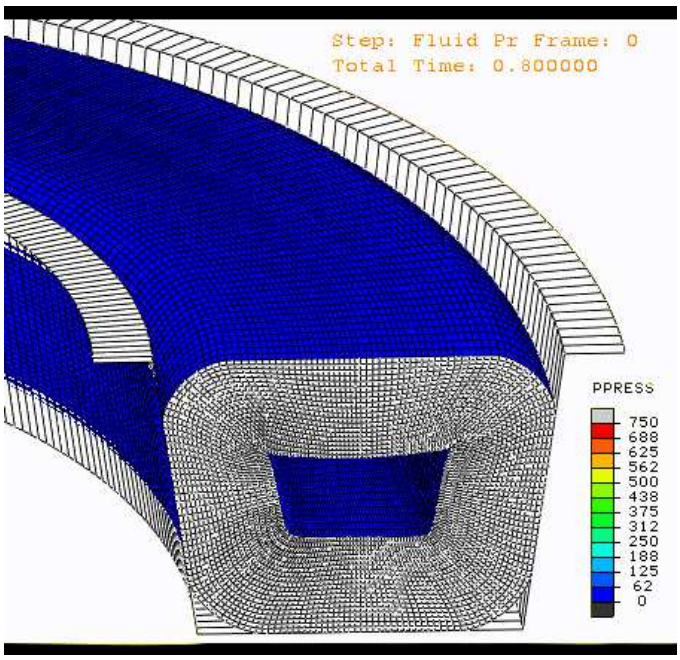


Figure 6: Evolution of fluid pressure PPRESS for procedure sequence 1 with pressure penetration load type 2.

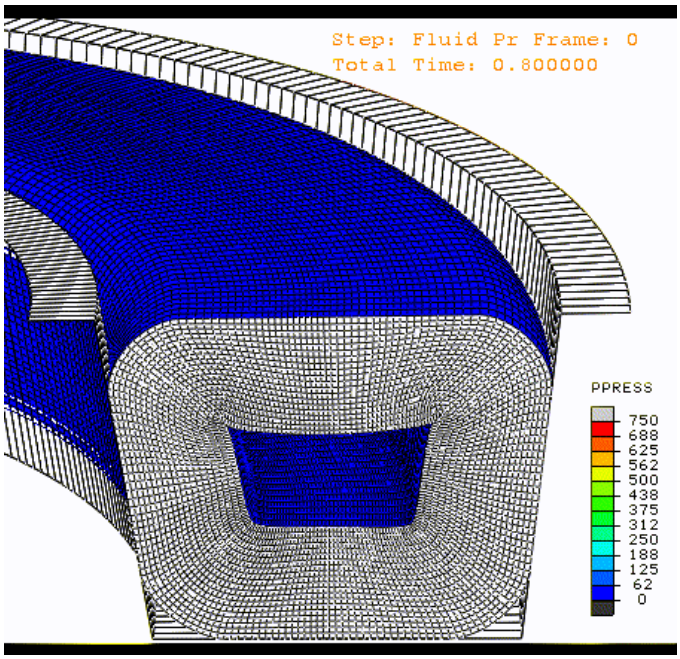


Figure 7: Evolution of fluid pressure PPRESS for procedure sequence 2 with pressure penetration load type 1.

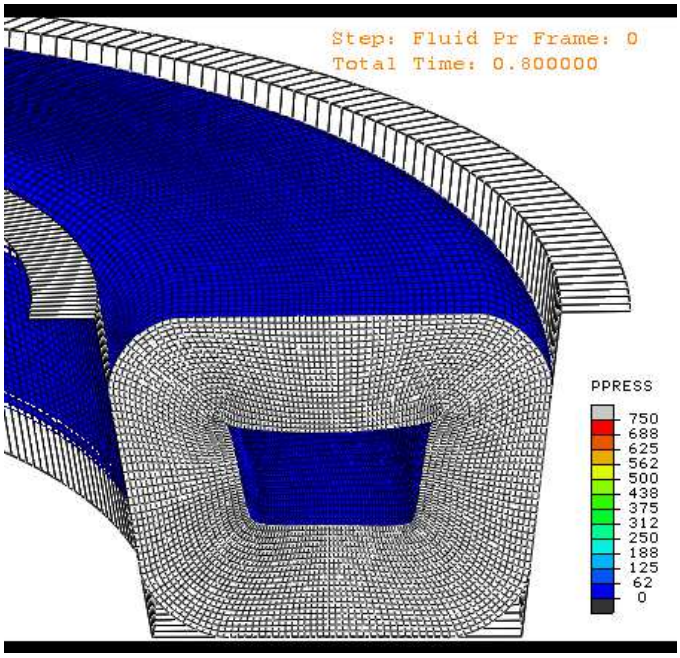


Figure 8: Evolution of fluid pressure PPRESS for procedure sequence 2 with pressure penetration load type 3.

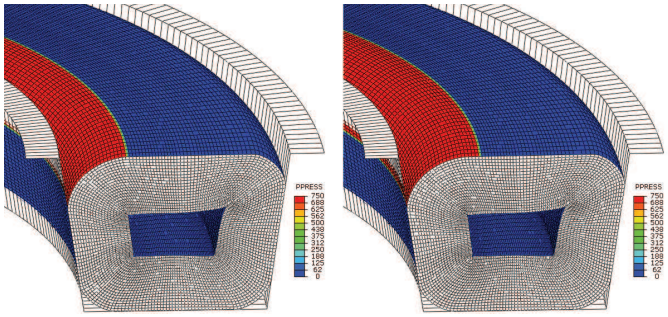


Figure 9: Fluid pressure PPRESS at the end of Step 2 for procedure sequence 1 with pressure penetration load type 1 (left) and 2 (right).

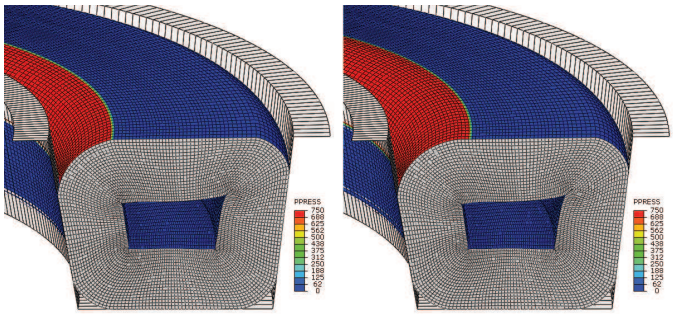


Figure 10: Fluid pressure PPRESS at the end of Step 2 for procedure sequence 2 with pressure penetration load type 1 (left) and 3 (right).

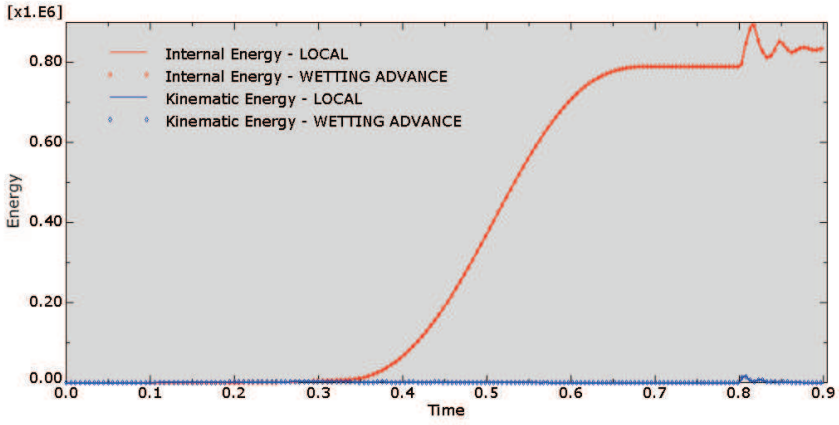


Figure 11: Kinematic energy (ALLKE) and Internal energy (ALLIE) history outputs for procedure sequence 1.

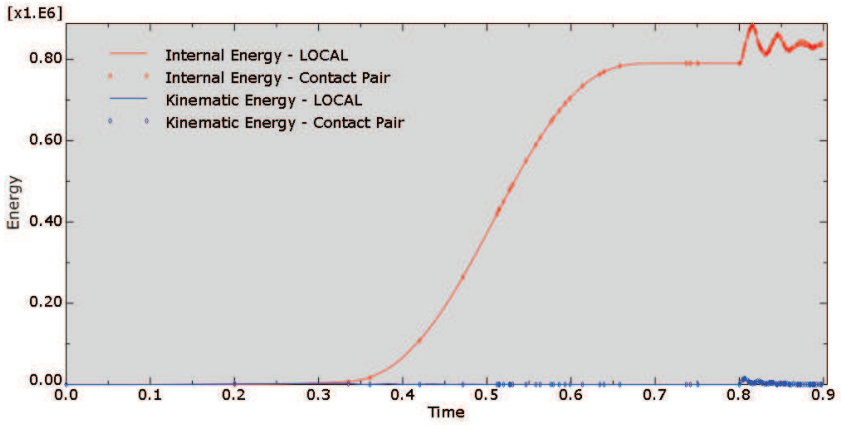


Figure 12: Kinematic energy (ALLKE) and Internal energy (ALLIE) history outputs for procedure sequence 2.

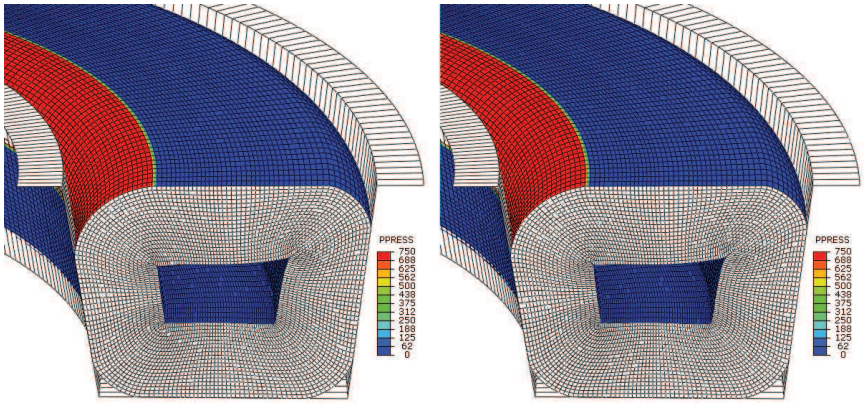


Figure 13: Fluid pressure PPRESS at the end of Step 2 for procedure sequence 1 (left) and 3 (right) with pressure penetration load type 1.

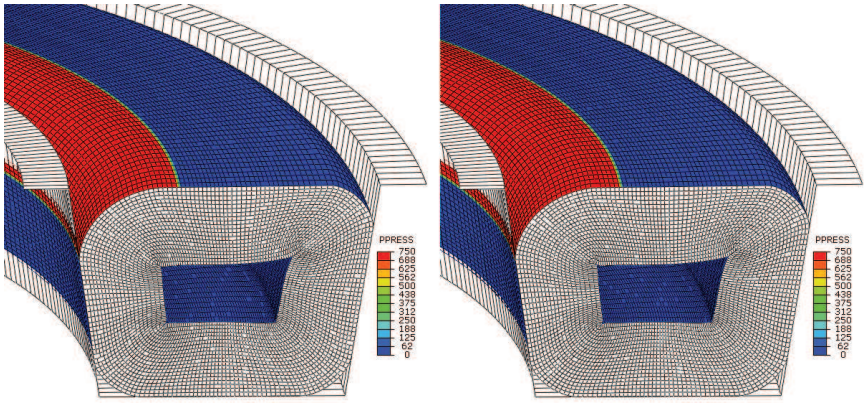


Figure 14: Fluid pressure PPRESS at the end of Step 2 for procedure sequence 1 (left) and 3 (right) with pressure penetration load type 2.

Buckling and collapse analyses

In this section:

- *Snap-through buckling analysis of circular arches*
- *Laminated composite shells: buckling of a cylindrical panel with a circular hole*
- *Buckling of a column with spot welds*
- *Elastic-plastic K-frame structure*
- *Unstable static problem: reinforced plate under compressive loads*
- *Buckling of an imperfection-sensitive cylindrical shell*

Snap-through buckling analysis of circular arches

Products: Abaqus/Standard

Overview

This example shows the postbuckling behavior of a structure whose response is unstable during part of its loading history.

It is often necessary to study the postbuckling behavior of a structure whose response is unstable during part of its loading history. Two of the models in this example illustrate the use of the modified Riks method, which is provided to handle such cases. The method is based on moving with fixed increments along the static equilibrium path in a space defined by the displacements and a proportional loading parameter. The actual load value may increase or decrease as the solution progresses. The modified Riks method implemented in Abaqus is described in *Modified Riks algorithm*.

The other two models illustrate the use of viscous damping. One example applies viscous damping as a feature of surface contact, which allows for the definition of a "viscous" pressure that is proportional to the relative velocity between the surfaces. The implementation of this option in Abaqus is described in *Contact pressure definition*. The other example applies volume proportional damping to the model. The implementation of this option is described in the automatic stabilization section of *Solving Nonlinear Problems*.

Three separate cases are considered here. The first is a clamped shallow arch subjected to a pressure load. Reference solutions for this case are given by Ramm (1981) and Sharafi and Popov (1971). The second case is the instability analysis of a clamped-hinged circular arch subjected to a point load. The exact analytical solution for this problem is given by DaDeppo and Schmidt (1975). The third case is a modification of the shallow arch problem in which the ends are pinned rather than clamped and the arch is depressed with a rigid punch.

Model and solution control

The shallow circular arch is shown in *Figure 1*. Since the deformation is symmetric, one-half of the arch is modeled. Ten elements of type B21 (linear interpolation beams) are used. A uniform pressure is first applied to snap the arch through. The loading is then reversed so that the behavior is also found as the pressure is removed.

The deep circular arch is shown in *Figure 2*. One end of the arch is clamped, and the other is hinged. A concentrated load is applied at the apex of the arch. The arch undergoes extremely large deflections but small strains. Because of the asymmetric boundary conditions, the arch will sway toward the hinged end and then collapse. The arch is almost inextensible for most of the response history. Sixty elements of type B31H are used. Hybrid elements are used because they are most suitable for problems such as this.

Solution controls are used to set a very tight convergence tolerance because the problem contains more than one equilibrium path. If tight tolerances are not used, the response might follow a path that is different from the one shown.

In the Riks procedure actual values of load magnitudes cannot be specified. Instead, they are computed as part of the solution, as the "load proportionality factor" multiplying the load magnitudes given on the loading data lines. User-prescribed load magnitudes serve only to define the direction and to estimate the magnitude of the initial increment of the load for a step. This initial load increment is the product of the ratio of the initial time increment to the time period and the load magnitudes given in the loading options. The user can terminate a Riks analysis by specifying either a maximum load proportionality factor or a maximum displacement at a node, or both. When a solution point is computed at which either of these limits is crossed, the analysis will stop. In any event, or if neither option is used, the analysis ends when the maximum number of increments for the step is exceeded.

In snap-through studies such as these, the structure can carry increasing load after a complete snap. Therefore, the analysis is terminated conveniently by specifying a maximum load proportionality factor.

For the clamped shallow arch the initial snap occurs at a pressure of about −1000 (force/length^2 units). Thus, −250 (force/length^2 units) seems to be a reasonable estimate for the first increment of load to be applied. Accordingly, an initial time increment of 0.05 is specified for a time period of 1.0 and a pressure load of −5000 (force/length^2 units). The solution will have been sufficiently developed at a pressure of about −2000 (force/length^2 units). Therefore, the analysis is terminated when the load proportionality factor exceeds 0.4.

To illustrate the use of Riks in several steps, a second step is included in which the pressure is taken off the arch so that it will snap back toward its initial configuration. At any point in a Riks analysis, the actual load is given by $P = P_0 + \lambda (P_{ref} - P_0)$, where P_0 is the load at the end of the previous step, P_{ref} is the load magnitude prescribed in the current step, and λ is the load proportionality factor. The arch is unloaded so that in the initial time increment, a pressure of approximately $0.15P_0$ is removed. Using an initial time increment of 0.05 in a time period of 1.0, a load of $P_{ref} = -2P_0$ is prescribed for this restarted step. Furthermore, we want the analysis to end when all the load is removed and the arch has returned to its initial configuration. Therefore, a displacement threshold of 0.0 is set for the center of the arch. The analysis terminates when this limit is crossed. Because Abaqus must pick up the load magnitude at the end of the initial Riks step to start the next step, any step following a Riks step can be done only as a restart job from the previous step.

For the deep clamped-hinged arch, the initial snap occurs at a load of about 900 (force units). The load magnitude specified is 100 (force units), and the maximum load proportionality factor is specified as 9.5.

The shallow arch depressed with a rigid punch is shown in *Figure 3*. The analysis uses the same model of the arch as the first problem. However, the end is pinned rather than clamped, and load is applied through the displacement of the punch. The pinned boundary condition makes the problem more unstable than the clamped-end case. A preliminary analysis in which the arch is depressed with a prescribed displacement of the midpoint of the arch shows that the force will become negative during snap-through. Thus, if the arch is depressed with a rigid punch, the Riks method will not help convergence because, at the moment of snap-through, the arch separates from the punch, and the movement of the punch no longer controls the displacement of the arch. Therefore, contact damping is introduced to aid in convergence. Viscous damping with surface contact adds a pressure that is proportional to the relative velocity to slow down the separation of the arch from the punch.

The viscous damping clearance is set to 10.0, and the fraction of the clearance interval is set to 0.9; the damping is constant for a clearance of up to 9.0. Since the arch is 4.0 units high, the distance traveled by the top of the arch from the initial position to the final snap-through position is 8.0 units. This distance is clearly larger than the clearance between the middle of the arch and the tip of the punch at any time during the analysis. Thus, the viscous damping is in effect for the whole period when the arch has separated from the punch.

To choose the viscous damping coefficient, note that it is given as pressure per relative velocity. The relevant pressure is obtained by dividing the approximate peak force (10000.0) by the contact area (1.0). The relevant velocity is obtained by dividing the distance over which the top of the arch travels (8.0 from initial to snapped position, which can be rounded to 10.0) by the time (approximately 1.0, the total time of the step). A small percentage (0.1%) of this value is used for the viscous damping coefficient:

$$\mu = 0.001\frac{Ft}{Al} = 0.001\frac{(10000.0)(1.0)}{(1.0)(10.)} = 1.0.$$

With $\mu =$ 1.0, the analysis runs to completion. Another analysis was run with a smaller value of $\mu =$ 0.1, but the viscous damping was not sufficient to enable the analysis to pass the point of snap-through. Thus, a damping coefficient of 1.0 was determined to be an appropriate value.

Automatic stabilization based on volume proportional damping is also considered for the shallow arch compressed with a rigid punch, as an alternative to contact damping. Two forms of automatic stabilization are considered: one with a constant damping factor that is chosen by default (see *Automatic Stabilization of Static Problems*

with a Constant Damping Factor), and one with an adaptive damping factor (see *Adaptive Automatic Stabilization Scheme*).

Results and discussion

The results for the clamped shallow arch are shown in *Figure 4*, where the downward displacement of the top of the arch is plotted as a function of the pressure. The algorithm obtains this solution in 12 increments, with a maximum of three iterations in an increment. At the end of 12 increments the displacement of the top of the arch is about 7.5 length units. This represents a complete snap through, as the original rise of the arch was 4 length units. *Figure 5* and *Figure 6* show a series of deformed configuration plots for this problem. Several other authors have examined this same case and have obtained essentially the same solution (see Ramm, 1981, and Sharafi and Popov, 1971).

The results for the deep clamped-hinged arch are shown in *Figure 7*, where the displacement of the top of the arch is plotted as a function of the applied load. *Figure 8* shows a series of deformed configuration plots for this problem. The arch collapses unstably at the peak load. Following this, the beam stiffens rapidly as the load increases. The ability of the Riks method to handle unstable response is well-illustrated by this example.

The results of the preliminary analysis of the prescribed displacement of a pinned shallow arch are shown in *Figure 9*, with the displacement of the top of the arch plotted as a function of the reaction force at that point. This plot shows the negative force that develops during snap-through. A series of deformed configuration plots for the pinned shallow arch depressed with a punch and with viscous damping introduced is shown in *Figure 10*, with one plot showing the arch separated from the punch. *Figure 11* is a plot of the force between the punch and the top of the arch. The force is positive until snap-through, when the arch separates from the punch and a negative viscous force develops. Once the snap-through is complete, the force drops to zero as the punch continues to move down while separated from the arch. When the punch contacts the arch, a positive force develops again.

Similar results are produced when the contact viscous damping is replaced by volume proportional damping (with either constant or adaptive damping coefficients). A sequence of configurations like *Figure 10* is obtained, in which separation of the arch from the punch occurs during snap-through. At the end of the analysis the amount of energy dissipated is similar to the amount dissipated with the viscous damping option.

You can use the **abaqus restartjoin** execution procedure to extract data from the output database created by a restart analysis and append the data to a second output database. For more information, see *Joining Output Database (.odb) Files from Restarted Analyses*.

Input files

snapbuckling_shallow_step1.inp	Initial analysis step for the shallow arch.
snapbuckling_shallow_unload.inp	Restart run to obtain the unloading response of the shallow arch.
snapbuckling_deep.inp	Deep arch.
snapbuckling_shallow_midpoint.inp	Shallow arch loaded by a fixed displacement of the midpoint.
snapbuckling_shallow_punch.inp	Shallow arch loaded by the displacement of a rigid punch.
snapbuckling_b21h_deep.inp	60 elements of type B21H used for the deep clamped-hinged arch analysis.
snapbuckling_b32h_deep.inp	30 elements of type B32H used for the deep clamped-hinged arch analysis.
snapbuckling_restart1.inp	Restart analysis of snapbuckling_shallow_step1.inp during the RIKS step.
snapbuckling_restart2.inp	Restart analysis of snapbuckling_restart1.inp during the RIKS step. This illustrates restarting an existing RIKS restart analysis.

snapbuckling_shallow_stabilize.inp	Same as snapbuckling_shallow_punch.inp with the surface contact viscous damping replaced by the volume proportional damping of **STATIC*, STABILIZE.
snapbuckling_shallow_stabilize_adap.inp	Same as snapbuckling_shallow_punch.inp with the surface contact viscous damping replaced by adaptive stabilization of **STATIC*, STABILIZE, ALLSDTOL.

References

- DaDeppo, D. A., and R. Schmidt, "Instability of Clamped-Hinged Circular Arches Subjected to a Point Load," *Transactions of the American Society of Mechanical Engineers*, Journal of Applied Mechanics, pp. 894–896, Dec. 1975.
- Ramm, E., "Strategies for Tracing the Nonlinear Response Near Limit Points," in *Nonlinear Finite Element Analysis in Structural Mechanics*, edited by W. Wunderlich, E. Stein and K. J. Bathe, Springer Verlag, Berlin, 1981.
- Sharifi, P., and E. P. Popov, "Nonlinear Buckling Analysis of Sandwich Arches," Proc. ASCE, Journal of the Engineering Mechanics Division, vol. 97, pp. 1397–1412, 1971.

Figures

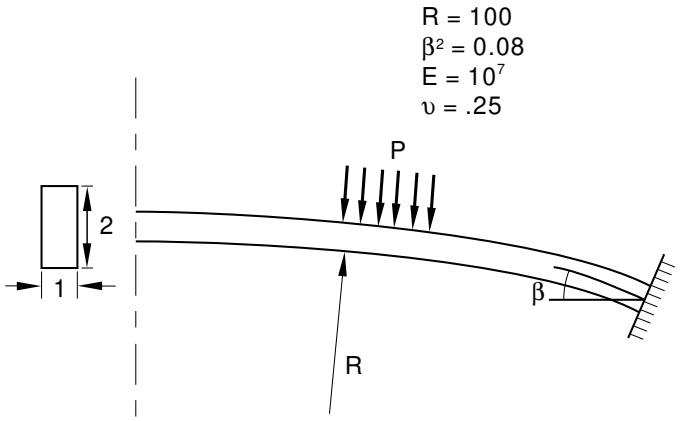


Figure 1: Clamped shallow circular arch.

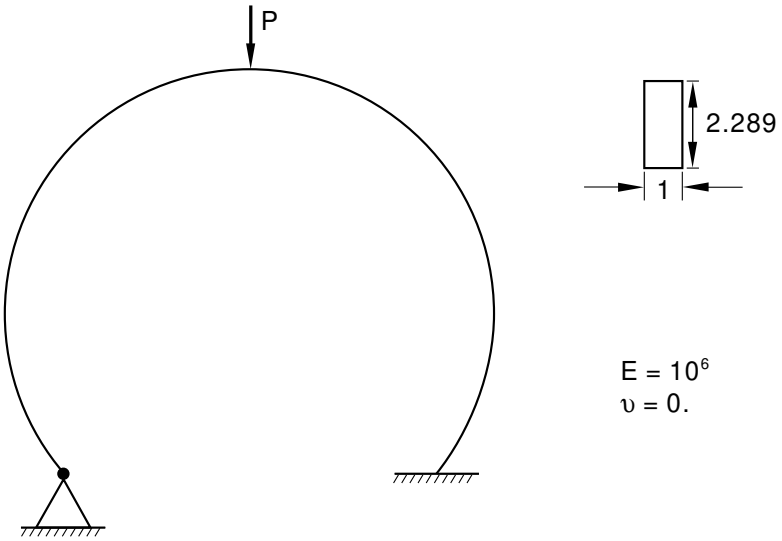


Figure 2: Deep clamped-hinged arch.

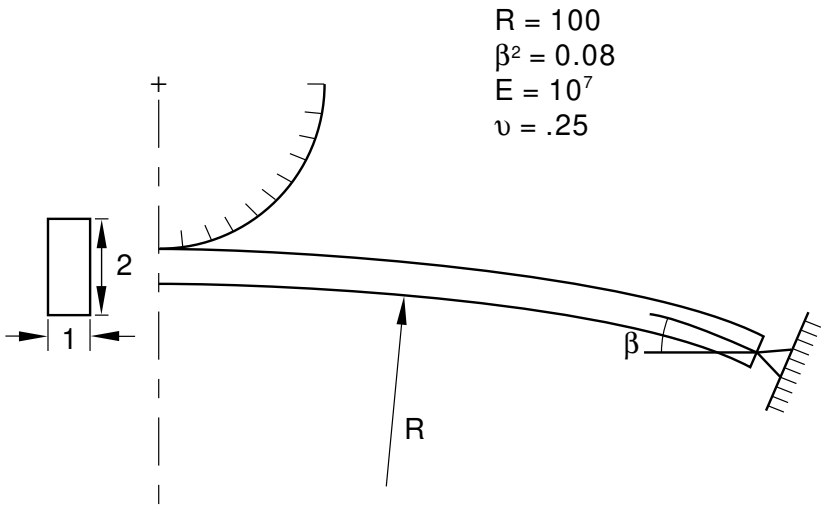


Figure 3: Pinned shallow arch with rigid punch.

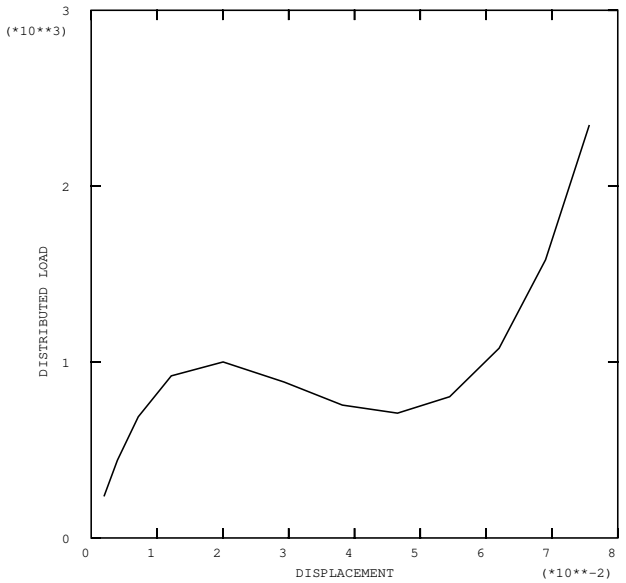


Figure 4: Load versus displacement curve for clamped shallow arch.

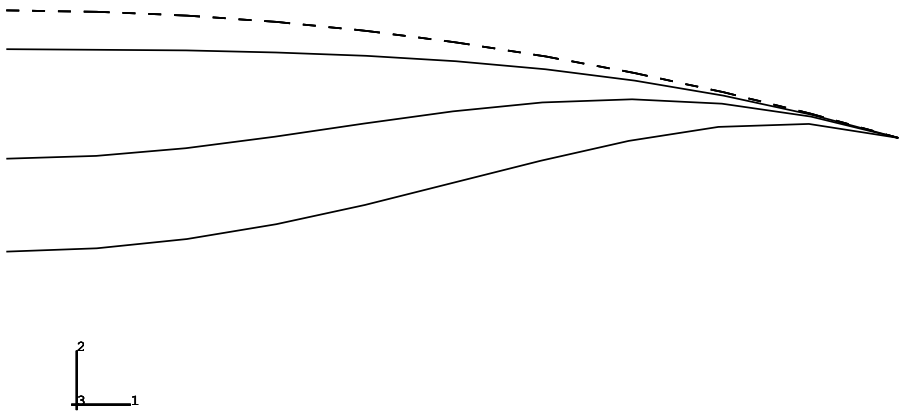


Figure 5: Deformed configuration plots for clamped shallow arch–Step 1.

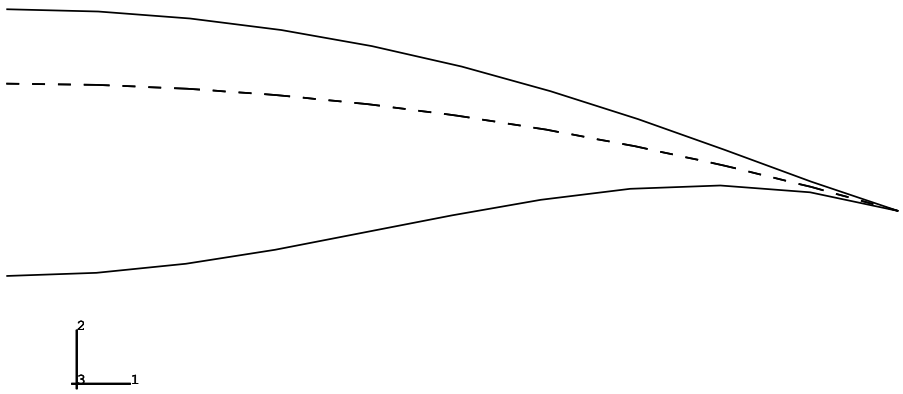


Figure 6: Deformed configuration plots for clamped shallow arch–Step 2.

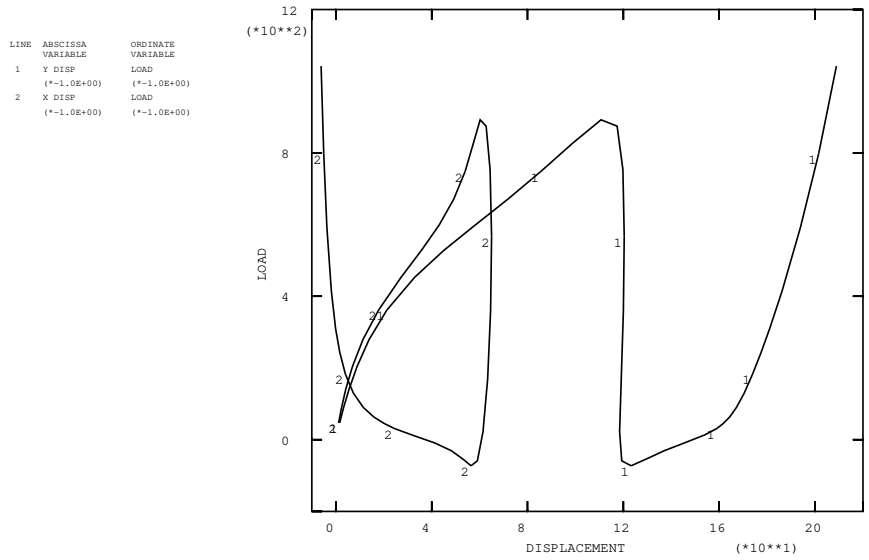


Figure 7: Load versus displacement curves for deep clamped-hinged arch.

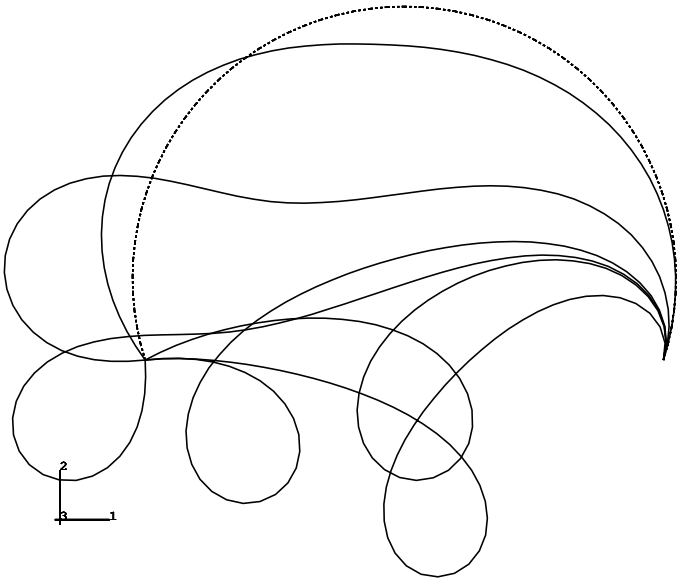


Figure 8: Deformed configuration plots for deep clamped-hinged arch.

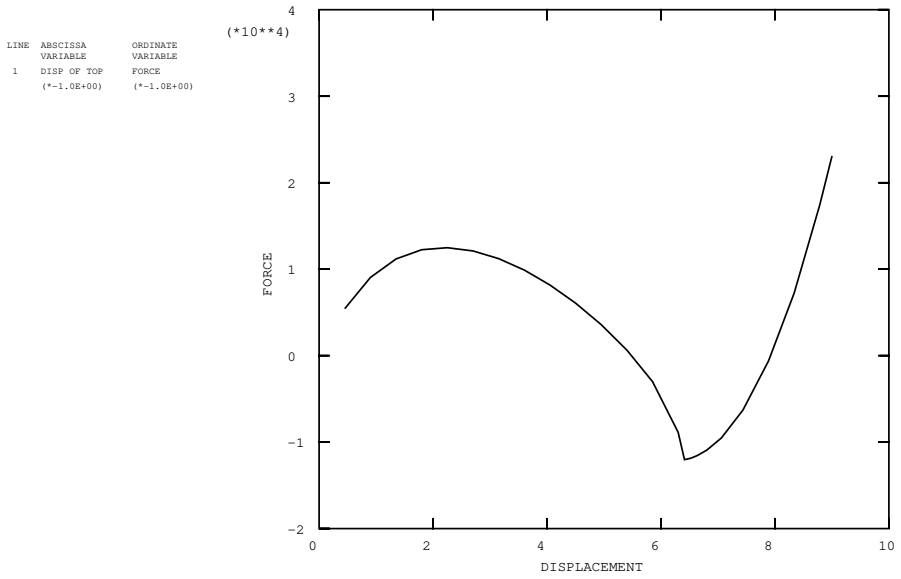


Figure 9: Force versus displacement curve for fixed displacement of pinned shallow arch.

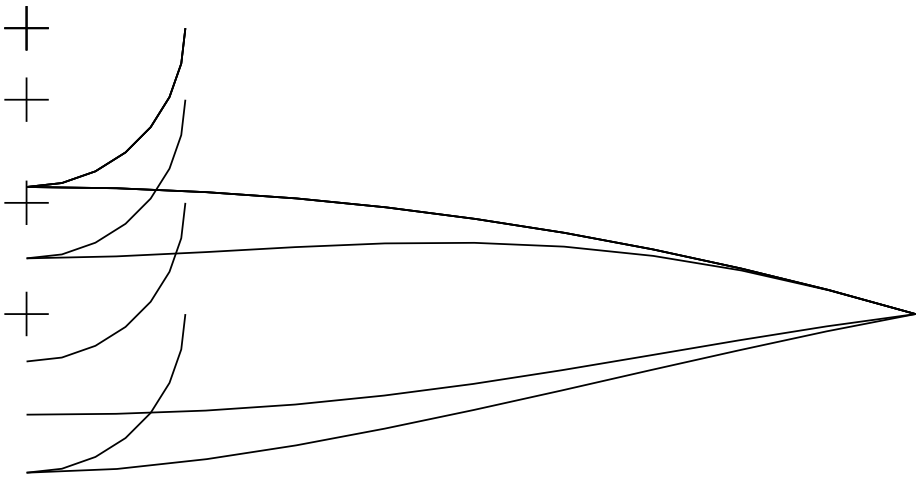


Figure 10: Deformed configuration plots for pinned arch depressed with rigid punch.

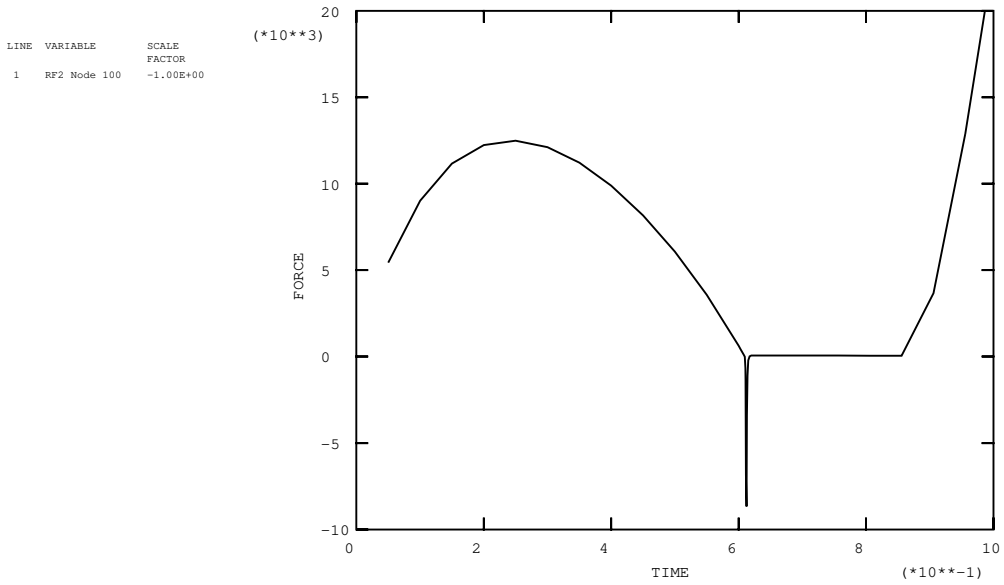


Figure 11: Force between the punch and the top of the pinned arch.

Laminated composite shells: buckling of a cylindrical panel with a circular hole

Products: Abaqus/Standard

Overview

This example illustrates modeling a thin, laminated composite shell in the presence of buckling.

The objective is to determine the strength of a thin, laminated composite shell, typical of shells used to form the outer surfaces of aircraft fuselages and rocket motors. Such analyses are complicated by the fact that these shells typically include local discontinuities—stiffeners and cutouts—which can induce substantial stress concentrations that can delaminate the composite material. In the presence of buckling this delamination can propagate through the structure to cause failure. In this example we study only the geometrically nonlinear behavior of the shell: delamination or other section failures are not considered. Some estimate of the possibility of material failure could presumably be made from the stresses predicted in the analyses reported here, but no such assessment is included in this example.

The example makes extensive use of material orientation in a general shell section to define the multilayered, anisotropic, laminated section. The various orientation options for shells are discussed in *Analysis of an anisotropic layered plate*.

General shell sections offer two methods of defining laminated sections: defining the thickness, material, and orientation of each layer or defining the equivalent section properties directly. The last method is particularly useful if the laminate properties are obtained directly from experiments or a separate preprocessor. This example uses both methods with a general shell section definition. Alternatively, you could use a shell section to analyze the model; however, because the material behavior is linear, no difference in solution would be obtained and the computational costs would be greater.

Geometry and model

The structure analyzed is shown in *Figure 1* and was originally studied experimentally by Knight and Starnes (1984). The test specimen is a cylindrical panel with a 355.6 mm (14 in) square platform and a 381 mm (15 in) radius of curvature, so that the panel covers a 55.6° arc of the cylinder. The panel contains a centrally located hole of 50.8 mm (2 in) diameter. The shell consists of 16 layers of unidirectional graphite fibers in an epoxy resin. Each layer is 0.142 mm (.0056 in) thick. The layers are arranged in the symmetric stacking sequence $\{\pm 45/90/0/0/90/\mp 45\}$ degrees repeated twice. The nominal orthotropic elastic material properties as defined by Stanley (1985) are

$E_{11} = 135$ kN/mm^2 $(19.6 \times 10^6$ lb/in$^2)$,

$E_{22} = 13$ kN/mm^2 $(1.89 \times 10^6$ lb/in$^2)$,

$G_{12} = G_{13} = 6.4$ kN/mm^2 $(.93 \times 10^6$ lb/in$^2)$,

$G_{23} = 4.3$ kN/mm^2 $(0.63 \times 10^6$ lb/in$^2)$,

$\nu_{12} = 0.38$,

where the 1-direction is along the fibers, the 2-direction is transverse to the fibers in the surface of the lamina, and the 3-direction is normal to the lamina.

The panel is fully clamped on the bottom edge, clamped except for axial motion on the top edge and simply supported along its vertical edges. Three analyses are considered. The first is a linear (prebuckling) analysis in which the panel is subjected to a uniform end shortening of 0.8 mm (.0316 in). The total axial force and the distribution of axial force along the midsection are used to compare the results with those obtained by Stanley (1985). The second analysis consists of an eigenvalue extraction of the first five buckling modes. The buckling

loads and mode shapes are also compared with those presented by Stanley (1985). Finally, a nonlinear load-deflection analysis is done to predict the postbuckling behavior, using the modified Riks algorithm. For this analysis an initial imperfection is introduced. The imperfection is based on the fourth buckling mode extracted during the second analysis. These results are compared with those of Stanley (1985) and with the experimental measurements of Knight and Starnes (1984).

The mesh used in Abaqus is shown in *Figure 2*. The anisotropic material behavior precludes any symmetry assumptions, hence the entire panel is modeled. The same mesh is used with the 4-node shell element (type S4R5) and also with the 9-node shell element (type S9R5); the 9-node element mesh, thus, has about four times the number of degrees of freedom as the 4-node element mesh. The 6-node triangular shell element STRI65 is also used; it employs two triangles for each quadrilateral element of the second-order mesh. Mesh generation is facilitated by specifying node fill and node mapping, as shown in the input data. In this model specification of the relative angle of orientation to define the material orientation within each layer, along with orthotropic elasticity in plane stress, makes the definition of the laminae properties straightforward.

The shell elements used in this example use an approximation to thin shell theory, based on a numerical penalty applied to the transverse shear strain along the element edges. These elements are not universally applicable to the analysis of composites since transverse shear effects can be significant in such cases and these elements are not designed to model them accurately. Here, however, the geometry of the panel is that of a thin shell; and the symmetrical lay-up, along with the relatively large number of laminae, tends to diminish the importance of transverse shear deformation on the response.

Relation between stress resultants and generalized strains

The shell section is most easily defined by giving the layer thickness, material, and orientation, in which case Abaqus preintegrates to obtain the section stiffness properties. However, the user can choose to input the section stiffness properties directly instead, as follows.

In Abaqus a lamina is considered as an orthotropic sheet in plane stress. The principal material axes of the lamina (see *Figure 3*) are longitudinal, denoted by L; transverse to the fiber direction in the surface of the lamina, denoted by T; and normal to the lamina surface, denoted by N. The constitutive relations for a general orthotropic material in the principal directions (L, T, N) are

$$\begin{Bmatrix} \sigma_L \\ \sigma_T \\ \tau_{LT} \\ \sigma_N \\ \tau_{LN} \\ \tau_{TN} \end{Bmatrix} = \begin{bmatrix} C_{11} & C_{12} & 0 & C_{14} & 0 & 0 \\ C_{12} & C_{22} & 0 & C_{24} & 0 & 0 \\ 0 & 0 & C_{33} & 0 & 0 & 0 \\ C_{14} & C_{24} & 0 & C_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & C_{55} & 0 \\ 0 & 0 & 0 & 0 & 0 & C_{66} \end{bmatrix} \begin{Bmatrix} \epsilon_L \\ \epsilon_T \\ \gamma_{LT} \\ \epsilon_N \\ \gamma_{LN} \\ \gamma_{TN} \end{Bmatrix}.$$

In terms of the data required to define orthotropic elasticity by specifying terms in the elastic stiffness matrix in Abaqus these are

$$\begin{aligned} &C_{11} = D_{1111}, \quad C_{12} = D_{1122}, \quad C_{14} = D_{1133}, \quad C_{22} = D_{2222} \\ &C_{24} = D_{2233}, \quad C_{33} = D_{1212}, \quad C_{44} = D_{3333}, \quad C_{55} = D_{1313} \\ &C_{66} = D_{2323} \end{aligned}$$

This matrix is symmetric and has nine independent constants. If we assume a state of plane stress, then σ_N is taken to be zero. This yields

$$\begin{Bmatrix} \sigma_L \\ \sigma_T \\ \tau_{LT} \\ \tau_{LN} \\ \tau_{TN} \end{Bmatrix} = \begin{bmatrix} Q_{11} & Q_{12} & 0 & 0 & 0 \\ Q_{12} & Q_{22} & 0 & 0 & 0 \\ 0 & 0 & Q_{33} & 0 & 0 \\ 0 & 0 & 0 & Q_{55} & 0 \\ 0 & 0 & 0 & 0 & Q_{66} \end{bmatrix} \begin{Bmatrix} \epsilon_L \\ \epsilon_T \\ \gamma_{LT} \\ \gamma_{LN} \\ \gamma_{TN} \end{Bmatrix},$$

where

$$\begin{aligned} Q_{11} &= C_{11} - \frac{C_{14}C_{14}}{C_{44}}, \\ Q_{12} &= C_{12} - \frac{C_{24}C_{14}}{C_{44}}, \\ Q_{22} &= C_{22} - \frac{C_{24}C_{24}}{C_{44}}, \\ Q_{33} &= C_{33}, \\ Q_{55} &= C_{55}, \\ Q_{66} &= C_{66}. \end{aligned}$$

The correspondence between these terms and the usual engineering constants that might be given for a simple orthotropic layer in a laminate is

$$\begin{aligned} Q_{11} &= \frac{E_1}{1-\nu_{12}\nu_{21}}, \\ Q_{12} &= \frac{\nu_{12}E_2}{1-\nu_{12}\nu_{21}} = \frac{\nu_{21}E_1}{1-\nu_{12}\nu_{21}}, \\ Q_{22} &= \frac{E_2}{1-\nu_{12}\nu_{21}}, \\ Q_{33} &= G_{12}, \\ Q_{55} &= G_{13}, \\ Q_{66} &= G_{23}. \end{aligned}$$

The parameters used on the right-hand side of the above equation are those that must be provided as part of the definition of orthotropic elasticity in plane stress.

If the $(\mathbf{1}, \mathbf{2}, \boldsymbol{N})$ system denotes the standard shell basis directions that Abaqus chooses by default, the local stiffness components must be rotated to this system to construct the lamina's contribution to the general shell section stiffness. Since $\boldsymbol{Q_{ij}}$ represent fourth-order tensors, in the case of a lamina they are oriented at an angle $\boldsymbol{\theta}$ to the standard shell basis directions used in Abaqus. Hence, the transformation is

$$
\begin{aligned}
\overline{Q}_{11} &= Q_{11}\cos^4\theta + 2\left(Q_{12}+2Q_{33}\right)\sin^2\theta\cos^2\theta + Q_{22}\sin^4\theta,\\
\overline{Q}_{12} &= \left(Q_{11}+Q_{22}-4Q_{33}\right)\sin^2\theta\cos^2\theta + Q_{12}\left(\sin^4\theta+\cos^4\theta\right),\\
\overline{Q}_{22} &= Q_{11}\sin^4\theta + 2\left(Q_{12}+2Q_{33}\right)\sin^2\theta\cos^2\theta + Q_{22}\cos^4\theta,\\
\overline{Q}_{13} &= \left(Q_{11}-Q_{12}-2Q_{33}\right)\sin\theta\cos^3\theta + \left(Q_{12}-Q_{22}+2Q_{33}\right)\sin^3\theta\cos\theta,\\
\overline{Q}_{23} &= \left(Q_{11}-Q_{12}-2Q_{33}\right)\sin^3\theta\cos\theta + \left(Q_{12}-Q_{22}+2Q_{33}\right)\sin\theta\cos^3\theta,\\
\overline{Q}_{33} &= \left(Q_{11}+Q_{22}-2Q_{12}-2Q_{33}\right)\sin^2\theta\cos^2\theta + Q_{33}\left(\sin^4\theta+\cos^4\theta\right),\\
\overline{Q}_{55} &= Q_{55}\cos^2\theta - Q_{66}\sin^2\theta,\\
\overline{Q}_{56} &= Q_{55}\sin\theta\cos\theta - Q_{66}\sin\theta\cos\theta,\\
\overline{Q}_{66} &= Q_{55}\sin^2\theta - Q_{66}\cos^2\theta,
\end{aligned}
$$

where $\overline{Q}_{ij}$ are the stiffness coefficients in the standard shell basis directions used by Abaqus.

Abaqus assumes that a laminate is a stack of laminae arranged with the principal directions of each layer in different orientations. The various layers are assumed to be rigidly bonded together. The section force and moment resultants per unit length in the normal basis directions in a given layer can be defined on this basis as

$$
\begin{aligned}
\left(N_1, N_2, N_{12}\right) &= \int_{-h/2}^{h/2} \left(\sigma_1, \sigma_2, \tau_{12}\right) dz,\\
\left(M_1, M_2, M_{12}\right) &= \int_{-h/2}^{h/2} \left(\sigma_1, \sigma_2, \tau_{12}\right) z\, dz,\\
\left(V_1, V_2\right) &= \int_{-h/2}^{h/2} \left(\tau_{13}, \tau_{23}\right) dz,
\end{aligned}
$$

where h is the thickness of the layer.

This leads to the relations

$$
\begin{Bmatrix} N_1\\ N_2\\ N_{12}\\ M_1\\ M_2\\ M_{12}\\ V_1\\ V_2 \end{Bmatrix} =
\begin{bmatrix}
A_{11} & A_{12} & A_{13} & B_{11} & B_{12} & B_{13} & 0 & 0\\
A_{12} & A_{22} & A_{23} & B_{12} & B_{22} & B_{23} & 0 & 0\\
A_{13} & A_{23} & A_{33} & B_{13} & B_{23} & B_{33} & 0 & 0\\
B_{11} & B_{12} & B_{13} & D_{11} & D_{12} & D_{13} & 0 & 0\\
B_{12} & B_{22} & B_{23} & D_{12} & D_{22} & D_{23} & 0 & 0\\
B_{13} & B_{23} & B_{33} & D_{13} & D_{23} & D_{33} & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & E_{11} & E_{12}\\
0 & 0 & 0 & 0 & 0 & 0 & E_{12} & E_{22}
\end{bmatrix}
\begin{Bmatrix} \epsilon_1\\ \epsilon_2\\ \gamma_{12}\\ \kappa_1\\ \kappa_2\\ \kappa_{12}\\ \gamma_{13}\\ \gamma_{23} \end{Bmatrix},
$$

where the components of this section stiffness matrix are given by

$$
\begin{aligned}
\left(A_{ij}, B_{ij}, D_{ij}\right) &= \int_{-h/2}^{h/2} \overline{Q}_{ij}^{m} \left(1, z, z^2\right)\, dz, \quad (i,j=1,2,3)\\
E_{ij} &= \int_{-h/2}^{h/2} \overline{Q}_{\alpha\beta}^{m} k_i k_j\, dz, \quad (i,j=1,2, \text{ and } \alpha, \quad \beta = i+4, j+4).
\end{aligned}
$$

Here m indicates a particular layer. Thus, the $\overline{Q}_{ij}^{m}$ depend on the material properties and fiber orientation of the mth layer. The k_i , $i=$ 1,2 parameters are the shear correction coefficients as defined by Whitney (1973). If

there are n layers in the lay-up, we can rewrite the above equations as a summation of integrals over the n laminae. The material coefficients will then take the form

$$A_{ij} = \sum_{m=1}^{n} \overline{Q}_{ij}^{m} \left(h_m - h_{m-1}\right),$$

$$B_{ij} = \frac{1}{2} \sum_{m=1}^{n} \overline{Q}_{ij}^{m} \left(h_m^2 - h_{m-1}^2\right),$$

$$D_{ij} = \frac{1}{3} \sum_{m=1}^{n} \overline{Q}_{ij}^{m} \left(h_m^3 - h_{m-1}^3\right),$$

$$E_{ij} = \sum_{m=1}^{n} \overline{Q}_{\alpha\beta}^{m} \left(h_m - h_{m-1}\right) k_i k_j,$$

where the h_m and h_{m-1} in these equations indicate that the mth lamina is bounded by surfaces $z = h_m$ and $z = h_{m-1}$. See *Figure 4* for the nomenclature.

These equations define the coefficients required for the direct input of the section stiffness matrix method in the general shell section. Only the $[A]$, $[B]$, and $[D]$ submatrices are needed for that option. The three terms in $[E]$, if required, are defined as part of the transverse shear stiffness. The section forces as defined above are in the normal shell basis directions.

Applying these equations to the laminate defined for this example leads to the following overall section stiffness:

$$[A] = \begin{bmatrix} 138.385 & 44.0189 & 0 \\ 44.0189 & 138.385 & 0 \\ 0 & 0 & 47.1831 \end{bmatrix} \text{ kN/mm}; \quad [B] = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix};$$

$$[D] = \begin{bmatrix} 55.670 & 21.638 & 2.138 \\ 21.638 & 58.521 & 2.138 \\ 2.138 & 2.138 & 23.004 \end{bmatrix} \text{ kN-mm}; \quad [E] = \begin{bmatrix} 12.2387 & 0 \\ 0 & 12.2387 \end{bmatrix} \text{ kN/mm},$$

or

$$[A] = \begin{bmatrix} 790.239 & 251.367 & 0 \\ 251.367 & 790.239 & 0 \\ 0 & 0 & 269.436 \end{bmatrix} \times 10^3 \text{ lb/in}; \quad [B] = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix};$$

$$[D] = \begin{bmatrix} 492.719 & 191.513 & 18.9245 \\ 191.513 & 517.951 & 18.9245 \\ 18.9245 & 18.9245 & 203.602 \end{bmatrix} \text{ lb-in}; \quad [E] = \begin{bmatrix} 49.573 & 0.002 \\ 0.002 & 52.967 \end{bmatrix} \times 10^3 \text{ lb/in}.$$

Results and discussion

The total axial force necessary to compress the panel 0.803 mm (0.0316 in) is 100.2 kN (22529 lb) for the mesh of S9R5 elements, 99.5 kN (22359 lb) for the mesh of S4R5 elements, and 100.3 kN (22547 lb) for the mesh of STRI65 elements. These values match closely with the result of 100 kN (22480 lb) reported by Stanley (1985). *Figure 5* shows the displaced configuration and a profile of axial force along the midsection of the panel (at $z = L/2$). It is interesting to note that the axial load is distributed almost evenly across the entire panel, with only a very localized area near the hole subjected to an amplified stress level. This suggests that adequate results for this linear analysis could also be obtained with a coarser mesh that has a bias toward the hole.

The second stage of the analysis is the eigenvalue buckling prediction. To obtain the buckling predictions with Abaqus, an eigenvalue buckling prediction step is run. In this step nominal values of load are applied. The magnitude that is used is not of any significance, since eigenvalue buckling is a linear perturbation procedure: the stiffness matrix and the stress stiffening matrix are evaluated at the beginning of the step without any of this load applied. The eigenvalue buckling prediction step calculates the eigenvalues that, multiplied with the applied load and added to any “base state” loading, are the predicted buckling loads. The eigenvectors associated with the eigenvalues are also obtained. This procedure is described in more detail in *Eigenvalue Buckling Prediction*.

The buckling predictions are summarized in *Table 1* and *Figure 6*. The buckling load predictions from Abaqus are higher than those reported by Stanley. The eigenmode predictions given by the mesh using element types S4R5, S9R5, and STRI65 are all the same and agree well with those reported by Stanley. Stanley makes several important observations that remain valid for the Abaqus results: (1) the eigenvalues are closely spaced; (2) nevertheless, the mode shapes vary significantly in character; (3) the first buckling mode bears the most similarity to the linear prebuckling solution; (4) there is no symmetry available that can be utilized for computational efficiency.

Following the eigenvalue buckling analyses, nonlinear postbuckling analysis is carried out by imposing an imperfection based on the fourth buckling mode. The maximum initial perturbation is 10% of the thickness of the shell. The load versus normalized displacement plots for the S9R5 mesh, the S4R5 mesh, and the STRI65 mesh are compared with the experimental results and those given by Stanley in *Figure 7*. The overall response prediction is quite similar for the Abaqus elements, although the general behavior predicted by Stanley is somewhat different. The Abaqus results show a peak load slightly above the buckling load predicted by the eigenvalue extraction, while Stanley's results show a significantly lower peak load. In addition, the Abaqus results show rather less loss of strength after the initial peak, followed quite soon by positive stiffness again. Neither the Abaqus results nor Stanley's results agree closely with the experimentally observed dramatic loss of strength after peak load. Stanley ascribes this to material failure (presumably delamination), which is not modeled in his analyses or in these.

Figure 8 shows the deformed configurations for the panel during its postbuckling response. The plots show the results for S4R5, but the pattern is similar for S9R5 and STRI65. The response is quite symmetric initially; but, as the critical load is approached, a nonsymmetric dimple develops and grows, presumably accounting for the panel's loss of strength. Later in the postbuckling response another wrinkle can be seen to be developing.

Input files

laminpanel_s9r5_prebuckle.inp	Prebuckling analysis for the 9-node (element type S9R5) mesh.
laminpanel_s9r5_buckle.inp	Eigenvalue buckling prediction using element type S9R5.
laminpanel_s9r5_postbuckle.inp	Nonlinear postbuckling analysis using element type S9R5.
laminpanel_s4r5_prebuckle.inp	Prebuckling analysis using element type S4R5.
laminpanel_s4r5_buckle.inp	Eigenvalue buckling prediction using element type S4R5.
laminpanel_s4r5_postbuckle.inp	Nonlinear postbuckling analysis using element type S4R5.
laminpanel_s4r5_node.inp	Nodal coordinate data for the imperfection imposed for the postbuckling analysis using element type S4R5.
laminpanel_s9r5_stri65_node.inp	Nodal coordinate data for the imperfection imposed for the postbuckling analysis using element types S9R5 and STRI65.
laminpanel_stri65_prebuckle.inp	Prebuckling analysis using element type STRI65.
laminpanel_stri65_buckle.inp	Eigenvalue buckling prediction using element type STRI65.
laminpanel_stri65_postbuckle.inp	Nonlinear postbuckling analysis using element type STRI65.
laminpanel_s4_prebuckle.inp	Prebuckling analysis using element type S4.
laminpanel_s4_buckle.inp	Eigenvalue buckling prediction using element type S4.
laminpanel_s4_postbuckle.inp	Nonlinear postbuckling analysis using element type S4.

References

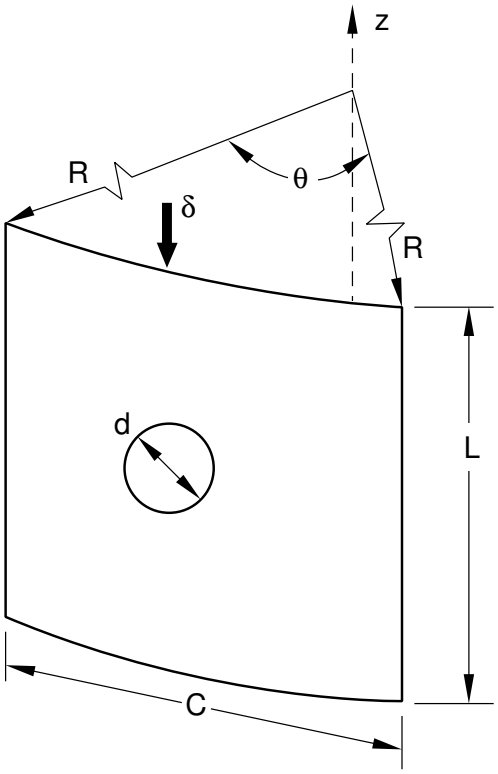
- Knight, N. F., and J. H. Starnes, Jr., "Postbuckling Behavior of Axially Compressed Graphite-Epoxy Cylindrical Panels with Circular Holes," *presented at the 1984 ASME Joint Pressure Vessels and Piping/Applied Mechanics Conference, San Antonio, Texas*, 1984.
- Stanley, G. M., Continuum-Based Shell Elements, Ph.D. Dissertation, Department of Mechanical Engineering, Stanford University, 1985.
- Whitney, J. M., "Shear Correction Factors for Orthotropic Laminates Under Static Loads," *Journal of Applied Mechanics*, Transactions of the ASME, vol. 40, pp. 302–304, 1973.

Tables

Table 1: Summary of buckling load predictions.

Mode 1	Stanley	107.0 kN (24054 lb)
	S9R5	113.4 kN (25501 lb)
	S4R5	115.5 kN (25964 lb)
	S4	114.3 kN (25696 lb)
	STRI65	113.8 kN (25579 lb)
Mode 2	Stanley	109.6 kN (24638 lb)
	S9R5	117.6 kN (26429 lb)
	S4R5	121.2 kN (27244 lb)
	S4	116.5 kN (26196 lb)
	STRI65	117.8 kN (26492 lb)
Mode 3	Stanley	116.2 kN (26122 lb)
	S9R5	120.3 kN (27049 lb)
	S4R5	124.7 kN (28042 lb)
	S4	124.1 kN (27889 lb)
	STRI65	121.1 kN (27217 lb)
Mode 4	Stanley	140.1 kN (31494 lb)
	S9R5	147.5 kN (33161 lb)
	S4R5	156.1 kN (35092 lb)
	S4	152.3 kN (34247 lb)
	STRI65	146.9 kN (33015 lb)
Mode 5	Stanley	151.3 kN (34012 lb)
	S9R5	171.3 kN (38512 lb)
	S4R5	181.5 kN (40800 lb)
	S4	184.2 kN (41413 lb)
	STRI65	172.8 kN (38843 lb)

Figures



GEOMETRIC PROPERTIES:

L = 355.6 mm (14 in)
C = 355.6 mm (14 in)
R = 381.0 mm (15 in)
d = 50.80 mm (2 in)
$\theta = 55.6°$
h_{layer} = 0.142 mm (.0056 in)

LOADING:

Uniform axial compression δ = 0.803 mm (.0316 in)

Figure 1: Geometry for cylindrical panel with hole.

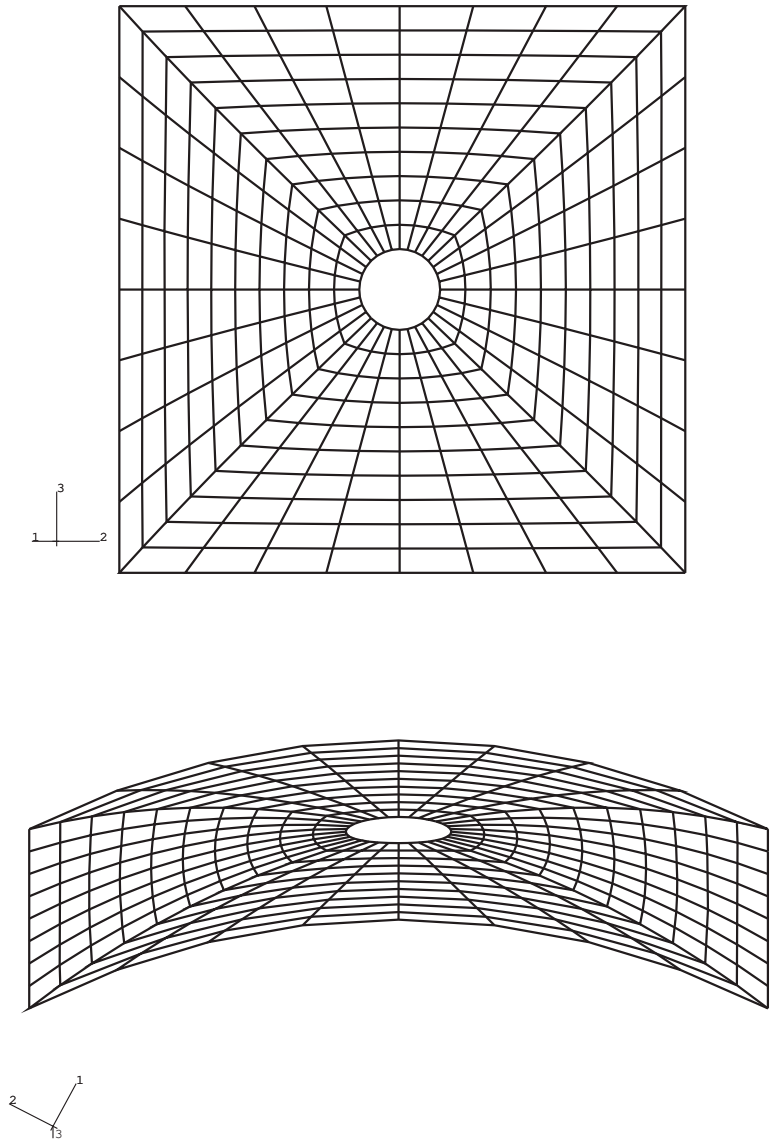


Figure 2: Mesh for cylindrical panel with hole.

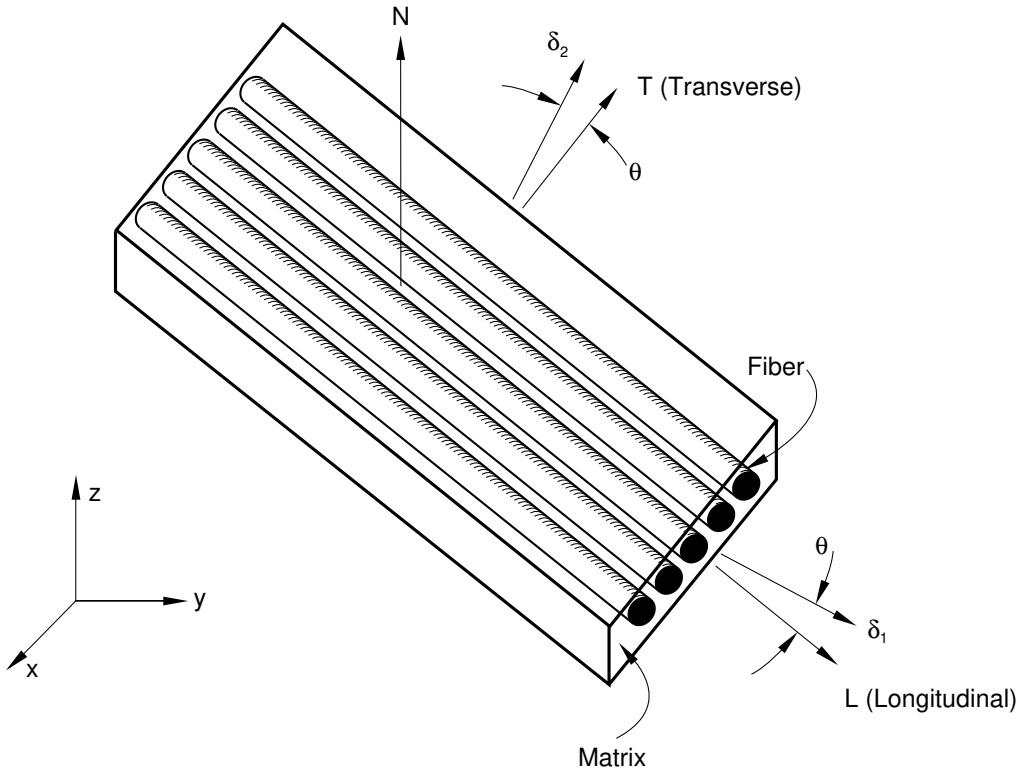


Figure 3: Typical lamina.

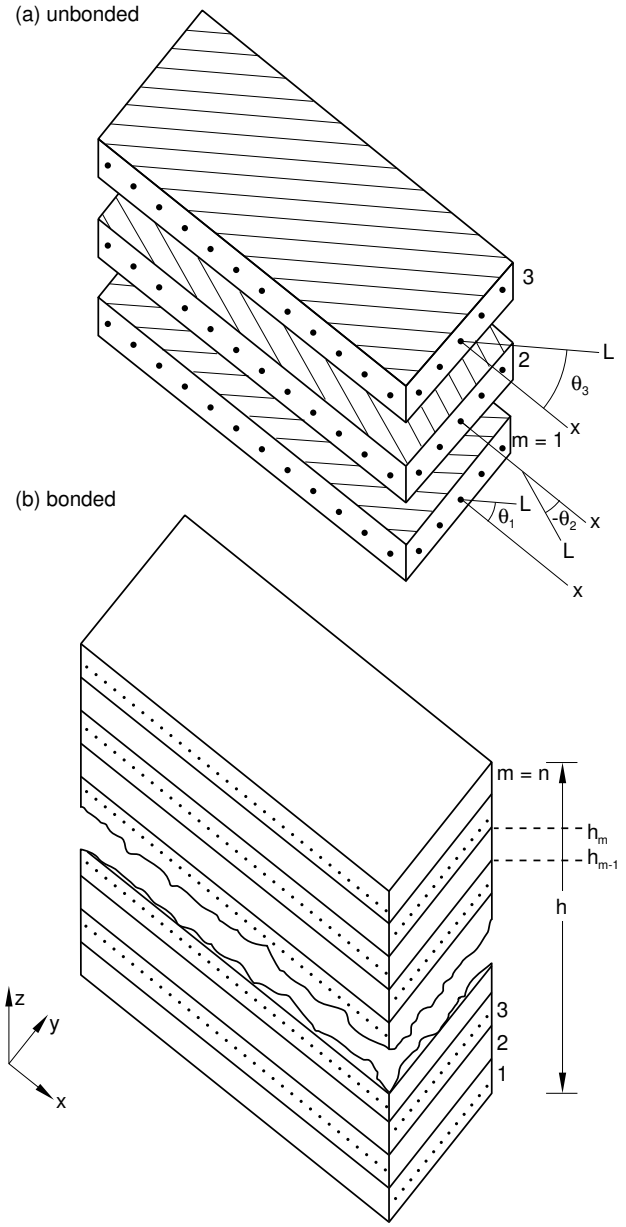
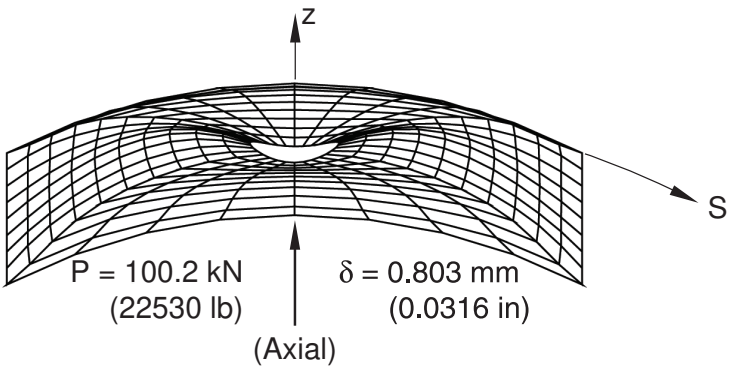


Figure 4: Typical laminate.



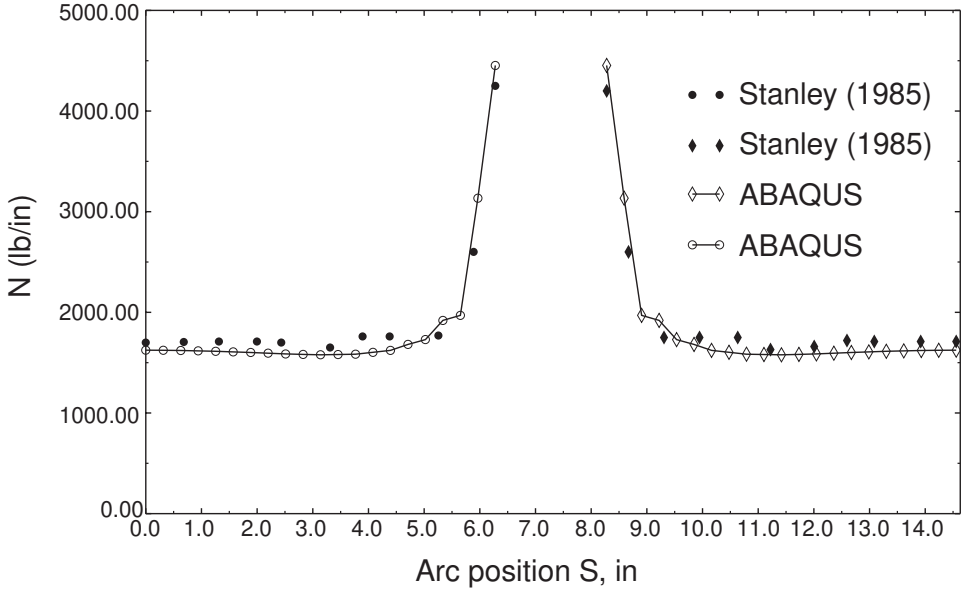
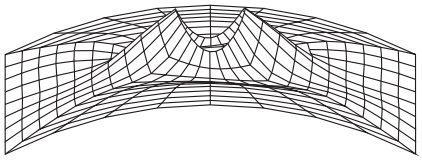
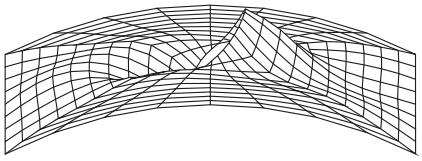


Figure 5: Displaced shape and axial force distribution.

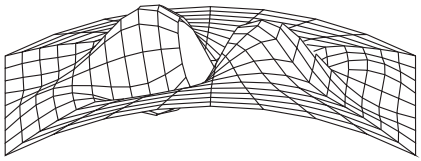
Buckling Mode 1



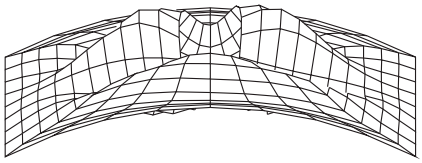
Buckling Mode 2



Buckling Mode 3



Buckling Mode 4



Buckling Mode 5

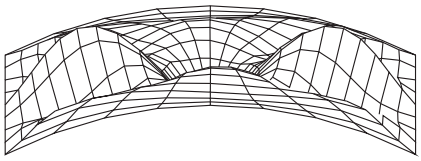


Figure 6: Buckling modes, element types S4R5, S9R5, and STRI65.

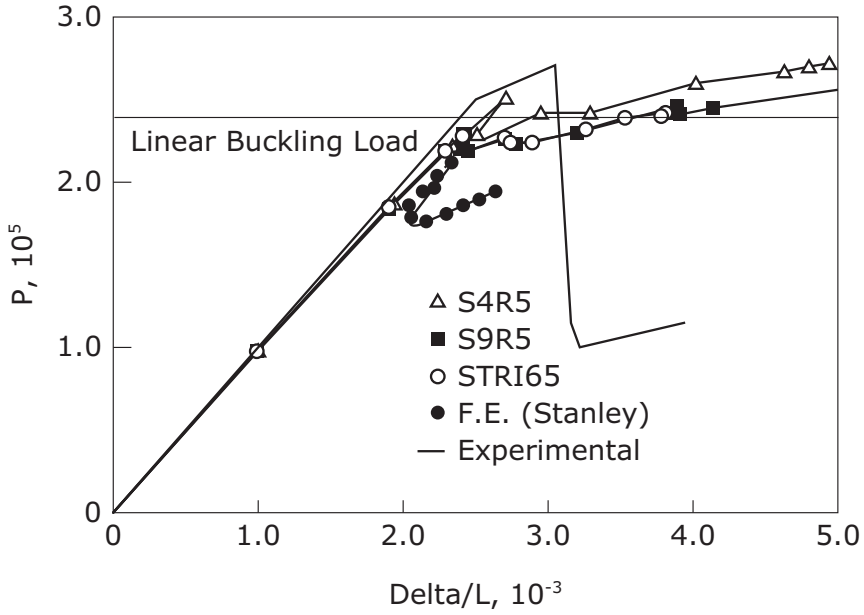


Figure 7: Load-displacement response.

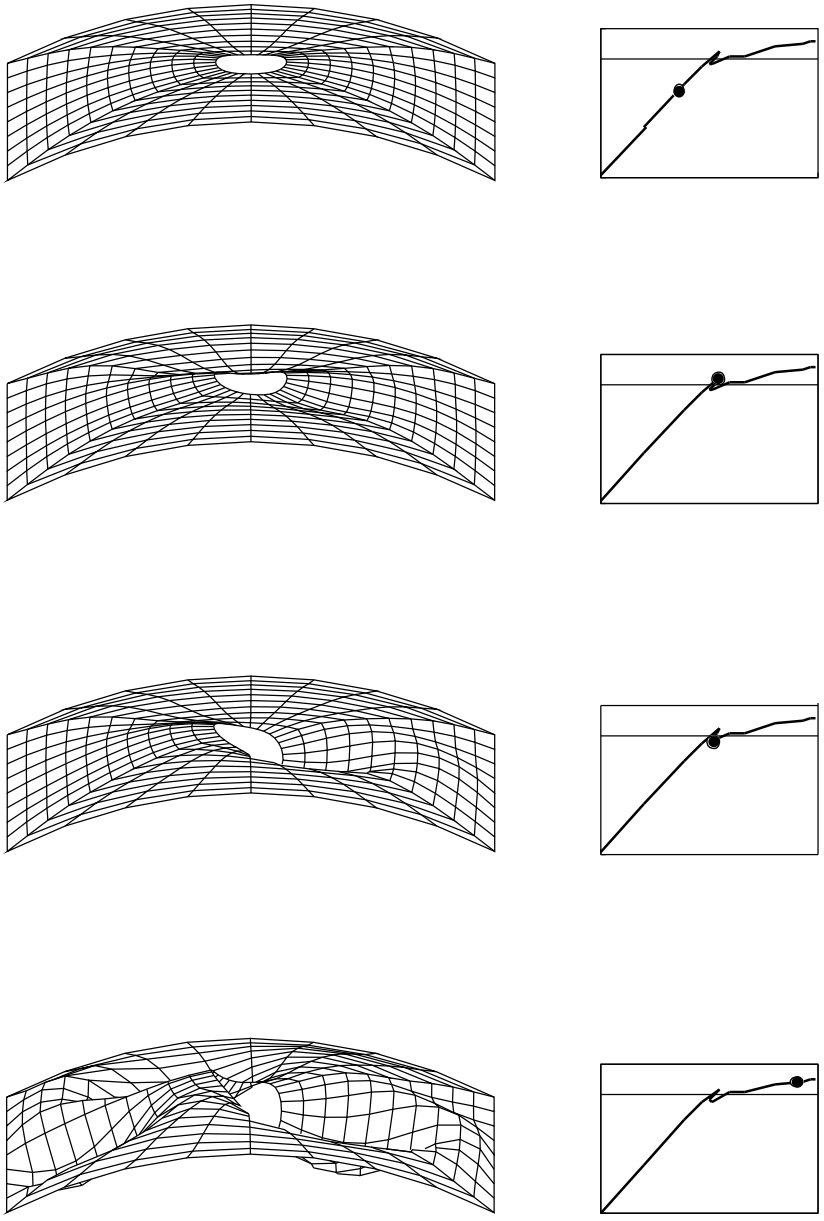


Figure 8: Postbuckling deformations: 10% h imperfection with S4R5.

Buckling of a column with spot welds

Products: Abaqus/Standard Abaqus/Explicit Abaqus/CAE

References:

- *Mesh-Independent Fasteners*

Overview

This example illustrates both a static and dynamic collapse of a steel column constructed by spot welding two channel sections.

This example is intended to illustrate the modeling of spot welds using mesh-independent spot weld modeling capabilities in Abaqus.

Problem description

The pillar is composed of two columns of different cross-sections, one box-shaped and the other W-shaped, welded together with spot welds (*Figure 1*). The top end of the pillar is connected to a rigid body, which makes the deformation of the pillar easy to control by manipulating the rigid body reference node. The box-shaped column is welded to the W-shaped column with five spot welds on either side of the box-shaped column.

The columns are both composed of aluminum-killed steel, which is assumed to satisfy the Ramberg-Osgood relation between true stress and logarithmic strain,

$$\varepsilon = \sigma/E + (\sigma/K)^n,$$

where Young's modulus (E) is 206.8 GPa, the reference stress value (K) is 0.510 GPa, and the work-hardening exponent (n) is 4.76. In the present Abaqus analyses the Ramberg-Osgood relation is approximated using elastic and plastic material properties. The material is assumed to be linear elastic up to a yield stress of 170.0 MPa, and the stress-strain curve beyond the yield stress is defined in piecewise linear segments using plastic material properties. Poisson's ratio is 0.3.

The spot welds are modeled in both Abaqus/Standard and Abaqus/Explicit using the mesh-independent fastener capability. Connector elements with CARTESIAN and CARDAN sections are used to define deformable fasteners. Alternatively, a BUSHING connection type could have been used. The element set that contains the connector elements is referenced in the mesh-independent fastener. The spot welds at nodes 5203, 15203, 25203, 35203, and 45203 are all located on the positive z-side of the box-shaped column, with node 5203 at the bottom end of the column and node 45203 at the top end of the column (see *Figure 2*). Spot welds at nodes 5211, 15211, 25211, 35211, and 45211 are all located on the negative z-side of the box-shaped column, with node 5211 at the bottom end of the column and node 45211 at the top end of the column. The surfaces of the box-shaped column and the W-shaped column are specified in the mesh-independent fastener. The spot welds are defined with a diameter of .002 m. The deformable behavior in the fastener is modeled using connector elasticity, with an elastic spring stiffness of 2×10^{11} N/m in translational as well as rotational components. For the Abaqus/Explicit analysis, spot weld damage and failure are modeled using connector damage behavior. A force-based coupled damage initiation criterion that uses a connector potential with both connector force and connector moment ingredients is used. (For further description of the connector potential used, see the spot weld example in *Connector Functions for Coupled Behavior*.) Damage initiates when the value of the potential exceeds 2×10^5 N. A post-damage-initiation equivalent displacement of 1×10^{-7} m is allowed. Once the post-damage-initiation equivalent displacement in a spot weld reaches this value, the spot weld ceases to carry any load. Both the continuum and structural coupling capabilities are used to define the fasteners.

A Python script is included that reproduces the model using the Scripting Interface in Abaqus/CAE. The script creates and assembles Abaqus/CAE parts and uses discrete fasteners to model the spot welds. The script creates an Abaqus/Standard model that is ready to be submitted for analysis from the Job module. The discrete fasteners created by the script result in the following differences compared with the mesh-independent, or point-based, fasteners used by the example input files:

- When you submit the Abaqus/CAE job for analysis, the discrete fasteners created by the Python script generate coupling constraints and distributing coupling constraints in the input file, together with connector elements. The example input files use mesh-independent fasteners to model point-based fasteners using connector elements.
- You must define the radius of influence when you create a discrete fastener using Abaqus/CAE. In contrast, the example input files allow Abaqus to compute a default value of the radius of influence based on the geometric properties of the fastener, the characteristic length of connected facets, and the type of weighting function selected.
- The input files share nodes between the pillar and the rigid body. To achieve similar behavior, the Python script creates tie constraints between the pillar and the rigid body.

For a description of the differences between discrete fasteners and point-based fasteners in Abaqus/CAE, see *About fasteners*.

Loading

The bottom of the pillar is fully fixed. In the Abaqus/Standard analysis the reference node for the rigid body at the top of the pillar moves 0.25 m in the y-direction, thus loading it in compression, together with a displacement of .02 m in the z-direction that shears it slightly. At the same time, the end of the pillar is rotated about the negative z-axis by 0.785 rad and rotated about the negative x-axis by 0.07 rad.

In the Abaqus/Explicit analyses the reference node for the rigid body at the top of the pillar moves at a constant velocity of 25 m/sec in the y-direction, thus loading it in compression, together with a velocity of 2 m/sec in the z-direction that shears it slightly. At the same time, the end of the pillar is rotated about the negative z-axis at 78.5 rad/sec and rotated about the negative x-axis at 7 rad/sec. This loading is applied by prescribing the velocities of the rigid body reference node that is attached to the top end of the compound pillar.

The analysis is carried out over 10 milliseconds.

Results and discussion

Figure 3 shows the deformed shape of the pillar after 5.0 msec in the Abaqus/Explicit analysis. *Figure 4* shows the deformed shape of the pillar after 10.0 msec. The spot welds in the mesh-independent Abaqus/Explicit analysis undergo damage and fail. For the current choice of parameters for the connector damage model, it is found that damage initiates in the spot welds at nodes 15203 through 45203 on the positive side of the box-shaped column and at nodes 15211 through 45211 on the negative side of the box-shaped column. However, the post-damage-initiation displacement is sufficient to cause ultimate failure of the spot welds at nodes 15203, 25203, 15211, and 25211 only. *Figure 5* illustrates the undamaged connector force CTF3 in the spot welds associated with reference nodes 25203 and 25211 as computed in the Abaqus/Standard analyses. *Figure 6* illustrates the damaged connector force CTF3 in the spot welds associated with reference nodes 25203 and 25211 as computed in the Abaqus/Explicit analyses. Forces in both spot welds drop to zero when ultimate failure occurs in the Abaqus/Explicit analyses. *Figure 7* shows the time history of the total kinetic energy, the total work done on the model, the total energy dissipated by friction, the total internal energy, and the total energy balance.

Input files

pillar_fastener_xpl.inp	Input data for the Abaqus/Explicit mesh-independent spot weld analysis.

pillar_fastener_structcoup_xpl.inp	Input data for the Abaqus/Explicit mesh-independent spot weld analysis using structural coupling in the fastener definitions.
pillar_fastener_std.inp	Input data for the Abaqus/Standard mesh-independent spot weld analysis.
pillar_fastener_structcoup_std.inp	Input data for the Abaqus/Standard mesh-independent spot weld analysis using structural coupling in the fastener definitions.
pillar_fastener_smslide_std.inp	Input data for the Abaqus/Standard mesh-independent spot weld analysis using small-sliding contact with shell thickness taken into account.
pillar_rest.inp	Input data used to test the restart capability with spot welds.

Python scripts

pillar_fastener_std.py	Script that creates a model with discrete fasteners using Abaqus/CAE.

Figures

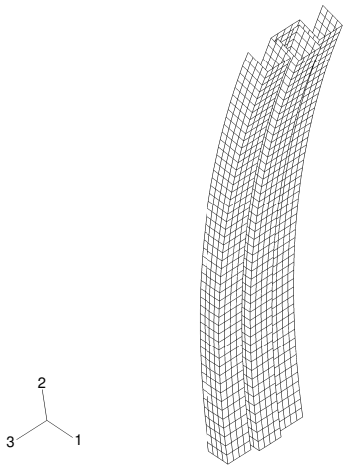


Figure 1: Initial configuration of the compound pillar.

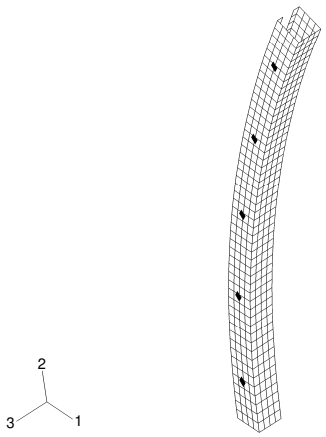


Figure 2: Initial configuration of the box-shaped column showing spot welds.

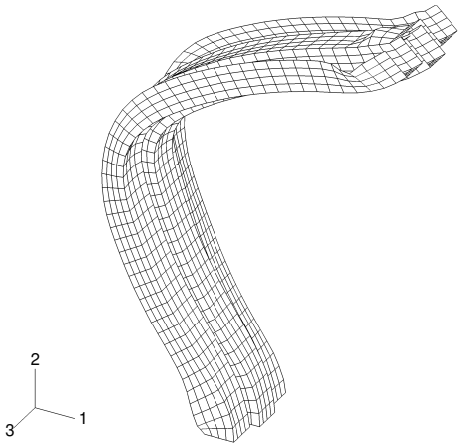


Figure 3: Deformed shape at 5.0 msec.

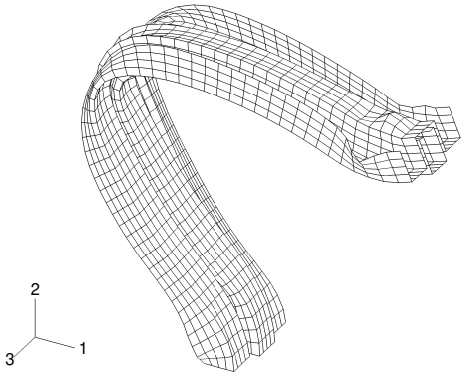


Figure 4: Deformed shape at 10.0 msec.

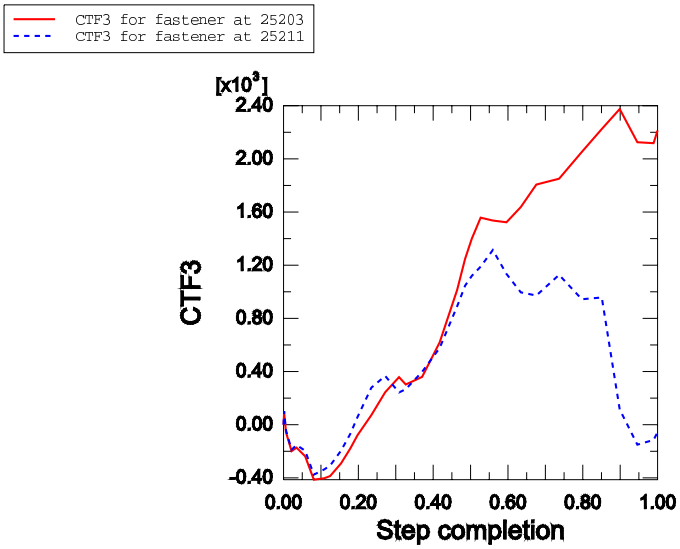


Figure 5: Connector force CTF3 in spot welds at reference nodes 25203 and 25211 for Abaqus/Standard.

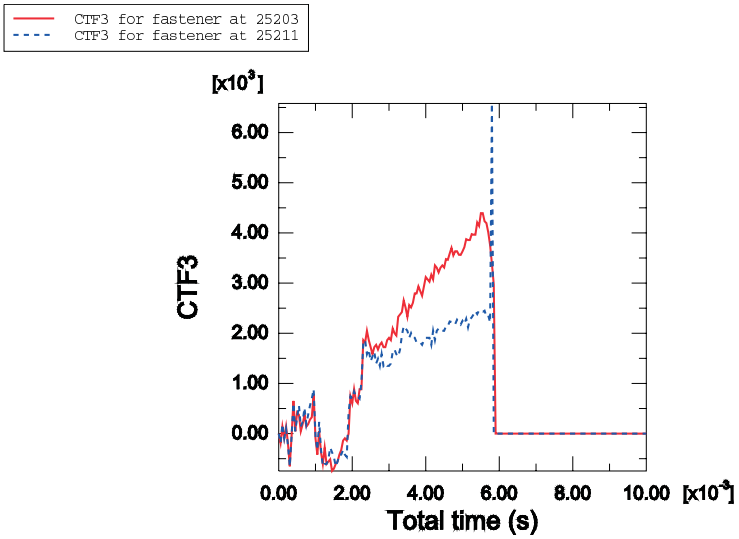


Figure 6: Connector force CTF3 in spot welds at reference nodes 25203 and 25211 for Abaqus/Explicit.

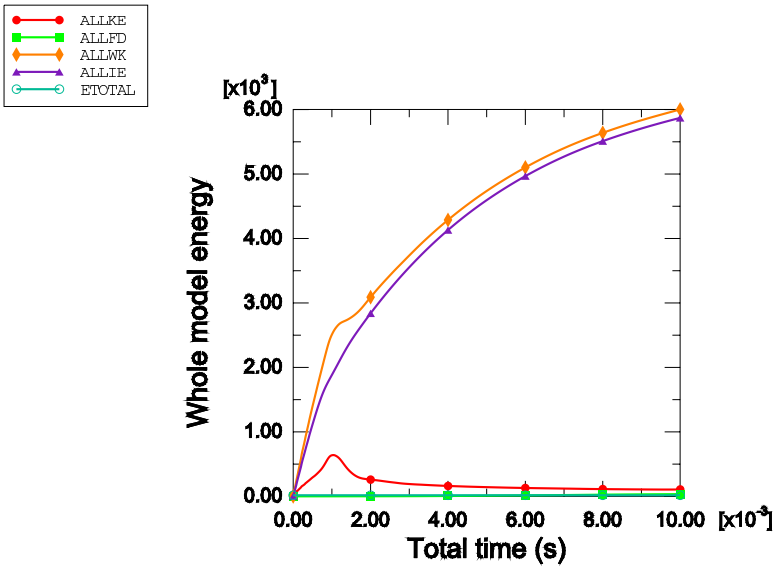


Figure 7: Time histories of the total kinetic energy, energy dissipated by friction, work done on the model, internal energy, and total energy.

Elastic-plastic K-frame structure

Products: Abaqus/Standard

References:

- *Frame Elements*

Overview

This example shows how frame elements (FRAME2D) can be used to model elastic, elastic-plastic, and buckling strut responses of individual members of frame-like structures.

The elastic response is defined by Euler-Bernoulli beam theory. The elastic-plastic response is modeled with nonlinear kinematic hardening plasticity concentrated at the element's ends, simulating the development of plastic hinges. The buckling strut response is a simplified, phenomenological representation of the highly nonlinear cross-section collapse and material yielding that takes place when slender members are loaded in compression .Therefore, frame elements can be elastic, elastic-plastic, behave as struts (with or without buckling), or switch during the analysis to strut behavior followed by postbuckling behavior. Both the elastic-plastic and buckling strut responses are simplifications of highly nonlinear responses. They are designed to approximate these complex responses with a single finite element representing a structural member between connections. For parts of the model where higher solution resolution is required, such as stress prediction, the model should be refined with beam elements.

The geometry in this example is a typical K-frame construction used in applications such as offshore structures (see *Figure 1*). A push-over analysis is performed to determine the maximum horizontal load that the structure can support before collapse results from the development of plastic hinges or buckling failure. During a push-over test, many structural members are loaded in compression. Slender members loaded in compression often fail due to geometric buckling, cross-section collapse, and/or material yielding. The buckling strut response, which models such compressive behavior, is added in separate simulations to investigate the effect of the compressive failure of critical members in the structure. A dead load is applied to the top of the structure representing the weight supported by the K-frame. Push-over analyses are either load or displacement control tests.

Geometry and model

The structure consists of 19 members between structural connections. Each finite element models a member of the frame. Hence, 19 frame elements are used: 17 elements with PIPE cross-sections of varying properties and 2 elements (the top platform) with I cross-sections. The plastic response of the elements is calculated from the yield stress of the material, using the plastic default values provided by Abaqus. (The default values for the plastic response are based on experiments with slender steel members. For details on the default values, see *Frame Section Behavior*.) The default plastic response includes mild hardening for axial forces and strong hardening for bending moments. The default hardening responses for a typical element in the model are shown in *Figure 2* and *Figure 3*.

A dead load of 444.8 kN (100,000 lb) is applied to the top of the K-frame, representing the part of the structure above the K-frame. Subsequently, the top platform is loaded or displaced horizontally. The load level or applied displacement is chosen to be large enough so that the entire structure fails by the formation of plastic hinges and, consequently, loses load-carrying capacity.

Three different models are investigated. A limit load is expected, since the goal of the analysis is to determine when the structure loses overall stiffness. Large- and small-displacement analyses are performed for all three models for comparison. Large-displacement analyses using frame elements are valid for large overall rotations but small strains, since frame elements assume that the strains are small. In the first model all elements use elastic-plastic material response. In the second model buckling is checked for all elements with PIPE cross-sections.

The ISO equation is used as a criteria for buckling, and the default Marshall strut envelope is followed for the postbuckling behavior. The buckling strut envelope is calculated from the yield stress of the material and the default Marshall Strut theory. (For details on the default buckling strut envelope, see *Frame Section Behavior*.) All frame members that use the buckling strut response check the ISO criteria for the switching-to-strut algorithm. In the third model the member that switches to strut behavior in the second model (element 7) is replaced by a frame element with buckling strut response from the beginning of the analysis. To proceed beyond the unstable phase of the response, the Riks static solution procedure is used in the elastic-plastic problems. To decrease the number of solution iterations, the solution controls are used in the elastic-plastic problem with large displacement, with the value of the ratio of the largest solution correction to the largest incremental solution set to 1.0, since displacement increments are very small after plasticity occurs.

Results and discussion

The structure is loaded or displaced to the point at which all load-carrying capacity is lost. In the first model with elastic-plastic frame elements, the results for the linear and nonlinear geometries compare as expected. That is, the limit load for the large-displacement analysis is reached at a load of 1141 kN (256,000 lb) as compared to a higher load of 1290 kN (291,000 lb) in the small-displacement analysis. The plastic hinge pattern is the same in both cases.

The second model uses the switching algorithm. It shows that element 7 first violates the ISO equation (buckles) at a prescribed displacement equal to 1.32 cm (0.52 in), before any elements form plastic hinges. The critical compressive force in this element is –318 kN (–71,400 lb). Next, plasticity develops at several elements, and the structure reaches its limit capacity. The frame elements with the switching algorithm predict the structural behavior in the most accurate way, since the possibility of buckling is checked for all elements in the model, and highly compressed members switch automatically to postbuckling behavior (see the plastic and buckled frame elements in *Figure 4*). When the structure can no longer support horizontal loading, the patterns of plastic hinges for linear and nonlinear geometry are the same. The results differ more for loads close to the limit load.

To investigate the effect of buckling, the first and the third (element 7 defined with buckling strut response from the beginning) models are compared (*kframe_loadcntrl_nlgeom.inp* and *kframe_dispcntrl_buckle_nlgeom.inp*). Load versus horizontal deflection curves for the large-displacement analyses are shown in *Figure 5*. Similar to the model with switching algorithm, first element 7 buckles. As the other members deform and absorb the load no longer carried by the buckled member, the structure regains stiffness and plasticity develops in other members. When seven members develop plastic hinges, the structure can no longer support additional horizontal loading. The limit load in the third model is only about 28% of the limit load in the model without buckling. The load-displacement curves for the switching algorithm and for the example with element 7 using buckling strut response compare well and are not shown.

Input files

kframe_loadcntrl_nlgeom.inp	Elastic-plastic analysis with load control; large-displacement analysis.
kframe_loadcntrl.inp	Elastic-plastic analysis with load control; small-displacement analysis.
kframe_dispcntrl_switch_nlgeom.inp	Elastic-plastic frame element with the switching algorithm and displacement control; large-displacement analysis.
kframe_dispcntrl_switch.inp	Elastic-plastic frame element with the switching algorithm and displacement control; small-displacement analysis.
kframe_dispcntrl_buckle_nlgeom.inp	Elastic-plastic and buckling strut response with load control; large-displacement analysis.
kframe_dispcntrl_buckle.inp	Elastic-plastic and buckling strut response with displacement control; small-displacement analysis.

Figures

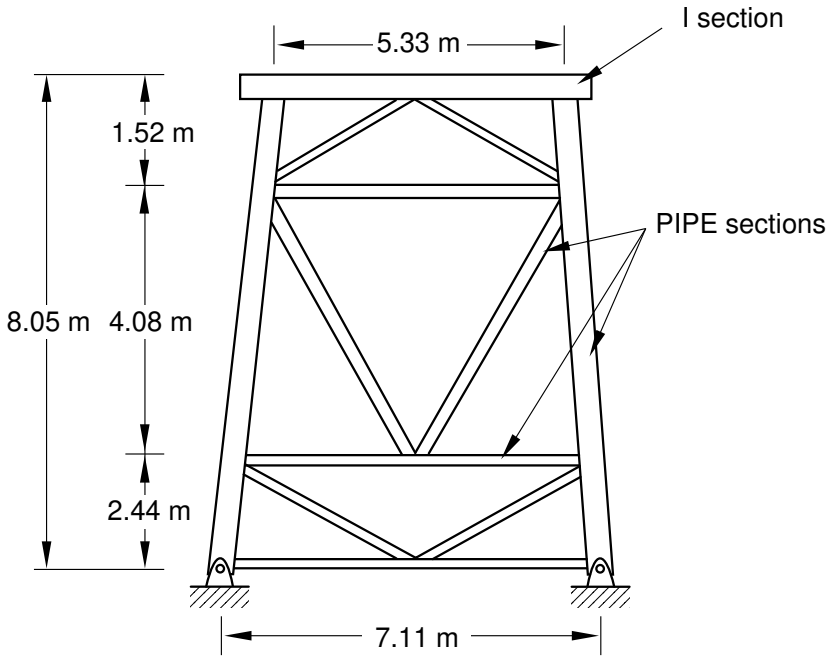


Figure 1: Two-dimensional K-frame structure.

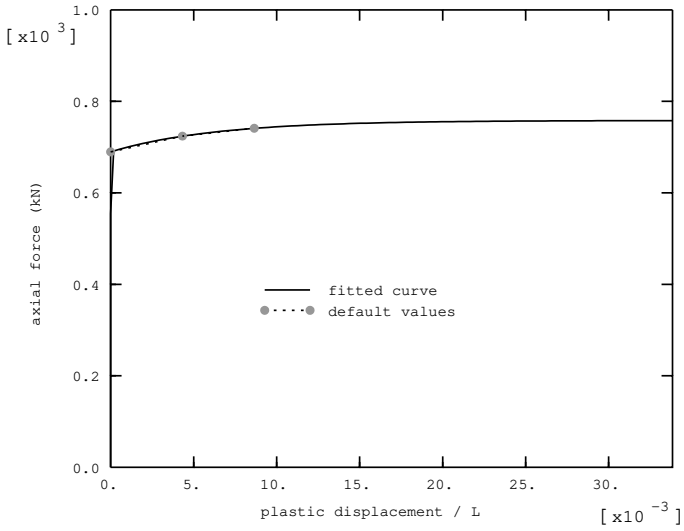


Figure 2: Default hardening response for axial force in a typical element with PIPE cross-section (element 7 in the model).

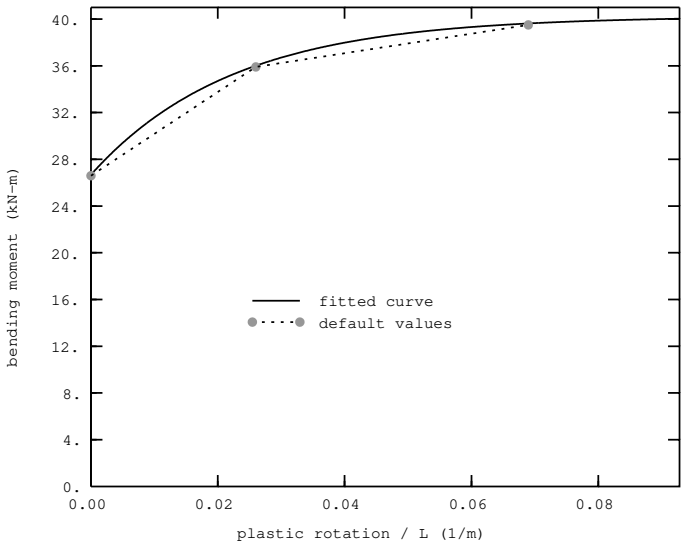


Figure 3: Default hardening response for bending moments in a typical element with PIPE cross-section (element 7 in the model).

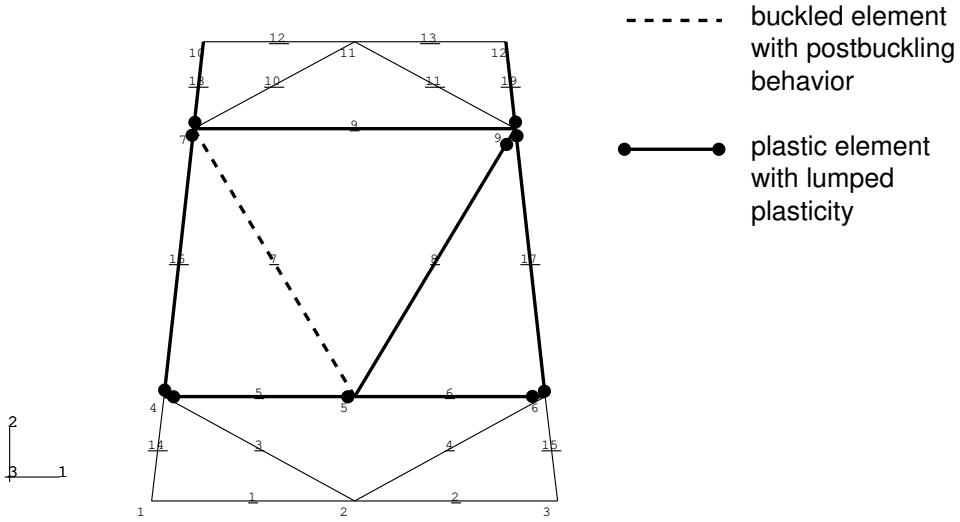


Figure 4: Results of analysis with switching algorithm: K-frame model with plastic and buckled elements.

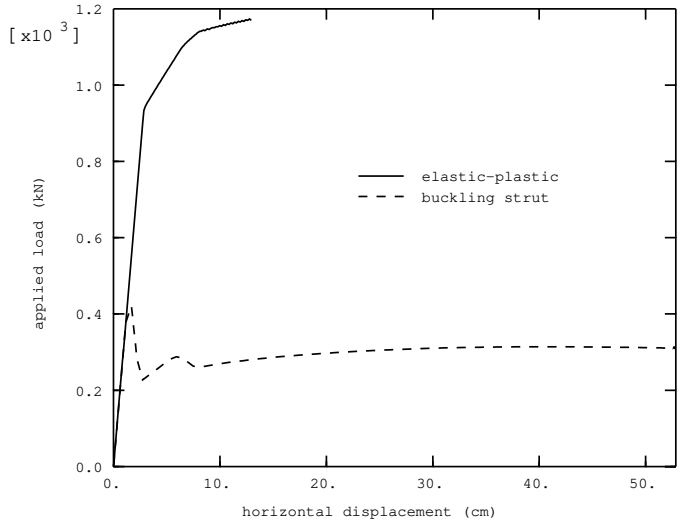


Figure 5: Applied force versus horizontal displacement of the load point for the elastic-plastic model and the model including buckling strut response.

Unstable static problem: reinforced plate under compressive loads

Products: Abaqus/Standard

Overview

This example demonstrates the use of automatic techniques to stabilize unstable static problems.

Geometrically nonlinear static problems can become unstable for a variety of reasons. Instability may occur in contact problems, either because of chattering or because contact intended to prevent rigid body motions is not established initially. Localized instabilities can also occur; they can be either geometrical, such as local buckling, or material, such as material softening.

This problem models a reinforced plate structure subjected to in-plane compressive loading that produces localized buckling. Structures are usually designed for service loads properly augmented by safety factors. However, it is quite often of interest to explore their behavior under extreme accident loads. This example looks into a submodel of a naval construction structure. It is a rectangular plate reinforced with beams in its two principal directions (*Figure 1*). The plate has symmetry boundary conditions along the longer edges and is pinned rigidly along the shorter sides. An in-plane load is applied to one of the pinned sides, compressing the plate. Gravity loads are also applied. The plate buckles under the load. The buckling is initially localized within each of the sections bounded by the reinforcements. At higher load levels the plate experiences global buckling in a row of sections closest to the applied load.

Standard analysis procedures typically provide the load at which the structure starts to buckle. The user may be interested in knowing the structure's additional load-carrying capacity. This information could translate, for instance, into knowing when the onset of global buckling takes place or how far into the structure damage propagates. In such situations more sophisticated analysis techniques are necessary. Arc length methods such as the Riks method available in Abaqus are global load-control methods that are suitable for global buckling and postbuckling analyses; they do not function well when buckling is localized. Alternatives are to analyze the problem dynamically or to introduce damping. In the dynamic case the strain energy released locally from buckling is transformed into kinetic energy; in the damping case this strain energy is dissipated. To solve a quasi-static problem dynamically is typically an expensive proposition. In this example the automatic stabilization capability in Abaqus, which applies volume proportional damping to the structure, is used.

Geometry and model

The model consists of a rectangular plate 10.8 m (425.0 in) long, 6.75 m (265.75 in) wide, and 5.0 mm (0.2 in) thick. This plate has several reinforcements in both the longitudinal and transverse directions (*Figure 1*). The plate represents part of a larger structure: the two longitudinal sides have symmetry boundary conditions, and the two transverse sides have pinned boundary conditions. In addition, springs at two major reinforcement intersections represent flexible connections to the rest of the structure. The mesh consists of S4 shell elements for both the plate and larger reinforcements and additional S3 shell and B31 beam elements for the remaining reinforcements. The entire structure is made of the same construction steel, with an initial flow stress of 235.0 MPa (34.0 ksi).

To provide stability to the numerical solution upon the anticipated buckling, automatic stabilization based on volume proportional damping is added to the model. Two forms of automatic stabilization are considered: one with a constant damping factor that is chosen by default (see *Automatic Stabilization of Static Problems with a Constant Damping Factor*), and one with an adaptive damping factor (see *Adaptive Automatic Stabilization Scheme*).

Results and discussion

The analysis consists of two steps. In the first step a gravity load perpendicular to the plane of the plate is applied. In the second step a longitudinal compressive load of 6.46×10^6 N (1.45×10^6 lbf) is applied to one of the pinned sides of the plate. All the nodes on that side are forced to move equally by means of multi-point constraints. The analysis is quasi-static, but buckling is expected.

Initially local out-of-plane buckling develops throughout the plate in an almost checkerboard pattern inside each one of the sections delimited by the reinforcements (*Figure 2*). Later, global buckling develops along a front of sections closer to the applied load (*Figure 3*). The evolution of the displacements produced by the applied load is very smooth (*Figure 4*) and does not reflect the early local instabilities in the structure. However, when the global instability develops, the curve becomes almost flat, indicating the complete loss of load-carrying capacity. An inspection of the model's energy content (*Figure 5* and *Figure 6*) reveals that while the load is increasing, the amount of dissipated energy is negligible. As soon as the load flattens out, the strain energy also flattens out (indicating a more or less constant load-carrying capacity), while the dissipated energy increases dramatically to absorb the work done by the applied loads.

Figure 7 shows the ratio of the dissipated energy to the total strain energy obtained using a constant damping factor versus using an adaptive damping factor.

Acknowledgements

SIMULIA would like to thank IRCN (France) for providing this example.

Input files

unstablestatic_plate.inp	Plate model.
unstablestatic_plate_stabil_adap.inp	Plate model with adaptive stabilization.
unstablestatic_plate_node.inp	Node definitions for the plate model.
unstablestatic_plate_elem.inp	Element definitions for the plate model.

Figures

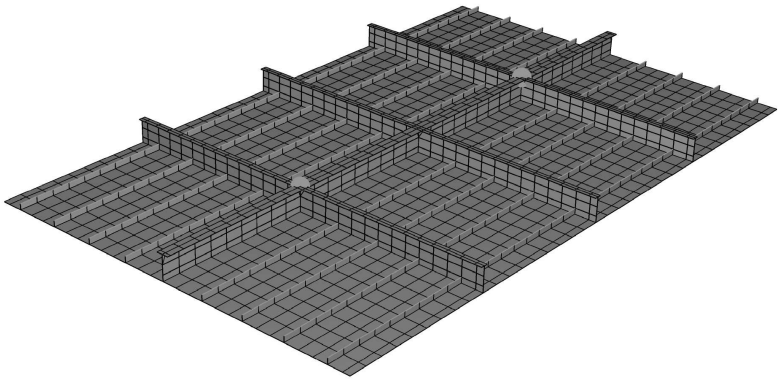


Figure 1: Reinforced plate initial mesh.

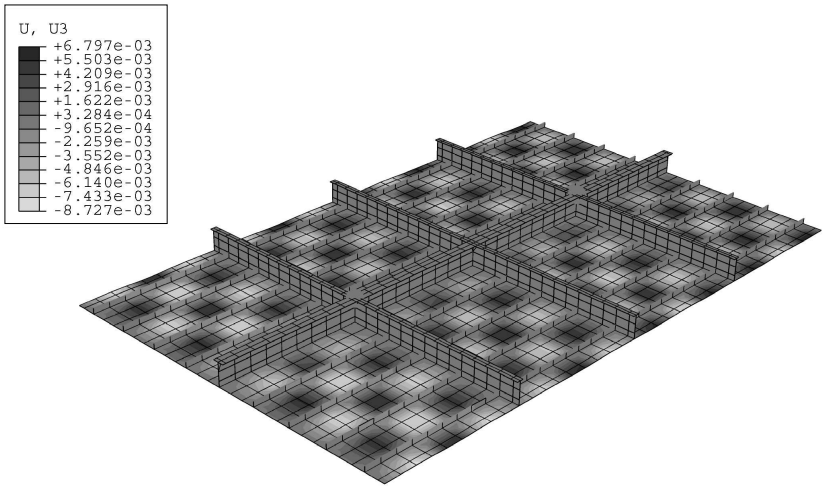


Figure 2: Plate localized buckling.

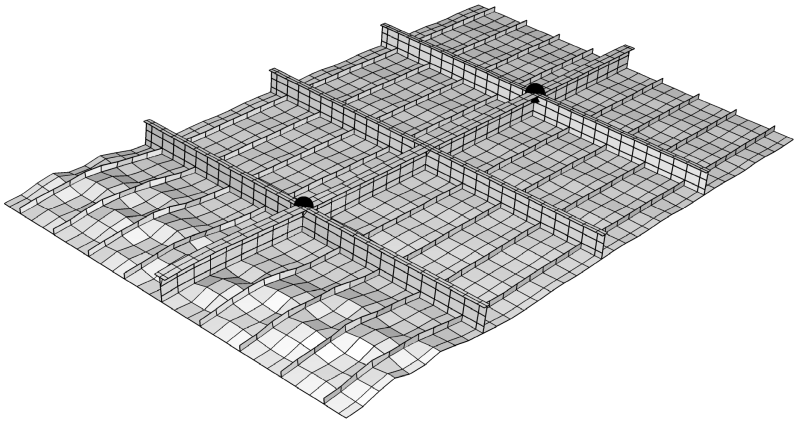


Figure 3: Plate global buckling.

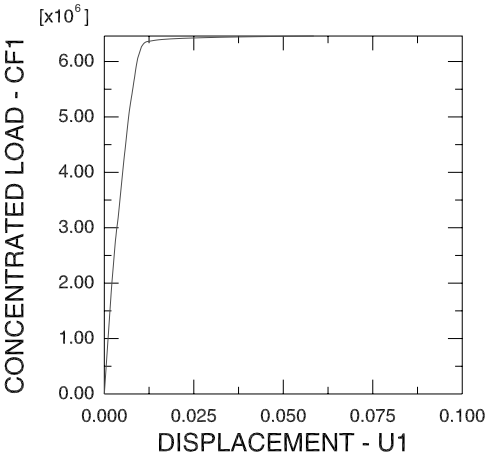


Figure 4: Plate load-displacement curve.

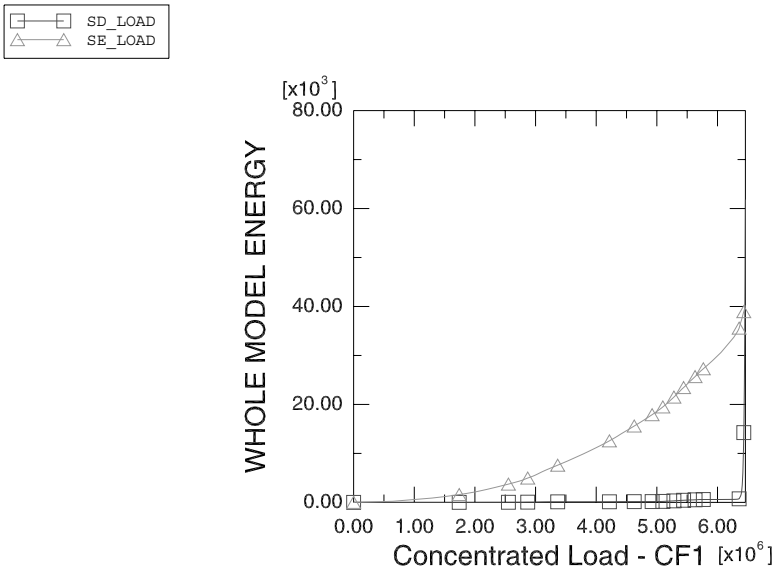


Figure 5: Dissipated and strain energies as functions of load.

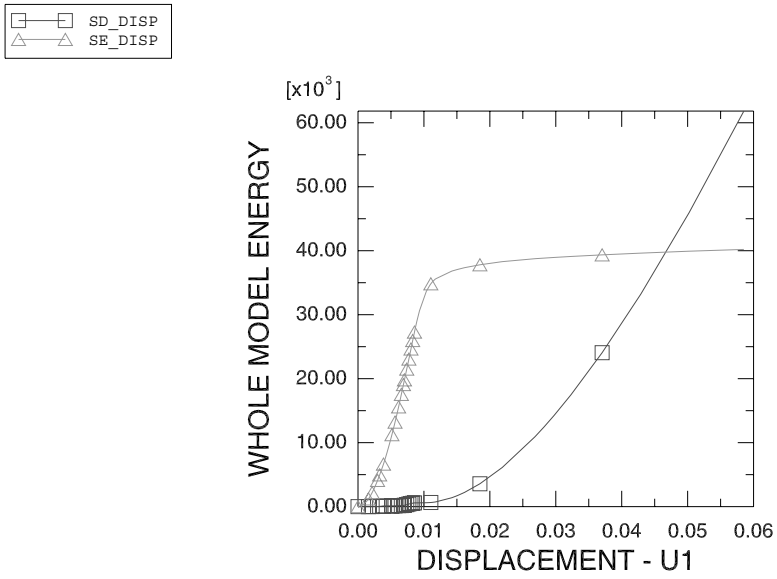


Figure 6: Dissipated and strain energies as functions of displacement.

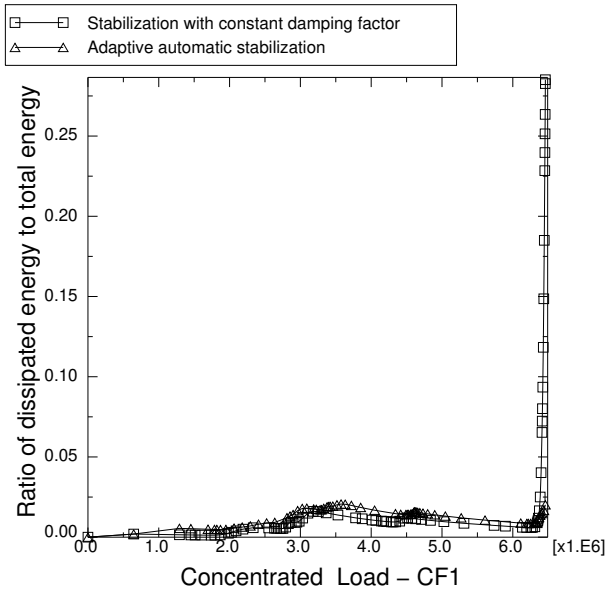


Figure 7: Ratio of dissipated energy to total strain energy using a constant damping factor and adaptive stabilization.

Buckling of an imperfection-sensitive cylindrical shell

Products: Abaqus/Standard

Overview

This example shows how to perform a postbuckling analysis using Abaqus for an imperfection-sensitive structure. A structure is imperfection sensitive if small changes in an imperfection change the buckling load significantly. Qualitatively, this behavior is characteristic of structures with closely spaced eigenvalues. For such structures the first eigenmode may not characterize the deformation that leads to the lowest buckling load. A cylindrical shell is chosen as an example of an imperfection-sensitive structure.

Geometry and model

The cylinder being analyzed is depicted in *Figure 1*. The cylinder is simply supported at its ends and is loaded by a uniform, compressive axial load. A uniform internal pressure is also applied to the cylinder. The material in the cylinder is assumed to be linear elastic. The thickness of the cylinder is 1/500 of its radius, so the structure can be considered to be a thin shell.

The finite element mesh uses the fully integrated S4 shell element. This element is based on a finite membrane strain formulation and is chosen to avoid hourglassing. A full-length model is used to account for both symmetric and antisymmetric buckling modes. A fine mesh, based on the results of a refinement study of the linear eigenvalue problem, is used. The convergence of the mesh density is based on the relative change of the eigenvalues as the mesh is refined. The mesh must have several elements along each spatial deformation wave; therefore, the level of mesh refinement depends on the modes with the highest wave number in the circumferential and axial directions.

Solution procedure

The solution strategy is based on introducing a geometric imperfection in the cylinder. In this study the imperfections are linear combinations of the eigenvectors of the linear buckling problem. If details of imperfections caused in a manufacturing process are known, it is normally more useful to use this information as the imperfection. However, in many instances only the maximum magnitude of an imperfection is known. In such cases assuming the imperfections are linear combinations of the eigenmodes is a reasonable way to estimate the imperfect geometry (Arbocz, 1987).

Determining the most critical imperfection shape that leads to the lowest collapse load of an axially compressed cylindrical shell is an open research issue. The procedure discussed in this example does not, therefore, claim to compute the lowest collapse load. Rather, this example discusses one approach that can be used to study the postbuckling response of an imperfection-sensitive structure.

The first stage in the simulation is a linear eigenvalue buckling analysis. To prevent rigid body motion, a single node is fixed in the axial direction. This constraint is in addition to the simply supported boundary conditions noted earlier and will not introduce an overconstraint into the problem since the axial load is equilibrated on opposing edges. The reaction force in the axial direction should be zero at this node.

The second stage involves introducing the imperfection into the structure using geometric imperfections. A single mode or a combination of modes is used to construct the imperfection. To compare the results obtained with different imperfections, the imperfection size must be fixed. The measure of the imperfection size used in this problem is the out-of-roundness of the cylinder, which is computed as the radial distance from the axis of the cylinder to the perturbed node minus the radius of the perfect structure. The scale factor associated with each eigenmode used to seed the imperfection is computed with a Fortran program. The program reads the results

file produced by the linear analysis and determines the scale factors so that the out-of-roundness of the cylinder is equal to a specified value. This value is taken as a fraction of the cylinder thickness.

The final stage of the analysis simulates the postbuckling response of the cylinder for a given imperfection. The primary objective of the simulation is to determine the static buckling load. The modified Riks method is used to obtain a solution since the problem under consideration is unstable. The Riks method can also be used to trace the unstable and stable solution branches of a buckled structure. However, with imperfection-sensitive structures the first buckling mode is usually catastrophic, so further continuation of the analysis is usually not undertaken. When using a static Riks step, the tolerance used for the force residual convergence criteria may need to be tightened to ensure that the solution algorithm does not retrace its original loading path once the limit point is reached. Simply restricting the maximum arc length allowed in an increment is normally not sufficient.

Parametric study

There are two factors that significantly alter the buckling behavior: the shape of the imperfection and the size of the imperfection. A convenient way to investigate the effects of these factors on the buckling response is to use the parametric study capabilities of Abaqus. A Python script file is used to perform the study. The script executes the linear analysis, runs the Fortran routine to create an input file with a specified imperfection size, and finally executes the postbuckling analysis.

Before executing the script, copy the Fortran routine *cylsh_maximp.f* to your work directory using the Abaqus**fetch** command,

```
abaqus fetch job=cylsh_maximp.f
```

and compile it using the Abaqus**make** command,

```
abaqus make job=cylsh_maximp.f
```

Parametrized template input data are used to generate variations of the parametric study. The script allows the analyst to vary the eigenmodes used to construct the imperfection, out-of-roundness measure, cylindrical shell geometry (radius, length, thickness), mesh density, material properties (Young's modulus and Poisson's ratio), etc. The results presented in the following section, however, are based on an analysis performed with a single set of parameters.

Results and discussion

The results for both the linear eigenvalue buckling and postbuckling analyses are discussed below.

Linear eigenvalue buckling

The Lanczos eigensolver is used to extract the linear buckling modes. This solver is chosen because of its superior accuracy and convergence rate relative to wavefront solvers for problems with closely spaced eigenvalues. *Table 1* lists the first 19 eigenvalues of the cylindrical shell. The eigenvalues are closely spaced with a maximum percentage difference of 1.3%.

The geometry, loading, and material properties of the cylindrical shell analyzed in this example are characterized by their axisymmetry. As a consequence of this axisymmetry the eigenmodes associated with the linear buckling problem will be either (1) axisymmetric modes associated with a single eigenvalue, including the possibility of eigenmodes that are axially symmetric but are twisted about the symmetry axis or (2) nonaxisymmetric modes associated with repeated eigenvalues (Wohlever, 1999). The nonaxisymmetric modes are characterized by sinusoidal variations (n-fold symmetry) about the circumference of the cylinder. For most practical engineering problems and as illustrated in *Table 1*, it is usually found that a majority of the buckling modes of the cylindrical shell are nonaxisymmetric.

The two orthogonal eigenmodes associated with each repeated eigenvalue span a two-dimensional space, and as a result any linear combination of these eigenmodes is also an eigenmode; i.e., there is no preferred direction.

Therefore, while the shapes of the orthogonal eigenmodes extracted by the eigensolver will always be the same and span the same two-dimensional space, the phase of the modes is not fixed and might vary from one analysis to another. The lack of preferred directions has consequences with regard to any imperfection study based upon a linear combination of nonaxisymmetric eigenmodes from two or more distinct eigenvalues. As the relative phases of eigenmodes change, the shape of the resulting imperfection and, therefore, the postbuckling response, also changes. To avoid this situation, postprocessing is performed after the linear buckling analysis on each of the nonaxisymmetric eigenmode pairs to fix the phase of the eigenmodes before the imperfection studies are performed. The basic procedure involves calculating a scaling factor for each of the eigenvectors corresponding to a repeated eigenvalue so that their linear combination generates a maximum displacement of 1.0 along the global *X*-axis. This procedure is completely arbitrary but ensures that the postbuckling response calculations are repeatable.

For the sake of consistency the maximum radial displacement associated with a unique eigenmode is also scaled to 1.0. These factors are further scaled to satisfy the out-of-roundness criterion mentioned earlier.

Postbuckling response

The modes used to seed the imperfection are taken from the first 19 eigenmodes obtained in the linear eigenvalue buckling analysis. Different combinations are considered: all modes, unique eigenmodes, and pairs of repeated eigenmodes. An imperfection size (i.e., out-of-roundness) of 0.5 times the shell thickness is used in all cases. The first limit-point on the load-displacement curve is recorded as the buckling load. The results indicate that the cylinder buckles at a much lower load than the value predicted by the linear analysis (i.e., the value predicted using only the lowest eigenmode of the system). An imperfection based on mode 1 (a unique eigenmode) results in a buckling load of about 90% of the predicted value. When the imperfection was seeded with a combination of all modes (1–19), a buckling load of 33% of the predicted value was obtained. *Table 2* lists the buckling loads predicted by Abaqus (as a fraction of linear eigenvalue buckling load) when different modes are used to seed the imperfection.

The smallest predicted buckling load in this study occurs when using modes 12 and 13 to seed the imperfection, yet the results obtained when the imperfection is seeded using all 19 modes indicate that a larger buckling load can be sustained. One possible explanation for this is that the solution strategy used in this study (discussed earlier) involves using a fixed value for the out-of-roundness of the cylinder as a measure of the imperfection size. Thus, when multiple modes are used to seed the imperfection, the overall effect of any given mode is less than it would be if only that mode were used to seed the imperfection. The large number of closely spaced eigenvalues and innumerable combinations of eigenmodes clearly demonstrates the difficulty of determining the collapse load of structures such as the cylindrical shell. In practice, designing imperfection-sensitive structures against catastrophic failure usually requires a combination of numerical and experimental results as well as practical building experience.

The final deformed configuration shown in *Figure 2* uses a displacement magnification factor of 5 and corresponds to using all the modes to seed the imperfection. Even though the cylinder appears to be very short, it can in fact be classified as a moderately long cylinder using the parameters presented in Chajes (1985). The cylinder exhibits thin wall wrinkling; the initial buckling shape can be characterized as dimples appearing on the side of the cylinder. The compression of the cylinder causes a radial expansion due to Poisson's effect; the radial constraint at the ends of the cylinder causes localized bending to occur at the ends. This would cause the shell to fold into an accordion shape. (Presumably this would be seen if self-contact was specified and the analysis was allowed to run further. This is not a trivial task, however, and modifications to the solution controls would probably be required. Such a simulation would be easier to perform with Abaqus/Explicit.) This deformed configuration is in accordance with the perturbed reference geometry, shown in *Figure 3*. To visualize the imperfect geometry, an imperfection size of 5.0 times the shell thickness (i.e., 10 times the value actually used in the analysis) was used to generate the perturbed mesh shown in this figure. The deformed configuration in the postbuckling analysis depends on the shape of the imperfection introduced into the structure. Seeding the structure with different combinations of modes and imperfection sizes produces different deformed configurations and buckling loads. As the results vary with the size and shape of the imperfection introduced into the structure, there is no solution to which the results from Abaqus can be compared.

The load-displacement curve for the case when the first 19 modes are used to seed the imperfection is shown in *Figure 4*. The figure shows the variation of the applied load (normalized with respect to the linear eigenvalue buckling load) versus the axial displacement of an end node. The peak load that the cylinder can sustain is clearly visible.

Input files

cylsh_buck.inp	Linear eigenvalue buckling problem.
cylsh_postbuck.inp	Postbuckling problem.
cylsh_maximp.f	Fortran program to compute the scaling factors for the imperfection size.
cylsh_script.psf	Python script to generate the parametrized input files.

References

- Arbocz, J., "Post-Buckling Behaviour of Structures: Numerical Techniques for More Complicated Structures," in *Lecture Notes in Physics*, Ed. H. Araki et al., Springer-Verlag, Berlin, 1987, pp. 84–142.
- Chajes, A., "Stability and Collapse Analysis of Axially Compressed Cylindrical Shells," in *Shell Structures: Stability and Strength*, Ed. R. Narayanan, Elsevier, New York, 1985, pp. 1–17.
- Wohlever, J. C., "Some Computational Aspects of a Group Theoretic Finite Element Approach to the Buckling and Postbuckling Analyses of Plates and Shells-of-Revolution," in *Computer Methods in Applied Mechanics and Engineering*, vol. 170, pp. 373–406, 1999.

Tables

Table 1: Eigenvalue estimates for the first 19 modes of the cylinder with and without internal pressure (p_i).

Mode number	Eigenvalue	
	$p_i = 0$	$p_i = 1$
1	11723	11723
2, 3	11724	11724
4, 5	11728	11728
6, 7	11734	11735
8, 9	11744	11746
10, 11	11757	11760
12, 13	11775	11779
14, 15	11798	11804
16, 17	11827	11835
18, 19	11864	11874

Table 2: Summary of predicted buckling loads for the cylinder with internal pressure ($p_i = 1$).

Mode used to seed the imperfection	Normalized buckling load
1	0.902

Mode used to seed the imperfection	Normalized buckling load
2, 3	0.647
4, 5	0.480
6, 7	0.355
8, 9	0.351
10, 11	0.340
12, 13	0.306
14, 15	0.323
16, 17	0.411
18, 19	0.422
All modes (1–19)	0.325

Figures

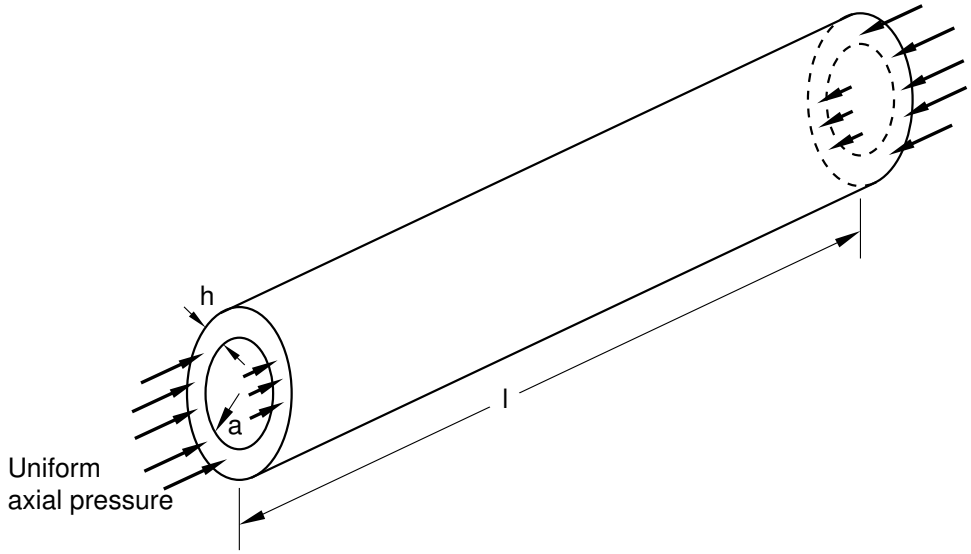


Figure 1: Cylindrical shell with uniform axial loading.

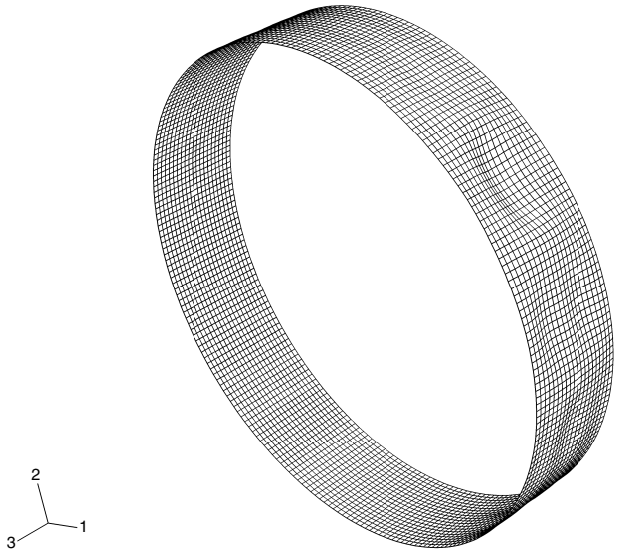


Figure 2: Final deformed configuration of the cylindrical shell (first 19 eigenmodes used to seed the imperfection; displacement magnification factor of 5.0; normalized end load = 0.29).



Figure 3: Perturbed geometry of the cylindrical shell (imperfection factor = 5 × thickness for illustration only; actual imperfection factor used = .5 × thickness).

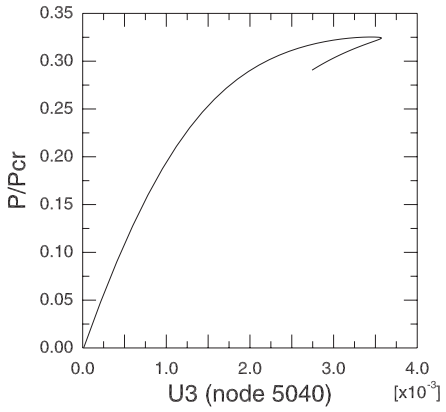


Figure 4: Normalized applied load versus axial displacement at node 5040 (first 19 modes used to seed the imperfection).

Forming analyses

In this section:

- *Upsetting of a cylindrical billet: quasi-static analysis with mesh-to-mesh solution mapping (Abaqus/Standard) and adaptive meshing (Abaqus/Explicit)*
- *Superplastic forming of a rectangular box*
- *Stretching of a thin sheet with a hemispherical punch*
- *Deep drawing of a cylindrical cup*
- *Extrusion of a cylindrical metal bar with frictional heat generation*
- *Rolling of thick plates*
- *Axisymmetric forming of a circular cup*
- *Cup/trough forming*
- *Forging with sinusoidal dies*
- *Forging with multiple complex dies*
- *Flat rolling: transient and steady-state*
- *Section rolling*
- *Ring rolling*
- *Axisymmetric extrusion: transient and steady-state*
- *Two-step forming simulation*
- *Upsetting of a cylindrical billet: coupled temperature-displacement and adiabatic analysis*
- *Unstable static problem: thermal forming of a metal sheet*
- *Inertia welding simulation using Abaqus/Standard and Abaqus/CAE*
- *Moldflow translation examples*

Upsetting of a cylindrical billet: quasi-static analysis with mesh-to-mesh solution mapping (Abaqus/Standard) and adaptive meshing (Abaqus/Explicit)

Products: Abaqus/Standard Abaqus/Explicit Abaqus/CAE

Overview

This example illustrates the use of the solution mapping capabilities of Abaqus/Standard and the adaptive meshing capabilities of Abaqus/Explicit in a metal forming application.
The analysis results are compared with the results of Taylor (1981). The same problem is also analyzed using the coupled temperature-displacement elements in *Upsetting of a cylindrical billet: coupled temperature-displacement and adiabatic analysis*. Coupled temperature-displacement elements are included in this example only for solution mapping verification purposes; no heat generation occurs in these elements for this example.

When the strains become large in a geometrically nonlinear analysis, the elements often become so severely distorted that they no longer provide a good discretization of the problem. When this occurs, it is necessary to map the solution onto a new mesh that is better designed to continue the analysis. In Abaqus/Standard the procedure is to monitor the distortion of the mesh—for example, by observing deformed configuration plots—and decide when the mesh needs to be mapped. When mesh distortion is so severe that a new mesh must be created, the new mesh can be generated using the mesh generation options in Abaqus/CAE. The output database is useful in this context since the current geometry of the model can be extracted from the data in the output database. Once a new mesh is defined, the analysis is continued by beginning a new problem using the solution from the old mesh at the point of mapping as initial conditions by specifying the step number and increment number at which the solution should be read from the previous analysis. Abaqus/Standard interpolates the solution from the old mesh onto the new mesh to begin the new problem. This technique provides considerable generality. For example, the new mesh might be more dense in regions of high-strain gradients and have fewer elements in regions that are moving rigidly—there is no restriction that the number of elements be the same or that element types agree between the old and new meshes. In a typical practical analysis of a manufacturing process, mesh-to-mesh solution mapping may have to be done several times because of the large shape changes associated with such a process.

Abaqus/Explicit has capabilities that allow automatic solution mapping using adaptive meshing. Therefore, the mapping process is easier since it is contained within the analysis and the user only has to decide how frequently remeshing should be done and what method to use to map the solution from the old mesh to the new mesh as the solution progresses. Abaqus/Explicit offers default choices for adaptive meshing that have been shown to work for a wide variety of problems. Finally, solution-dependent meshing is used to concentrate mesh refinement areas of evolving boundary curvature. This counteracts the tendency of the basic smoothing methods to reduce the mesh refinement near concave boundaries where solution accuracy is important.

Geometry and model

The geometry is the standard test case of Lippmann (1979) and is defined in *Upsetting of a cylindrical billet: coupled temperature-displacement and adiabatic analysis*. It is a circular billet, 30 mm long, with a radius of 10 mm, compressed between two flat, rigid dies that are defined to be perfectly rough.

The mesh used to begin the analysis is shown in *Figure 1*. The finite element model is axisymmetric and includes the top half of the billet only since the middle surface of the billet is a plane of symmetry. In both the Abaqus/Standard and Abaqus/Explicit simulations, element type CAX4R is used: this is a 4-node quadrilateral with a single integration point and “hourglass control” to control spurious mechanisms caused by the fully reduced integration. The element is chosen here because it is relatively inexpensive for problems involving nonlinear constitutive behavior since the material calculations are only done at one point in each element. In addition, in the Abaqus/Standard simulations element types CGAX4R, CGAX4T, CAX4P, and CAX4I are also

used to model the billet; in the Abaqus/Explicit simulations element type CAX6M is also used to model the billet.

The contact between the top and lateral exterior surfaces of the billet and the rigid die is modeled with a contact pair in Abaqus/Standard and with general contact in Abaqus/Explicit. The billet surface is specified as a surface definition in the model. The rigid die is modeled in a variety of different ways as described in *Table 1*. The mechanical interaction between the contact surfaces is assumed to be nonintermittent, rough frictional contact. Therefore, the contact property includes two additional specifications: rough friction to enforce a no-slip constraint between the two surfaces, and a no-separation contact pressure-overclosure relationship to ensure that separation does not occur once contact has been established. All the contact simulations in Abaqus/Standard use the node-to-surface formulation except one case where the surface-to-surface formulation is introduced.

Table 1 summarizes the different analysis cases that are studied. The column headings indicate whether the problem was analyzed using Abaqus/Standard and/or Abaqus/Explicit.

For Case 1 several different analyses are performed to compare the different section control options available in Abaqus/Explicit and to evaluate the effects of mesh refinement for the billet modeled with CAX4R elements. A coarse mesh (analysis COARSE_SS) and a fine mesh (analysis FINE_SS) are analyzed with the pure stiffness form of hourglass control. A coarse mesh (analysis COARSE_CS) is analyzed with the combined hourglass control. A coarse mesh (analysis COARSE_ENHS) and a fine mesh (analysis FINE_ENHS) are analyzed with the hourglass control based on the enhanced strain method. The default section controls, using the integral viscoelastic form of hourglass control, are tested on a coarse mesh (analysis COARSE) and a fine mesh (analysis FINE). Since this is a quasi-static analysis, the viscous hourglass control option should not be used. All other cases use the default section controls.

The Abaqus/Standard analyses for Case 1 compare the two hourglass control options and evaluate the effect of mesh refinement for the billet modeled with CAX4R elements. A coarse mesh (analysis COARSE_S) and a fine mesh (analysis FINE_S) are analyzed with the pure stiffness form of hourglass control. A coarse mesh (analysis COARSE_EH) and a fine mesh (analysis FINE_EH) are analyzed with hourglass control based on the enhanced strain method. A coarse mesh (analysis COARSE_EHG) with CGAX4R elements is also analyzed with hourglass control based on the enhanced strain method for comparison purposes.

No mesh convergence studies have been done, but the agreement with the results given in Lippmann (1979) suggests that the meshes used here are good enough to provide reasonable predictions of the overall force on the dies.

Material

The material model assumed for the billet is that given in Lippmann (1979). Young's modulus is 200 GPa, Poisson's ratio is 0.3, and the density is 7833 kg/m^3. A rate-independent von Mises elastic-plastic material model is used, with a yield stress of 700 MPa and a hardening slope of 0.3 GPa.

Boundary conditions and loading

The kinematic boundary conditions are symmetry on the axis (nodes at $r = 0$, in node set `AXIS`, have $u_r = 0$ prescribed) and symmetry about $z = 0$ (all nodes at $z = 0$, in node set `MIDDLE`, have $u_z = 0$ prescribed). The node on the top surface of the billet that lies on the symmetry axis is not part of the node set `AXIS` to avoid overconstraint: the radial motion of this node is already constrained by a no slip frictional constraint (see *Common Difficulties Associated with Contact Modeling in Abaqus/Standard* and *Common Difficulties Associated with Contact Modeling Using Contact Pairs in Abaqus/Explicit*).

In Abaqus/Standard the rigid die is displaced by −9 mm in the axial direction using a displacement boundary condition. In Abaqus/Explicit the u_z-displacement of the rigid die is prescribed using a velocity boundary condition whose value is ramped up to a velocity of 20 m/s and then held constant until the die has moved a total of 9 mm. The total simulation time of the Abaqus/Explicit analysis is 0.55 millisec, and the loading rate is slow

enough to be considered quasi-static. In both Abaqus/Standard and Abaqus/Explicit the radial and rotational degrees of freedom of the rigid die are constrained.

For all cases the analyses are done in two steps so that the first step can be stopped at a die displacement corresponding to 44% upsetting; the second step carries the analysis to 60% upsetting. In the Abaqus/Standard simulations the solution mapping analysis restarts from the end of the first step with a new mesh and proceeds until 60% upsetting is achieved.

Mesh-to-mesh solution mapping in Abaqus/Standard

The interpolation technique used in solution mapping is a two-step process. First, values of all solution variables are obtained at the nodes of the old mesh by extrapolating the values from the integration points to the nodes of each element and averaging those values over all elements abutting each node. The second step is to locate each integration point in the new mesh with respect to the old mesh (this assumes all integration points in the new mesh lie within the bounds of the old mesh: warning messages are issued if this is not so, and new model solution variables at the integration point are set to zero). The variables are then interpolated from the nodes of the element in the old mesh to the location in the new mesh. All solution variables are interpolated automatically in this way so that the solution can proceed on the new mesh. Whenever a model is mapped, it can be expected that there will be some discontinuity in the solution because of the change in the mesh. If the discontinuity is significant, it is an indication that the meshes are too coarse or that the mapping should have been done at an earlier stage before too much distortion occurred.

Extracting two-dimensional profiles and remeshing using Abaqus/CAE

The model is built and meshed using Abaqus/CAE. The solution mapping for the Abaqus/Standard analysis is done by extracting the two-dimensional profile of the deformed billet from the output; the user must enter commands into the command line interface at the bottom of the Abaqus/CAE main window. To extract the deformed geometry from the output database as an orphan mesh part, use the command PartFromOdb, which takes the following arguments:

name

The name of the orphan mesh part to be created.

odb

The output database object returned from the command `openOdb`.

instance

The name of the part instance in the initial model in capital letters.

shape

Determines whether to import the part in its UNDEFORMED or DEFORMED shape.

The command PartFromOdb returns a Part object that is passed to the command Part2DGeomFrom2DMesh. This command creates a geometric Part object from the orphan mesh imported earlier. It takes the following arguments:

name

The name of the part to be created.

part

The part object returned from the command PartFromOdb.

featureAngle

A float specifying the angle (in degrees) between line segments that triggers a break in the geometry.

Once the profile of the deformed part has been created, the user can switch to the Mesh module, remesh the part, and write out the new node and element definitions to be used in the mapping analysis. The Python script file *billet_rezone.py* is included to demonstrate the process described above.

Adaptive meshing in Abaqus/Explicit

Adaptive meshing consists of two fundamental tasks: creating a new mesh, and remapping the solution variables from the old mesh to the new mesh with a process called advection. A new mesh is created at a specified frequency for each adaptive mesh domain. The mesh is found by sweeping iteratively over the adaptive mesh domain and moving nodes to smooth the mesh. The process of mapping solution variables from an old mesh to a new mesh is referred to as an advection sweep. At least one advection sweep is performed in every adaptive mesh increment. The methods used for advecting solution variables to the new mesh are consistent; monotonic; (by default) accurate to the second order; and conserve mass, momentum, and energy. This example problem uses the default settings for adaptive mesh domains.

Results and discussion

The following discussion focuses primarily on the results for Case 1, where the billet is modeled with CAX4R elements, the rigid die is modeled using an analytical rigid surface, and the pure stiffness hourglass control is used in Abaqus/Explicit. The deformed meshes at 44% billet upsetting (73.3% of the total die displacement) are shown in *Figure 2*, *Figure 3*, and *Figure 4*. The folding of the top outside surface of the billet onto the die is clearly visible. In Abaqus/Standard (*Figure 2*) severe straining and element distortion can be seen through the center of the specimen. At this point the Abaqus/Standard mesh is mapped. The new mesh for the mapped model is shown in *Figure 3*. *Figure 4* clearly indicates the benefits of adaptive meshing as the mesh used in Abaqus/Explicit has very little distortion.

The final configurations at 60% billet upsetting are shown in *Figure 5* and *Figure 6*. Both the Abaqus/Standard and Abaqus/Explicit results compare well, and the meshes appear only slightly distorted. Similarly, the equivalent plastic strain magnitudes compare well (*Figure 7* and *Figure 8*).

Figure 9 is a plot of upsetting force versus vertical displacement at the rigid surface reference node. The results of both the Abaqus/Standard and the Abaqus/Explicit analyses show excellent agreement with the rate-independent results obtained by Taylor (1981). Also worth noting is that the mapping in Abaqus/Standard does not appear to have a significant effect on the total upsetting force.

Figure 10 is a plot of upsetting force versus vertical displacement at the rigid surface reference node with the section control options identified in *Table 2*. The curves obtained using CAX4R and CAX6M elements are very close and agree well with the rate-independent results obtained by Taylor (1981). The results from the COARSE_SS analysis are virtually the same as the results from the FINE analysis but at a much reduced cost; therefore, such analysis options are recommended for this problem. The results for all the other cases (which use the default section controls but different rigid surface models) are the same as the results for Case 1 using the default section controls.

Input files

Abaqus/Standard input files

billet_case1_std_coarse.inp	Original COARSE CAX4R mesh using STIFFNESS hourglass control.
billet_coarse_nodes.inp	Node definitions for original COARSE mesh.
billet_coarse_elem.inp	Element definitions for original COARSE mesh.
billet_case1_std_coarse_rez.inp	Mapped COARSE CAX4R mesh.
billet_coarse_nodes_rez.inp	Node definitions for mapped COARSE mesh.
billet_coarse_elem_rez.inp	Element definitions for mapped COARSE mesh.
billet_case1_std_coarse_eh.inp	Original COARSE CAX4R mesh using ENHANCED hourglass control.
billet_case1_std_fine.inp	Original FINECAX4R mesh using STIFFNESS hourglass control.
billet_case1_std_fine_rez.inp	Mapped FINECAX4R mesh.
billet_case1_std_fine_eh.inp	Original FINECAX4R mesh using ENHANCED hourglass control.
billet_case1_std_coarse_cax4i.inp	Original COARSECAX4I mesh.
billet_case1_std_coarse_cax4i_surf.inp	Original COARSECAX4I mesh using surface-to-surface contact formulation.
billet_case1_std_coarse_cax4i_rez.inp	Mapped COARSECAX4I mesh.
billet_case1_std_coarse_cgax4r.inp	Original COARSECGAX4R mesh.
billet_case1_std_coarse_cgax_eh.inp	Original COARSECGAX4R mesh using ENHANCED hourglass control.
billet_case1_std_coarse_cgax4r_rez.inp	Mapped COARSECGAX4R mesh.
billet_case1_std_coarse_cgax4t.inp	Original COARSECGAX4T mesh.
billet_case1_std_coarse_cgax4t_rez.inp	Mapped COARSECGAX4T mesh.
billet_case1_std_coarse_cax4p.inp	Original COARSECAX4P mesh.
billet_case1_std_coarse_cax4p_rez.inp	Mapped COARSECAX4P mesh.
billet_rezone.py	Python script showing an example of the command usage to extract the geometric profile of the deformed mesh from an output database.
billet_case2_std.inp	Original COARSE CAX4R mesh.
billet_case2_std_rez.inp	Mapped COARSE CAX4R mesh.
billet_case3_std.inp	Original COARSE CAX4R mesh.
billet_case3_std_rez.inp	Mapped COARSE CAX4R mesh.
billet_case6_std.inp	Original COARSE CAX4R mesh.
billet_case6_std_rez.inp	Mapped COARSE CAX4R mesh.

Abaqus/Explicit input files

billet_case1_xpl_coarse.inp	COARSE CAX4R mesh using RELAX STIFFNESS hourglass control.
billet_case1_xpl_coarse_ss.inp	COARSE CAX4R mesh using STIFFNESS hourglass control.

billet_case1_xpl_coarse_cs.inp	COARSE CAX4R mesh using COMBINED hourglass control.
billet_case1_xpl_coarse_enhs.inp	COARSE CAX4R mesh using ENHANCED hourglass control.
billet_case1_xpl_fine.inp	FINE CAX4R mesh using RELAX STIFFNESS hourglass control.
billet_case1_xpl_fine_ss.inp	FINE CAX4R mesh using STIFFNESS hourglass control.
billet_case1_xpl_fine_cs.inp	FINE CAX4R mesh using COMBINED hourglass control.
billet_case1_xpl_fine_enhs.inp	FINE CAX4R mesh using ENHANCED hourglass control.
billet_case1_xpl_coarse_cax6m.inp	COARSE CAX6M mesh using RELAX STIFFNESS hourglass control.
billet_case1_xpl_fine_cax6m.inp	FINE CAX6M mesh using RELAX STIFFNESS hourglass control.
billet_case2_xpl.inp	COARSE CAX4R mesh using RELAX STIFFNESS hourglass control.
billet_case3_xpl.inp	COARSE CAX4R mesh using RELAX STIFFNESS hourglass control.
billet_case4_xpl.inp	COARSE CAX4R mesh using RELAX STIFFNESS hourglass control.
billet_case5_xpl.inp	COARSE CAX4R mesh using RELAX STIFFNESS hourglass control.
billet_case6_xpl.inp	COARSE CAX4R mesh using RELAX STIFFNESS hourglass control.
billet_case7_xpl.inp	COARSE CAX4R mesh using RELAX STIFFNESS hourglass control.

References

- Lippmann, H., Metal Forming Plasticity, Springer-Verlag, Berlin , 1979.
- Taylor, L. M., “A Finite Element Analysis for Large Deformation Metal Forming Problems Involving Contact and Friction,” *Ph.D. Thesis, U. of Texas at Austin*, 1981.

Tables

Table 1: Cases describing the modeling of the rigid die.

Case	Description	STD	XPL
1	The die is modeled as an analytical rigid surface using a planar analytical surface and a rigid body constraint. The rigid surface is associated with a rigid body by its specified reference node.	Yes	Yes
2	Axisymmetric rigid elements of type RAX2 are used to model the rigid die.	Yes	Yes
3	The die is modeled with RAX2 elements, as in Case 2. However, the die is assigned a mass by specifying point masses at the nodes of the RAX2 elements.	Yes	Yes
4	The rigid die is modeled with RAX2 elements, as in Case 2. The rigid elements are assigned a thickness and density values such that the mass of the die is the same as in Case 3.	No	Yes
5	The die is modeled with RAX2 elements, as in Case 2. In this case the thickness of the rigid elements is interpolated from the thickness specified at the nodes. The same thickness value is prescribed as in Case 4.	No	Yes
6	Axisymmetric shell elements of type SAX1 are used to model the die, and they are included in the rigid body definition.	Yes	Yes

Case	Description	STD	XPL
7	The die is modeled with axisymmetric shell elements of type SAX1 and with axisymmetric rigid elements of type RAX2. The deformable elements are included in the rigid body definition. Both element types have the same thickness and density as in Case 4.	No	Yes

Table 2: Analysis options for Case 1 using CAX4R elements.

Analysis Label	Mesh Type	Hourglass Control	Analysis Type
COARSE_SS	coarse	stiffness	XPL
FINE_SS	fine	stiffness	XPL
COARSE_CS	coarse	combined	XPL
COARSE	coarse	relax stiffness	XPL
FINE	fine	relax stiffness	XPL
COARSE_ENHS	coarse	enhanced	XPL
FINE_ENHS	fine	enhanced	XPL
COARSE_S	coarse	stiffness	STD
FINE_S	fine	stiffness	STD
COARSE_EH	coarse	enhanced	STD
FINE_EH	fine	enhanced	STD
COARSE_EHG	coarse	enhanced	STD

Figures

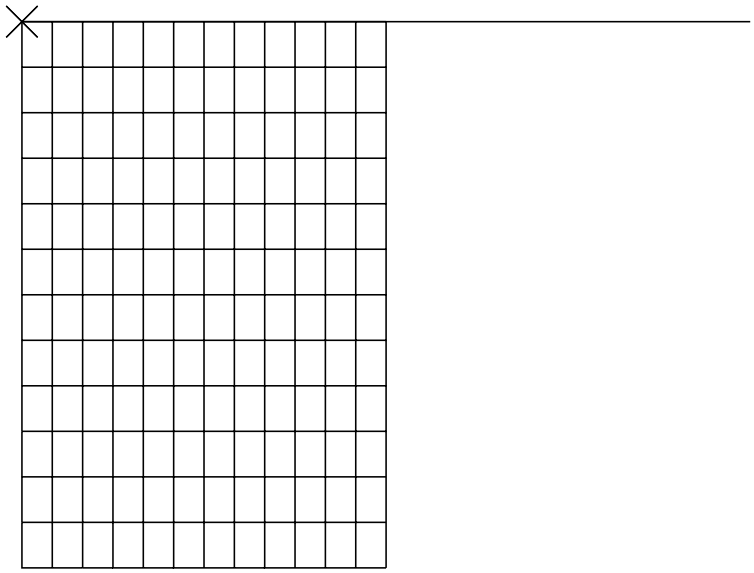


Figure 1: Axisymmetric upsetting example: initial mesh.

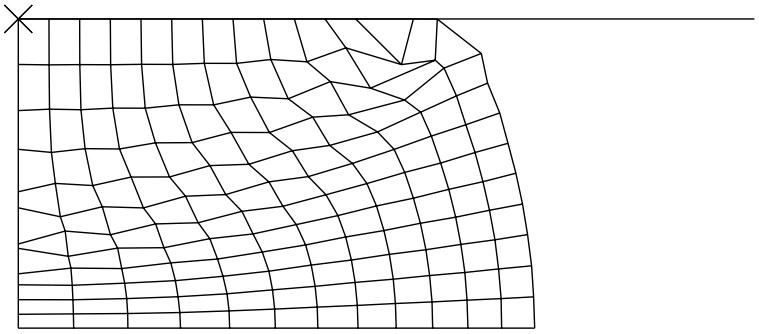


Figure 2: Abaqus/Standard: Deformed configuration at 44% upset (original mesh).

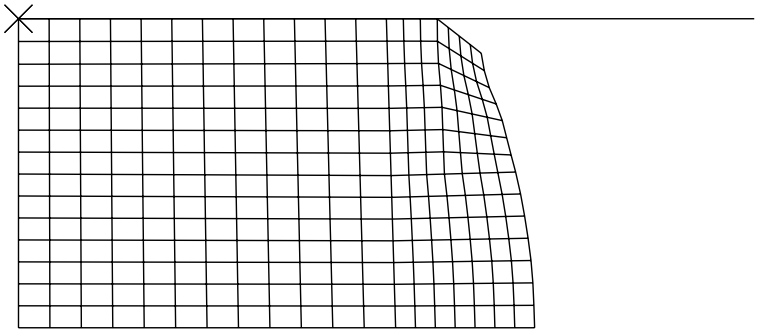


Figure 3: Abaqus/Standard: New mesh at 44% upset.

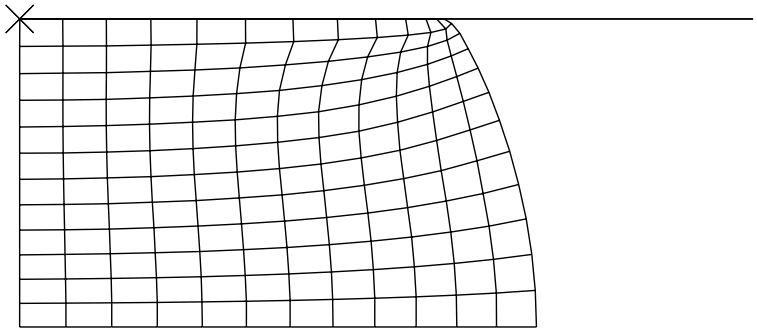


Figure 4: Abaqus/Explicit: Deformed configuration at 44% upset (CAX4R elements).

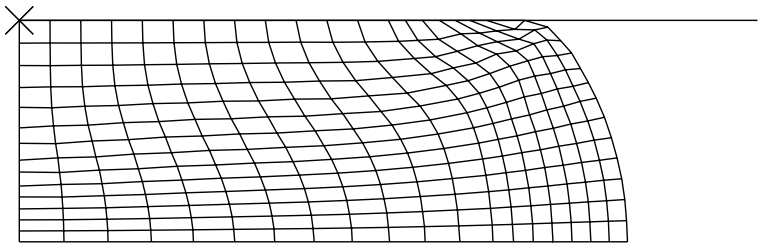


Figure 5: Abaqus/Standard: New mesh at 60% upset.

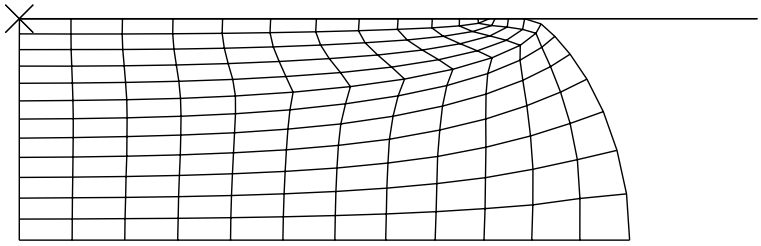


Figure 6: Abaqus/Explicit: Deformed mesh at 60% upset (CAX4R elements).

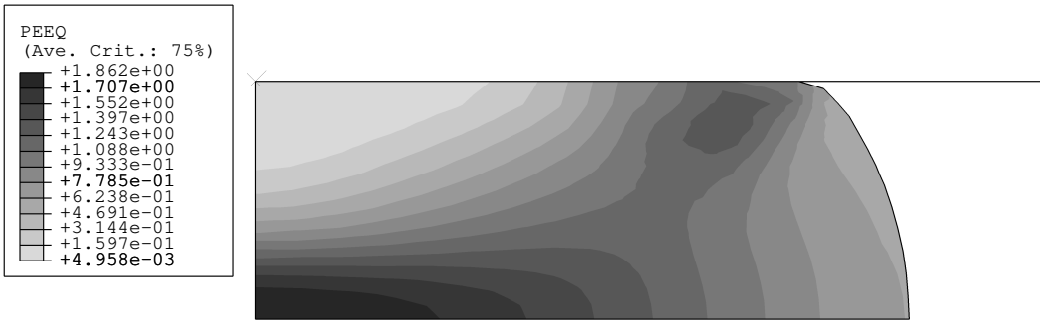


Figure 7: Abaqus/Standard: Plastic strain of new mesh at 60% upset.

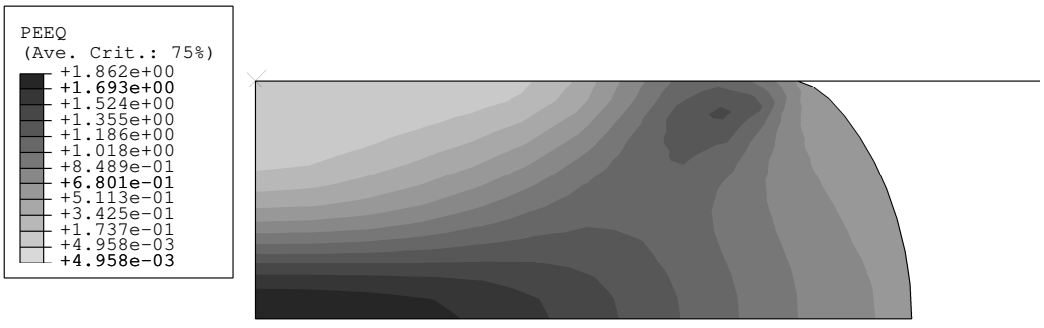


Figure 8: Abaqus/Explicit: Plastic strain at 60% upset (CAX4R elements).

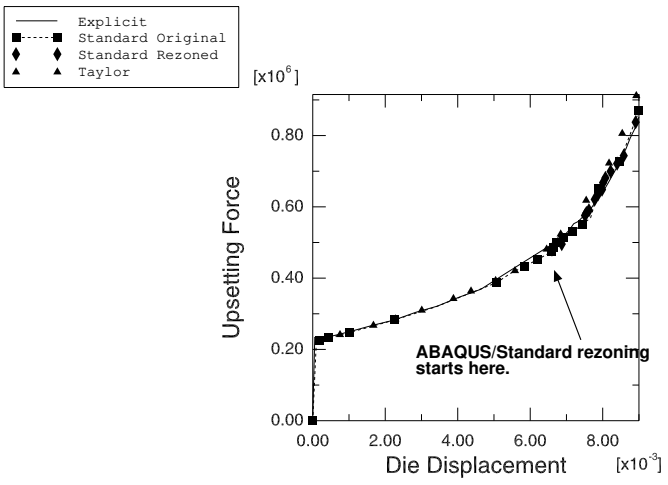


Figure 9: Force-deflection response for cylinder upsetting.

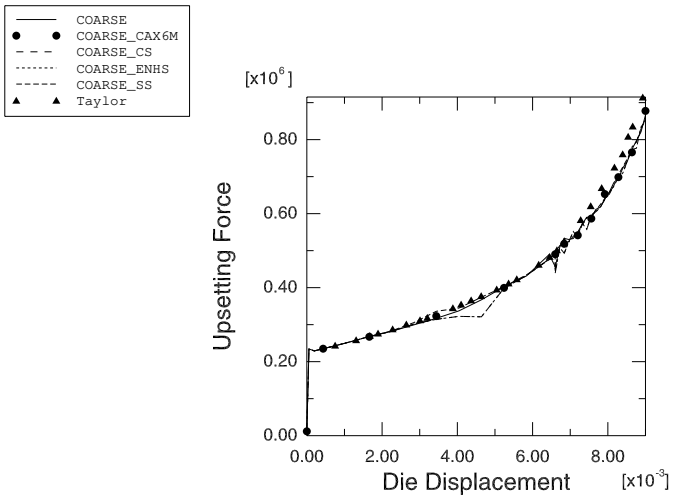


Figure 10: Force-deflection response for cylinder upsetting. Comparison of Abaqus/Explicit hourglass controls.

Superplastic forming of a rectangular box

Products: Abaqus/Standard

Overview

In this example we consider the superplastic forming of a rectangular box. The example illustrates the use of rigid elements to create a smooth three-dimensional rigid surface.

Superplastic metals exhibit high ductility and very low resistance to deformation and are, thus, suitable for forming processes that require very large deformations. Superplastic forming has a number of advantages over conventional forming methods. Forming is usually accomplished in one step rather than several, and intermediate annealing steps are usually unnecessary. This process allows the production of relatively complex, deep-shaped parts with quite uniform thickness. Moreover, tooling costs are lower since only a single die is usually required. Drawbacks associated with this method include the need for tight control of temperature and deformation rate. Very long forming times make this method impractical for high volume production of parts.

A superplastic forming process usually consists of clamping a sheet against a die whose surface forms a cavity of the shape required. Gas pressure is then applied to the opposite surface of the sheet, forcing it to acquire the die shape.

Rigid surface

A rigid faceted surface can be created from an arbitrary mesh of three-dimensional rigid elements (either triangular R3D3 or quadrilateral R3D4 elements). See *Analytical Rigid Surface Definition* for a discussion of smoothing of main surfaces. Abaqus automatically smoothes any discontinuous surface normal transitions between the surface facets.

Solution-dependent amplitude

One of the main difficulties in superplastically forming a part is the control of the processing parameters. The temperature and the strain rates that the material experiences must remain within a certain range for superplasticity to be maintained. The former is relatively easy to achieve. The latter is more difficult because of the unknown distribution of strain rates in the part. The manufacturing process must be designed to be as rapid as possible without exceeding a maximum allowable strain rate at any material point. For this purpose Abaqus has a feature that allows the loading (usually the gas pressure) to be controlled by means of a solution-dependent amplitude and a target maximum creep strain rate. In the loading options the user specifies a reference value. The amplitude definition requires an initial, a minimum, and a maximum load multiplier. During a quasi-static procedure Abaqus will then monitor the maximum creep strain rate and compare it with the target value. The load amplitude is adjusted based on this comparison. This controlling algorithm is simple and relatively crude. The purpose is not to follow the target values exactly but to obtain a practical loading schedule.

Geometry and model

The example treated here corresponds to superplastic forming of a rectangular box whose final dimensions are 1524 mm (60 in) long by 1016 mm (40 in) wide by 508 mm (20 in) deep with a 50.8 mm (2 in) flange around it. All fillet radii are 101.6 mm (4 in). The box is formed by means of a uniform fluid pressure.

A quarter of the blank is modeled using 704 membrane elements of type M3D4. These are fully integrated bilinear membrane elements. The initial dimensions of the blank are 1625.6 mm (64 in) by 1117.6 mm (44 in), and the thickness is 3.175 mm (0.125 in). The blank is clamped at all its edges. The flat initial configuration of the membrane model is entirely singular in the normal direction unless it is stressed in biaxial tension. This

difficulty is prevented by applying a small biaxial initial stress of 6.89 kPa (1 lb/in^2) by means of the initial stress conditions.

The female die is modeled as a rigid body and is meshed with rigid R3D3 elements. The rigid surface can be defined by grouping together those faces of the 231 R3D3 elements used to model the die that face the contact direction. See *Figure 1* for an illustration of the rigid element mesh.

To avoid having points "fall off" the rigid surface during the course of the analysis, more than a quarter of the die has been modeled, as shown in *Figure 2*. It is always a good idea to extend the rigid surface far enough so that contacting nodes will not slide off the main surface.

By default, Abaqus generates a unique normal to the rigid surface at each node point, based on the average of the normals to the elements sharing each node. There are times, however, when the normal to the surface should be specified directly. This is discussed in *Node Definition*. In this example the flange around the box must be flat to ensure compatibility between the originally flat blank and the die. Therefore, an outer normal (0, 1, 0) has been specified at the 10 nodes that make up the inner edge of the flange. This is done by entering the direction cosines after the node coordinates. The labels of these 10 vertices are 9043, 9046, 9049, 9052, 9089, 9090, 9091, 9121, 9124, and 9127; and their definitions can be found in *superplasticbox_node.inp*.

Material

The material in the blank is assumed to be elastic-viscoplastic, and the properties roughly represent the 2004 (Al-6Cu-0.4Zr)-based commercial superplastic aluminum alloy Supral 100 at 470°C. It has a Young's modulus of 71 GPa (10.3 × 10^6 lb/in^2) and a Poisson's ratio of 0.34. The flow stress is assumed to depend on the plastic strain rate according to

$$\sigma^f = A\left(\dot{\varepsilon}^{pl}\right)^{1/2},$$

where A is 179.2 MPa (26. × 10^3 lb/in^2) and the time is in seconds.

Loading and controls

We perform two analyses to compare constant pressure loading and a pressure schedule automatically adjusted to achieve a maximum strain rate of 0.02/sec. In the constant load case the prestressed blank is subjected to a rapidly applied external pressure of 68.8 kPa (10 lb/in^2), which is then held constant for 3000 sec until the box has been formed. In the second case the prestressed blank is subjected to a rapidly applied external pressure of 1.38 kPa (0.2 lb/in^2). The pressure schedule is then chosen by Abaqus.

The initial application of the pressures is assumed to occur so quickly that it involves purely elastic response. This is achieved by using the static procedure. The creep response is developed in a second step using a quasi-static procedure.

During the quasi-static step an accuracy tolerance controls the time increment and, hence, the accuracy of the transient creep solution. Abaqus compares the equivalent creep strain rate at the beginning and the end of an increment. The difference should be less than this tolerance divided by the time increment. Otherwise, the increment is reattempted with a smaller time increment. The usual guideline for setting this accuracy tolerance is to decide on an acceptable error in stress and convert it to an error in strain by dividing by the elastic modulus. For this problem we assume that moderate accuracy is required and choose this tolerance as 0.5%. In general, larger tolerance values allow Abaqus to use larger time increments, resulting in a less accurate and less expensive analysis.

In the automatic scheduling analysis the pressure refers to an amplitude that allows for a maximum pressure of 1.38 MPa (200 lb/in^2) and a minimum pressure of 0.138 kPa (0.02 lb/in^2). The target creep strain rate is a constant entered using creep strain rate control. The node-to-surface (default) and surface-to-surface contact formulations are considered for this case where the creep strain rate is used to control the pressure amplitude. In the node-to-surface contact formulation the thickness of the blank is ignored and the blank is positioned such that

its midsurface is used in the contact calculations as an approximation. However, the surface-to-surface contact formulation explicitly accounts for the blank thickness, as in reality.

Results and discussion

Figure 3 through *Figure 5* show a sequence of deformed configurations during the automatically controlled forming process. The stages of deformation are very similar in the constant load process. However, the time necessary to obtain the deformation is much shorter with automatic loading—the maximum allowable pressure is reached after 83.3 seconds. The initial stages of the deformation correspond to inflation of the blank because there is no contact except at the edges of the box. Contact then occurs at the box's bottom, with the bottom corners finally filling. Although there is some localized thinning at the bottom corners, with strains on the order of 100%, these strains are not too much larger than the 80% strains seen on the midsides.

Figure 6 and *Figure 7* show the equivalent plastic strain at the end of the process using the surface-to-surface and the node-to-surface contact formulations, respectively. The differences in the results are primarily due to differences in the way that the blank thickness is handled. Consequently, results from the surface-to-surface contact formulation are more reliable since the blank thickness is considered.

Figure 8 shows the evolution in time of the ratio between the maximum creep strain rate found in the model and the target creep strain rate for the two contact formulations. The load applied initially produces a low maximum creep strain rate at the beginning of the analysis. At the end the maximum creep strain rate falls substantially as the die cavity fills up. Although the curve appears very jagged, it indicates that the maximum peak strain rate is always relatively close to the target value. This is quite acceptable in practice. *Figure 9* shows the pressure schedule that Abaqus calculates for this problem. For most of the time, while the sheet does not contact the bottom of the die, the pressure is low. Once the die starts restraining the deformation, the pressure can be increased substantially without producing high strain rates. Again, the differences in the pressure schedule toward the end of the simulation are due primarily to the differences in the handling of the blank thickness.

Input files

superplasticbox_constpress.inp	Constant pressure main analysis.
superplasticbox_autopress.inp	Automatic pressurization main analysis.
superplasticbox_autopress_surf.inp	Automatic pressurization main analysis using the surface-to-surface contact formulation.
superplasticbox_node.inp	Node definitions for the rigid elements.
superplasticbox_element.inp	Element definitions for the rigid R3D3 elements.

Figures

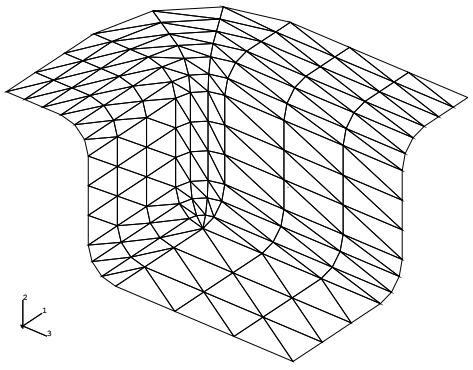


Figure 1: Rigid surface for die.

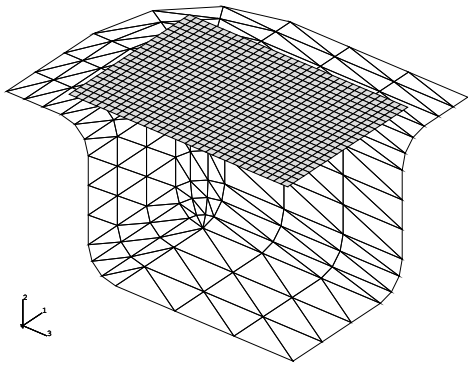


Figure 2: Initial position of blank with respect to die.

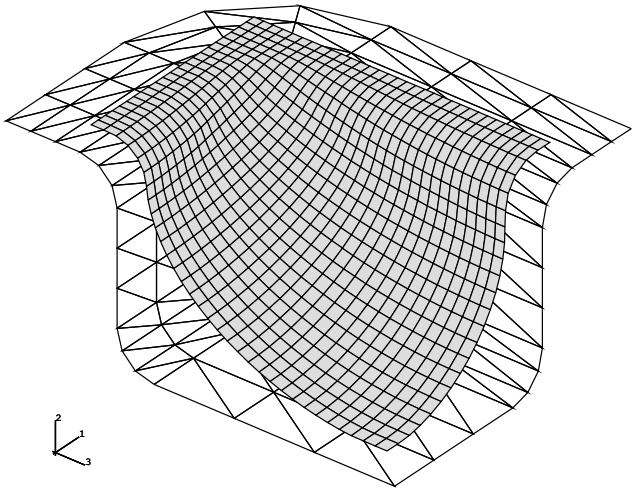


Figure 3: Automatic loading: deformed configuration after 34 sec in Step 2 using the node-to-surface contact formulation.

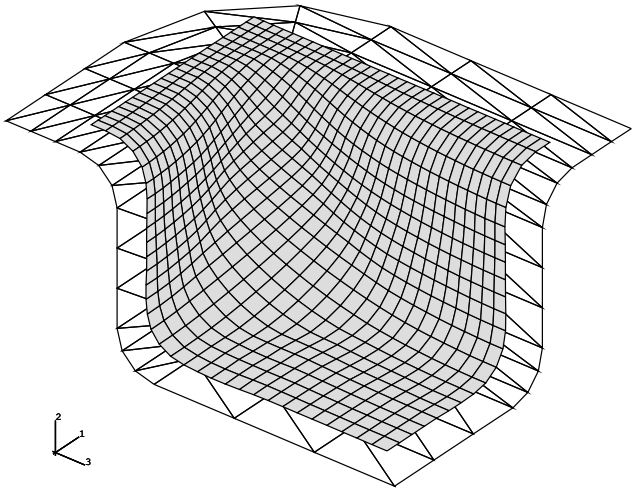


Figure 4: Automatic loading: deformed configuration after 63 sec in Step 2 using the node-to-surface contact formulation.

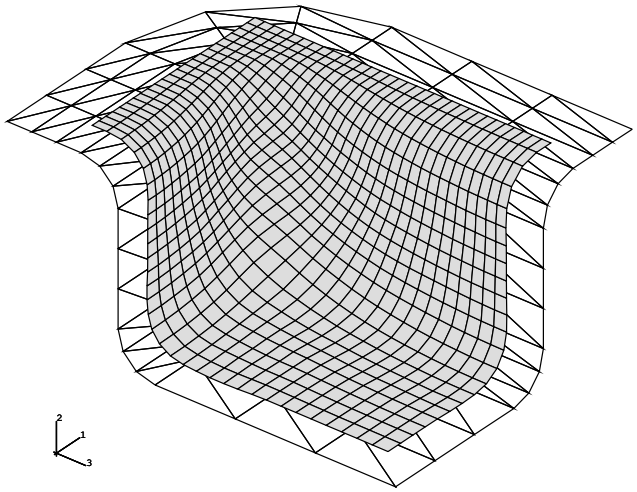


Figure 5: Automatic loading: deformed configuration after 83 sec in Step 2 using the node-to-surface contact formulation.

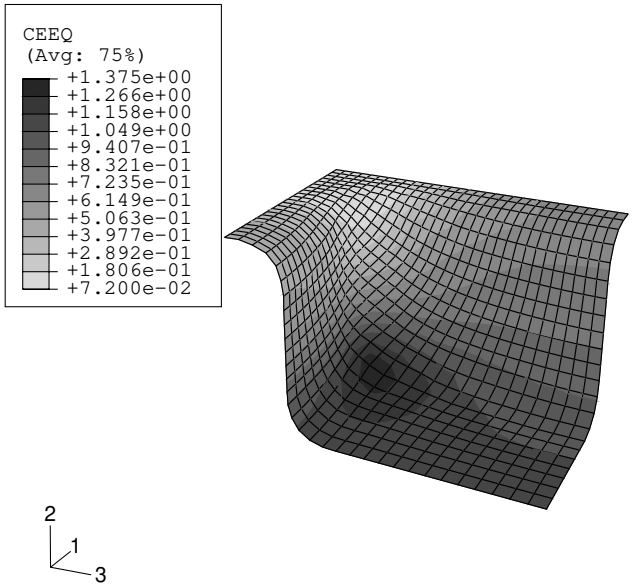


Figure 6: Automatic loading: inelastic strain in the formed box using the surface-to-surface contact formulation.

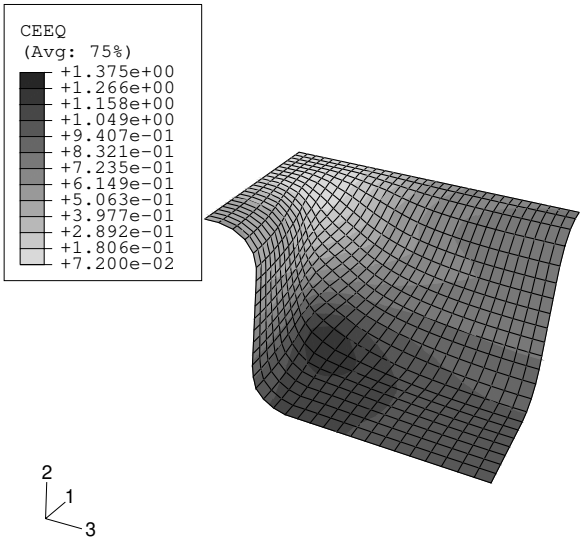


Figure 7: Automatic loading: inelastic strain in the formed box using the node-to-surface contact formulation.

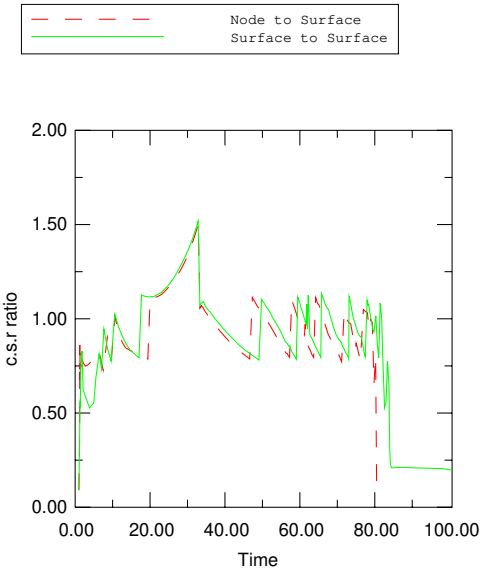


Figure 8: History of ratio between maximum creep strain rate and target creep strain rate.

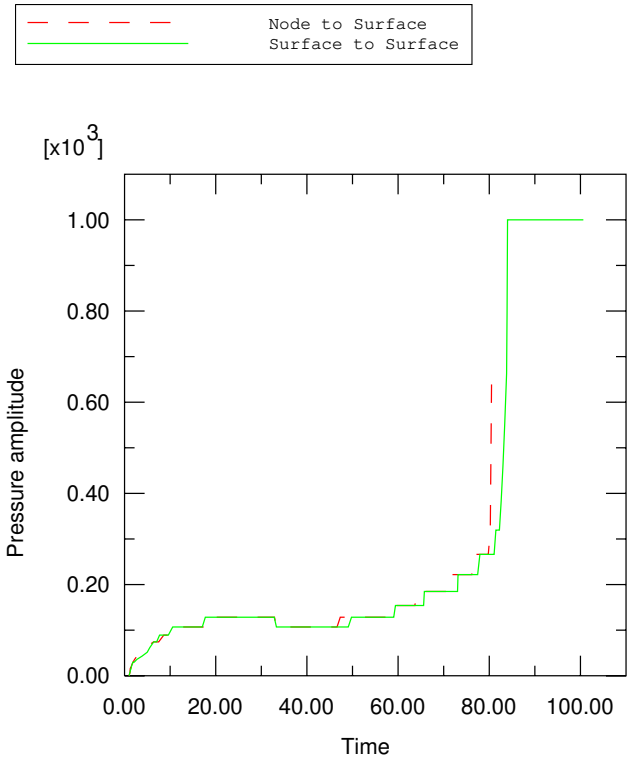


Figure 9: History of pressure amplitude.

Stretching of a thin sheet with a hemispherical punch

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example shows the stretching of a thin circular sheet with a hemispherical punch.

Stamping of sheet metals by means of rigid punches and dies is a standard manufacturing process. In most bulk forming processes the loads required for the forming operation are often the primary concern. However, in sheet forming the prediction of strain distributions and limit strains (which define the onset of local necking) are most important. Such analysis is complicated in that it requires consideration of large plastic strains during deformation, an accurate description of material response including strain hardening, the treatment of a moving boundary that separates the region in contact with the punch head from the unsupported one, and the inclusion of friction between the sheet and the punch head.

Geometry and model

The geometry of this problem is shown in *Figure 1*. The sheet being stretched has a clamping radius, r_0, of 59.18 mm. The radius of the punch, r_p, is 50.8 mm; the die radius, r_d, is 6.35 mm; and the initial thickness of the sheet, t_0, is 0.85 mm. Such a sheet has been tested experimentally by Ghosh and Hecker (1975) and has been analyzed by Wang and Budiansky (1978) using an axisymmetric membrane shell finite element formulation. The analysis is conducted statically in Abaqus/Standard and dynamically in Abaqus/Explicit such that inertial forces are relatively small. The initial configuration for the analysis is shown in *Figure 2*.

The sheet, the punch, and the die are modeled as separate parts, each instanced once. As an axisymmetric problem in Abaqus/Standard the sheet is modeled using 50 elements of type SAX1 (or MAX1) or 25 elements of type SAX2 (or MAX2). The Abaqus/Explicit model uses 50 elements of type SAX1. Mesh convergence studies (not reported here) have been done and indicate that these meshes give acceptably accurate results for most of the values of interest. To test the three-dimensional membrane and shell elements in Abaqus/Standard, a 10° sector is modeled using 100 elements of type S4R, S4, SC8R, or M3D4R or 25 elements of type M3D9R. All these meshes are reasonably fine; they are used to obtain good resolution of the moving contact between the sheet and the dies. In the Abaqus/Standard shell models nine integration points are used through the thickness of the sheet to ensure the development of yielding and elastic-plastic bending response; in Abaqus/Explicit five integration points are used through the thickness of the sheet.

The rigid punch and die are modeled in Abaqus/Standard as analytical rigid surfaces with a surface in conjunction with a rigid body constraint. The top and bottom surfaces of the sheet are defined with surface definitions. In Abaqus/Explicit the punch and die are modeled as rigid bodies; the surface of the punch and die are modeled either by analytical rigid surfaces or RAX2 elements. In the Abaqus/Explicit analyses the rigid surfaces are offset from the blank by half the thickness of the blank because the contact algorithm in Abaqus/Explicit takes the shell thickness into account. Similarly, when the surface-to-surface contact formulation in Abaqus/Standard for contact pairs or general contact is used, the blank thickness is considered by default and the rigid surfaces are offset from the blank by half the blank thickness, consistent with the physical reality. However, most of the input files presented in this section use the node-to-surface contact pair formulation in Abaqus/Standard. In these cases the shell thickness is ignored, and the mid-surface of the shell is used in the contact calculations as an approximation.

Material properties

The material (aluminum-killed steel) is assumed to satisfy the Ramberg-Osgood relation between true stress and logarithmic strain,

$$\varepsilon = \sigma/E + (\sigma/K)^n,$$

where Young's modulus, E, is 206.8 GPa; the reference stress value, K, is 0.510 GPa; and the work-hardening exponent, n, is 4.76. In the present Abaqus analyses the Ramberg-Osgood relation is approximated using an elastic-plastic material. The material is assumed to be linear elastic up to a yield stress of 170.0 MPa, and the stress-strain curve beyond the yield stress is defined in piecewise linear segments. Poisson's ratio is 0.3.

The membrane element models in Abaqus/Standard are inherently unstable in a static analysis unless some prestress is present in the elements prior to the application of external loading. Therefore, an equibiaxial initial stress condition equal to 5% of the initial yield stress is prescribed for the membrane elements in Abaqus/Standard.

Contact interactions

The contact between the sheet and the rigid punch and the rigid die is modeled with a contact pair and with general contact. The mechanical interaction between the contact surfaces is assumed to be frictional contact, with a coefficient of friction of 0.275.

Loading

The Abaqus/Standard analysis is carried out in five steps; the Abaqus/Explicit analysis is carried out in four steps. In Abaqus/Explicit the velocity of the punch head is prescribed using a velocity boundary condition whose amplitude is ramped up to 30 m/s at 1.24 milliseconds during the first step and then kept constant until time reaches 1.57 milliseconds at the end of the second step. It is then ramped down to zero at a time of 1.97 milliseconds at the end of the third step. In the first three steps of the Abaqus/Standard and Abaqus/Explicit analyses, the punch head is moved toward the sheet through total distances of 18.6 mm, 28.5 mm, and 34.5 mm, respectively. The purpose of these three steps is to compare the results with those provided experimentally by Ghosh and Hecker for these punch displacements. More typically the punch move through its entire travel in one step.

Two final steps are included in the Abaqus/Standard analysis. In the first step, the metal sheet is held in place and contact is removed from the model. In the second step, the original boundary conditions for the metal sheet are reintroduced for springback analysis. However, this springback step is not included for the analyses using membrane elements, since these elements do not have any bending stiffness and residual bending stress is often a key determinant of springback.

In the final step of the Abaqus/Explicit analysis, the punch head is moved away from the sheet for springback analysis. A viscous pressure load is applied to the surface of the shell during this step to damp out transient wave effects so that quasi-static equilibrium can be reached quickly. This effect happens within approximately 2 milliseconds from the start of unloading. The coefficient of viscous pressure is chosen to be 0.35 MPa sec/m, approximately 1% of the value of ρc_d, where ρ is the material density of the sheet and c_d is the dilatational wave speed. A value of viscous pressure of ρc_d would absorb all the energy in a pressure wave. For typical structural problems choosing a small percentage of this value provides an effective way of minimizing ongoing dynamic effects. Static equilibrium is reached when residual stresses in the sheet are reasonably constant over time.

Results and discussion

Figure 2 shows the initial, undeformed profile of the blank, the die, and the punch. *Figure 3* illustrates the deformed sheet and the punch and the die. *Figure 4* shows a plot of the same system after the punch is lifted back, showing the springback of the sheet.

Figure 5 shows the distribution of nominal values of radial and circumferential membrane strain in the sheet for an 18.6 mm punch head displacement. *Figure 6* shows the strain distributions at a punch head displacement of 28.5 mm, and *Figure 7* shows the strain distributions at a punch head displacement of 34.5 mm. The strain distributions for the SAX1 models compare well with those obtained experimentally by Ghosh and Hecker

(1975) and those obtained numerically by Wang and Budiansky (1978), who used a membrane shell finite element formulation. The important phenomenon of necking during stretching is reproduced at nearly the same location, although slightly different strain values are obtained. Draw beads are used to hold the edge of the sheet in the experiment, but in this analysis the sheet is simply clamped at its edge. Incorporation of the draw bead boundary conditions may further improve the correlation with the experimental data.

A spike can be observed in the radial strain distribution toward the edge of the sheet in some of the Abaqus/Standard shell models. This strain spike is the result of the sheet bending around the die. The spike is not present in the membrane element models since they possess no bending stiffness. All of the Abaqus/Standard results presented in *Figure 5* through *Figure 9* use the node-to-surface contact pair formulation. Similar results are obtained when using the surface-to-surface contact pair or the general contact formulations.

The results obtained with the axisymmetric membrane models are compared with those obtained from the axisymmetric shell models and are found to be in good agreement.

These analyses assume a value of 0.275 for the coefficient of friction. Ghosh and Hecker do not give a value for their experiments, but Wang and Budiansky assume a value of 0.17. The coefficient of friction has a marked effect on the peak strain during necking and may be a factor contributing to the discrepancy of peak strain results during necking. The values used in these analyses have been chosen to provide good correlation with the experimental data.

The distributions of the residual stresses on springback of the sheet are shown in *Figure 8* and *Figure 9*.

Input files

Abaqus/Standard input files

thinsheetstretching_m3d4r.inp	Element type M3D4R.
thinsheetstretching_m3d4r_surf.inp	Element type M3D4R using surface-to-surface contact while accounting for shell thickness.
thinsheetstretching_m3d4r_gcontsd.inp	Element type M3D4R using general contact.
thinsheetstretching_m3d9r.inp	Element type M3D9R.
thinsheetstretching_max1.inp	Element type MAX1.
thinsheetstretching_max2.inp	Element type MAX2.
thinsheetstretching_s4.inp	Element type S4.
thinsheetstretching_s4r.inp	Element type S4R.
thinsheetstretching_s4r_po.inp	**POST OUTPUT* analysis.
thinsheetstretching_sc8r.inp	Element type SC8R.
thinsheetstretching_sax1.inp	Element type SAX1.
thinsheetstretching_sax2.inp	Element type SAX2.
thinsheetstretching_restart.inp	Restart of thinsheetstretching_sax2.inp.

Abaqus/Explicit input files

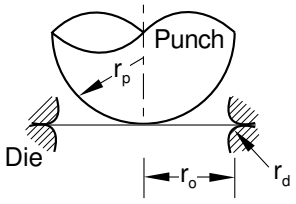
hemipunch_anl.inp	Model using analytical rigid surfaces to describe the rigid surface.
hemipunch.inp	Model using rigid elements to describe the rigid surface.

References

- Ghosh, A. K., and S. S. Hecker, “Failure in Thin Sheets Stretched Over Rigid Punches,” *Metallurgical Transactions*, vol. 6A, pp. 1065–1074, 1975.

• Wang, N. M., and B. Budiansky, “Analysis of Sheet Metal Stamping by a Finite Element Method,” *Journal of Applied Mechanics*, vol. 45, pp. 73–82, 1978.

Figures



r_o = 59.18 mm
r_p = 50.8 mm
r_d = 6.35 mm

Figure 1: Configuration and dimensions for hemispherical punch stretching.

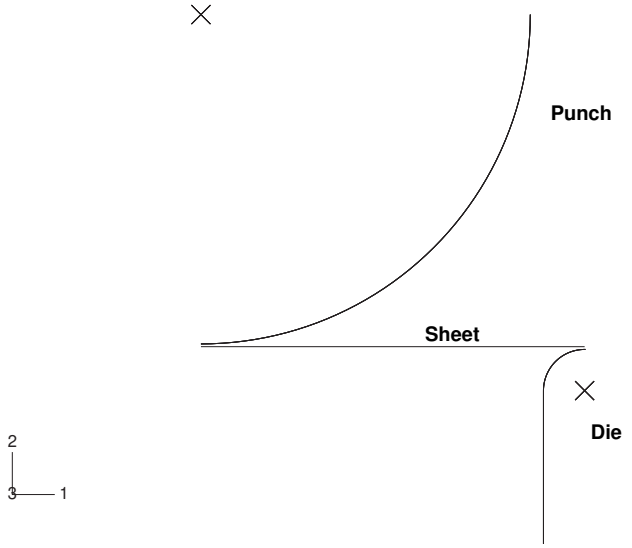


Figure 2: Initial configuration.

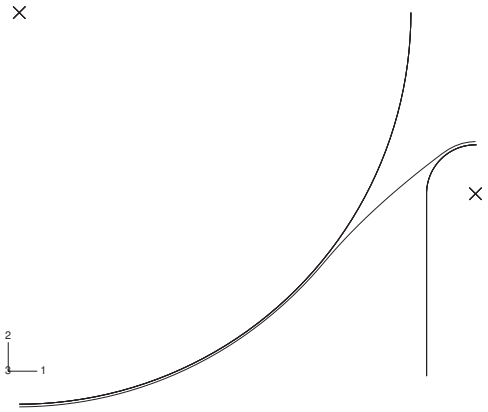


Figure 3: Configuration for punch head displacement of 34.5 mm, Abaqus/Explicit.

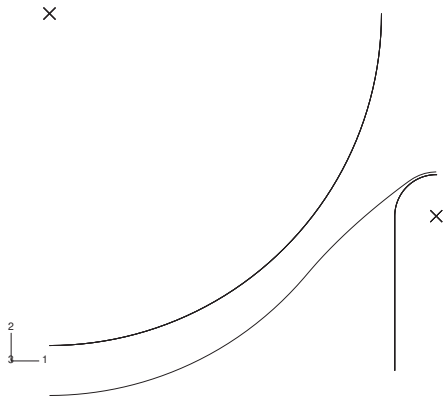


Figure 4: Final configuration after springback, Abaqus/Explicit.

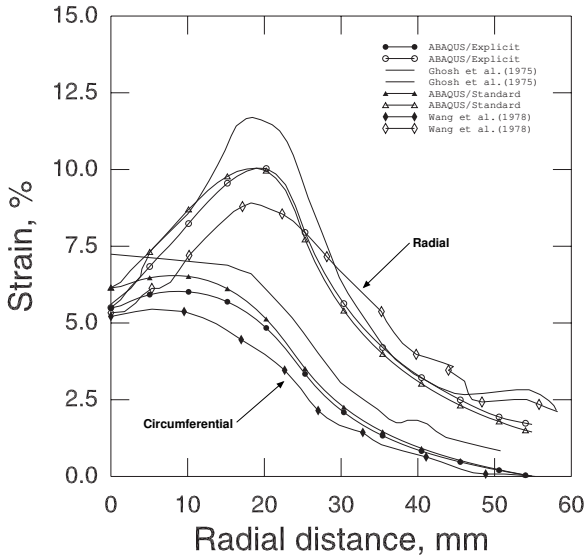


Figure 5: Strain distribution for punch head displacement of 18.6 mm.

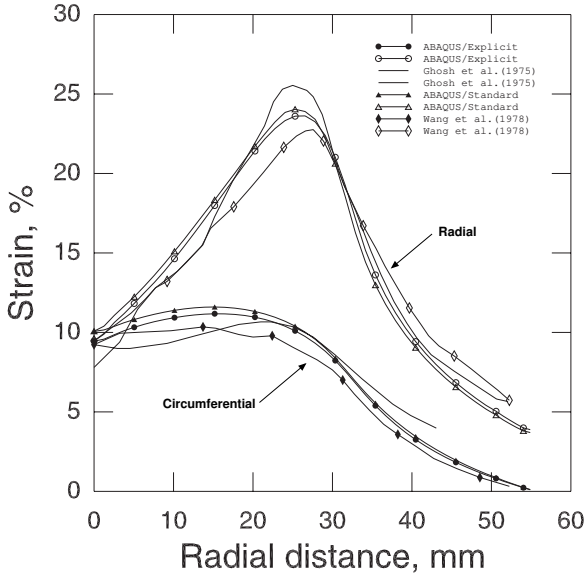


Figure 6: Strain distribution for punch head displacement of 28.5 mm.

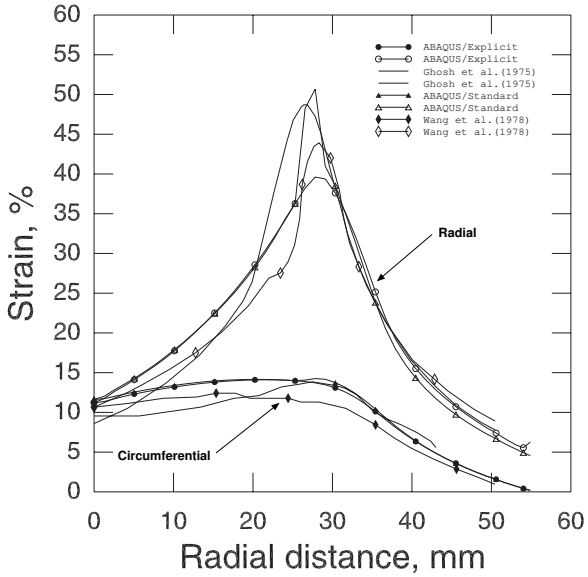


Figure 7: Strain distribution for punch head displacement of 34.5 mm.

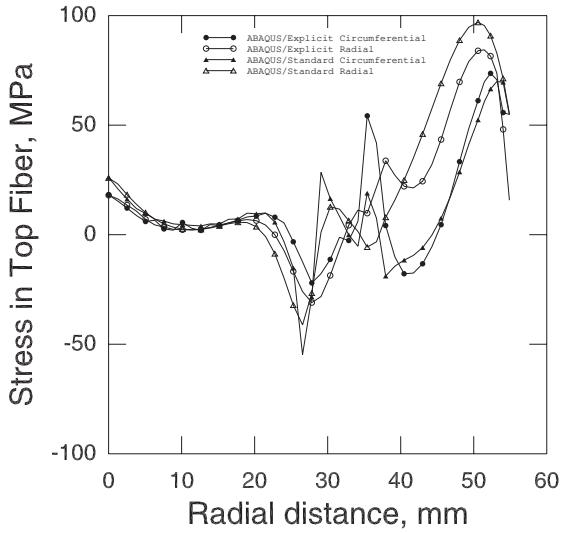


Figure 8: Residual stress on top surface after springback.

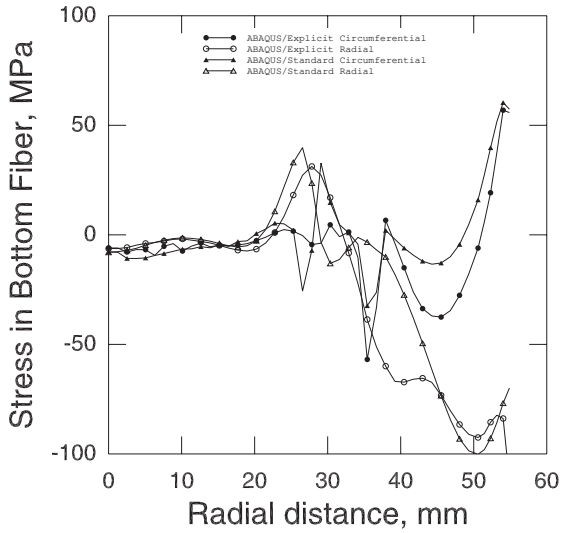


Figure 9: Residual stress on bottom surface after springback.

Deep drawing of a cylindrical cup

Products: Abaqus/Standard

Overview

This example illustrates the deep drawing of a sheet metal cylindrical cup.

Deep drawing of sheet metal is an important manufacturing technique. In the deep drawing process, a "blank" of sheet metal is clamped by a blank holder against a die. A punch is then moved against the blank, which is drawn into the die. Unlike the operation described in the hemispherical punch stretching example (*Stretching of a thin sheet with a hemispherical punch*), the blank is not assumed to be fixed between the die and the blank holder; rather, the blank is drawn from between these two tools. The ratio of drawing versus stretching is controlled by the force on the blank holder and the friction conditions at the interface between the blank and the blank holder and the die. Higher force or friction at the blank/die/blank holder interface limits the slip at the interface and increases the radial stretching of the blank. In certain cases drawbeads, shown in *Figure 1*, are used to restrain the slip at this interface even further.

To obtain a successful deep drawing process, it is essential to control the slip between the blank and its holder and die. If the slip is restrained too much, the material will undergo severe stretching, thus potentially causing necking and rupture. If the blank can slide too easily, the material will be drawn in completely and high compressive circumferential stresses will develop, causing wrinkling in the product. For simple shapes like the cylindrical cup here, a wide range of interface conditions will give satisfactory results. But for more complex, three-dimensional shapes, the interface conditions need to be controlled within a narrow range to obtain a good product.

During the drawing process the response is determined primarily by the membrane behavior of the sheet. For axisymmetric problems in particular, the bending stiffness of the metal yields only a small correction to the pure membrane solution, as discussed by Wang and Tang (1988). In contrast, the interaction between the die, the blank, and the blank holder is critical. Thus, thickness changes in the sheet material must be modeled accurately in a finite element simulation, since they will have a significant influence on the contact and friction stresses at the interface. In these circumstances the most suitable elements in Abaqus are the 4-node reduced-integration axisymmetric quadrilateral, CAX4R; the first-order axisymmetric shell element, SAX1; the first-order axisymmetric membrane element, MAX1; the first-order finite-strain quadrilateral shell element, S4R; the fully integrated general-purpose finite-membrane-strain shell element, S4; and the 8-node continuum shell element, SC8R.

Membrane effects and thickness changes are modeled properly with CAX4R. However, the bending stiffness of the element is low. The element does not exhibit "locking" due to incompressibility or parasitic shear. It is also very cost-effective. For shells and membranes the thickness change is calculated from the assumption of incompressible deformation of the material.

Geometry and model

The geometry of the problem is shown in *Figure 2*. The circular blank being drawn has an initial radius of 100 mm and an initial thickness of 0.82 mm. The punch has a radius of 50 mm and is rounded off at the corner with a radius of 13 mm. The die has an internal radius of 51.25 mm and is rounded off at the corner with a radius of 5 mm. The blank holder has an internal radius of 56.25 mm.

The blank is modeled using 40 elements of type CAX4R or 31 elements of type SAX1, MAX1, S4R, S4, or SC8R. An 11.25° wedge of the circular blank is used in the three-dimensional S4R and S4 models. These meshes are rather coarse for this analysis. However, since the primary interest in this problem is to study the membrane effects, the analysis will still give a fair indication of the stresses and strains occurring in the process.

The contact between the blank and the rigid punch, the rigid die, and the rigid blank holder is modeled with a contact pair in most cases. The top and bottom surfaces of the blank are defined as surfaces in the model. The

rigid punch, the die, and the blank holder are modeled as analytical rigid surfaces. The mechanical interaction between the contact surfaces is assumed to be frictional contact. Therefore, friction is used in conjunction with the various contact property definitions to specify coefficients of friction.

At the start of the analysis for the CAX4R model, the blank is positioned precisely on top of the die and the blank holder is precisely in touch with the top surface of the blank. The punch is positioned 0.18 mm above the top surface of the blank.

In the case of shells and membranes, the positioning of the blank depends on the contact formulation used. Node-to-surface and surface-to-surface contact formulations are available in Abaqus/Standard. For the node-to-surface formulation, the shell/membrane thickness is modeled using an exponential pressure-overclosure relationship (*Contact Pressure-Overclosure Relationships*). The blank holder is positioned a fixed distance above the blank. This fixed distance is the distance at which the contact pressure is set to zero using an exponential pressure-overclosure relationship. However, the surface-to-surface contact formulation (which applies to both contact pairs and general contact) automatically takes thickness into account, and the need for specifying pressure overclosure relations is eliminated. Examples of the surface-to-surface contact formulation with shell elements are provided in this problem.

Material properties

The material (aluminum-killed steel) is assumed to satisfy the Ramberg-Osgood relation between true stress and logarithmic strain:

$$\epsilon = (\sigma/K)^{1/n}.$$

The reference stress value, K, is 513 MPa; and the work-hardening exponent, n, is 0.223. The Young's modulus is 211 GPa, and the Poisson's ratio is 0.3. An initial yield stress of 91.3 MPa is obtained with these data. The stress-strain curve is defined in piecewise linear segments in the metal plasticity specification, up to a total (logarithmic) strain level of 107%.

The coefficient of friction between the interface and the punch is taken to be 0.25; and that between the die and the blank holder is taken as 0.1, the latter value simulating a certain degree of lubrication between the surfaces. The stiffness method of sticking friction is used in these analyses. The numerics of this method make it necessary to choose an acceptable measure of relative elastic slip between mating surfaces when sticking should actually be occurring. The basis for the choice is as follows. Small values of elastic slip best simulate the actual behavior but also result in a slower convergence of the solution. Permission of large relative elastic displacements between the contacting surfaces can cause higher strains at the center of the blank. In these runs we let Abaqus choose the allowable elastic slip, which is done by determining a characteristic interface element length over the entire mesh and multiplying by a small fraction to get an allowable elastic slip measure. This method typically gives a fairly small amount of elastic slip.

Although the material in this process is fully isotropic, the local coordinate system is used with the CAX4R elements to define a local orientation that is coincident initially with the global directions. The reason for using this option is to obtain the stress and strain output in more natural coordinates: if the local coordinate system is used in a geometrically nonlinear analysis, stress and strain components are given in a corotational framework. Hence, in our case throughout the motion, S11 will be the stress in the r–z plane in the direction of the middle surface of the cup. S22 will be the stress in the thickness direction, S33 will be the hoop stress, and S12 will be the transverse shear stress, which makes interpreting the results considerably easier. This orientation definition is not necessary with the SAX1 or MAX1 elements since the output for shell and membrane elements is already given in the local shell system. For the SAX1 and MAX1 model, S11 is the stress in the meridional direction and S22 is the circumferential (hoop) stress. An orientation definition would normally be needed for the S4R and S4 models but can be avoided by defining the wedge in such a manner that the single integration point of each element lies along the global x-axis. Such a model definition, along with appropriate kinematic boundary conditions, keeps the local stress output definitions for the shells as S11 being the stress in the meridional plane and S22 the hoop stress. There should be no in-plane shear, S12, in this problem. A transformation is used in the S4R and S4 models to impose boundary constraints in a cylindrical system.

Loading

The entire analysis is carried out in five steps. In the first step the blank holder is pushed onto the blank with a prescribed displacement to establish contact. In the shell models this displacement roughly corresponds to zero clearance across the interface.

In the second step the boundary condition is removed and replaced by the applied force of 100 kN on the blank holder. This force is kept constant during Steps 2 and 3. This technique of simulating the clamping process is used to avoid potential problems with rigid body modes of the blank holder, since there is no firm contact between the blank holder, the blank, and the die at the start of the process. The two-step procedure creates contact before the blank holder is allowed to move freely.

In the third step the punch is moved toward the blank through a total distance of 60 mm. This step models the actual drawing process. During this step the time incrementation parameters are set to improve efficiency for severely discontinuous behavior associated with frictional contact.

The last two steps are used to simulate springback. In the fourth step all the nodes in the model are fixed in their current positions and the contact pairs are removed from the model. This is the most reliable method for releasing contact conditions. In the fifth, and final, step the regular set of boundary conditions is reinstated and the springback is allowed to take place. This part of the analysis with the CAX4R elements is included to demonstrate the feasibility of the unloading procedure only and is not expected to produce realistic results, since the reduced-integration elements have a purely elastic bending behavior. The springback is modeled with more accuracy in the shell element models.

Results and discussion

Figure 3 shows deformed shapes that are predicted at various stages of the drawing process for the CAX4R model. The profiles show that the metal initially bends and stretches and is then drawn in over the surface of the die. The distributions of radial and circumferential strain for all three models and thickness strain for the CAX4R model are shown in *Figure 4*. The thickness for the shell or membrane models can be monitored with output variable STH (current shell or membrane thickness). The thickness does not change very much: the change ranges from approximately −12% in the cylindrical part to approximately +16% at the edge of the formed cup. Relatively small thickness changes are usually desired in deep drawing processes and are achieved because the radial tensile strain and the circumferential compressive strain balance each other.

The drawing force as a function of punch displacement for various element types is shown in *Figure 5*, where the curves are seen to match closely. Similarly, the drawing force as a function of punch displacement with S4R elements using the node-to-surface and surface-to-surface contact pair formulations is shown in *Figure 6*. The differences in the reaction force history are due to consideration of the blank thickness explicitly in the surface-to-surface contact formulation as compared to the node-to-surface contact formulation where a pressure-overclosure relationship is specified. In all of the cases, oscillations in the force history are seen. These oscillations are a result of the rather coarse mesh—each oscillation represents an element being drawn over the corner of the die. Compared to the shell models, the membrane model predicts a smaller punch force for a given punch displacement. Thus, toward the end of the analysis the results for punch force versus displacement for the MAX1 model are closer to those for the CAX4R model.

The deformed shape after complete unloading is shown in *Figure 7*, superimposed on the deformed shape under complete loading. The analysis shows the lip of the cup springing back strongly after the blank holder is removed for the CAX4R model. No springback is evident in the shell models. As was noted before, this springback in the CAX4R model is not physically realistic: in the first-order reduced-integration elements an elastic “hourglass control” stiffness is associated with the “bending” mode, since this mode is identical to the “hourglass” mode exhibited by this element in continuum situations. In reality the bending of the element is an elastic-plastic process, so that the springback is likely to be much less. A better simulation of this aspect would be achieved by using several elements through the thickness of the blank, which would also increase the cost of the analysis.

The springback results for the shell models do not exhibit this problem and are clearly more representative of the actual elastic-plastic process.

Input files

deepdrawcup_cax4r.inp	CAX4R model.
deepdrawcup_cax4r_surf.inp	CAX4R model using surface-to-surface contact.
deepdrawcup_cax4i.inp	Model using the incompatible mode element, CAX4I, as an alternative to the CAX4R element. In contrast to the reduced-integration, linear isoparametric elements such as the CAX4R element, the incompatible mode elements have excellent bending properties even with one layer of elements through the thickness (see *Geometrically nonlinear analysis of a cantilever beam*) and have no hourglassing problems. However, they are computationally more expensive.
deepdrawcup_s4.inp	S4 model.
deepdrawcup_s4_surf.inp	S4 model using surface-to-surface contact.
deepdrawcup_s4_gcontsd.inp	S4 model using general contact.
deepdrawcup_s4r.inp	S4R model.
deepdrawcup_s4r_surf.inp	S4R model using surface-to-surface contact.
deepdrawcup_sc8r.inp	SC8R model.
deepdrawcup_sax1.inp	SAX1 model.
deepdrawcup_postoutput.inp	**POST OUTPUT* analysis of deepdrawcup_sax1.inp.
deepdrawcup_max1.inp	MAX1 model.
deepdrawcup_mgax1.inp	MGAX1 model.

References

- Wang, N. M., and S. C. Tang, “Analysis of Bending Effects in Sheet Forming Operations,” International Journal for Numerical Methods in Engineering, vol. 25, pp. 253–267, January 1988.

Figures

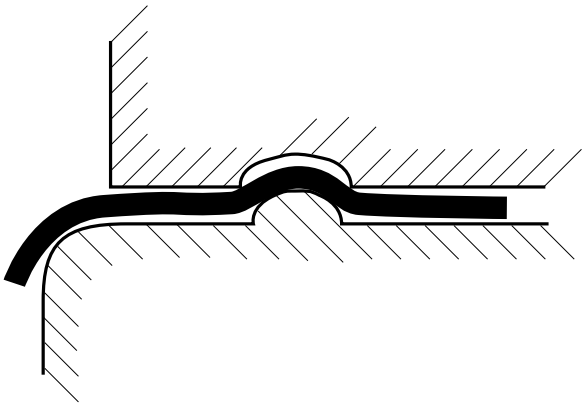


Figure 1: A typical drawbead used to limit slip between the blank and die.

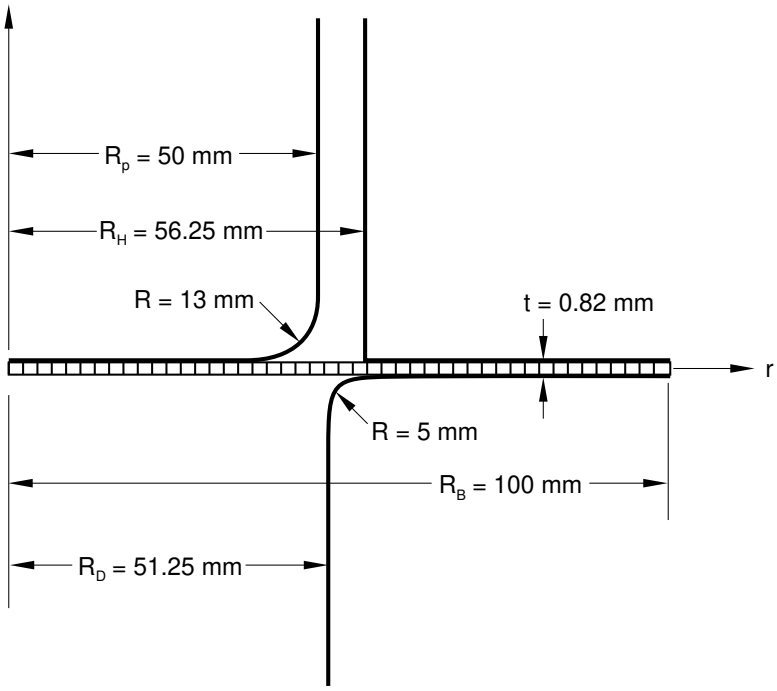


Figure 2: Geometry and mesh for the deep drawing problem.

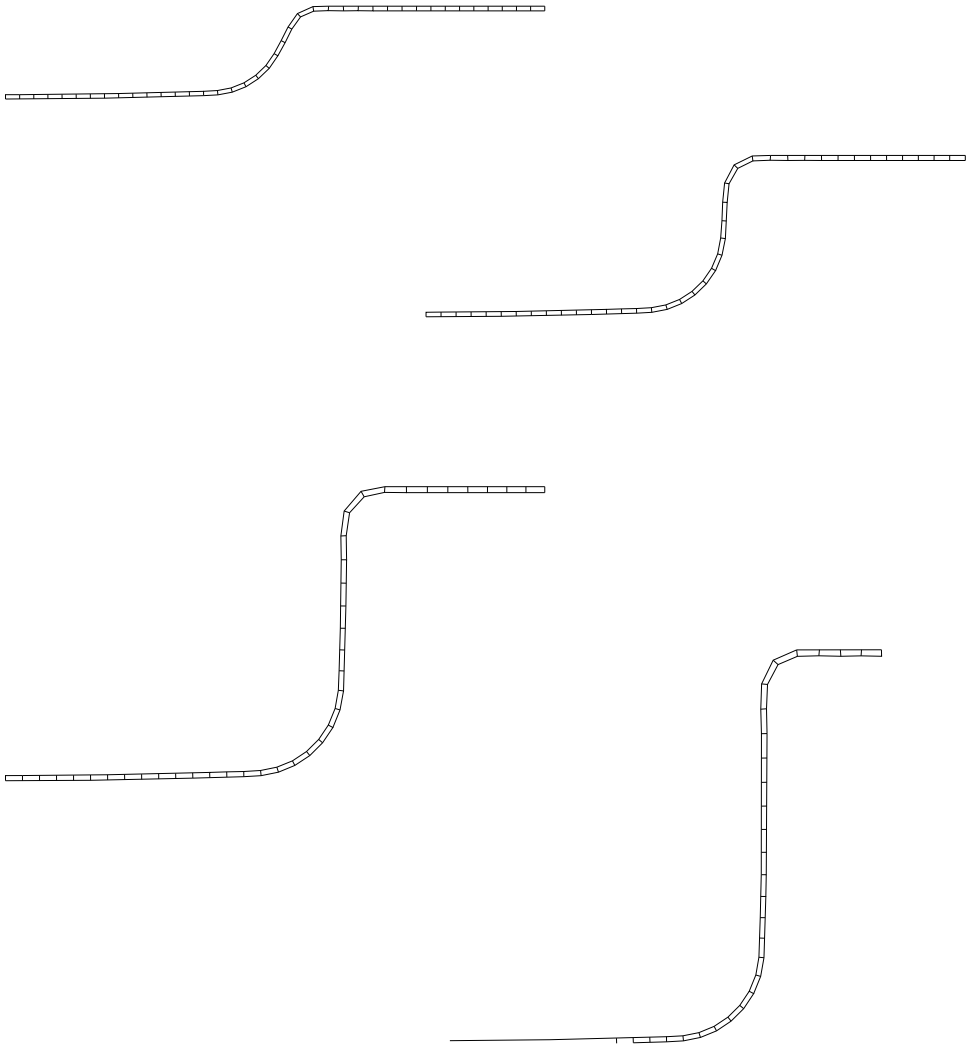


Figure 3: Deformed shapes at various stages of the analysis.

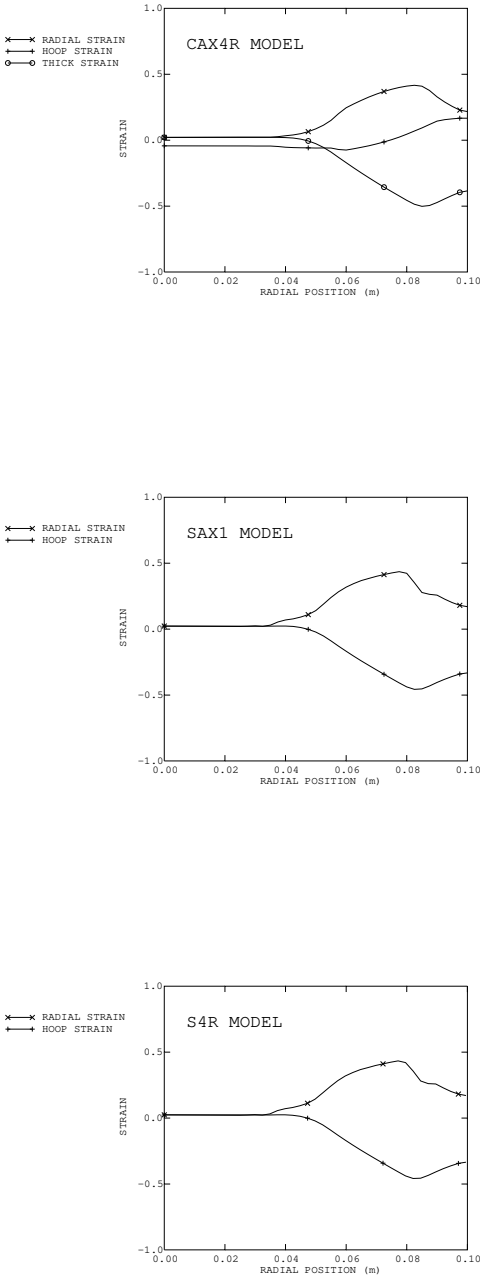


Figure 4: Strain distribution at the end of the deep drawing step.

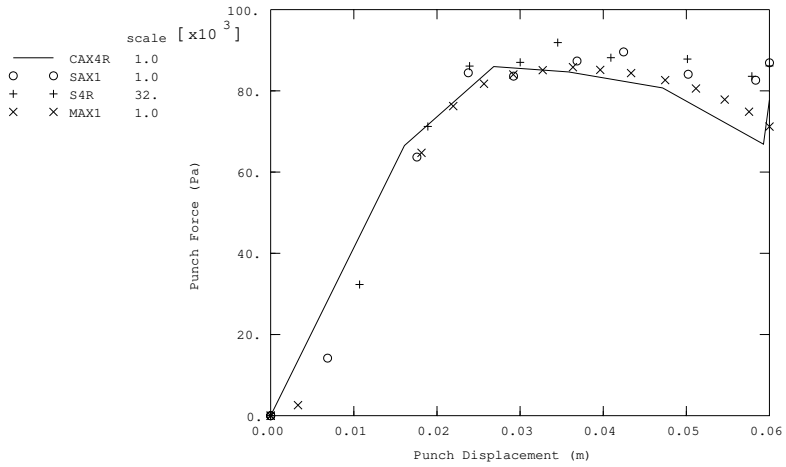


Figure 5: Punch force versus punch displacement using the node-to-surface contact formulation.

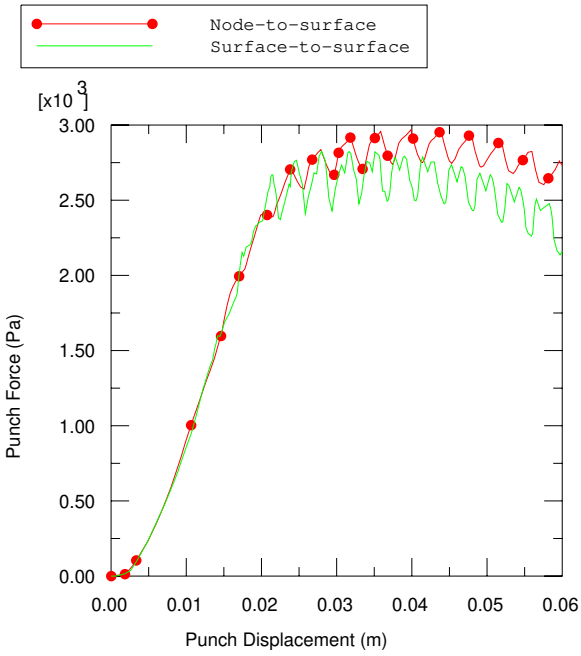


Figure 6: Comparison of punch force versus punch displacement with S4R elements for different contact formulations.

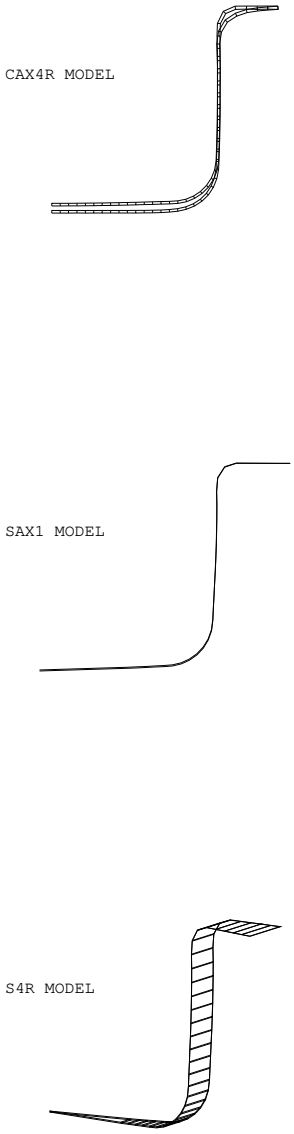


Figure 7: Deformed shape after unloading.

Extrusion of a cylindrical metal bar with frictional heat generation

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example shows how to simulate extrusion problems using Abaqus.
In this particular problem the radius of an aluminum cylindrical bar is reduced 33% by an extrusion process. The generation of heat due to plastic dissipation inside the bar and the frictional heat generation at the workpiece/die interface are considered.

Geometry and model

The bar has an initial radius of 100 mm and is 300 mm long. *Figure 1* shows half of the cross-section of the bar, modeled with first-order axisymmetric elements (CAX4T and CAX4RT elements in Abaqus/Standard and CAX4RT elements in Abaqus/Explicit).

In the primary analysis in both Abaqus/Standard and Abaqus/Explicit, heat transfer between the deformable bar and the rigid die is not considered, although frictional heating is included. A fully coupled temperature-displacement analysis is performed with the die kept at a constant temperature. In addition, an adiabatic analysis is presented using Abaqus/Standard without accounting for frictional heat generation. The node-to-surface (default) and surface-to-surface contact pair formulations as well as general contact in Abaqus/Standard are presented. In the case of Abaqus/Explicit, penalty and kinematic contact formulations are used in the definition of contact interactions.

Various techniques are used to model the rigid die. In Abaqus/Standard the die is modeled with CAX4T elements made into an isothermal rigid body using an isothermal rigid body and with an analytical rigid surface. In Abaqus/Explicit the die is modeled with an analytical rigid surface and discrete rigid elements (RAX2). The fillet radius is set to 0.075 for models using an analytical rigid surface to smoothen the die surface.

The Abaqus/Explicit simulations are also performed with Arbitrary Lagrangian-Eulerian (ALE) adaptive meshing and enhanced hourglass control.

Material model and interface behavior

The material model is chosen to reflect the response of a typical commercial purity aluminum alloy. The material is assumed to harden isotropically. The dependence of the flow stress on the temperature is included, but strain rate dependence is ignored. Instead, representative material data at a strain rate of 0.1 $\sec^{-1}$ are selected to characterize the flow strength.

The interface is assumed to have no conductive properties. Coulomb friction is assumed for the mechanical

behavior, with a friction coefficient of 0.1. Gap heat generation is used to specify the fraction, f_g, of total heat generated by frictional dissipation that is transferred to the two bodies in contact. Half of this heat is conducted into the workpiece, and the other half is conducted into the die. Furthermore, 90% of the nonrecoverable work due to plasticity is assumed to heat the work material.

Boundary conditions, loading, and solution control

In the first step, the bar is moved to a position where contact is established and slipping of the workpiece against the die begins. In the second step, the bar is extruded through the die to realize the extrusion process. This is accomplished by prescribing displacements to the nodes at the top of the bar. In the third step, contact is deactivated in preparation for the cool down portion of the simulation. In Abaqus/Standard this is performed in a single step:

the bar is allowed to cool down using film conditions, and deformation is driven by thermal contraction during the fourth step.

Volume proportional damping is applied to two of the analyses that are considered. In one case the automatic stabilization scheme with a constant damping factor is used. A nondefault damping density is chosen so that a converged and accurate solution is obtained. In another case the adaptive automatic stabilization scheme with a default damping density is used. In this case the damping factor is automatically adjusted based on the convergence history.

In Abaqus/Explicit the cool down simulation is broken into two steps: the first introduces viscous pressure to damp out dynamic effects and, thus, allow the bar to reach static equilibrium quickly; the balance of the cool down simulation is performed in a fifth step. The relief of residual stresses through creep is not analyzed in this example.

In Abaqus/Explicit mass scaling is used to reduce the computational cost of the analysis; nondefault hourglass control is used to control the hourglassing in the model. The default integral viscoelastic approach to hourglass control generally works best for problems where sudden dynamic loading occurs; a stiffness-based hourglass control is recommended for problems where the response is quasi-static. A combination of stiffness and viscous hourglass control is used in this problem.

For purposes of comparison a second problem is also analyzed, in which the first two steps of the previous analysis are repeated in a static analysis with the adiabatic heat generation capability. The adiabatic analysis neglects heat conduction in the bar. Frictional heat generation must also be ignored in this case. This problem is analyzed only in Abaqus/Standard.

Results and discussion

The following discussion centers around the results obtained with Abaqus/Standard. The results of the Abaqus/Explicit simulation are in close agreement with those obtained with Abaqus/Standard for both the node-to-surface and surface-to-surface contact pair formulations.

Figure 2 shows the deformed configuration after Step 2 of the analysis. *Figure 3* and *Figure 4* show contour plots of plastic strain and the von Mises stress at the end of Step 2 for the fully coupled analysis using CAX4RT elements. These plots show good agreement between the results using the two contact formulations in Abaqus/Standard. The plastic deformation is most severe near the surface of the workpiece, where plastic strains exceed 100%. The peak stresses occur in the region where the diameter of the workpiece narrows down due to deformation and also along the contact surface. *Figure 5* compares nodal temperatures obtained at the end of Step 2 using the surface-to-surface contact formulation in Abaqus/Standard with those obtained using kinematic contact in Abaqus/Explicit. In both cases CAX4RT elements are used. The results from both of the analyses match very well even though mass scaling is used in Abaqus/Explicit for computational savings. The peak temperature occurs at the surface of the workpiece because of plastic deformation and frictional heating. The peak temperature occurs immediately after the radial reduction zone of the die. This is expected for two reasons. First, the material that is heated by dissipative processes in the reduction zone will cool by conduction as the material progresses through the postreduction zone. Second, frictional heating is largest in the reduction zone because of the larger values of shear stress in that zone.

Similar results were obtained with the two types of stabilization considered. Adaptive automatic stabilization is generally preferred because it is easier to use. It is often necessary to specify a nondefault damping factor for the stabilization approach with a constant damping factor; whereas, with an adaptive damping factor, the default settings are typically appropriate.

Figure 6 compares results of a thermally coupled analysis with an adiabatic analysis using the surface-to-surface contact formulation in Abaqus/Standard. If we ignore the zone of extreme distortion at the end of the bar, the temperature increase on the surface is not as large for the adiabatic analysis because of the absence of frictional heating. As expected, the temperature field contours for the adiabatic heating analysis, shown in *Figure 6*, are very similar to the contours for plastic strain from the thermally coupled analysis, shown in *Figure 3*.

As noted earlier, excellent agreement is observed for the results obtained with Abaqus/Explicit (using both the default and enhanced hourglass control) and Abaqus/Standard. *Figure 7* compares the effects of ALE adaptive meshing on the element quality. The results obtained with ALE adaptive meshing show significantly reduced mesh distortion. The material point in the bar that experiences the largest temperature rise during the course of the simulation is indicated (node 2029 in the model without adaptivity). *Figure 8* compares the results obtained for the temperature history of this material point using Abaqus/Explicit with the results obtained using the two contact formulations in Abaqus/Standard. Again, a very good match between the results is obtained.

Input files

Abaqus/Standard input files

metalbarextrusion_coupled_fric.inp	Contact pair modeling of thermally coupled extrusion using CAX4T elements with frictional heat generation.
metalbarextrusion_stabil.inp	Contact pair modeling of thermally coupled extrusion using CAX4T elements with frictional heat generation and automatic stabilization, user-defined damping.
metalbarextrusion_stabil_adap.inp	Contact pair modeling of thermally coupled extrusion using CAX4T elements with frictional heat generation and adaptive automatic stabilization, default damping.
metalbarextrusion_coupled_fric_surf.inp	Contact pair modeling of thermally coupled extrusion using CAX4T elements with frictional heat generation and the surface-to-surface contact formulation.
metalbarextrusion_coupled_fric_gcontsd.inp	General contact modeling of thermally coupled extrusion using CAX4T elements with frictional heat generation.
metalbarextrusion_s_coupled_fric_cax4rt.inp	Contact pair modeling of thermally coupled extrusion using CAX4RT elements with frictional heat generation.
metalbarextrusion_s_coupled_fric_cax4rt_surf.inp	Contact pair modeling of thermally coupled extrusion using CAX4RT elements with frictional heat generation and the surface-to-surface contact formulation.
metalbarextrusion_adiab.inp	Contact pair modeling of extrusion with adiabatic heat generation and without frictional heat generation.
metalbarextrusion_adiab_surf.inp	Contact pair modeling of extrusion with adiabatic heat generation and without frictional heat generation using the surface-to-surface contact formulation.

Abaqus/Explicit input files

metalbarextrusion_x_cax4rt.inp	Thermally coupled extrusion with frictional heat generation and without ALE adaptive meshing; die modeled with an analytical rigid surface; kinematic mechanical contact.
metalbarextrusion_x_cax4rt_enh.inp	Thermally coupled extrusion with frictional heat generation and without ALE adaptive meshing; die modeled with an analytical rigid surface; kinematic mechanical contact; enhanced hourglass control.
metalbarextrusion_xad_cax4rt.inp	Thermally coupled extrusion with frictional heat generation and ALE adaptive meshing; die modeled with an analytical rigid surface; kinematic mechanical contact.
metalbarextrusion_xad_cax4rt_enh.inp	Thermally coupled extrusion with frictional heat generation and ALE adaptive meshing; die modeled with an analytical rigid surface; kinematic mechanical contact; enhanced hourglass control.

metalbarextrusion_xd_cax4rt.inp	Thermally coupled extrusion with frictional heat generation and without ALE adaptive meshing; die modeled with RAX2 elements; kinematic mechanical contact.
metalbarextrusion_xd_cax4rt_enh.inp	Thermally coupled extrusion with frictional heat generation and without ALE adaptive meshing; die modeled with RAX2 elements; kinematic mechanical contact; enhanced hourglass control.
metalbarextrusion_xp_cax4rt.inp	Thermally coupled extrusion with frictional heat generation and without ALE adaptive meshing; die modeled with an analytical rigid surface; penalty mechanical contact.
metalbarextrusion_xp_cax4rt_enh.inp	Thermally coupled extrusion with frictional heat generation and without ALE adaptive meshing; die modeled with an analytical rigid surface; penalty mechanical contact; enhanced hourglass control.

Figures

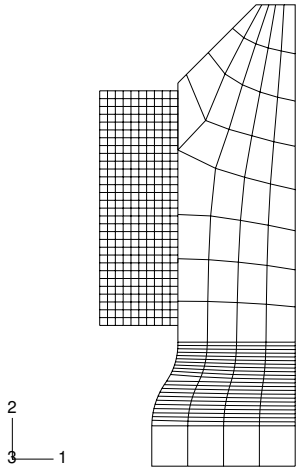


Figure 1: Mesh and geometry: axisymmetric extrusion with meshed rigid die, Abaqus/Standard.

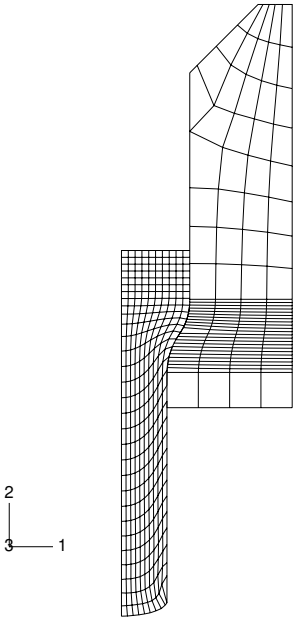


Figure 2: Deformed configuration: Step 2, Abaqus/Standard.

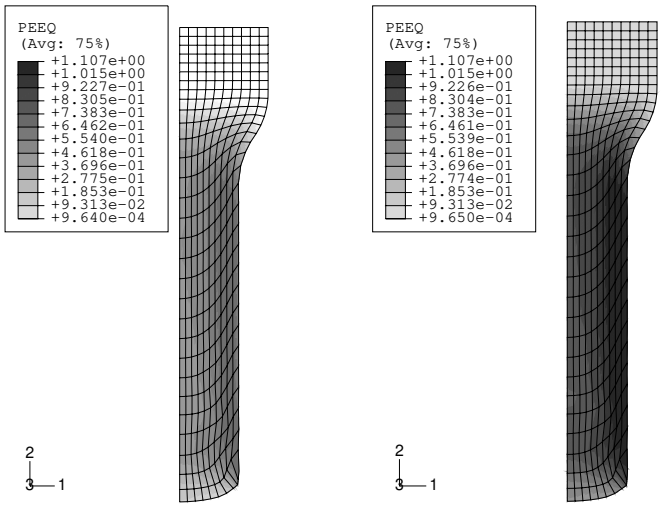


Figure 3: Plastic strain contours: Step 2, thermally coupled analysis (frictional heat generation), Abaqus/Standard (surface-to-surface contact formulation, left; node-to-surface contact formulation, right).

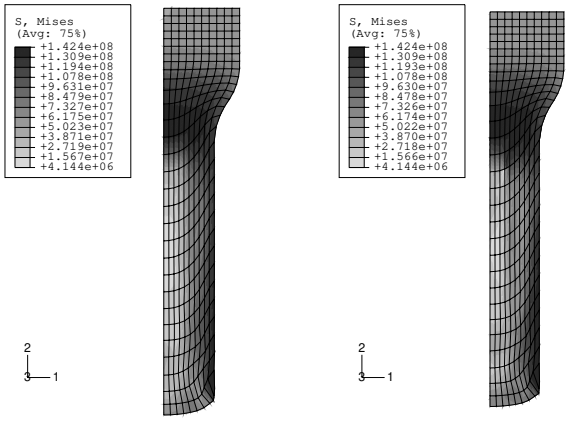


Figure 4: Von Mises stress contours: Step 2, thermally coupled analysis (frictional heat generation), Abaqus/Standard (surface-to-surface contact formulation, left; node-to-surface contact formulation, right).

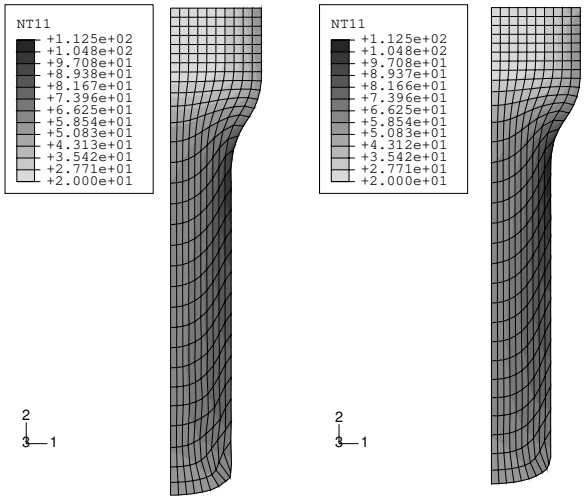


Figure 5: Temperature contours: Step 2, thermally coupled analysis (frictional heat generation); surface-to-surface contact formulation in Abaqus/Standard, left; Abaqus/Explicit, right.

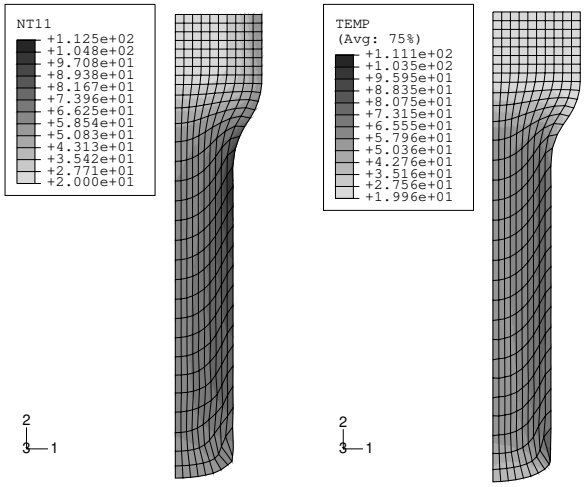


Figure 6: Temperature contours: Step 2, Abaqus/Standard using surface-to-surface contact formulation; thermally coupled analysis, left; adiabatic heat generation (without heat generation due to friction), right.

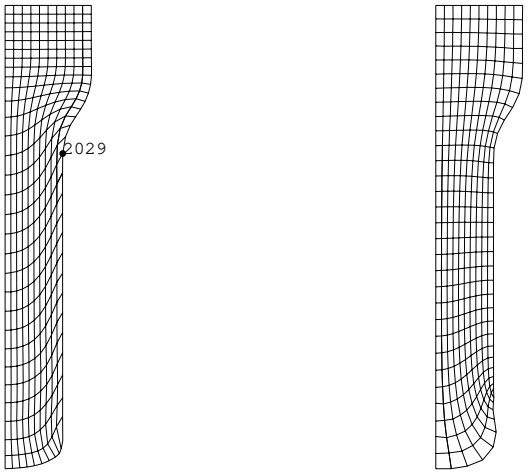


Figure 7: Deformed shape of the workpiece: Abaqus/Explicit; without adaptive remeshing, left; with ALE adaptive remeshing, right.

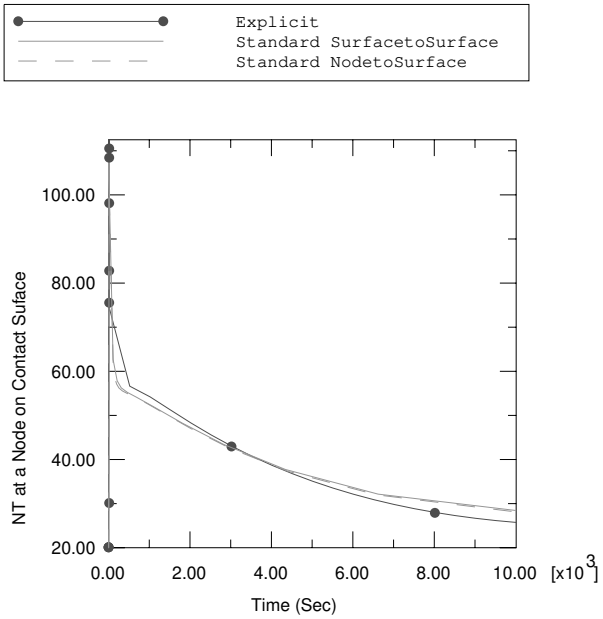


Figure 8: Temperature history of a node on the contact surface (nonadaptive result).

Rolling of thick plates

Products: Abaqus/Explicit

Overview

This example illustrates how Abaqus/Explicit can be used to model the rolling process, even though rolling is normally applied at relatively low speeds.

Hot rolling is a basic manufacturing technique used to transform preformed shapes into a form suitable for further processing. Rolling processes can be divided into different categories, depending on the complexity of metal flow and on the geometry of the rolled product. Finite element computations are used increasingly to analyze the elongation and spread of the material during rolling (Kobayashi, 1989). Although the forming process is often carried out at low roll speed, this example shows that a considerable amount of engineering information can be obtained by using the explicit dynamics procedure in Abaqus/Explicit to model the process.

The rolling process is first investigated using plane strain computations. These results are used to choose the modeling parameters associated with the more computationally expensive three-dimensional analysis.

Since rolling is normally performed at relatively low speeds, it is natural to assume that static analysis is the proper modeling approach. Typical rolling speeds (surface speed of the roller) are on the order of 1 m/sec. At these speeds inertia effects are not significant, so the response—except for rate effects in the material behavior—is quasi-static. Representative rolling geometries generally require three-dimensional modeling, resulting in very large models, and include nonlinear material behavior and discontinuous effects—contact and friction. Because the problem size is large and the discontinuous effects dominate the solution, the explicit dynamics approach is often less expensive computationally and more reliable than an implicit quasi-static solution technique.

The computer time involved in running a simulation using explicit time integration with a given mesh is directly proportional to the time period of the event. This is because numerical stability considerations restrict the time increment to

$$\Delta t \le \min\left(L^{el}\sqrt{\frac{\rho}{\lambda+2\mu}}\right),$$

where the minimum is taken over all elements in the mesh, L^{el} is a characteristic length associated with an element, ρ is the density of the material in the element, and λ and μ are the effective Lamé's constants for the material in the element. Since this condition effectively means that the time increment can be no larger than the time required to propagate a stress wave across an element, the computer time involved in running a quasi-static analysis can be very large. The cost of the simulation is directly proportional to the number of time increments required, $n = T/\Delta t$ if Δt remains constant, where T is the time period of the event being simulated. (Δt will not remain constant in general, since element distortion will change L^{el} and nonlinear material response will change the effective Lamé constants and density; but the assumption is acceptable for the purposes of this discussion.) Thus,

$$n = T\max\left(\frac{1}{L^{el}}\sqrt{\frac{\lambda+2\mu}{\rho}}\right).$$

To reduce n, we can speed up the simulation compared to the time of the actual process; that is, we can artificially reduce the time period of the event, T. This will introduce two possible errors. If the simulation speed is increased too much, the inertia forces will be larger and will change the predicted response (in an extreme case the problem will exhibit wave propagation response). The only way to avoid this error is to find a speedup that is not too large. The other error is that some aspects of the problem other than inertia forces—for example, material behavior—might also be rate dependent. This implies that we cannot change the actual time period of the event being modeled. But we can

see a simple equivalent—artificially increasing the material density, ρ, by a factor f^2 reduces n to n/f, just as decreasing T to T/f does. This concept, which is called "mass scaling," reduces the ratio of the event time to the time for wave propagation across an element while leaving the event time fixed, thus allowing treatment of rate-dependent material and other behaviors while having exactly the same effect on inertia forces as speeding up the time of simulation. Mass scaling is attractive because it allows us to treat rate-dependent quasi-static problems efficiently. But we cannot take it too far or we allow the inertia forces to dominate and, thus, change the solution. This example illustrates the use of mass scaling and shows how far we can take it for a practical case.

The trial and error method works well for most generic quasi-static problems; however, for rolling processes Abaqus/Explicit can set the mass scaling factor automatically based on the rolling geometry and mesh properties. An acceptable value for the stable time increment is calculated, and the appropriate mass scaling factor is applied on an element-by-element basis. The value of the stable time increment is based on the average element length in the rolling direction, the average velocity of the product through the rollers, and the number of nodes in the cross-section of the mesh.

Problem description

A steel plate of an original square cross-section of 40 mm by 40 mm and a length of 92 mm is reduced to a 30 mm height by rolling through one roll stand. The radius of the rollers is 170 mm. The single roller in the model (taking advantage of symmetry) is assumed to be rigid and is modeled as an analytical rigid surface. The isotropic hardening yield curve of the steel is taken from Kopp and Dohmen (1990). Isotropic elasticity is assumed, with Young's modulus of 150 GPa and Poisson's ratio of 0.3. The strain hardening is described using 11 points on the yield stress versus plastic strain curve, with an initial yield stress of 168.2 MPa and a maximum yield stress of 448.45 MPa. No rate dependence or temperature dependence is taken into account.

Coulomb friction is assumed between the roller and the plate, with a friction coefficient of 0.3. Friction plays an important role in this process, as it is the only mechanism by which the plate is pulled through the roll stand. If the friction coefficient is too low, the plate cannot be drawn through the roll stand. Initially, when a point on the surface of the plate has just made contact with the roller, the roller surface is moving faster than the point on the surface of the plate and there is a relative slip between the two surfaces. As the point on the plate is drawn into the process zone under the roller, it moves faster and, after a certain distance, sticks to the roller. As the point on the surface of the plate is pushed out of the process zone, it picks up speed and begins to move faster than the roller. This causes slip in the opposite direction before the point on the surface of the sheet finally loses contact with the roller.

For plane strain computations a half-symmetry model with CPE4R elements is used. For the three-dimensional computations a one-quarter symmetry model with C3D8R elements is used. The roller is modeled with analytical rigid surfaces for both the two-dimensional and three-dimensional cases. For quasi-static rolling problems perfectly round analytical surfaces can provide a more accurate representation of the revolved roller geometry, improve computational efficiency, and reduce noise when compared to element-based rigid surfaces.

The roller is rotated through 32° at a constant angular velocity of 1 revolution per second (6.28 rad/sec), which corresponds to a roller surface speed of 1.07 m/sec. The plate is given an initial velocity in the global x-direction. The initial velocity is chosen to match the x-component of velocity of the roller at the point of first contact. This choice of initial velocity results in a net acceleration of zero in the x-direction at the point of contact and minimizes the initial impact between the plate and the roller. This minimizes the initial transient disturbance.

In all but one of the analyses performed in this example, the masses of all elements in the model are scaled by factors of either 110, 2758, or 68962. These scaling factors translate into effective roller surface speeds of 11.2 m/sec, 56.1 m/sec, and 280.5 m/sec. An alternative, but equivalent, means of mass scaling could be achieved by scaling the material mass density by the aforementioned factors. In one analysis, automatic mass scaling is used.

The element formulation for the two-dimensional (using CPE4R elements) and three-dimensional (using C3D8R elements) analyses uses the pure stiffness form of hourglass control. The element formulation is selected using section controls. In addition, the three-dimensional model (using C3D8R elements) uses the centroidal kinematic

formulation. These options are economical yet provide the necessary level of accuracy for this class of problems. Two- and three-dimensional analyses using the default hourglass control formulation, the combined hourglass control formulation, and the enhanced hourglass control formulation are included for comparison. For the three-dimensional case, both the orthogonal kinematic formulation and the centroidal kinematic formulation are considered.

For the sole purpose of testing the performances of the modified triangular and tetrahedral elements, the problem is also analyzed in two dimensions using CPE6M elements and in three dimensions using C3D10M elements.

Results and discussion

Table 1 shows the effective rolling speeds and the relative CPU cost of the cases using the element formulations recommended for this problem. The relative costs are normalized with respect to the CPU time for the two-dimensional model (using CPE4R elements) with the intermediate mass scaling value. In addition, *Table 2* compares the relative CPU cost and accuracy between the different element formulations of the solid elements using the intermediate mass scaling value.

Plane strain rolling (CPE4R elements)

A plane strain calculation allows the user to resolve a number of modeling questions in two dimensions before attempting a more expensive three-dimensional calculation. In particular, an acceptable effective mass scaling factor for running the transient dynamics procedure can be determined.

Figure 1 through *Figure 3* show contours of equivalent plastic strain for the three mass scaling factors using the stiffness hourglass control. *Figure 4* through *Figure 6* show contours of shear stress for the same cases. These results show that there is very little difference between the lowest and the intermediate mass scaling cases. All the results are in good agreement with the quasi-static analysis results obtained with Abaqus/Standard. The results of the largest mass scaling case show pronounced dynamic effects. *Table 1* shows the relative run time of the quasi-static calculation, and *Table 2* compares the different element formulations at the same level of mass scaling. The intermediate mass scaling case gives essentially the same results as the quasi-static calculation, using about one-thirteenth of the CPU time. In addition to the savings provided by mass scaling, more computational savings are achieved using the stiffness hourglass control element formulation; the results for this formulation compare well to the results for the computationally more expensive element formulations.

Three-dimensional rolling (C3D8R elements)

We have ascertained with the two-dimensional calculations that mass scaling by a factor of 2758 gives results that are essentially the same as a quasi-static solution. *Figure 7* shows the distribution of the equivalent plastic strain of the deformed sheet for the three-dimensional case using the centroidal kinematic formulation and stiffness hourglass control. *Figure 8* shows the distribution of the equivalent plastic strain of the deformed sheet for the three-dimensional case using the default section controls (average strain kinematic and relax stiffness hourglass). *Table 1* compares this three-dimensional case with the plane strain, quasi-static, and three-dimensional automatic mass scaling cases; and *Table 2* compares the five different three-dimensional element formulations included here with the two-dimensional cases at the same level of mass scaling. The accuracy for all five element formulations tested is very similar for this problem, but significant savings are realized in the three-dimensional analyses when using more economical element formulations.

Analyses using CPE6M and C3D10M elements

The total number of nodes in the CPE6M model is identical to the number in the CPE4R model. The number of nodes in the C3D10M model is 3440 (compared to 3808 in the C3D8R model). The analyses using the CPE6M and C3D10M elements use a mass scaling factor of 2758. *Figure 9* and *Figure 10* show the distribution of the equivalent plastic strain of the plate for the two-dimensional and three-dimensional cases, respectively. The results are in reasonably good agreement with other element formulations. However, the CPU costs are higher since the modified triangular and tetrahedral elements use more than one integration point in each element and

the stable time increment size is somewhat smaller than in analyses that use reduced-integration elements with the same node count. For the mesh refinements used in this problem, the CPE6M model takes about twice the CPU time as the CPE4R model, while the C3D10M model takes about 5.75 times the CPU time as the C3D8R model.

Input files

roll2d330_anl_ss.inp	Two-dimensional case (using CPE4R elements) with a mass scaling factor of 2758 and the STIFFNESS hourglass control.
roll3d330_rev_anl_css.inp	Three-dimensional case (using C3D8R elements) with a mass scaling factor of 2758, an analytical rigid surface of TYPE=REVOLUTION, and the CENTROID kinematic and STIFFNESS hourglass section control options.
roll2d66_anl_ss.inp	Two-dimensional case (using CPE4R elements) with a mass scaling factor of 110 using the STIFFNESS hourglass control.
roll2d330_anl_cs.inp	Two-dimensional case (using CPE4R elements) with a mass scaling factor of 2758 using the COMBINED hourglass control.
roll2d330_anl_enhs.inp	Two-dimensional case (using CPE4R elements) with a mass scaling factor of 2758 using the ENHANCED hourglass control.
roll2d330_cs.inp	Two-dimensional case (using CPE4R elements) with a mass scaling factor of 2758 using the COMBINED hourglass control and rigid elements.
roll3d330_css.inp	Three-dimensional case (using C3D8R elements) with a mass scaling factor of 2758, rigid elements, and the CENTROID kinematic and STIFFNESS hourglass section control options.
roll3d330_css_gcont.inp	Three-dimensional case (using C3D8R elements) with a mass scaling factor of 2758, rigid elements, the CENTROID kinematic and STIFFNESS hourglass section control options, and the general contact capability.
roll3d330_ocs.inp	Three-dimensional case (using C3D8R elements) with a mass scaling factor of 2758, rigid elements, and the ORTHOGONAL kinematic and COMBINED hourglass section control options.
roll3d330_ocs_gcont.inp	Three-dimensional case (using C3D8R elements) with a mass scaling factor of 2758, rigid elements, the ORTHOGONAL kinematic and COMBINED hourglass section control options, and the general contact capability.
roll2d1650_anl_ss.inp	Two-dimensional case (using CPE4R elements) with a mass scaling factor of 68962 using STIFFNESS hourglass control.
roll3d330_rev_anl_ocs.inp	Three-dimensional model (using C3D8R elements) with a mass scaling factor of 2758, an analytical rigid surface of TYPE=REVOLUTION, and the ORTHOGONAL kinematic and COMBINED hourglass section control options.
roll3d330_rev_anl_oenhs.inp	Three-dimensional model (using C3D8R elements) with a mass scaling factor of 2758, an analytical rigid surface of TYPE=REVOLUTION, and the ORTHOGONAL kinematic and ENHANCED hourglass section control options.
roll3d330_rev_anl_cenhs.inp	Three-dimensional model (using C3D8R elements) with a mass scaling factor of 2758, an analytical rigid surface of TYPE=REVOLUTION, and the CENTROID kinematic and ENHANCED hourglass section control options.
roll3d330_rev_anl.inp	Three-dimensional model (using C3D8R elements) with a mass scaling factor of 2758, an analytical rigid surface of TYPE=REVOLUTION, and the default section control options.
roll3d_auto_rev_anl_css.inp	Three-dimensional case (using C3D8R elements) with automatic mass scaling, an analytical rigid surface of TYPE=REVOLUTION, and the CENTROID kinematic and STIFFNESS hourglass section control options.

roll3d330_cyl_anl.inp	Three-dimensional model (using C3D8R elements) with a mass scaling factor of 2758, an analytical rigid surface of TYPE=CYLINDER, and the default section control options.
roll2d66.inp	Two-dimensional model (using CPE4R elements) with a mass scaling factor of 110 and the default section controls.
roll2d330.inp	Two-dimensional model (using CPE4R elements) with a mass scaling factor of 2758 and the default section controls.
roll2d1650.inp	Two-dimensional model (using CPE4R elements) with a mass scaling factor of 68962 and the default section controls.
roll3d330.inp	Three-dimensional model using rigid elements and the default section controls.
roll3d330_gcont.inp	Three-dimensional model using rigid elements, the default section controls, and the general contact capability.
roll2d66_anl.inp	Two-dimensional model (using CPE4R elements) with a mass scaling factor of 110, analytical rigid surfaces, and the default section controls.
roll2d330_anl.inp	Two-dimensional model (using CPE4R elements) with a mass scaling factor of 2758, analytical rigid surfaces, and the default section controls.
roll2d1650_anl.inp	Two-dimensional model (using CPE4R elements) with a mass scaling factor of 68962, analytical rigid surfaces, and the default section controls.
roll2d_impl_qs.inp	Implicit, quasi-static, two-dimensional model (using CPE4R elements) with analytical rigid surfaces.
roll2d330_anl_cpe6m.inp	Two-dimensional case (using CPE6M elements) with a mass scaling factor of 2758.
roll3d330_anl_c3d10m.inp	Three-dimensional case (using C3D10M elements) with a mass scaling factor of 2758.
roll3d_medium.inp	Additional mesh refinement case (using C3D8R elements) included for the sole purpose of testing the performance of the code.
roll3d_medium_gcont.inp	Additional mesh refinement case (using C3D8R elements) with the general contact capability.
roll3d_medium_gcont_cms.inp	Additional mesh refinement case (using C3D8R elements) with the general contact capability and contact mass scaling activated.

References

- Kobayashi, S., S. I. Oh, and T. Altan, Metal Forming and the Finite Element Method, Oxford University Press, 1989.
- Kopp, R., and P. M. Dohmen, "Simulation und Planung von Walzprozessen mit Hilfe der Finite-Elemente-Methode (FEM)," *Stahl U. Eisen*, no. 7, pp. 131–136, 1990.

Tables

Table 1: Analysis cases and relative CPU costs. (The two-dimensional explicit analyses all use CPE4R elements and stiffness hourglass control. The three-dimensional explicit analyses use C3D8R elements and the centroidal kinematic and stiffness hourglass section controls.)

Analysis Type	Mass Scaling Factor	Effective Roll Surface Speed (m/sec)	Relative CPU Time
Explicit, plane strain	110.3	11.2	4.99
Explicit, plane strain	2758.5	56.1	1.00

Analysis Type	Mass Scaling Factor	Effective Roll Surface Speed (m/sec)	Relative CPU Time
Explicit, plane strain	68961.8	280.5	0.21
Implicit, plane strain		quasi-static	13.4
Explicit, 3D	2758.5	56.1	13.8
Explicit, 3D	automatic	~96	9.5

Table 2: Explicit section controls tested (mass scaling factor=2758.5). CPE4R and C3D8R elements are employed for the two-dimensional and three-dimensional cases, respectively. Spread values are reported for the half-model at node 24015.

Analysis Type	Section Controls		Relative CPU Time	Spread (mm)
	Kinematic	Hourglass		
Explicit, plane strain	n/a	stiffness	1.00	n/a
Explicit, plane strain	n/a	relax stiffness	1.11	n/a
Explicit, plane strain	n/a	combined	1.04	n/a
Explicit, plane strain	n/a	enhanced	1.02	n/a
Explicit, 3D	average strain	relax stiffness	20.8	2.06
Explicit, 3D	orthogonal	combined	17.1	2.07
Explicit, 3D	centroidal	stiffness	13.8	2.10
Explicit, 3D	centroidal	enhanced	14.8	2.10
Explicit, 3D	orthogonal	enhanced	17.3	2.10

Figures

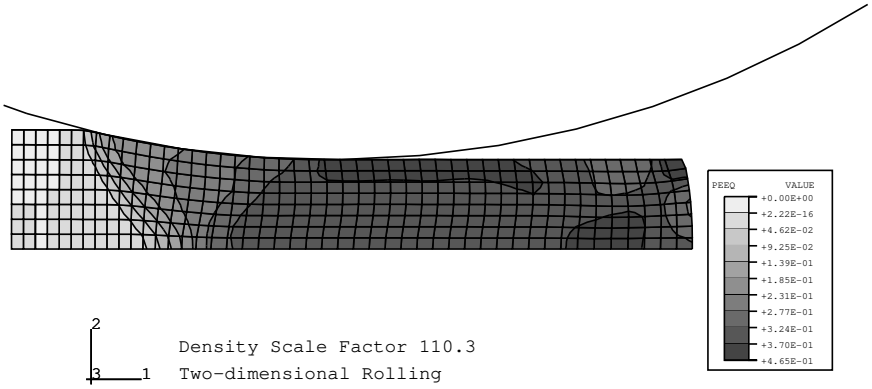


Figure 1: Equivalent plastic strain for the plane strain case (CPE4R) with stiffness hourglass control (mass scaling factor=110.3).

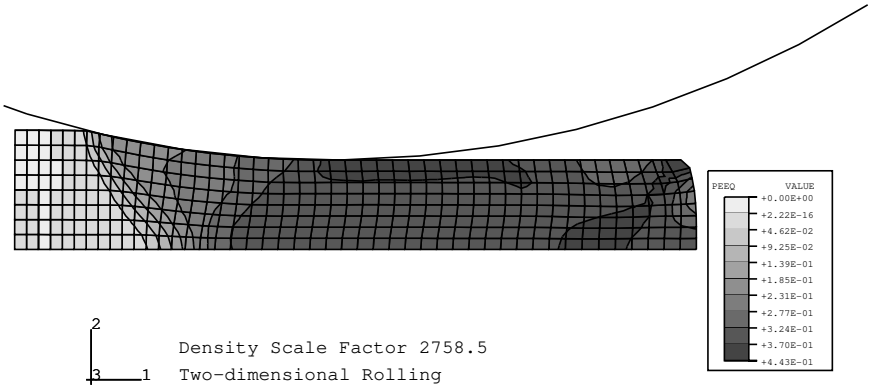


Figure 2: Equivalent plastic strain for the plane strain case (CPE4R) with stiffness hourglass control (mass scaling factor=2758.5).

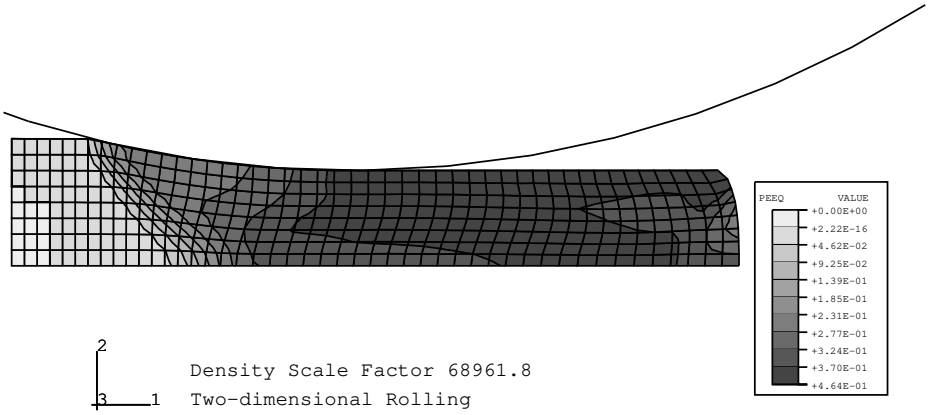


Figure 3: Equivalent plastic strain for the plane strain case (CPE4R) with stiffness hourglass control (mass scaling factor=68961.8).

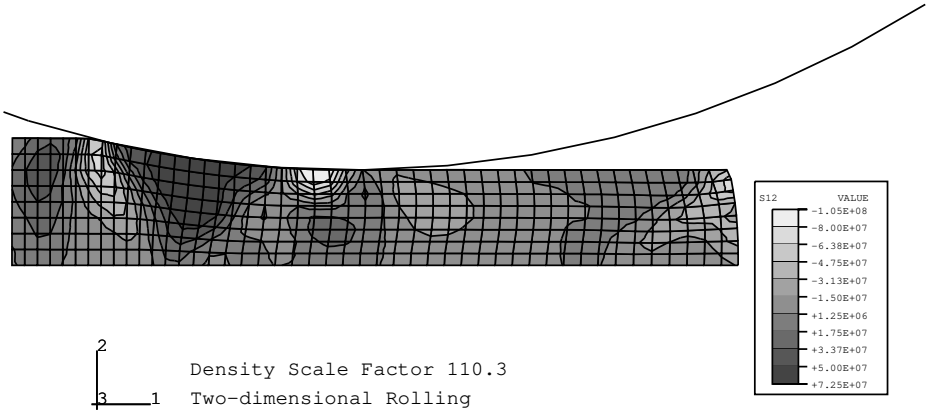


Figure 4: Shear stress for the plane strain case (CPE4R) with stiffness hourglass control (mass scaling factor=110.3).

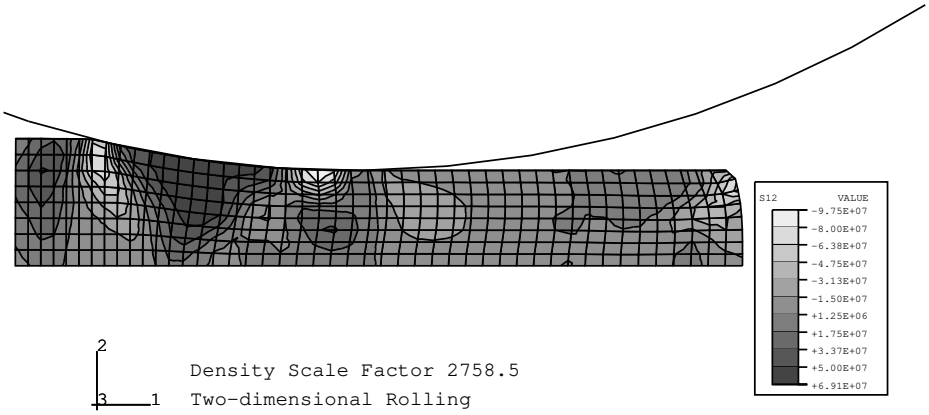


Figure 5: Shear stress for the plane strain case (CPE4R) with stiffness hourglass control (mass scaling factor=2758.5).

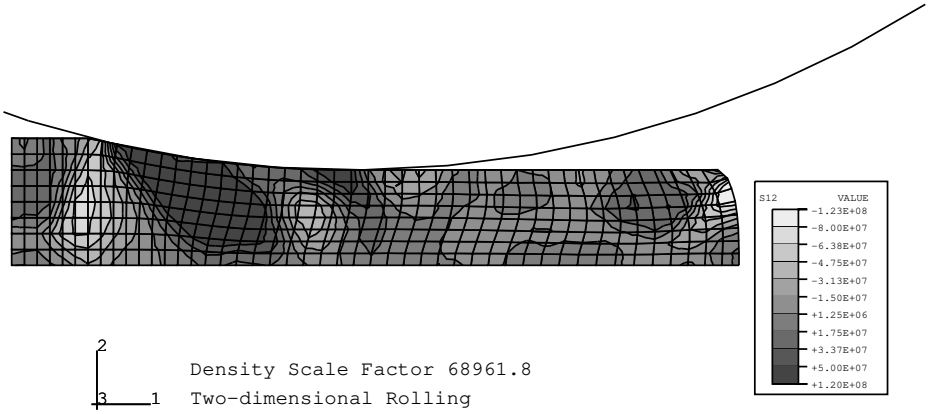


Figure 6: Shear stress for the plane strain case (CPE4R) with stiffness hourglass control (mass scaling factor=68961.8).

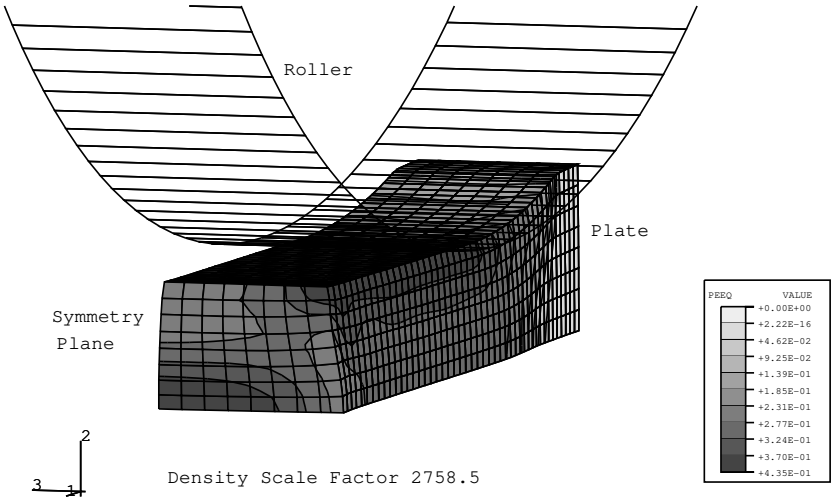


Figure 7: Equivalent plastic strain for the three-dimensional case (C3D8R) using the centroidal kinematic and stiffness hourglass section controls (mass scaling factor=2758.5).

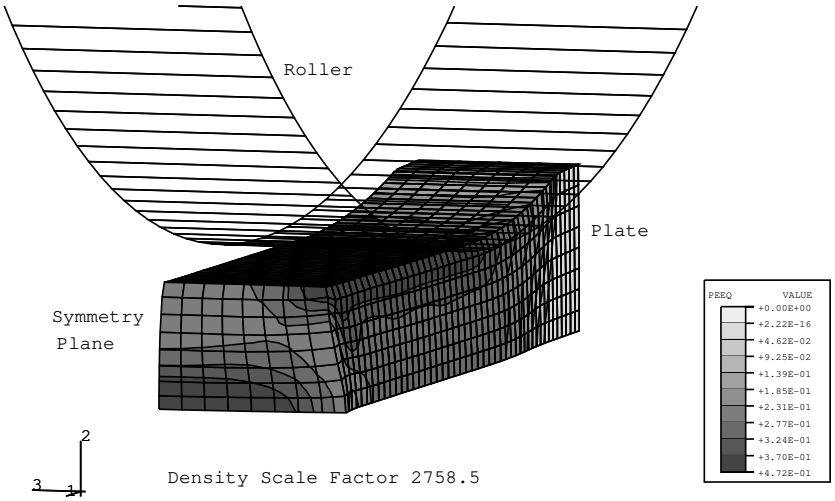


Figure 8: Equivalent plastic strain for the three-dimensional case (C3D8R) using the average strain kinematic and relax stiffness hourglass section controls (mass scaling factor=2758.5).

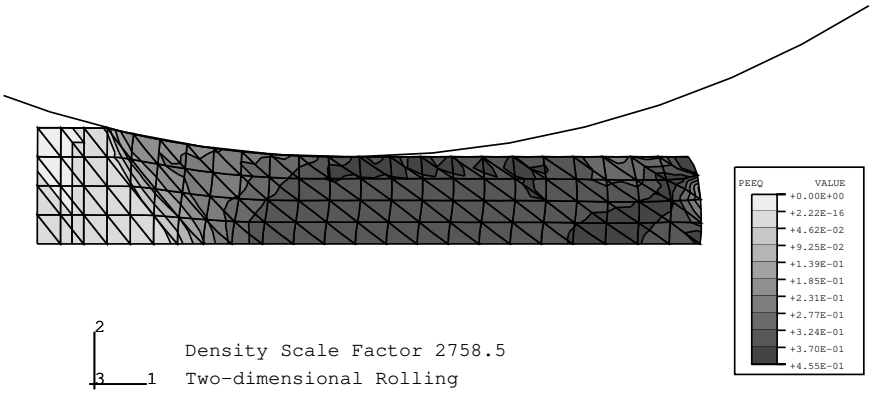


Figure 9: Equivalent plastic strain for the plane strain case (CPE6M) (mass scaling factor=2758.5).

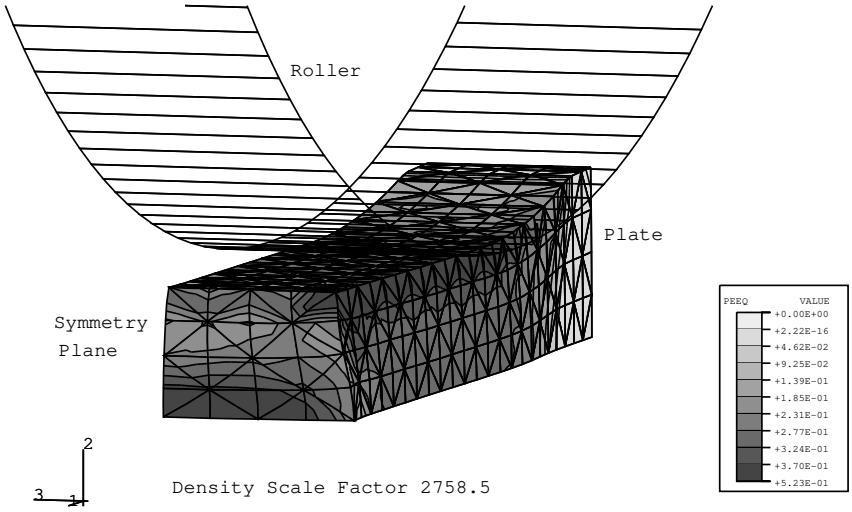


Figure 10: Equivalent plastic strain for the three-dimensional case (C3D10M) (mass scaling factor=2758.5).

Axisymmetric forming of a circular cup

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example shows the hydroforming of a circular cup using an axisymmetric model.
In this example a two-stage forming sequence is used, with annealing between the stages. Two analysis methods are used: in one the entire process is analyzed using Abaqus/Explicit; in the other the forming sequences are analyzed with Abaqus/Explicit, while the springback analyses are run in Abaqus/Standard. Here, the import capability is used to transfer results between Abaqus/Explicit and Abaqus/Standard and vice versa.

Problem description

The model consists of a deformable blank and three rigid dies. The blank has a radius of 150.0 mm, is 1.0 mm thick, and is modeled using axisymmetric shell elements, SAX1. The coefficient of friction between the blank and the dies is taken to be 0.1. Dies 1 and 2 are offset from the blank by half of the thickness of the blank, because the contact algorithm takes into account the shell thickness. To avoid pinching of the blank while die 3 is put into position for the second forming stage, the radial gap between dies 2 and 3 is set to be 20% bigger than the initial shell thickness. *Figure 1* and *Figure 2* show the initial geometry of the model.

The three dies are modeled with either two-dimensional analytical rigid surfaces or RAX2 rigid elements. An analytical rigid surface can yield a more accurate representation of two-dimensional curved punch geometries and result in computational savings. Contact pressure can be viewed on the specimen surface, and the reaction force is available at the rigid body reference node. In addition, both the kinematic (default) and penalty contact formulations are tested. Results for the kinematic contact formulation using rigid elements are presented here.

The blank is made of aluminum-killed steel, which is assumed to satisfy the Ramberg-Osgood relation between true stress and logarithmic strain,

$$\epsilon = (\sigma/K)^{1/n},$$

with a reference stress value (K) of 513 MPa and work-hardening exponent (n) of 0.223. Isotropic elasticity is assumed, with Young's modulus of 211 GPa and Poisson's ratio of 0.3. With these data an initial yield stress of 91.3 MPa is obtained. The stress-strain behavior is defined by piecewise linear segments matching the Ramberg-Osgood curve up to a total (logarithmic) strain level of 107%, with Mises yield, isotropic hardening, and no rate dependence.

The analysis that is performed entirely within Abaqus/Explicit consists of six steps. In the first step contact is defined between the blank and dies 1 and 2. Both dies remain fixed while a distributed load of 10 MPa in the negative z-direction is ramped onto the blank. This load is then ramped off in the second step, allowing the blank to spring back to an equilibrium state.

The third step is an annealing step. The annealing procedure in Abaqus/Explicit sets all appropriate state variables to zero. These variables include stresses, strains (excluding the thinning strain for shells, membranes, and plane stress elements), plastic strains, and velocities. There is no time associated with an annealing step. The process occurs instantaneously.

In the fourth step contact is defined between the blank and die 3 and contact is removed between the blank and die 1. Die 3 moves down vertically in preparation for the next pressure loading.

In the fifth step another distributed load is applied to the blank in the positive z-direction, forcing the blank into die 3. This load is then ramped off in the sixth step to monitor the springback of the blank.

To obtain a quasi-static response, an investigation was conducted to determine the optimum rate for applying the pressure loads and removing them. The optimum rate balances the computational time against the accuracy of the results; increasing the loading rate will reduce the computer time but lead to less accurate quasi-static results.

The analysis that uses the import capability consists of four runs. The first run is identical to Step 1 of the Abaqus/Explicit analysis described earlier. In the second run the Abaqus/Explicit results for the first forming stage are imported into Abaqus/Standard without updating the reference configuration and with an import of the material state for the first springback analysis. The third run imports the results of the first springback analysis into Abaqus/Explicit for the subsequent annealing process and the second forming stage. By updating the reference configuration and not importing the material state, this run begins with no initial stresses or strains, effectively simulating the annealing process. The final run imports the results of the second forming stage into Abaqus/Standard for the second springback analysis.

Results and discussion

Figure 3 to *Figure 5* show the results of the analysis conducted entirely within Abaqus/Explicit using the rigid element approach and the kinematic contact formulation. *Figure 3* shows the deformed shape at the end of Step 2, after the elastic springback. *Figure 4* shows the deformed shape at the end of the analysis, after the second elastic springback. Although it is not shown here, the amount of springback observed during the unloading steps is negligible. *Figure 5* shows a contour plot of the shell thickness (STH) at the end of the analysis. The thickness of the material at the center of the cup has been reduced by about 20%, while the thickness at the edges of the cup has been increased by about 10%.

The results obtained using the import capability to perform the springback analyses in Abaqus/Standard are nearly identical, as are those obtained using analytical rigid surfaces and/or penalty contact formulations.

You can use the **abaqus restartjoin** execution procedure to extract data from the output database created by a restart analysis and append the data to a second output database. For more information, see *Joining Output Database (.odb) Files from Restarted Analyses*.

Input files

axiform.inp	Abaqus/Explicit analysis that uses rigid elements and kinematic contact. This file is also used for the first step of the analysis that uses the import capability.
axiform_anl.inp	Model using analytical rigid surfaces and kinematic contact.
axiform_pen.inp	Model using rigid elements and penalty contact.
axiform_anl_pen.inp	Model using analytical rigid surfaces and penalty contact.
axiform_sprbk1.inp	First springback analysis using the import capability.
axiform_form2.inp	Second forming analysis using the import capability.
axiform_sprbk2.inp	Second springback analysis using the import capability.
axiform_restart.inp	Restart of axiform.inp included for the purpose of testing the restart capability.
axiform_rest_anl.inp	Restart of axiform_anl.inp included for the purpose of testing the restart capability.

Figures

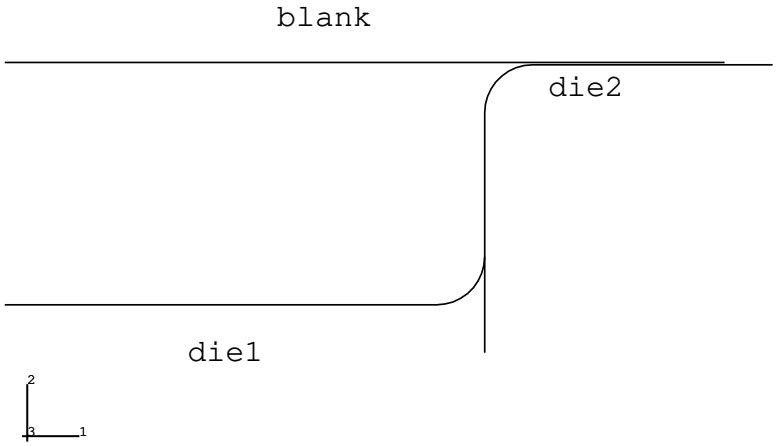


Figure 1: Configuration at the beginning of stage 1.

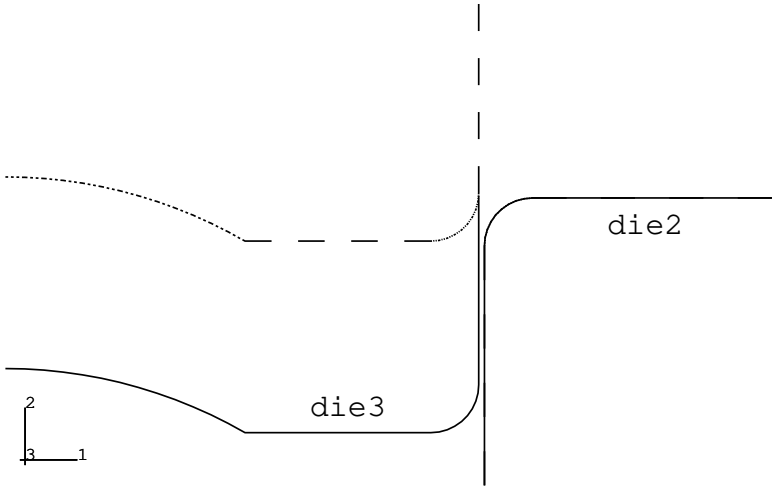


Figure 2: Configuration of dies in forming stage 2. (The dotted line shows the initial position of die 3.)



Figure 3: Deformed configuration after the first forming stage.

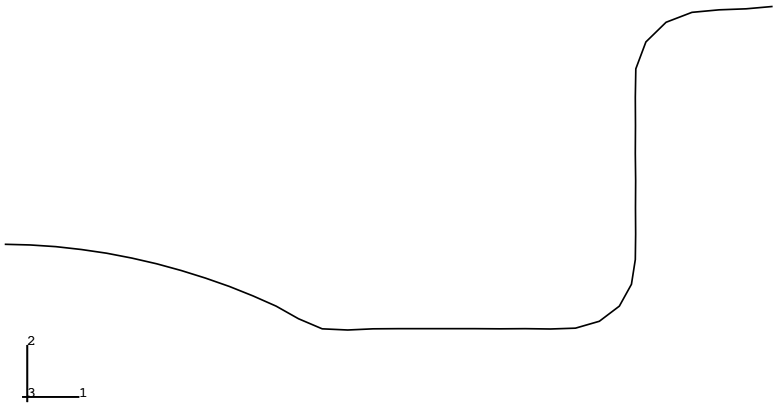


Figure 4: Final configuration.

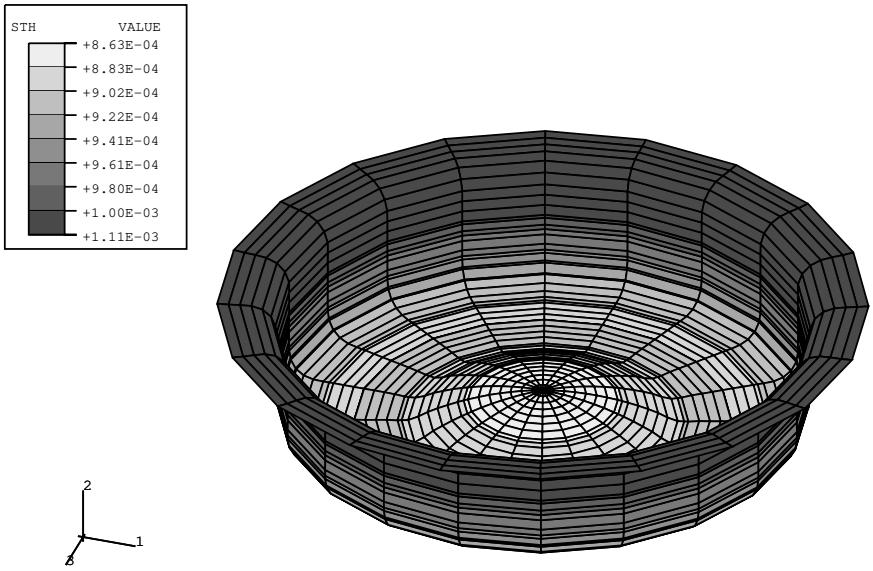


Figure 5: Contour plot of shell thickness.

Cup/trough forming

Products: Abaqus/Explicit

Overview

This example shows the use of adaptive meshing in forging problems that include large amounts of shearing at the tool-blank interface; a cup and a trough are formed.

Problem description

Three different geometric models are considered, as shown in *Figure 1*. Each model consists of a rigid punch, a rigid die, and a deformable blank. The outer top and bottom edges of the blank are cambered, which facilitates the flow of material against the tools. The punch and die have semicircular cross-sections; the punch has a radius of 68.4 mm, and the die has a radius of 67.9 mm. The blank is modeled as a von Mises elastic, perfectly plastic material with a Young's modulus of 4000 MPa and a yield stress of 5 MPa. The Poisson's ratio is 0.21; the density is 1.E−4 tonne/mm^3.

In each case the punch is moved 61 mm, while the die is fully constrained. A smooth amplitude curve is used to ramp the punch velocity to a maximum, at which it remains constant. The smoothing of the velocity promotes a quasi-static response to the loading.

Case 1: Axisymmetric model for cup forming

The blank is meshed with CAX4R elements and measures 50 × 64.77 mm. The punch and the die are modeled as analytical rigid surfaces using connected line segments. Symmetry boundary conditions are prescribed at r=0. The finite element model is shown in *Figure 2*.

Case 2: Three-dimensional model for trough forming

The blank is meshed with C3D8R elements and measures 50 × 64.7 × 64.7 mm. The punch and the die are modeled as three-dimensional cylindrical analytical rigid surfaces. Symmetry boundary conditions are applied at the x=0 and z=0 planes. The finite element model of the blank is shown in *Figure 3*.

Case 3: Three-dimensional model for cup forming

The blank is meshed with C3D8R elements. A 90° wedge of the blank with a radius of 50 mm and a height of 64.7 mm is analyzed. The punch and the die are modeled as three-dimensional revolved analytical rigid surfaces. Symmetry boundary conditions are applied at the x=0 and y=0 planes. The finite element model of the blank is shown in *Figure 4*.

Adaptive meshing

A single adaptive mesh domain that incorporates the entire blank is used for each model. Symmetry planes are defined as Lagrangian boundary regions (the default), and contact surfaces are defined as sliding boundary regions (the default). Since this problem is quasi-static with relatively small amounts of deformation per increment, the default values for frequency, mesh sweeps, and other adaptive mesh parameters and controls are sufficient.

Results and discussion

Figure 5 through *Figure 7* show the mesh configuration at the end of the forging simulation for Cases 1–3. In each case a quality mesh is maintained throughout the simulation. As the blank flattens out, geometric edges and corners that exist at the beginning of the analysis are broken and adaptive meshing is allowed across them. The eventual breaking of geometric edges and corners is essential for this type of problem to minimize element distortion and optimize element aspect ratios.

For comparison purposes *Figure 8* shows the deformed mesh for a pure Lagrangian simulation of Case 1 (the axisymmetric model). The mesh is clearly better when continuous adaptive meshing is used. Several diamond-shaped elements with extremely poor aspect ratios are formed in the pure Lagrangian simulation. Adaptive meshing improves the element quality significantly, especially along the top surface of the cup where solution gradients are highest. *Figure 9* and *Figure 10* show contours of equivalent plastic strain at the completion of the forging for the adaptive meshing and pure Lagrangian analyses of Case 1, respectively. Overall plastic strains compare quite closely. Slight differences exist only along the upper surface, where the pure Lagrangian mesh becomes very distorted at the end of the simulation. The time histories of the vertical punch force for the adaptive and pure Lagrangian analyses agree closely for the duration of the forging, as shown in *Figure 11*.

Input files

ale_cupforming_axi.inp	Case 1.
ale_cupforming_axinodes.inp	External file referenced by Case 1.
ale_cupforming_axielements.inp	External file referenced by Case 1.
ale_cupforming_cyl.inp	Case 2.
ale_cupforming_sph.inp	Case 3.
lag_cupforming_axi.inp	Lagrangian solution of Case 1.

Figures

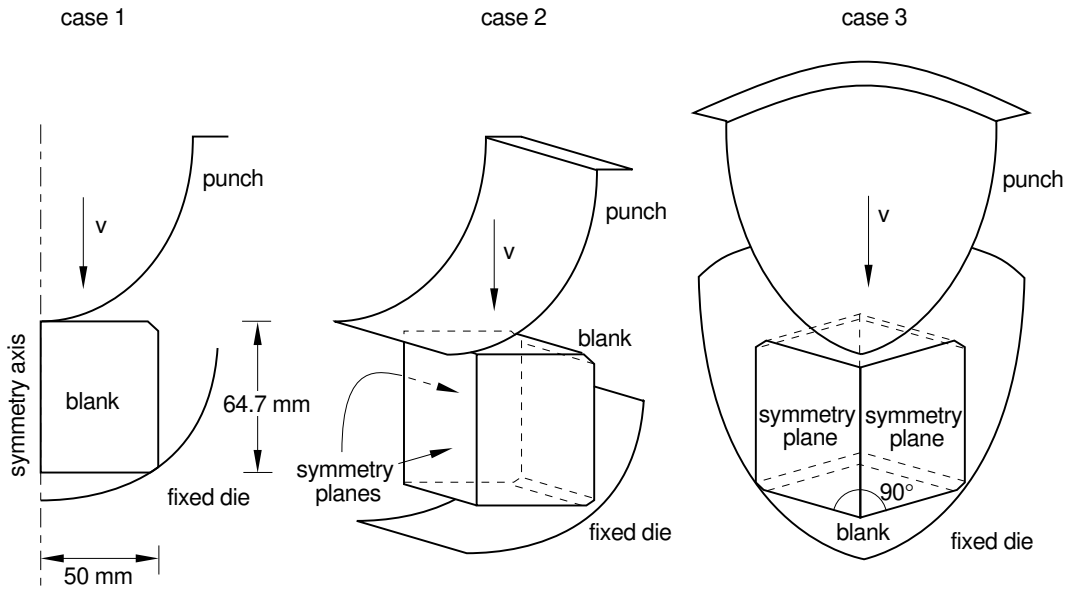


Figure 1: Model geometries for each case.

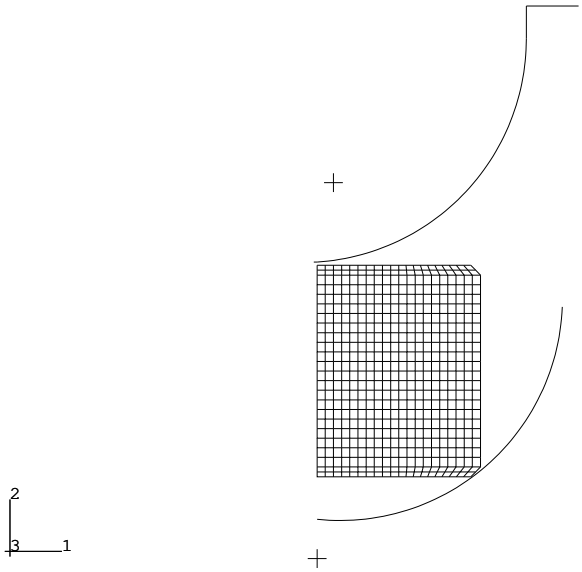


Figure 2: Undeformed mesh for Case 1.

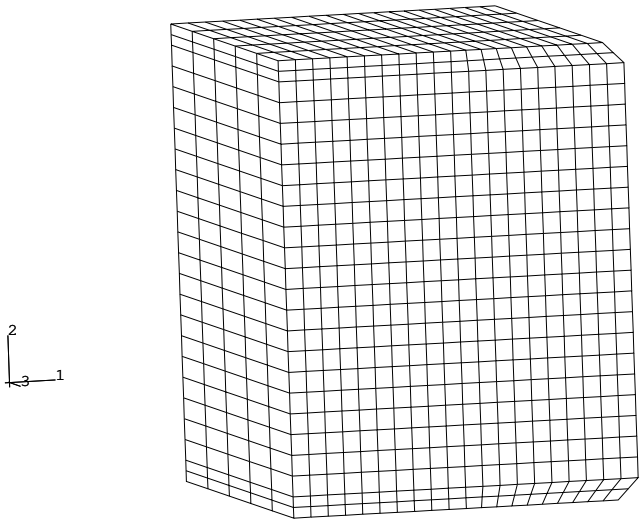


Figure 3: Undeformed mesh for Case 2.

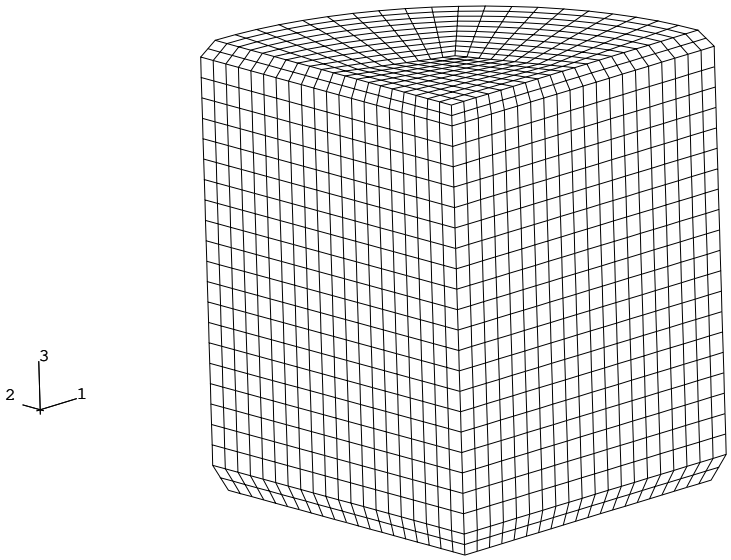


Figure 4: Undeformed mesh for Case 3.

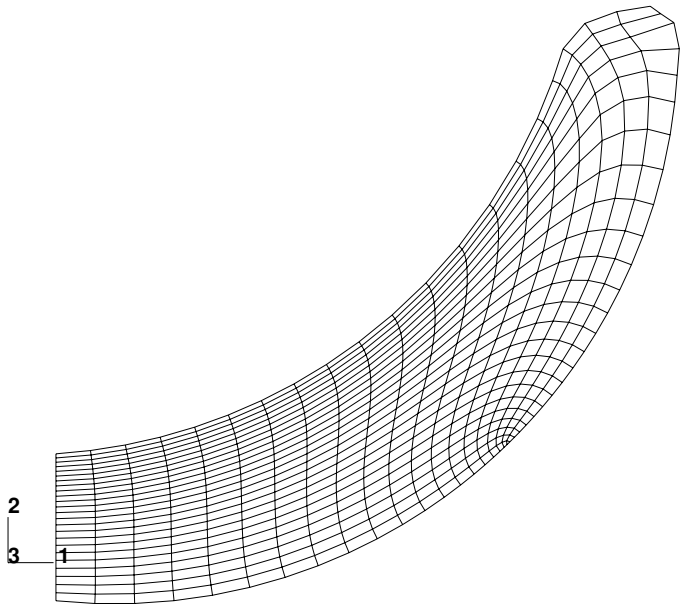


Figure 5: Deformed mesh for Case 1.

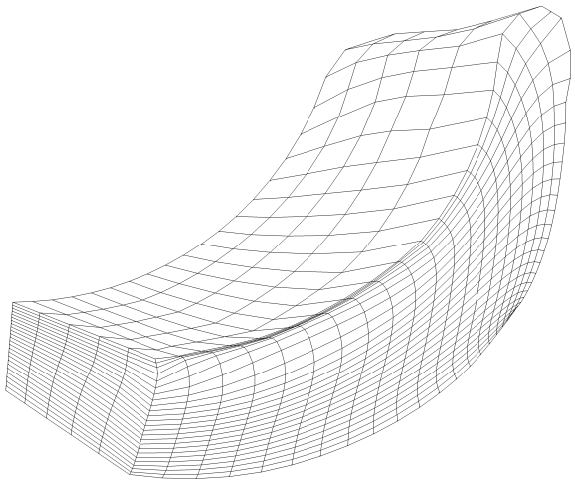


Figure 6: Deformed mesh for Case 2.

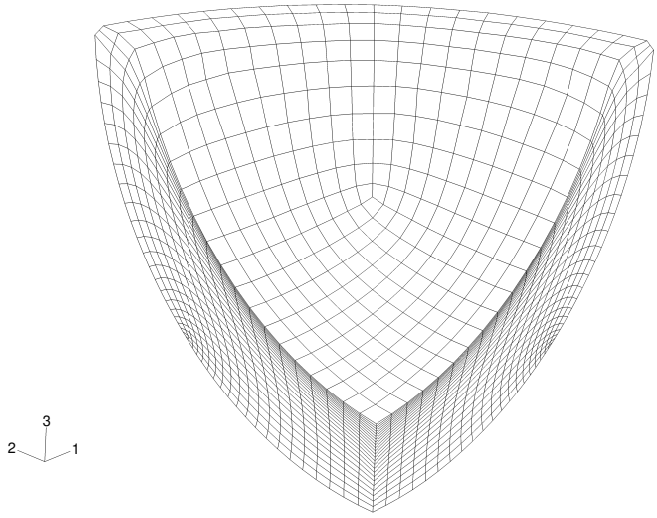


Figure 7: Deformed mesh for Case 3.

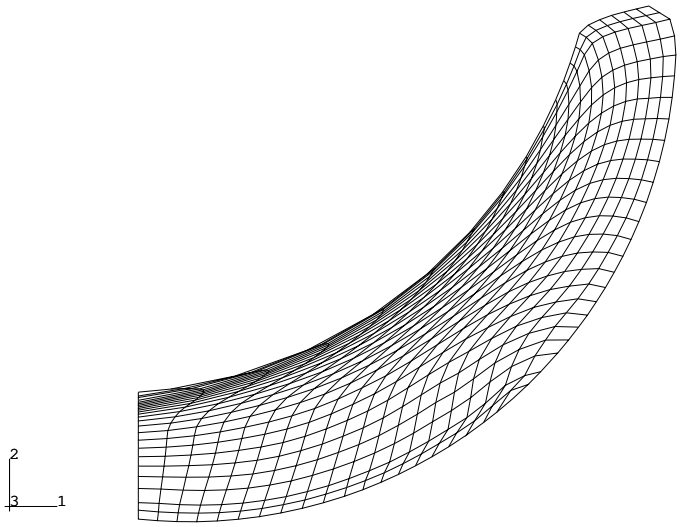


Figure 8: Deformed mesh for Case 1 using a pure Lagrangian formulation.

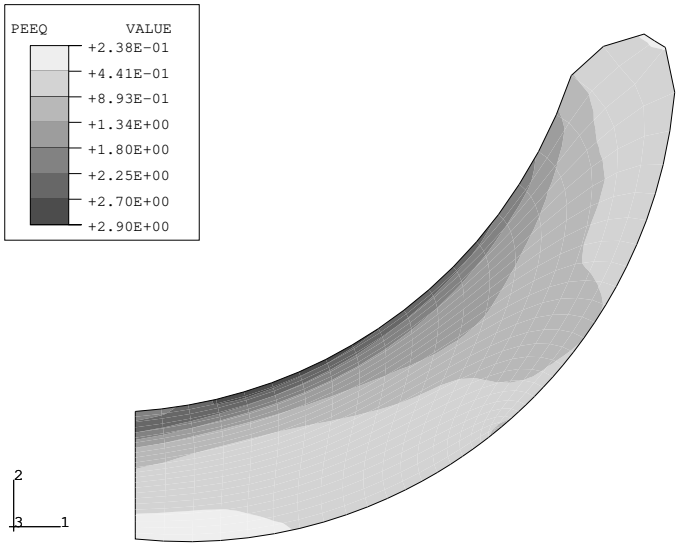


Figure 9: Contours of equivalent plastic strain for Case 1 using adaptive meshing.

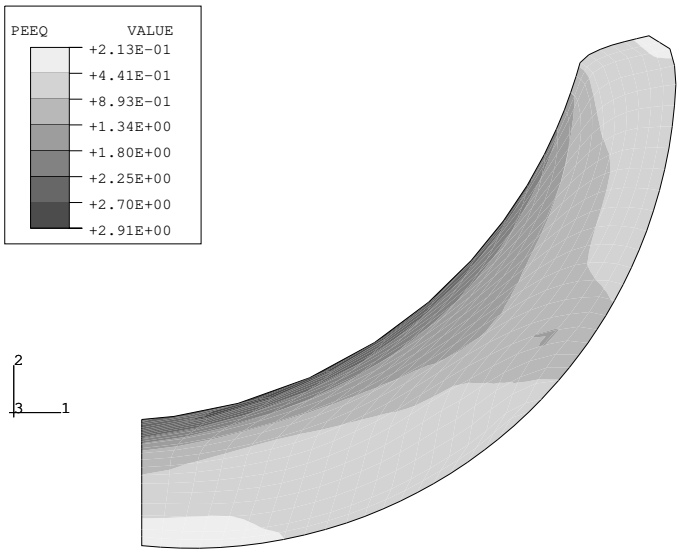


Figure 10: Contours of equivalent plastic strain for Case 1 using a pure Lagrangian fomulation.

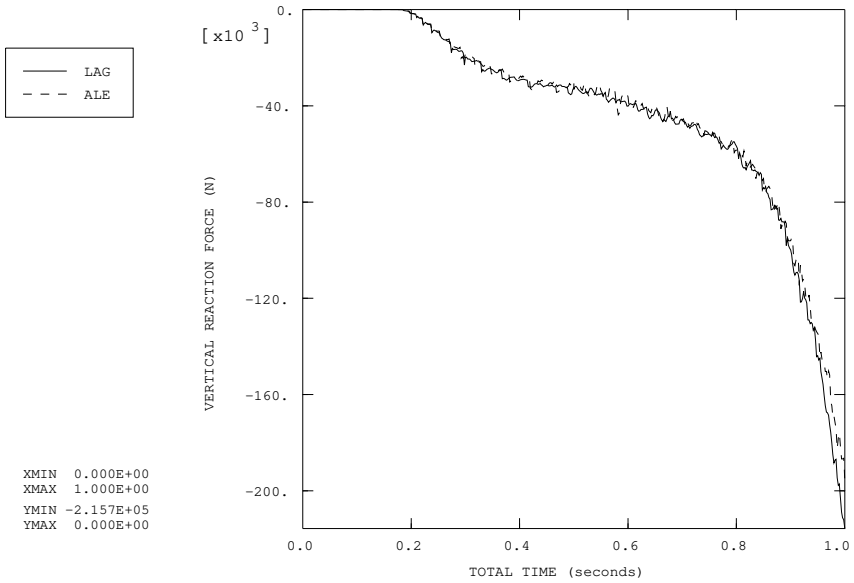


Figure 11: Comparison of time histories for the vertical punch force for Case 1.

Forging with sinusoidal dies

Products: Abaqus/Explicit

Overview

This example shows the use of adaptive meshing in forging problems that incorporate geometrically complex dies and involve substantial material flow.

Problem description

Three different geometric models are considered, as shown in *Figure 1*. Each model consists of a rigid die and a deformable blank. The cross-sectional shape of the die is sinusoidal with an amplitude and a period of 5 and 10 mm, respectively. The blank is steel and is modeled as a von Mises elastic-plastic material with a Young's modulus of 200 GPa, an initial yield stress of 100 MPa, and a constant hardening slope of 300 MPa. Poisson's ratio is 0.3; the density is 7800 kg/m^3.

In all cases the die is moved downward vertically at a velocity of 20000 mm/sec and is constrained in all other degrees of freedom. The total die displacement is 7.6 mm for Case 1, 6.7 mm for Case 2, and 5.6 mm for Case 3. These displacements represent the maximum possible given the refinement and topology of the initial mesh (if the quality of the mesh is retained for the duration of the analysis). Although each analysis uses a sinusoidal die, the geometries and flow characteristics of the blank material are quite different for each problem.

Case 1: Axisymmetric model

The blank is meshed with CAX4R elements and measures 20 × 10 mm. The dies are modeled as analytical rigid surfaces comprised of connected line segments. The bottom of the blank is constrained in the z-direction, and symmetry boundary conditions are prescribed at r=0. The initial configuration of the blank and the die is shown in *Figure 2*.

Case 2: Three-dimensional model

The blank is meshed with C3D8R elements and measures 20 × 10 × 10 mm. The dies are modeled as three-dimensional cylindrical analytical rigid surfaces. The bottom of the blank is constrained in the y-direction, and symmetry boundary conditions are applied at the x=0 and z=10 planes. The finite element model of the blank and the die is shown in *Figure 3*.

Case 3: Three-dimensional model

The blank is meshed with C3D8R elements and measures 20 × 10 × 20 mm. The dies are modeled as three-dimensional revolved analytical rigid surfaces. The bottom of the blank is constrained in the y-direction, and symmetry boundary conditions are applied at the x=0 and z=10 planes. The finite element model of the blank and the die is shown in *Figure 4*. The revolved die is displaced upward in the figure from its initial position for clarity.

Adaptive meshing

A single adaptive mesh domain that incorporates the entire blank is used for each model. Symmetry planes are defined as Lagrangian boundary regions (the default), and contact surfaces are defined as sliding boundary regions (the default). Because the material flow for each of the geometries is substantial, the frequency and the intensity of adaptive meshing must be increased to provide an accurate solution. The frequency at which adaptive

meshing is to be performed is reduced from the default of 10 to 5 for all cases. The default number of 1 mesh sweep is used for case 2, and this number is increased to 3 for case 1 and case 3.

Results and discussion

Figure 5 and *Figure 6* show the deformed mesh and contours of equivalent plastic strain at the completion of the forming step for Case 1. Adaptive meshing maintains reasonable element shapes and aspect ratios. This type of forging problem cannot typically be solved using a pure Lagrangian formulation. *Figure 7* shows the deformed mesh for Case 2. A complex, doubly curved deformation pattern is formed on the free surface as the material spreads under the die. Element distortion appears to be reasonable. *Figure 8* and *Figure 9* show the deformed mesh and contours of equivalent plastic strain for Case 3. Although the die is a revolved geometry, the three-dimensional nature of the blank gives rise to fairly complex strain patterns that are symmetric with respect to the planes of quarter symmetry.

Input files

ale_sinusoid_forgingaxi.inp	Case 1.
ale_sinusoid_forgingaxisurf.inp	External file referenced by Case 1.
ale_sinusoid_forgingcyl.inp	Case 2.
ale_sinusoid_forgingrev.inp	Case 3.

Figures

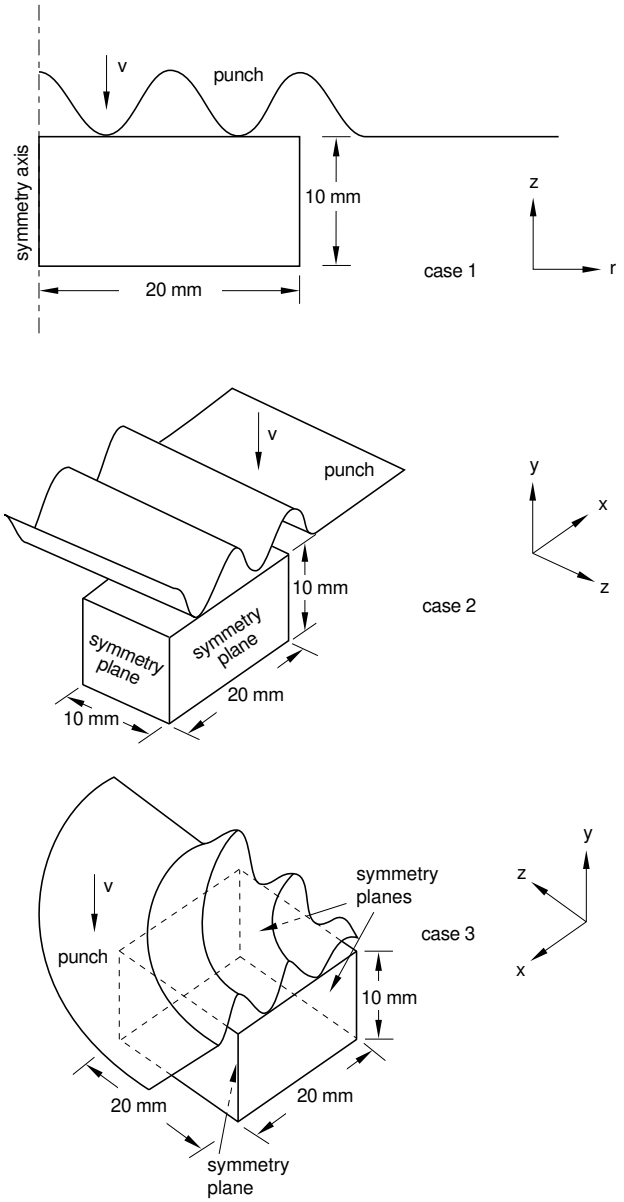


Figure 1: Model geometries for each of the three cases.

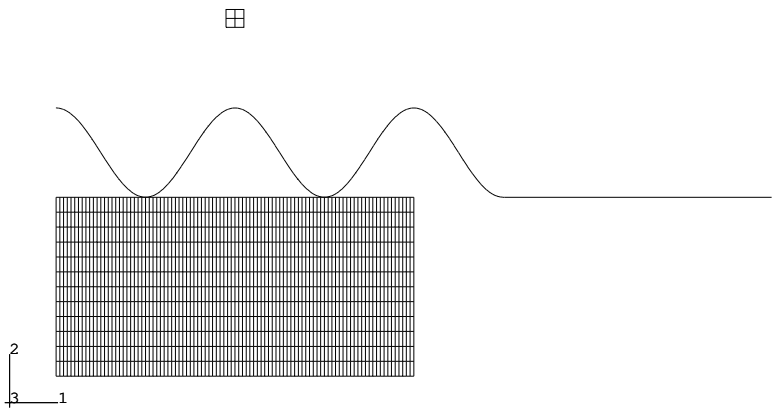


Figure 2: Initial configuration for Case 1.

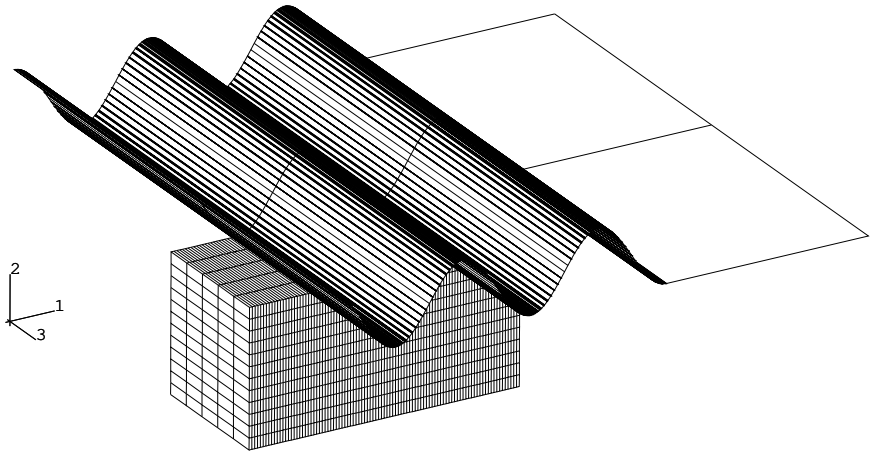


Figure 3: Initial configuration for Case 2.

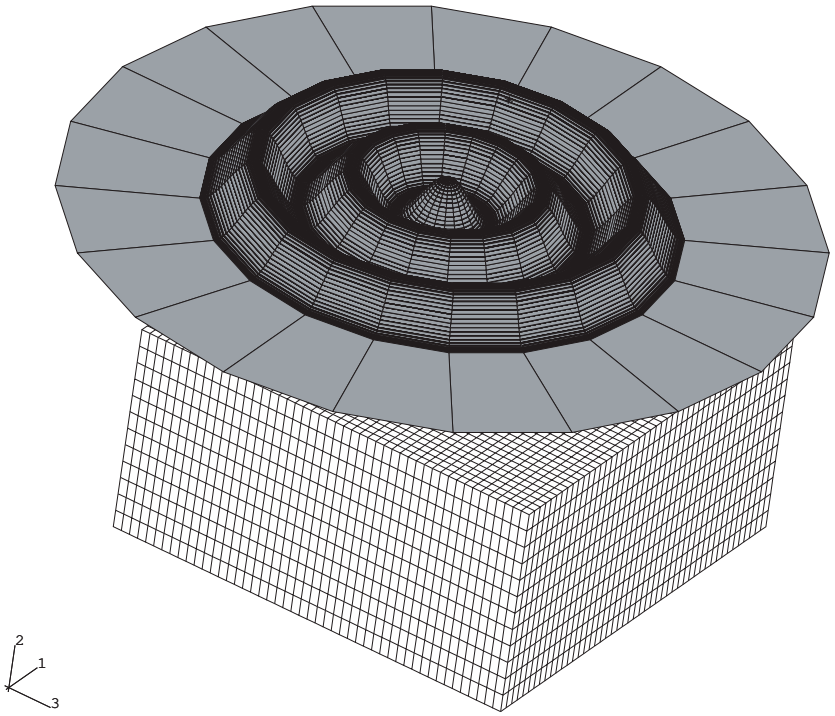


Figure 4: Initial configuration for Case 3.

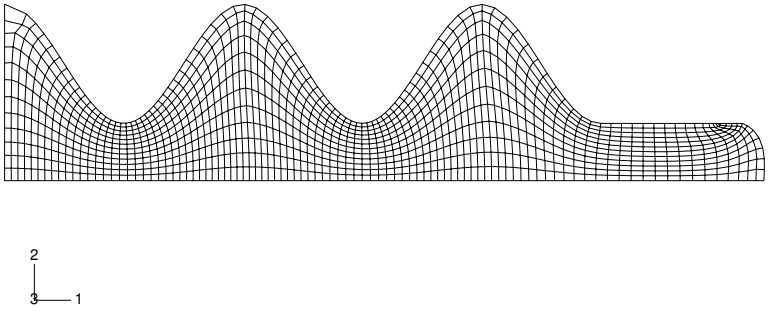


Figure 5: Deformed mesh for Case 1.

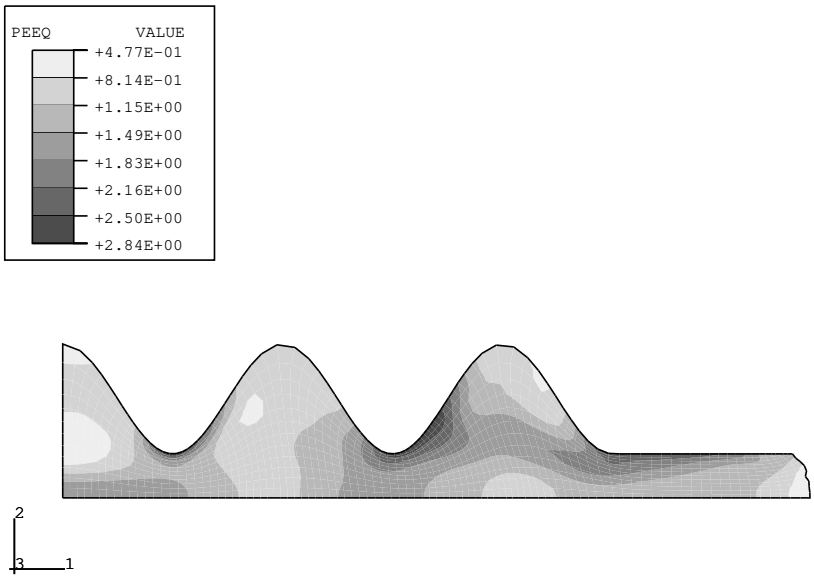


Figure 6: Contours of equivalent plastic strain for Case 1.

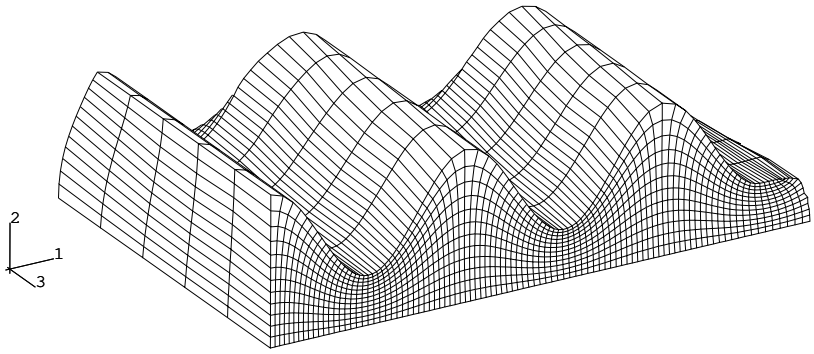


Figure 7: Deformed mesh for Case 2.

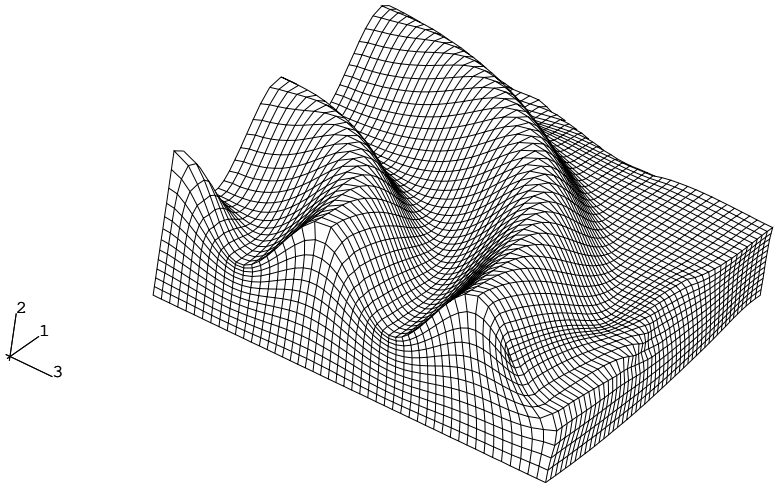


Figure 8: Deformed mesh for Case 3.

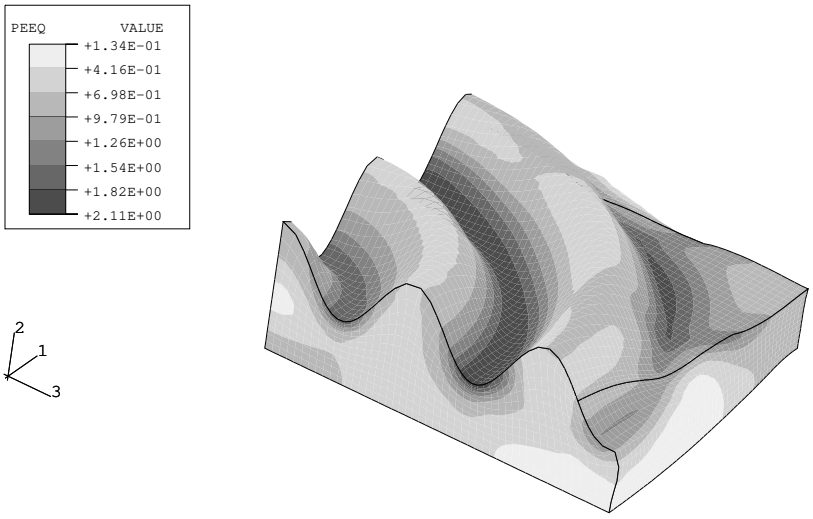


Figure 9: Contours of equivalent plastic strain for Case 3.

Forging with multiple complex dies

Products: Abaqus/Explicit

Overview

This example illustrates the use of adaptive meshing in forging problems that use multiple geometrically complex dies. The problem is based on a benchmark presented at the "FEM–Material Flow Simulation in the Forging Industry" workshop.

Problem description

The benchmark problem is an axisymmetric forging, but in this example both axisymmetric and three-dimensional geometric models are considered. For the axisymmetric models the default hourglass formulation (HOURGLASS=RELAX STIFFNESS) and the enhanced strain hourglass formulation (HOURGLASS=ENHANCED) are considered. For the three-dimensional geometric models the pure stiffness hourglass formulation (HOURGLASS=STIFFNESS) and the enhanced strain hourglass formulation with the orthogonal kinematic formulation (KINEMATIC SPLIT=ORTHOGONAL) are considered. Each model is shown in *Figure 1*. Both models consist of two rigid dies and a deformable blank. The blank's maximum radial dimension is 895.2 mm, and its thickness is 211.4 mm. The outer edge of the blank is rounded to facilitate the flow of material through the dies. The blank is modeled as a von Mises elastic-plastic material with a Young's modulus of 200 GPa, an initial yield stress of 360 MPa, and a constant hardening slope of 30 MPa. The Poisson's ratio is 0.3; the density is 7340 kg/m^3.

Both dies are fully constrained, with the exception of the top die, which is moved 183.4 mm downward at a constant velocity of 166.65 mm/s.

Case 1: Axisymmetric model

The blank is meshed with CAX4R elements. A fine discretization is required in the radial direction because of the geometric complexity of the dies and the large amount of material flow that occurs in that direction. Symmetry boundary conditions are prescribed at r=0. The dies are modeled as TYPE=SEGMENTS analytical rigid surfaces. The initial configuration is shown in *Figure 2*.

Case 2: Three-dimensional model

The blank is meshed with C3D8R elements. A 90° wedge of the blank is analyzed. The level of mesh refinement is the same as that used in the axisymmetric model. Symmetry boundary conditions are applied at the x=0 and z=0 planes. The dies are modeled as TYPE=REVOLUTION analytical rigid surfaces. The initial configuration of the blank only is shown in *Figure 3*. Although the tools are not shown in the figure, they are originally in contact with the blank.

Adaptive meshing

A single adaptive mesh domain that incorporates the entire blank is used for each model. Symmetry planes are defined as Lagrangian boundary regions (the default), and contact surfaces are defined as sliding boundary regions (the default). Since this problem is quasi-static with relatively small amounts of deformation per increment, the defaults for frequency, mesh sweeps, and other adaptive mesh parameters and controls are sufficient.

Results and discussion

Figure 4 and *Figure 5* show the deformed mesh for the axisymmetric case using the default hourglass control formulation (HOURGLASS=RELAX STIFFNESS) at an intermediate stage ($t =$ 0.209 s) and in the final configuration ($t =$ 0.35 s), respectively. The elements remain well shaped throughout the entire simulation, with the exception of the elements at the extreme radius of the blank, which become very coarse as material flows radially during the last 5% of the top die's travel. *Figure 6* shows contours of equivalent plastic strain at the completion of forming. *Figure 7* and *Figure 8* show the deformed mesh for the three-dimensional case using the pure stiffness hourglass control (HOURGLASS=STIFFNESS) and the orthogonal kinematic formulation (KINEMATIC SPLIT=ORTHOGONAL) at $t =$ 0.209 and $t =$ 0.35, respectively. Although the axisymmetric and three-dimensional mesh smoothing algorithms are not identical, the elements in the three-dimensional model also remain well shaped until the end of the analysis, when the same behavior that is seen in the two-dimensional model occurs. Contours of equivalent plastic strain for the three-dimensional model (not shown) are virtually identical to those shown in *Figure 6*.

Input files

ale_duckshape_forgingaxi.inp	Case 1 using the default hourglass formulation (HOURGLASS=RELAX STIFFNESS).
ale_duckshape_forgingaxi_enhs.inp	Case 1 using the enhanced strain hourglass formulation (HOURGLASS=ENHANCED).
ale_duckshape_forg_axind.inp	External file referenced by the Case 1 analyses.
ale_duckshape_forg_axiel.inp	External file referenced by the Case 1 analyses.
ale_duckshape_forg_axiset.inp	External file referenced by the Case 1 analyses.
ale_duckshape_forg_axirs.inp	External file referenced by the Case 1 analyses.
ale_duckshape_forgingrev.inp	Case 2 using the pure stiffness hourglass formulation (HOURGLASS=STIFFNESS) and the orthogonal kinematic formulation (KINEMATIC SPLIT=ORTHOGONAL).
ale_duckshape_forgingrev_oenhs.inp	Case 2 using the enhanced strain hourglass formulation (HOURGLASS=ENHANCED) and the orthogonal kinematic formulation (KINEMATIC SPLIT=ORTHOGONAL).

References

- Industrieverband Deutscher Schmieden e.V. (IDS), “Forging of an Axisymmetric Disk,” FEM–Material Flow Simulation in the Forging Industry, Hagen, Germany, October 1997.

Figures

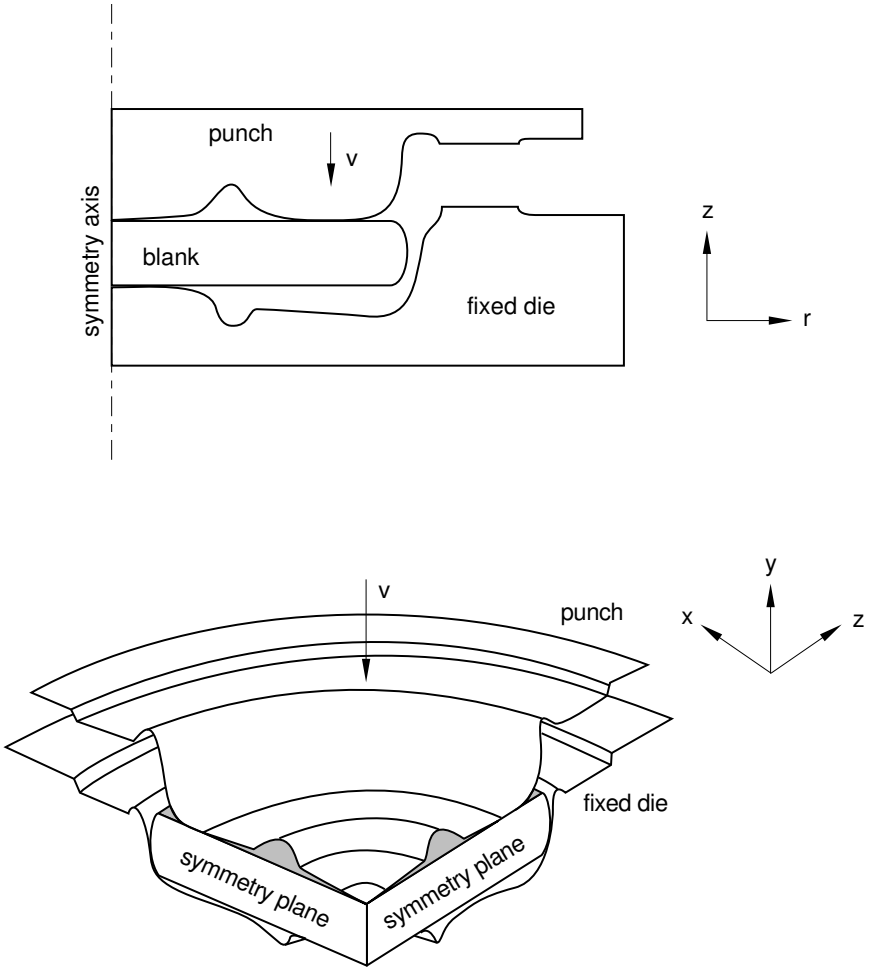


Figure 1: Axisymmetric and three-dimensional model geometries.

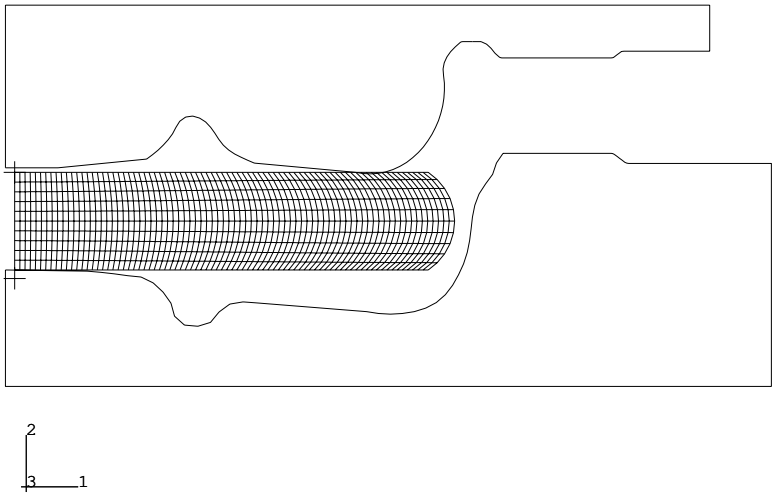


Figure 2: Initial configuration for the axisymmetric model.

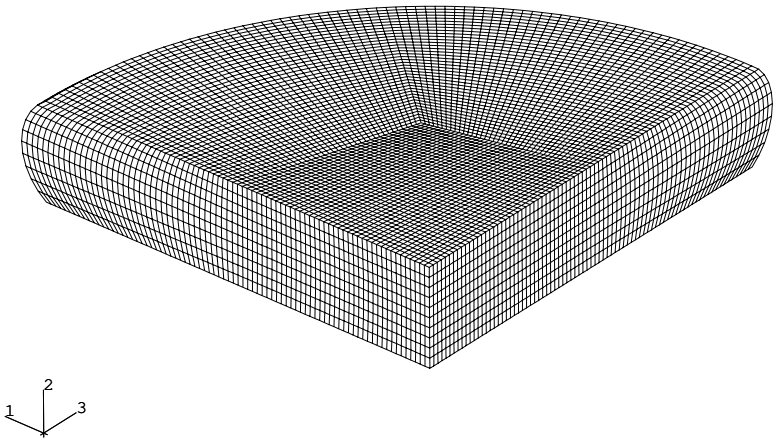


Figure 3: Initial configuration mesh for the three-dimensional model.

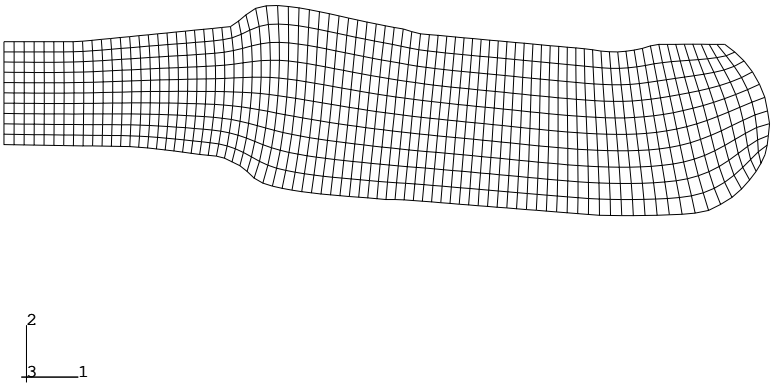


Figure 4: The deformed mesh for the axisymmetric model using the default hourglass formulation (HOURGLASS=RELAX STIFFNESS) at an intermediate stage.

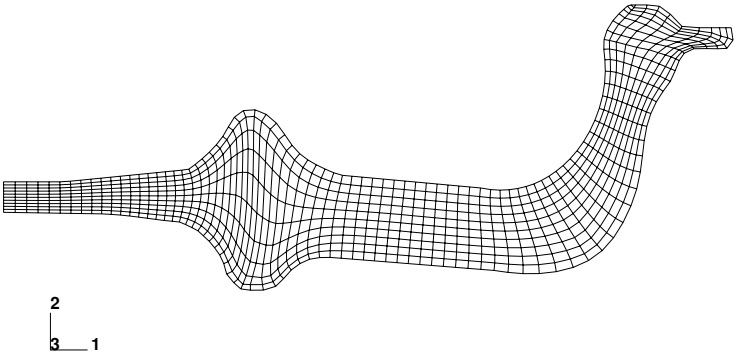


Figure 5: The deformed mesh for the axisymmetric model using the default hourglass formulation (HOURGLASS=RELAX STIFFNESS) at the end of forming.

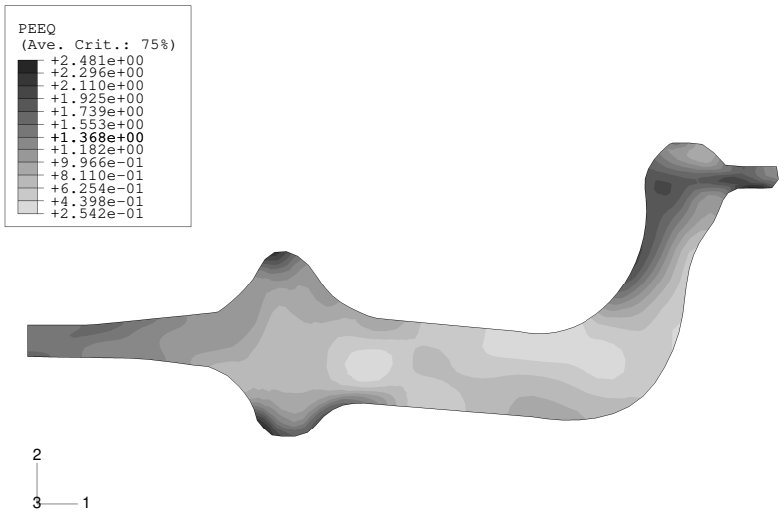


Figure 6: Contours of equivalent plastic strain for the axisymmetric model using the default hourglass formulation (HOURGLASS=RELAX STIFFNESS) at the end of forming.

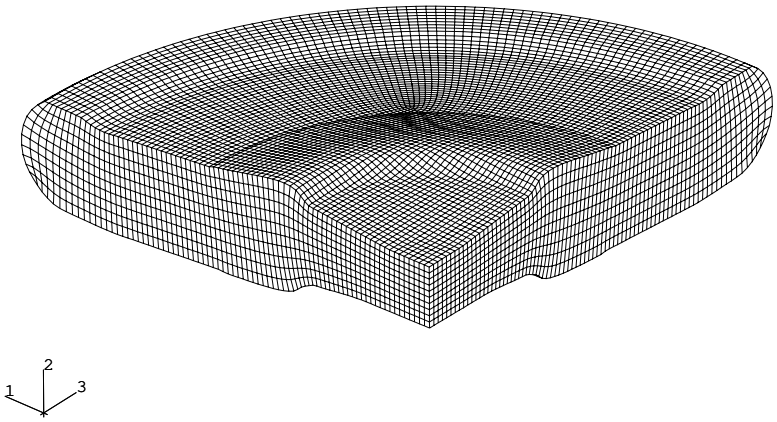


Figure 7: The deformed mesh for the three-dimensional model using the pure stiffness hourglass formulation (HOURGLASS=STIFFNESS) and the orthogonal kinematic formulation (KINEMATIC SPLIT=ORTHOGONAL) at an intermediate stage.

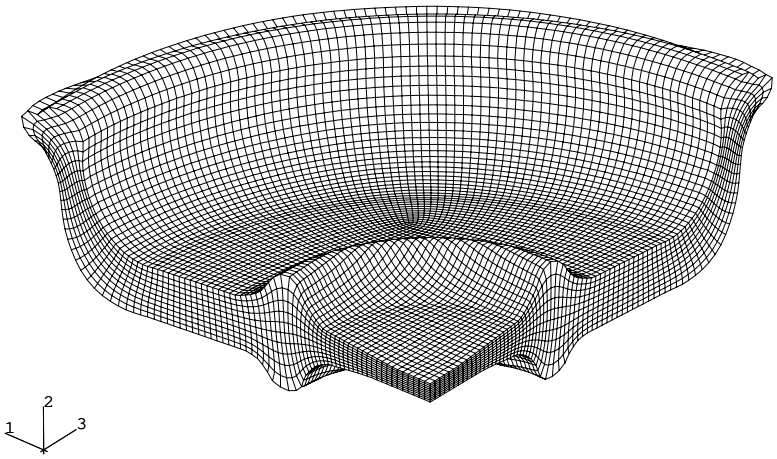


Figure 8: The deformed mesh for the three-dimensional model using the pure stiffness hourglass formulation (HOURGLASS=STIFFNESS) and the orthogonal kinematic formulation (KINEMATIC SPLIT=ORTHOGONAL) at the end of forming.

Flat rolling: transient and steady-state

Products: Abaqus/Explicit

Overview

This example illustrates the use of adaptive meshing to simulate a rolling process using both transient and steady-state approaches.

A transient flat rolling simulation, as shown in *Figure 1*, is performed using three different methods: a "pure" Lagrangian approach, an adaptive meshing approach using a Lagrangian domain, and a mixed Eulerian-Lagrangian adaptive meshing approach in which material upstream from the roller is drawn from an Eulerian inflow boundary but the downstream end of the blank is handled in a Lagrangian manner. In addition, a steady-state flat rolling simulation is performed using an Eulerian adaptive mesh domain as a control volume and defining inflow and outflow Eulerian boundaries. Solutions using each approach are compared.

Problem description

For each analysis case quarter symmetry is assumed; the model consists of a rigid roller and a deformable blank. The blank is meshed with C3D8R elements. The cylindrical roller is modeled as an analytical rigid surface. The radius of the cylinder is 175 mm. Symmetry boundary conditions are prescribed on the right (z=0 plane) and bottom (y=0 plane) faces of the blank.

Coulomb friction with a friction coefficient of 0.3 is assumed between the roller and the plate. All degrees of freedom are constrained on the roller except rotation about the z-axis, where a constant angular velocity of 6.28 rad/sec is defined. For each analysis case the blank is given an initial velocity of 0.3 m/s in the x-direction to initiate contact.

The blank is steel and is modeled as a von Mises elastic-plastic material with isotropic hardening. The Young's modulus is 150 GPa, and the initial yield stress is 168.2 MPa. The Poisson's ratio is 0.3; the density is 7800 kg/m^3. The masses of all blank elements are scaled by a factor of 2750 at the beginning of the step so that the analysis can be performed more economically. This scaling factor represents an approximate upper bound on the mass scaling possible for this problem, above which significant inertial effects would be generated.

The criteria for stopping the rolling analyses based on the achievement of a steady-state condition is defined. The criteria used require the satisfaction of the steady-state detection norms of equivalent plastic strain, spread, force, and torque within the default tolerances. The exit plane for each norm is defined as the plane passing through the center of the roller with the normal to the plane coincident with the rolling direction. For Case 1 through Case 3 the steady-state detection norms are evaluated as each plane of elements passes the exit plane. Case 4 requires uniform sampling since the initial mesh is roughly stationary due to the initial geometry and the inflow and outflow Eulerian boundaries.

The finite element models used for each analysis case are shown in *Figure 2*. A description of each model and the adaptive meshing techniques used follows:

Case 1: Transient simulation—pure Lagrangian approach

The blank is initially rectangular and measures 224 × 20 × 50 mm. No adaptive meshing is performed. The analysis is run until steady-state conditions are achieved.

Case 2: Transient simulation—Lagrangian adaptive mesh domain

The finite element model for this case is identical to that used for Case 1, with the exception that a single adaptive mesh domain that incorporates the entire blank is defined to allow continuous adaptive meshing. Symmetry planes are defined as Lagrangian surfaces (the default), and the contact surface on the blank is defined as a sliding surface (the default). The analysis is run until steady-state conditions are achieved.

Case 3: Transient simulation—mixed Eulerian-Lagrangian approach

This analysis is performed on a relatively short initial blank measuring $65 \times 20 \times 50$ mm. Material is continuously drawn by the action of the roller on the blank through an inflow Eulerian boundary defined on the upstream end. The blank is meshed with the same number of elements as in Cases 1 and 2 so that similar aspect ratios are obtained as the blank lengthens and steady-state conditions are achieved.

An adaptive mesh domain is defined that incorporates the entire blank. Because it contains at least one Eulerian surface, this domain is considered Eulerian for the purpose of setting parameter defaults. However, the analysis model has both Lagrangian and Eulerian aspects. The amount of material flow with respect to the mesh will be large at the inflow end and small at the downstream end of the domain. To account for the Lagrangian motion of the downstream end, change the adaptive mesh controls for this problem so that adaptive meshing is performed based on the positions of the nodes at the start of the current adaptive mesh increment. To mesh the inflow end accurately and to perform the analysis economically, the frequency is set to 5 and the number of mesh sweeps is set to 5.

As in Case 2, symmetry planes are defined as Lagrangian boundary regions (the default), and the contact surface on the blank is defined as a sliding boundary region (the default). In addition, the boundary on the upstream end is defined as an Eulerian surface. Adaptive mesh constraints are defined on the Eulerian surface using adaptive mesh constraints to hold the inflow surface mesh completely fixed while material is allowed to enter the domain normal to the surface. An equation constraint is used to ensure that the velocity normal to the inflow boundary is uniform across the surface. The velocity of nodes in the direction tangential to the inflow boundary surface is constrained.

Case 4: Steady-state simulation—Eulerian adaptive mesh domain

This analysis employs a control volume approach in which material is drawn from an inflow Eulerian boundary and is pushed out through an outflow boundary by the action of the roller. The blank geometry for this analysis case is defined such that it approximates the shape corresponding to the steady-state solution: this geometry can be thought of as an “initial guess” to the solution. The blank initially measures 224 mm in length and 50 mm in width and has a variable thickness such that it conforms to the shape of the roller. The surface of the blank transverse to the rolling direction is not adjusted to account for the eventual spreading that will occur in the steady-state solution. Actually, any reasonable initial geometry will reach a steady state, but geometries that are closer to the steady-state geometry often allow a solution to be obtained in a shorter period of time.

As in the previous two cases an adaptive mesh domain is defined on the blank, symmetry planes are defined as Lagrangian surfaces (the default), and the contact surface is defined as a sliding surface (the default). Inflow and outflow Eulerian surfaces are defined on the ends of the blank using the same techniques as in Case 3, except that for the outflow boundary adaptive mesh constraints are applied only normal to the boundary surface and no material constraints are applied tangential to the boundary surface.

To improve the computational efficiency of the analysis, the frequency of adaptive meshing is increased to every fifth increment because the Eulerian domain undergoes very little overall deformation and the material flow speed is much less than the material wave speed. This frequency will cause the mesh at Eulerian boundaries to drift slightly. However, the amount of drift is extremely small and does not accumulate. There is no need to increase the mesh sweeps because this domain is relatively stationary and, by default, adaptive meshing is performed based on the nodal positions of the original mesh. Very little mesh smoothing is required.

Results and discussion

The final deformed configurations of the blank for each of the three transient cases are shown in *Figure 3*. The transient cases have reached a steady-state solution and have been terminated based on the criteria given in the steady-state detection definition. Steady-state conditions are determined to have been reached when the reaction forces and moments on the roller have stabilized and the cross-sectional shape and distribution of equivalent plastic strain under the roller become constant over time. When using a steady-state detection definition, these conditions imply that the force, moment, spread, and equivalent plastic strain norms have stabilized such that the changes in the norms over three consecutive sampling intervals have fallen below the user-prescribed tolerances. See *Steady-State Detection* for a detailed discussion on the definition of the norms. Contours of equivalent plastic strain for each of the three transient cases are in good agreement and are shown in the final configuration of each blank in *Figure 4*. *Figure 5* shows the initial and final mesh configurations at steady state. With the exception of Case 3 all analyses were terminated using the default steady-state norm tolerances. Case 3 required that the force and torque norm tolerances be increased from .005 to .01 due to the force and torque at the roller being rather noisy.

To compare the results from the transient and steady-state approaches, the steady-state detection norms are summarized for each case in *Table 1*. The table shows a comparison of the values of the steady-state detection norms after the analyses have been terminated. The only significant difference is in the value of the spread norm for Case 4, which is higher than the others. The spread norm is defined as the largest of the second principle moments of inertia of the workpiece's cross-section. Since the spread norm is a cubic function of the lateral deformation of the workpiece, rather small differences in displacements between the test cases can lead to significant differences in the spread norms.

Time history plots of the steady-state detection norms are also shown. *Figure 9* and *Figure 10* show time history plots of the steady-state force and torque norms, respectively, for all cases. The force and torque norms are essentially running averages of the force and moment on the roller and show good agreement for all four test cases. *Figure 7* and *Figure 8* show time history plots of the steady-state equivalent plastic strain and spread norms, respectively, for all cases. The equivalent plastic strains norms are in good agreement for all cases.

Input files

lag_flatrolling.inp	Case 1 with contact pairs.
lag_flatrolling_gcont.inp	Case 1 with general contact.
ale_flatrolling_noeuler.inp	Case 2.
ale_flatrolling_inlet.inp	Case 3.
ale_flatrolling_inletoutlet.inp	Case 4.

Tables

Table 1: Comparison of steady-state detection norms.

Formulation	Spread norm	Effective plastic strain norm	Force norm	Torque norm
Case 1	1.349 E–7	.8037	–1.43 E6	3.59 E4
Case 2	1.369 E–7	.8034	–1.43 E6	3.55 E4
Case 3	1.365 E–7	.8018	–1.43 E6	3.61 E4
Case 4	1.485 E–7	.8086	–1.40 E6	3.65 E4

Figures

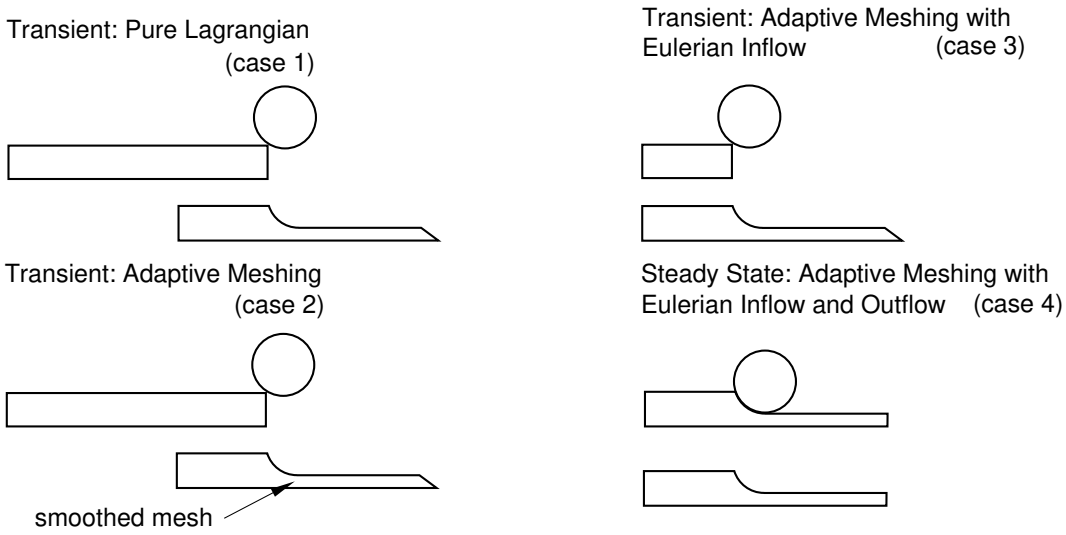


Figure 1: Diagram illustrating the four analysis approaches used in this problem.

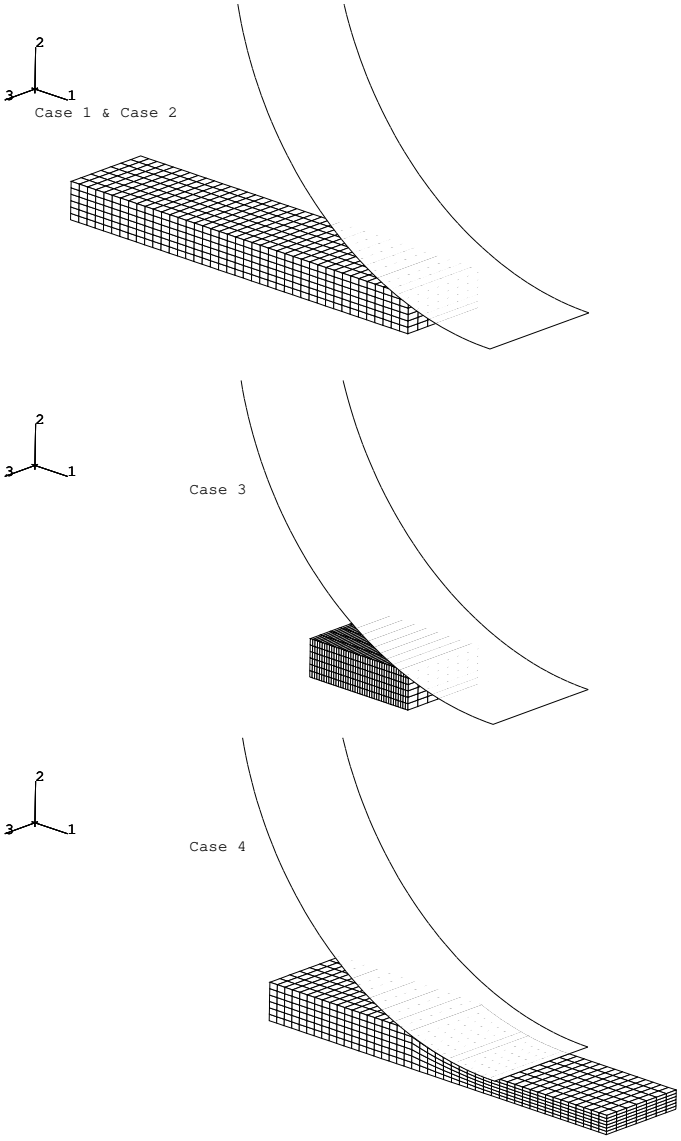


Figure 2: Initial configurations for each case.

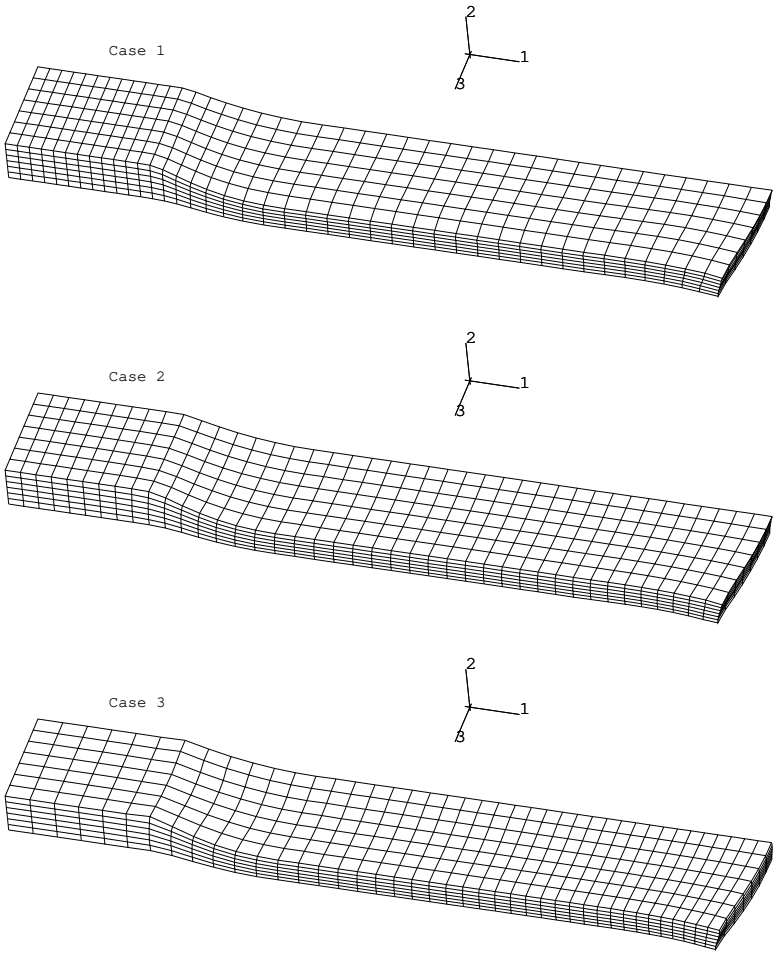


Figure 3: Deformed mesh for Cases 1–3.

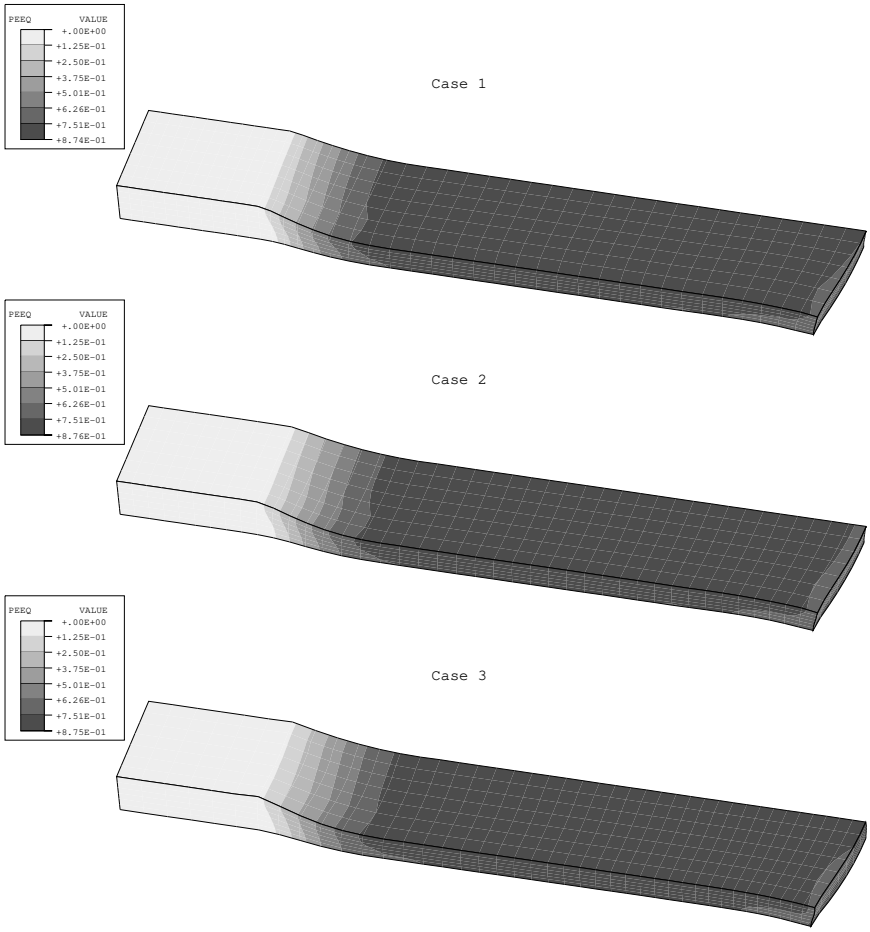


Figure 4: Contours of equivalent plastic strain for Cases 1–3.

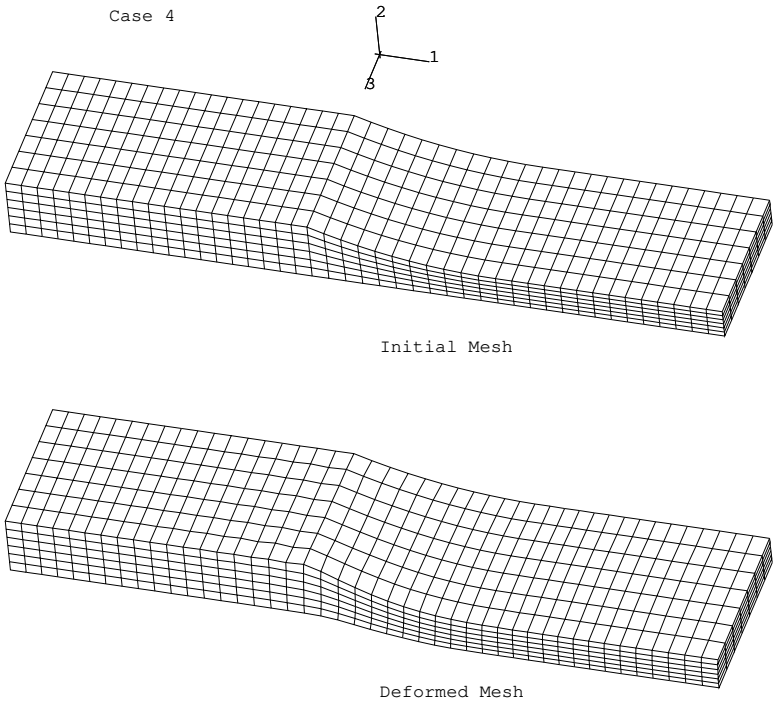


Figure 5: Deformed mesh for Case 4 (shown with initial mesh for comparison).

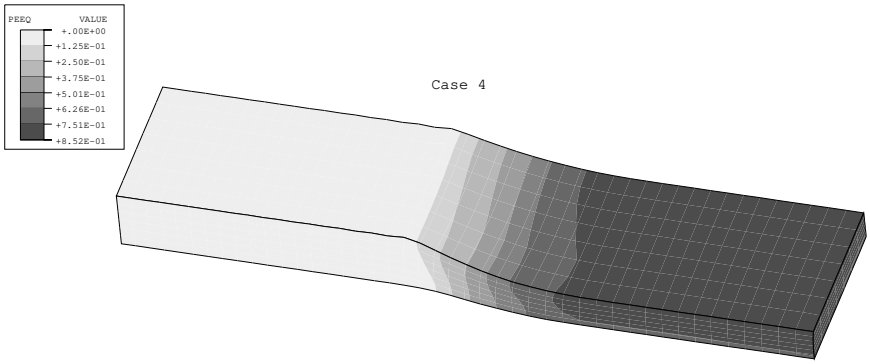


Figure 6: Contours of equivalent plastic strain for Case 4.

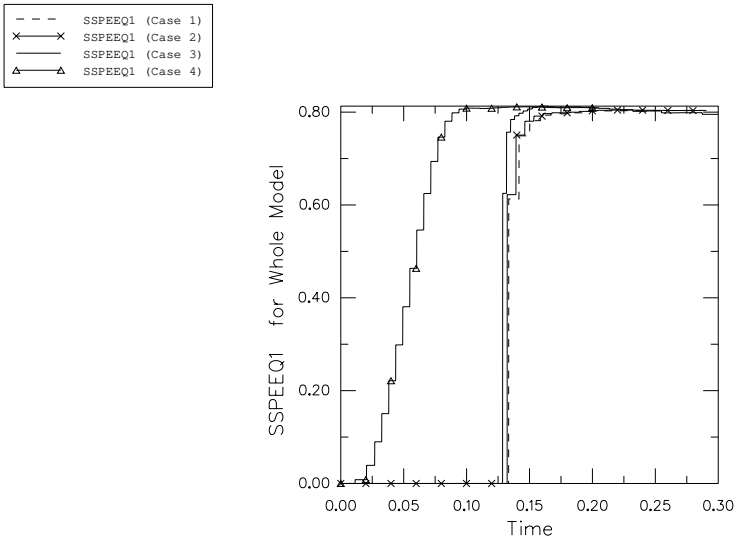


Figure 7: Comparison of equivalent plastic strain norm versus time for all cases.

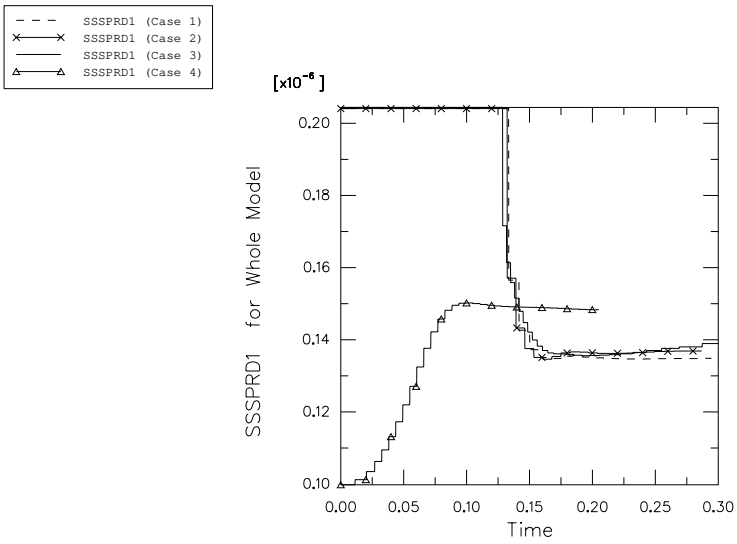


Figure 8: Comparison of spread norm versus time for all cases.

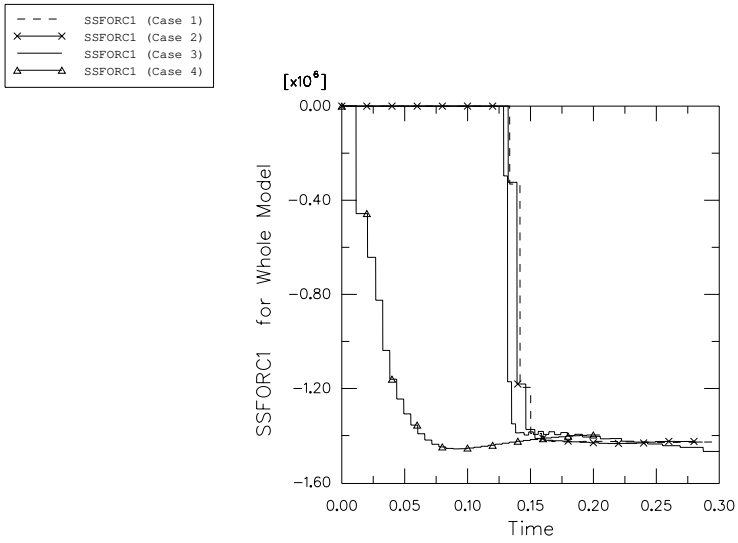


Figure 9: Comparison of force norm versus time for all cases.

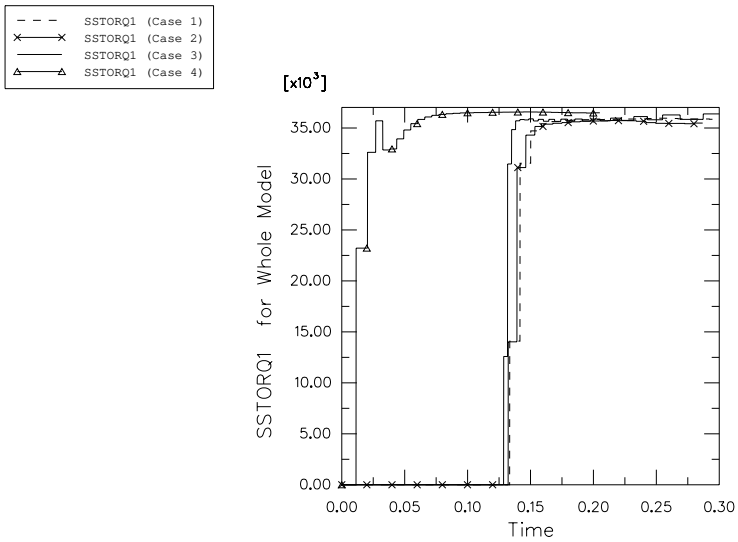


Figure 10: Comparison of torque norm versus time for all cases.

Section rolling

Products: Abaqus/Explicit

Overview

This example illustrates the use of adaptive meshing in a transient simulation of section rolling. Results are compared to a pure Lagrangian simulation.

Problem description

This analysis shows a stage in the rolling of a symmetric I-section. Because of the cross-sectional shape of the I-section, two planes of symmetry exist and only a quarter of the section needs to be modeled. The quarter-symmetry model, shown in *Figure 1*, consists of two rigid rollers and a blank. Roller 1 has a radius of 747 mm, and roller 2 has a radius of 452 mm. The blank has a length of 850 mm, a web half-width of 176.7 mm, a web half-thickness of 24 mm, and a variable flange thickness.

The finite element model is shown in *Figure 2*. The blank is meshed with C3D8R elements. Symmetry boundary conditions are applied on the y and z symmetry planes of the blank. The rollers are modeled as three-dimensional revolved analytical rigid surfaces. Roller 1 has all degrees of freedom constrained except rotation about the z-axis, where a constant angular velocity of 5 rad/sec is specified. Roller 2 has all degrees of freedom constrained except rotation about the y-axis. An initial velocity of 4187.0 mm/sec in the negative x-direction is applied to the blank to initiate contact between the blank and the rollers. This velocity corresponds to the velocity of the rollers at the point of initial contact.

Variable mass scaling is used to scale the masses of all the blank elements so that a desired minimum stable time increment is achieved initially and the stable time increment does not fall below this minimum throughout the analysis. The loading rates and mass scaling definitions are such that a quasi-static solution is generated.

The blank is steel and is modeled as a von Mises elastic-plastic material with a Young's modulus of 212 GPa, an initial yield stress of 80 MPa, and a constant hardening slope of 258 MPa. Poisson's ratio is 0.281; the density is 7833 kg/m^3. Coulomb friction with a friction coefficient of 0.3 is assumed between the rollers and the blank.

Adaptive meshing

Adaptive meshing can improve the solution and mesh quality for section rolling problems that involve large deformations. A single adaptive mesh domain that incorporates the entire blank is defined. Symmetry planes are defined as Lagrangian boundary regions (the default), and the contact surface on the blank is defined as a sliding boundary region (the default). The default values are used for all adaptive mesh parameters and controls.

Results and discussion

Figure 3 shows the deformed configuration of the blank when continuous adaptive meshing is used. For comparison purposes a pure Lagrangian simulation is performed. *Figure 4* shows the deformed configuration for a pure Lagrangian simulation. The mesh at the flange-web interface is distorted in the Lagrangian simulation, but the mesh remains nicely proportioned in the adaptive mesh analysis. A close-up view of the deformed configuration of the blank is shown for each analysis in *Figure 5* and *Figure 6* to highlight the differences in mesh quality. Contours of equivalent plastic strain for each analysis are shown in *Figure 7* and *Figure 8*. The plastic strain distributions are very similar.

Figure 9 and *Figure 10* show time history plots for the y-component of reaction force and the reaction moment about the z-axis, respectively, for roller 1. The results for the adaptive mesh simulation compare closely to those for the pure Lagrangian simulation.

Input files

ale_rolling_section.inp	Analysis that uses adaptive meshing.
ale_rolling_sectionnode.inp	External file referenced by the adaptive mesh analysis.
ale_rolling_sectionelem.inp	External file referenced by the adaptive mesh analysis.
ale_rolling_sectionnelset.inp	External file referenced by the adaptive mesh analysis.
ale_rolling_sectionsurf.inp	External file referenced by the adaptive mesh analysis.
lag_rolling_section.inp	Lagrangian analysis using contact pairs.
lag_rolling_section_gcont.inp	Lagrangian analysis using general contact.

Figures

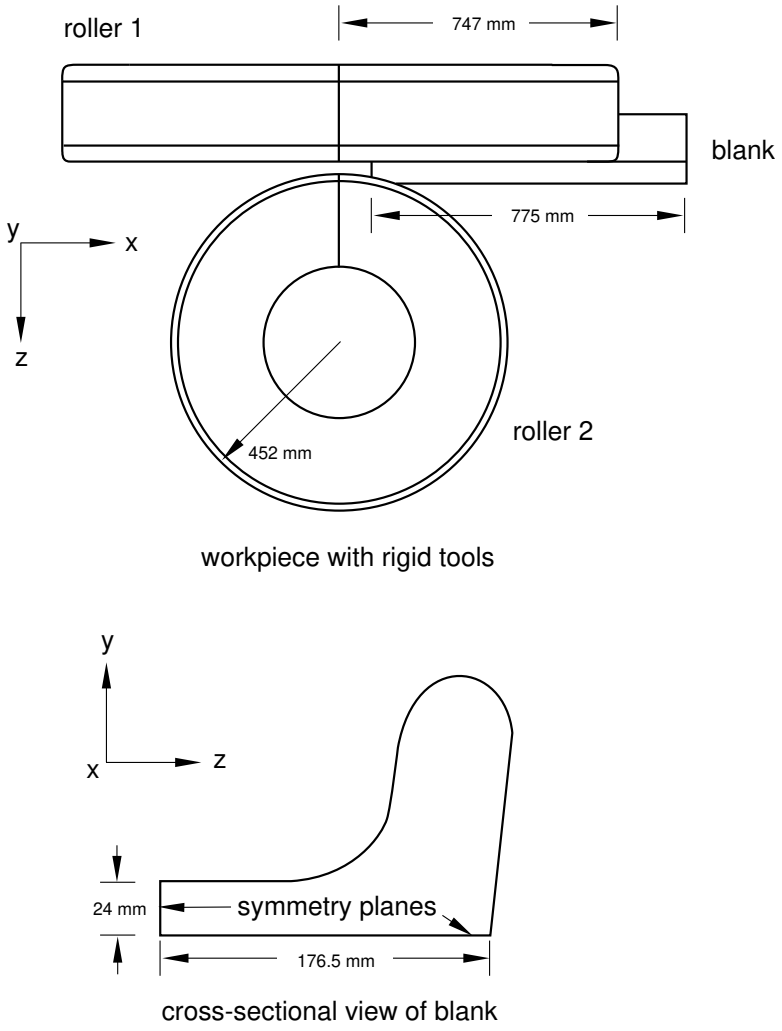


Figure 1: Geometry of the quarter-symmetry blank and the rollers.

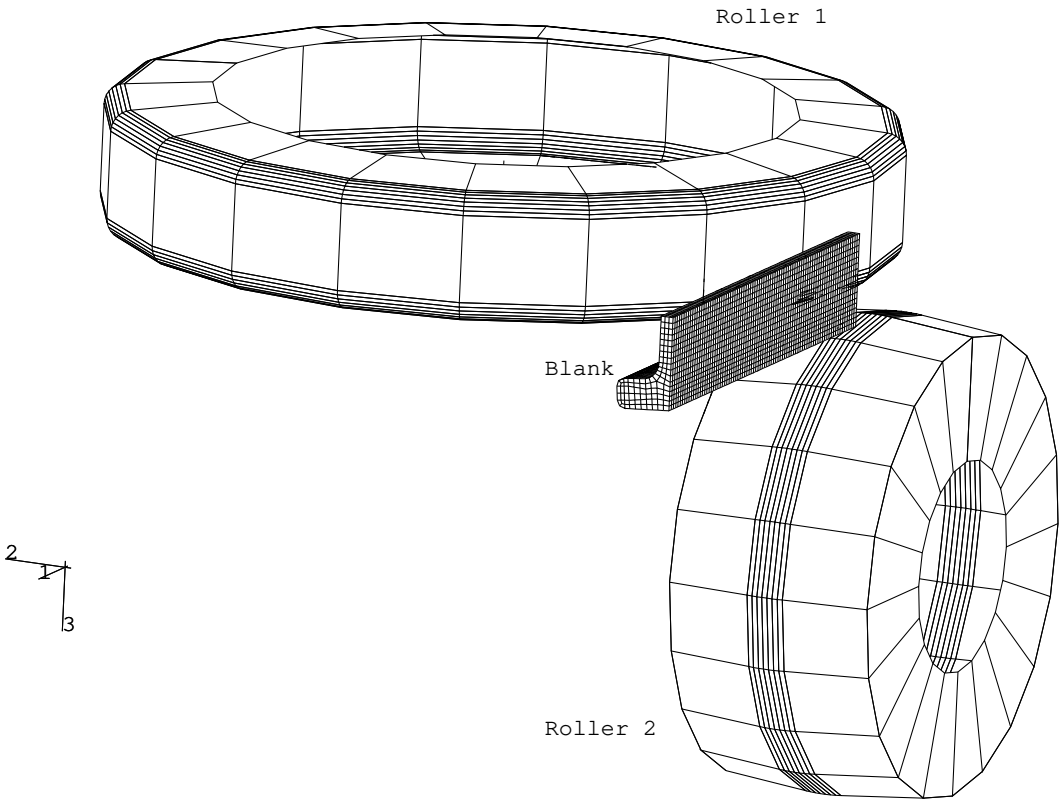


Figure 2: Quarter-symmetry finite element model.

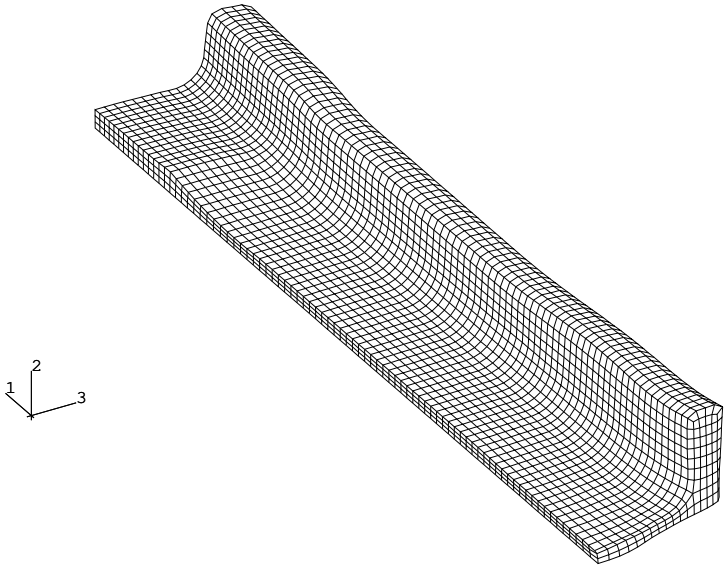


Figure 3: Deformed blank for the adaptive mesh simulation.

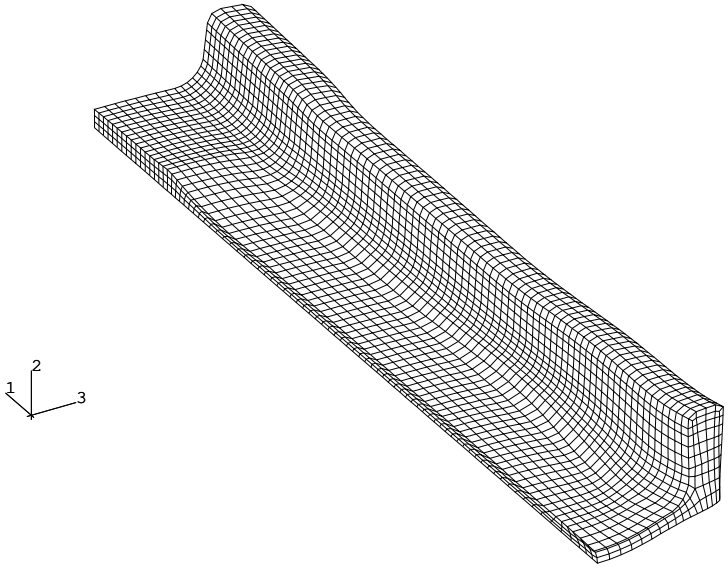


Figure 4: Deformed blank for the pure Lagrangian simulation.

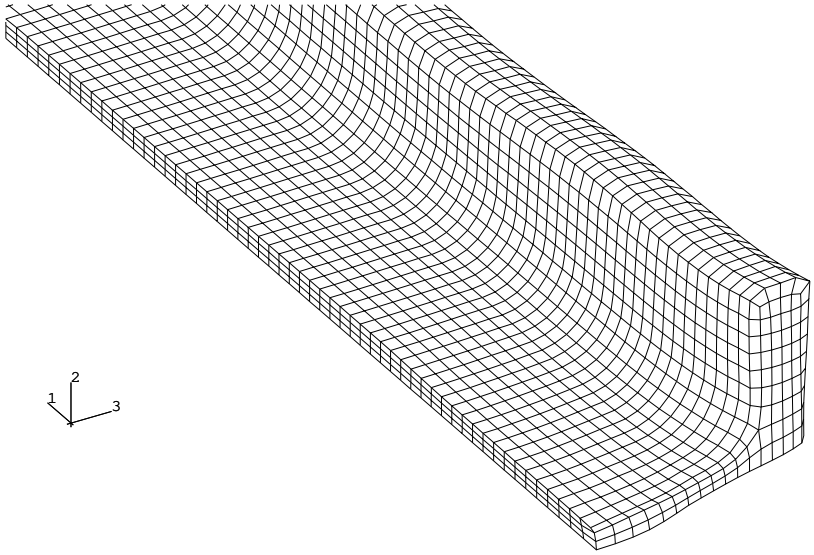


Figure 5: Close-up of the deformed blank for the adaptive mesh simulation.

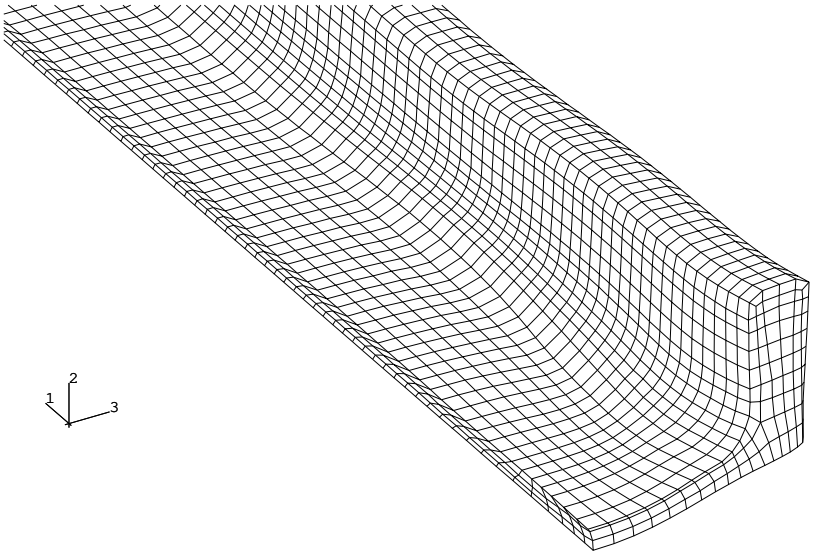


Figure 6: Close-up of the deformed blank for the pure Lagrangian simulation.

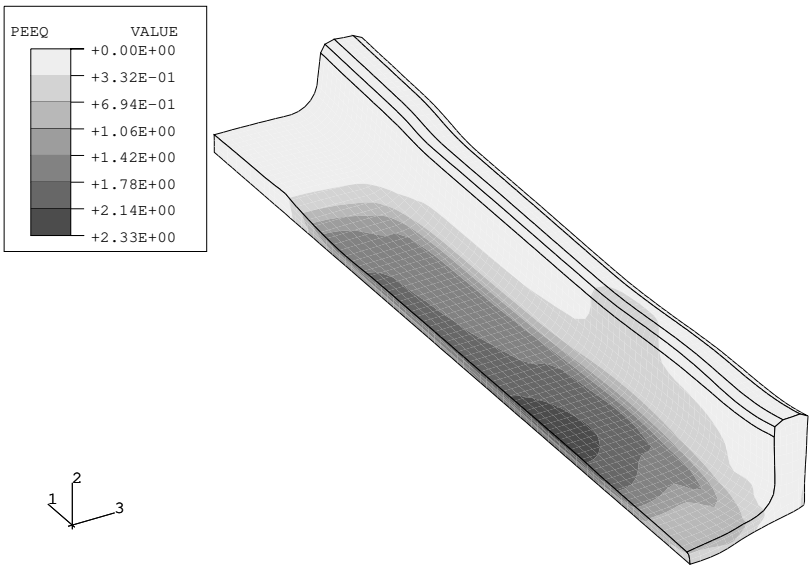


Figure 7: Contours of equivalent plastic strain for the adaptive mesh simulation.

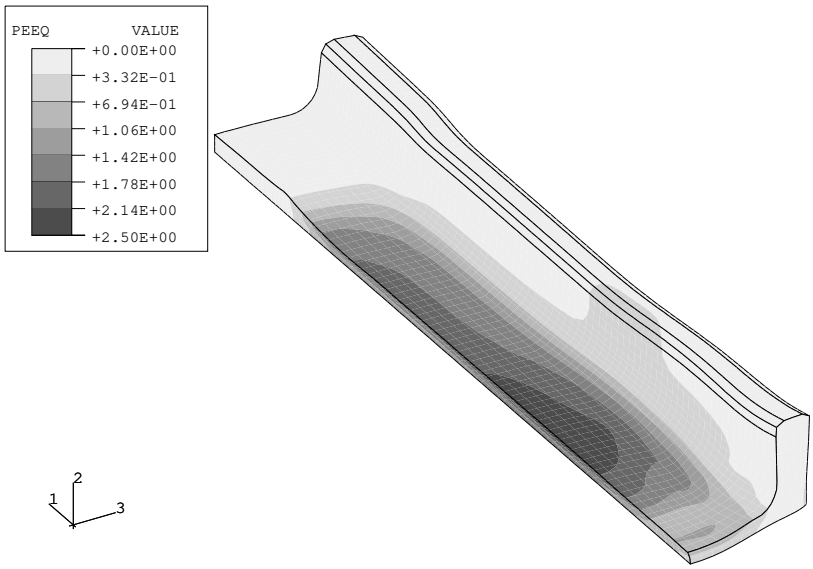


Figure 8: Contours of equivalent plastic strain for the pure Lagrangian simulation.

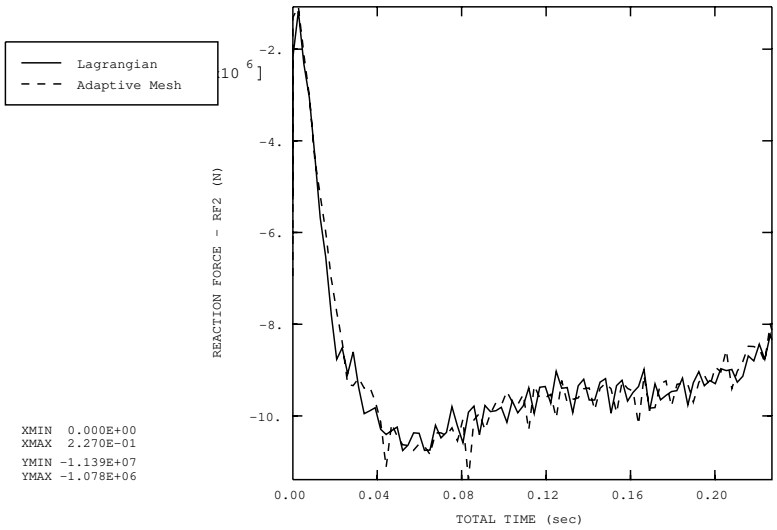


Figure 9: Time history of the reaction force in the y-direction at the reference node of Roller 1.

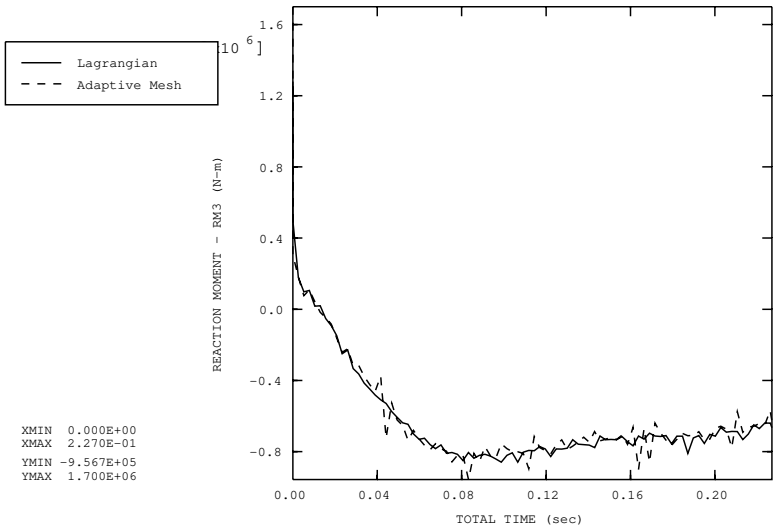


Figure 10: Time history of the reaction moment about the z-axis at the reference node of Roller 1.

Ring rolling

Products: Abaqus/Explicit

Overview

This example illustrates the use of adaptive meshing in a two-dimensional rolling simulation. Results are compared to those obtained using a pure Lagrangian approach.

Problem description

Ring rolling is a specialized process typically used to manufacture parts with revolved geometries such as bearings. The three-dimensional rolling setup usually includes a freely mounted, idle roll; a continuously rotating driver roll; and guide rolls in the rolling plane. Transverse to the rolling plane, conical rolls are used to stabilize the ring and provide a forming surface in the out-of-plane direction. In this example a two-dimensional, plane stress idealization is used that ignores the effect of the conical rolls. A schematic diagram of the ring and the surrounding tools is shown in *Figure 1*.

The driver roll has a diameter of 680 mm, and the idle and guide rolls have diameters of 102 mm. The ring has an initial inner diameter of 127.5 mm and a thickness of 178.5 mm. The idle and driver rolls are arranged vertically and are in contact with the inner and outer surfaces of the ring, respectively. The driver roll is rotated around its stationary axis, while the idle roll is moved vertically downward at a specified feed rate. For this simulation the x–y motion of the guide rolls is determined *a priori* and is prescribed so that the rolls remain in contact with the ring throughout the analysis but do not exert appreciable force on it. In practice the guide rolls are usually connected through linkage systems, and their motion is a function of both force and displacement.

The ring is meshed with CPS4R elements, as shown in *Figure 2*. The ring is steel and is modeled as a von Mises elastic-plastic material with a Young's modulus of 150 GPa, an initial yield stress of 168.7 MPa, and a constant hardening slope of 884 MPa. The Poisson's ratio is 0.3; the density is 7800 kg/m^3.

The analysis is run so that the ring completes approximately 20 revolutions (16.5 seconds). The rigid rolls are modeled as analytical rigid surfaces using connected line segments. The driver roll is rotated at a constant angular velocity of 3.7888 rad/sec about the z-axis, while the idle roll has a constant feed rate of 4.9334 mm/sec and is free to rotate about the z-axis. All other degrees of freedom for the driver and idle rolls are constrained. A friction coefficient of 0.5 is defined at the blank-idle roll and blank-drive roll interfaces. Frictionless contact is used between the ring and guide rolls, and the rotation of the guide rolls is constrained since the actual guide rolls are free to rotate and exert negligible torque on the ring.

To obtain an economical solution, the masses of all elements in the ring are scaled by a factor of 2500. This scaling factor represents a reasonable upper limit on the mass scaling possible for this problem, above which significant inertial effects would be generated. Furthermore, since the two-dimensional model does not contain the conical rolls, the ring oscillates from side to side even under the action of the guide rolls. An artificial viscous pressure of 300 MPa sec/m is applied on the inner and outer surfaces of the ring to assist the guide rolls in preserving the circular shape of the ring. The pressure value was chosen by trial and error.

Adaptive meshing

A single adaptive mesh domain that incorporates the ring is defined. Contact surfaces on the ring are defined as sliding boundary regions (the default). Because of the large number of increments required to simulate 20 revolutions, the deformation per increment is very small. Therefore, the frequency of adaptive meshing is changed from the default of 10 to every 50 increments. The cost of adaptive meshing at this frequency is negligible compared to the underlying analysis cost.

Results and discussion

Figure 3 shows the deformed configuration of the ring after completing 20 revolutions with continuous adaptive meshing. High-quality element shapes and aspect ratios are maintained throughout the simulation. *Figure 4* shows the deformed configuration of the ring when a pure Lagrangian simulation is performed. The pure Lagrangian mesh is distorted, especially at the inner radius where elements become skewed and very small in the radial direction.

Figure 5 and *Figure 6* show time history plots for the y-component of reaction force on the idle roll and the reaction moment about the z-axis for the driver roll, respectively, for both the adaptive mesh and pure Lagrangian approaches. Although the final meshes are substantially different, the roll force and torque match reasonably well.

For both the adaptive and pure Lagrangian solutions the plane stress idealization used here results in very localized through-thickness straining at the inner and outer radii of the ring. This specific type of localized straining is unique to plane stress modeling and does not occur in ring rolling processes. It is also not predicted by a three-dimensional finite element model. If adaptivity is used and refined meshing is desired to capture strong gradients at the inner and outer extremities, the initially uniform mesh can be replaced with a graded mesh. Although not shown here, a graded mesh concentrates element refinement in areas of strong gradients. You can specify in the adaptive mesh controls that the initial mesh gradation should be preserved while distortions are reduced as the analysis evolves.

Input files

ale_ringroll_2d.inp	Analysis that uses adaptive meshing.
ale_ringroll_2dnode.inp	External file referenced by the adaptive mesh analysis.
ale_ringroll_2delem.inp	External file referenced by the adaptive mesh analysis.
guideamp.inp	External file referenced by the adaptive mesh analysis.
lag_ringroll_2d.inp	Lagrangian analysis.

Figures

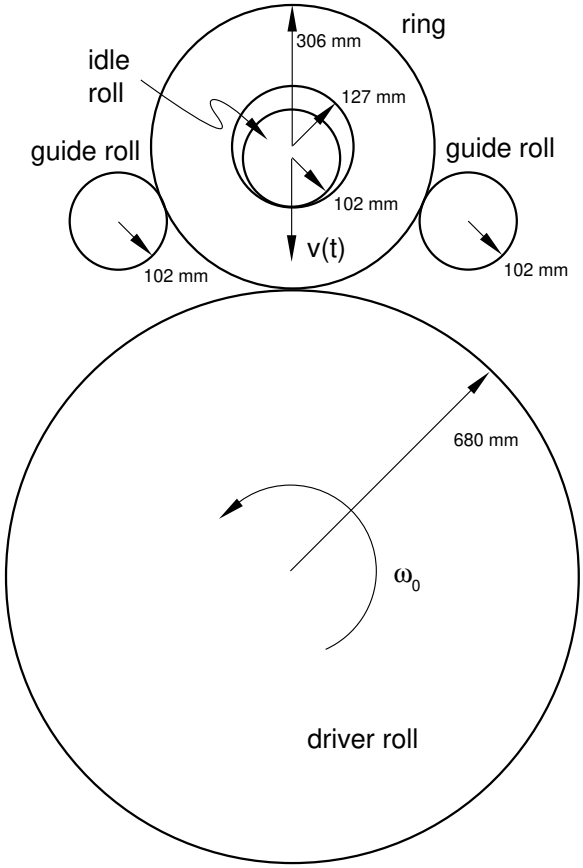


Figure 1: Model geometry for the two-dimensional ring rolling analysis.

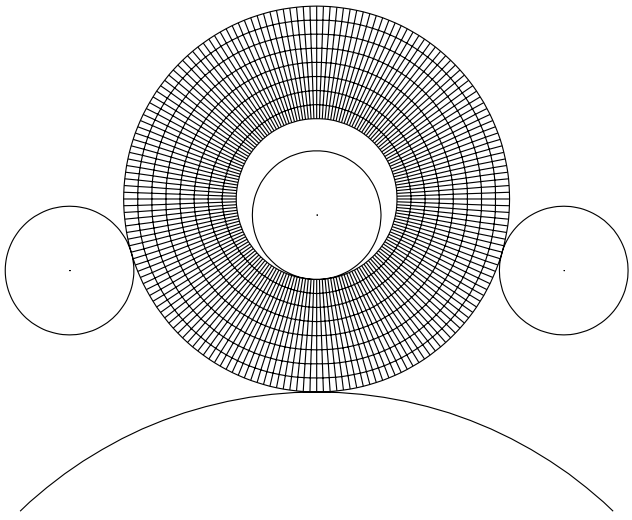


Figure 2: Initial mesh configuration.

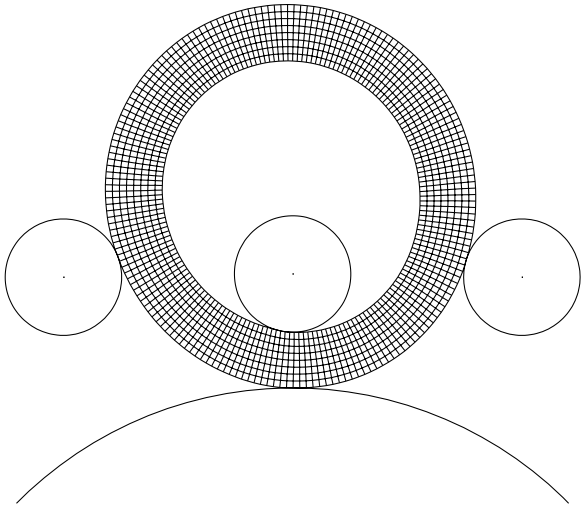


Figure 3: Deformed configuration after 20 revolutions using adaptive meshing.

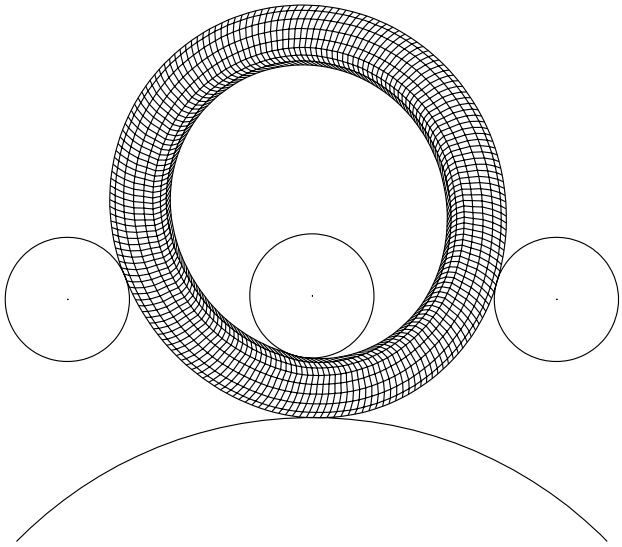


Figure 4: Deformed configuration after 20 revolutions using a pure Lagrangian approach.

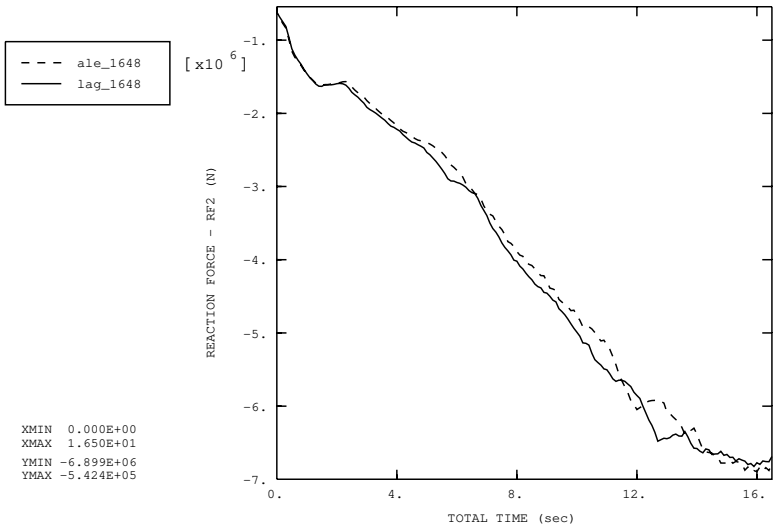


Figure 5: Time history of the reaction force in the y-direction for the idle roll.

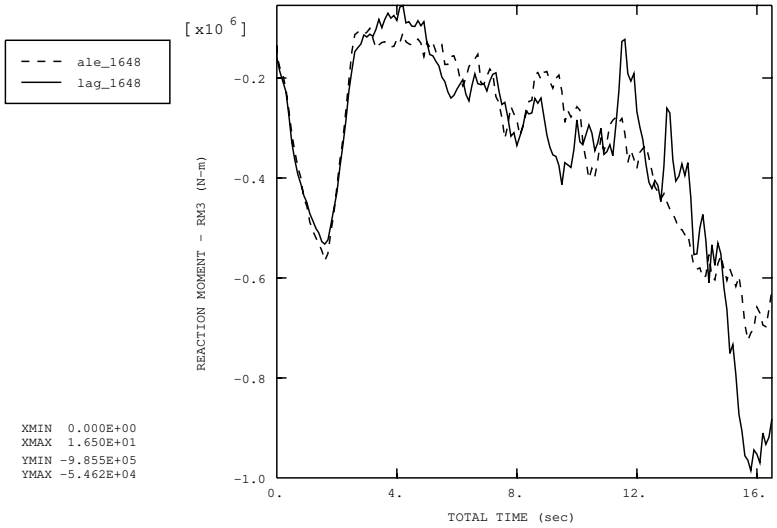


Figure 6: Time history of the reaction moment about the z-axis for the driver roll.

Axisymmetric extrusion: transient and steady-state

Products: Abaqus/Explicit

Overview

This example illustrates the use of adaptive meshing in simulations of extrusion processes with three axisymmetric analysis cases.
First, a transient simulation is performed for a backward, flat-nosed die, extrusion geometry using adaptivity on a Lagrangian mesh domain. Second, a transient simulation is performed on the analogous forward, square die, extrusion geometry, also using adaptivity on a Lagrangian mesh domain. Finally, a steady-state simulation is performed for the forward extrusion geometry using adaptivity on an Eulerian mesh domain.

Problem description

The model configurations for the three analysis cases are shown in *Figure 1*. Each of the models is axisymmetric and consists of one or more rigid tools and a deformable blank. The rigid tools are modeled as analytical rigid surfaces of connected line segments. All contact surfaces are assumed to be well-lubricated and, thus, are treated as frictionless. The blank is made of aluminum and is modeled as a von Mises elastic-plastic material with isotropic hardening. The Young's modulus is 38 GPa, and the initial yield stress is 27 MPa. The Poisson's ratio is 0.33; the density is 2672 kg/m^3.

Case 1: Transient analysis of a backward extrusion

The model geometry consists of a rigid die, a rigid punch, and a blank. The blank is meshed with CAX4R elements and measures 28 × 89 mm. The blank is constrained along its base in the z-direction and at the axis of symmetry in the r-direction. Radial expansion is prevented by contact between the blank and the die. The punch and the die are fully constrained, with the exception of the prescribed vertical motion of the punch. The punch is moved downward 82 mm to form a tube with wall and endcap thicknesses of 7 mm each. The punch velocity is specified using a smooth amplitude so that the response is essentially quasi-static.

The deformation that occurs in extrusion problems, especially in those that involve flat-nosed die geometries, is extreme and requires adaptive meshing. Since adaptive meshing in Abaqus/Explicit works with the same mesh topology throughout the step, the initial mesh must be chosen such that the mesh topology will be suitable for the duration of the simulation. A simple meshing technique has been developed for extrusion problems such as this. In two dimensions it uses a four-sided, mapped mesh domain that can be created with nearly all finite element mesh preprocessors. The vertices for the four-sided, mapped mesh are shown in *Figure 1* and are denoted A, B, C, and D. Two vertices are located on either side of the extrusion opening, the third is in the corner of the dead material zone (the upper left corner of the blank), and the fourth vertex is located in the diagonally opposite corner. A 10 × 60 element mesh using this meshing technique is created for this analysis case and is shown in *Figure 2*. The mesh refinement is oriented such that the fine mesh along sides AB and DC will move up along the extruded walls as the punch is moved downward.

An adaptive mesh domain is defined that incorporates the entire blank. Because of the extremely large distortions expected in the backward extrusion simulation, three mesh sweeps, instead of the default value of one, are specified for each adaptive mesh increment. The default adaptive meshing frequency of 10 is used. Alternatively, a higher frequency could be specified to perform one mesh sweep per adaptive mesh increment. However, this method would result in a higher computational cost because of the increased number of advection sweeps it would require.

A substantial amount of initial mesh smoothing is performed by increasing the number of mesh sweeps to be performed at the beginning of the step to 100. The initially smoothed mesh is shown in *Figure 2*. Initial smoothing

reduces the distortion of the mapped mesh by rounding out corners and easing sharp transitions before the analysis is performed; therefore, it allows the best mesh to be used throughout the analysis.

Case 2: Transient analysis of a forward extrusion

The model geometry consists of a rigid die and a blank. The blank geometry and the mesh are identical to those described for Case 1, except that the mapped mesh is reversed with respect to the vertical plane so that the mesh lines are oriented toward the forward extrusion opening. The blank is constrained at the axis of symmetry in the *r*-direction. Radial expansion is prevented by contact between the blank and the die. The die is fully constrained. The blank is pushed up 19 mm by prescribing a constant velocity of 5 m/sec for the nodes along the bottom of the blank. As the blank is pushed up, material flows through the die opening to form a solid rod with a 7 mm radius.

Adaptive meshing for Case 2 is defined in a similar manner as for Case 1. The undeformed mesh configurations, before and after initial mesh smoothing, are shown in *Figure 3*.

Case 3: Steady-state analysis of a forward extrusion

The model geometry consists of a rigid die, identical to the die used for Case 2, and a blank. The blank geometry is defined such that it closely approximates the shape corresponding to the steady-state solution: this geometry can be thought of as an “initial guess” to the solution. As shown in *Figure 4*, the blank is discretized with a simple graded pattern that is most refined near the die fillet. No special mesh is required for the steady-state case since minimal mesh motion is expected during the simulation. The blank is constrained at the axis of symmetry in the *r*-direction. Radial expansion of the blank is prevented by contact between it and the die.

An adaptive mesh domain is defined that incorporates the entire blank. Because the Eulerian domain undergoes very little overall deformation and the material flow speed is much less than the material wave speed, the frequency of adaptive meshing is changed to 5 from the default value of 1 to improve the computational efficiency of the analysis.

The outflow boundary is assumed to be traction-free and is located far enough downstream to ensure that a steady-state solution can be obtained. This boundary is cast as an Eulerian boundary region. A multi-point constraint is defined on the outflow boundary to keep the velocity normal to the boundary uniform. The inflow boundary is defined using an Eulerian boundary condition to prescribe a velocity of 5 m/sec in the vertical direction. Adaptive mesh constraints are defined on both the inflow and outflow boundaries to fix the mesh in the vertical direction. This effectively creates a stationary control volume with respect to the inflow and outflow boundaries through which material can pass.

Results and discussion

The results for each analysis case are described below.

Case 1

The use of the mapped meshing technique along with adaptive meshing allows the backward extrusion analysis to run to completion, creating the long tube with an endcap. Three plots of the deformed mesh at various times are shown in *Figure 5*. These plots clearly show how the quality of the mesh is preserved for the majority of the simulation. Despite the large amount of deformation involved, the mesh remains smooth and concentrated in the areas of high strain gradients. Extreme deformation and thinning at the punch fillet occurs near the end of the analysis. This thinning can be reduced by increasing the fillet radius of the punch. Corresponding contours of equivalent plastic strain are plotted in *Figure 6*. The plastic strains are highest along the inner surface of the tube.

Case 2

Adaptive meshing enables the transient forward extrusion simulation to proceed much further than would be possible using a pure Lagrangian approach. After pushing the billet 19 mm through the die, the analysis cannot be continued because the elements become too distorted. Since the billet material is essentially incompressible and the cross-sectional area of the die opening at the top is 1/16 of the original cross-sectional area of the billet, a rod measuring approximately 304 mm (three times the length of the original billet) is formed.

Three plots of the deformed mesh at various times in the transient forward extrusion are shown in *Figure 7*. As in the backward extrusion case, the plots show that the quality of the mesh is preserved for a majority of the simulation. The last deformed shape has been truncated for clarity because the extruded column becomes very long and thin. Contours of equivalent plastic strain at similar times are shown in *Figure 8*. The plastic strain distribution developing in the vertical column does not reach a steady-state value, even at a height of 304 mm. The steady-state results reported in the discussion for Case 3 show that a steady-state solution based on the equivalent plastic strain distribution is not reached until much later. An absolute steady-state solution cannot be reached until the material on the upstream side of the dead material zone first passes along that zone and through the die opening. The dead material zone is roughly the shape of a triangle and is located in the upper right-hand corner of the die.

Case 3

The steady-state solution to the forward extrusion analysis is obtained at an extruded column height of 800 mm, which corresponds to pushing the billet 50 mm through the die. Thus, this analysis runs 2.5 times longer than Case 2.

Contours of equivalent plastic strain in the middle and at the end of the simulation are shown in *Figure 9*. Time histories of the equivalent plastic strains on the outer edge of the extruded column at the outflow boundary and 27.5 mm below the outflow boundary are shown in *Figure 10*. The plastic strains at both locations converge to the same value by the end of the simulation, which indicates that the solution has reached a steady state. The final mesh configuration is shown in *Figure 11*. The mesh undergoes very little change from the beginning to the end of the analysis because of the accurate initial guess made for the steady-state domain shape and the ability of the adaptive meshing capability in Abaqus/Explicit to retain the original mesh gradation.

As a further check on the accuracy of the steady-state simulation and the conservation properties of adaptive meshing, a time history of the velocity at the outflow boundary is shown in *Figure 12*. The velocity reaches a steady value of approximately 80 m/s, which is consistent with the incompressible material assumption and the 1/16 ratio of the die opening to the billet size.

Input files

ale_extrusion_back.inp	Case 1.
ale_extrusion_backnode.inp	Node data for Case 1.
ale_extrusion_backelem.inp	Element data for Case 1.
ale_extrusion_forward.inp	Case 2.
ale_extrusion_forwardnode.inp	Node data for Case 2.
ale_extrusion_forwardelem.inp	Element data for Case 2.
ale_extrusion_eulerian.inp	Case 3.
ale_extrusion_euleriannode.inp	Node data for Case 3.
ale_extrusion_eulerianelem.inp	Element data for Case 3.

Figures

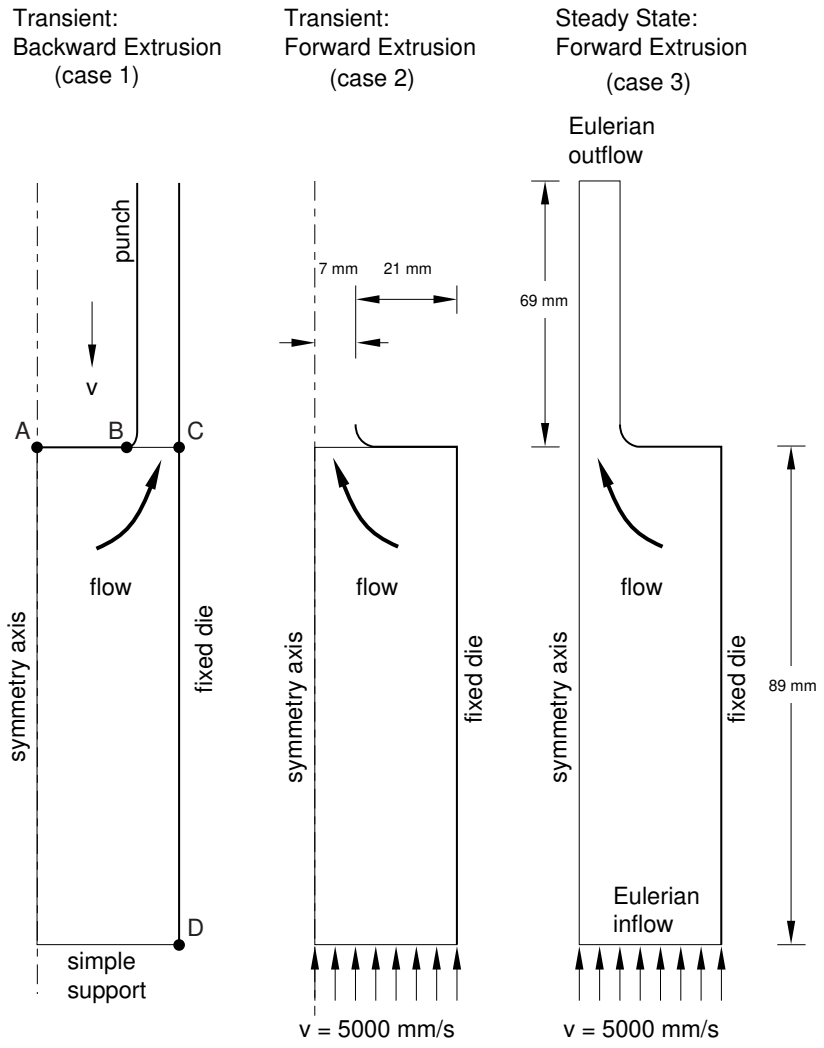


Figure 1: Axisymmetric model geometries used in the extrusion analysis.

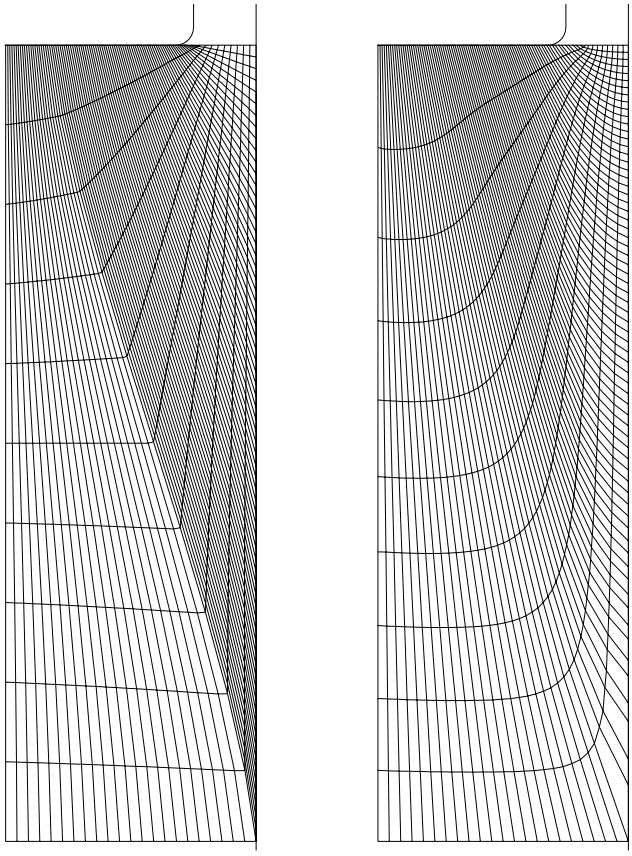


Figure 2: Undeformed configuration for Case 1, before and after initial smoothing.

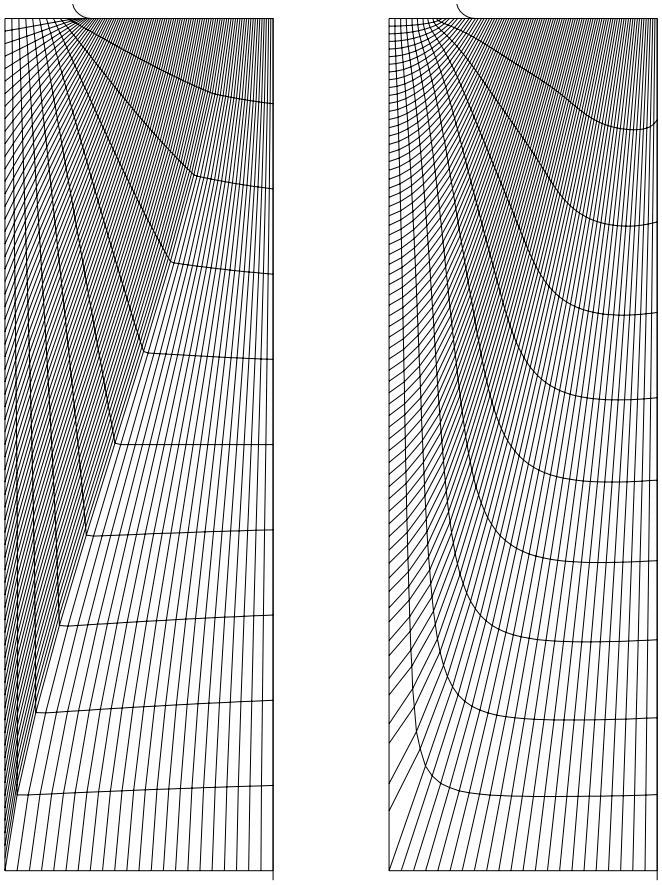


Figure 3: Undeformed configuration for Case 2, before and after initial smoothing.

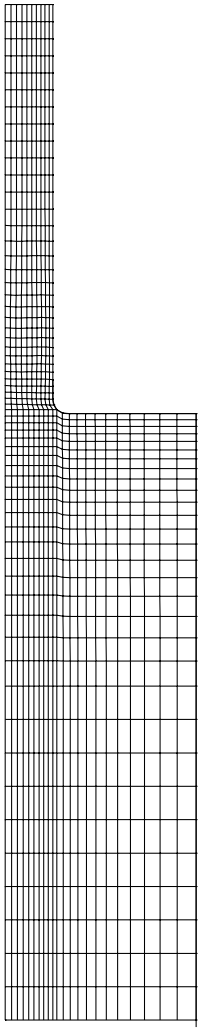


Figure 4: Undeformed configuration for Case 3.

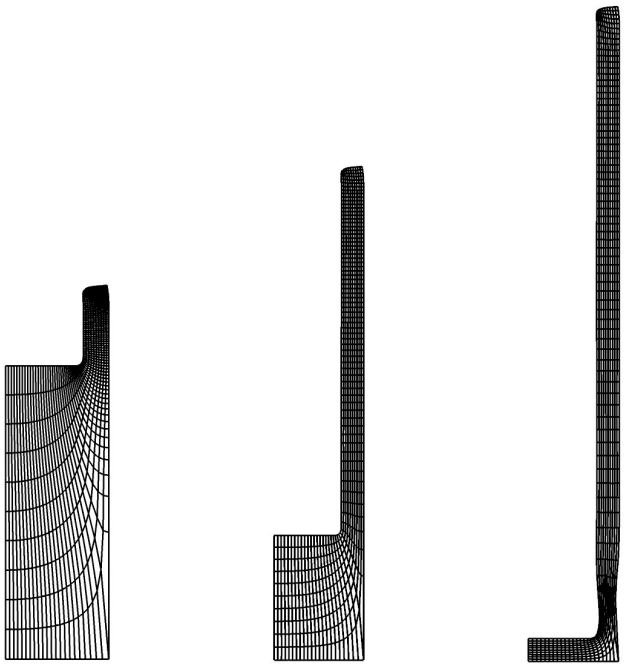


Figure 5: Deformed mesh at various times for Case 1.

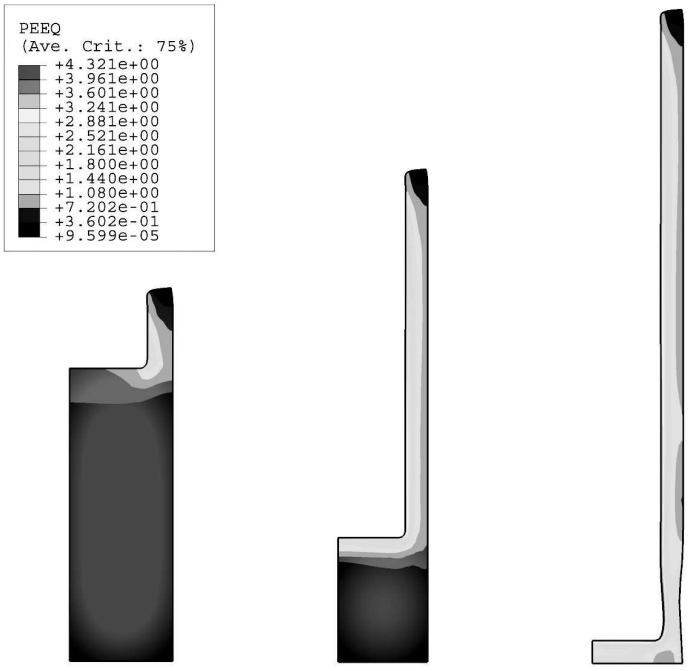


Figure 6: Contours of equivalent plastic strain at various times for Case 1.

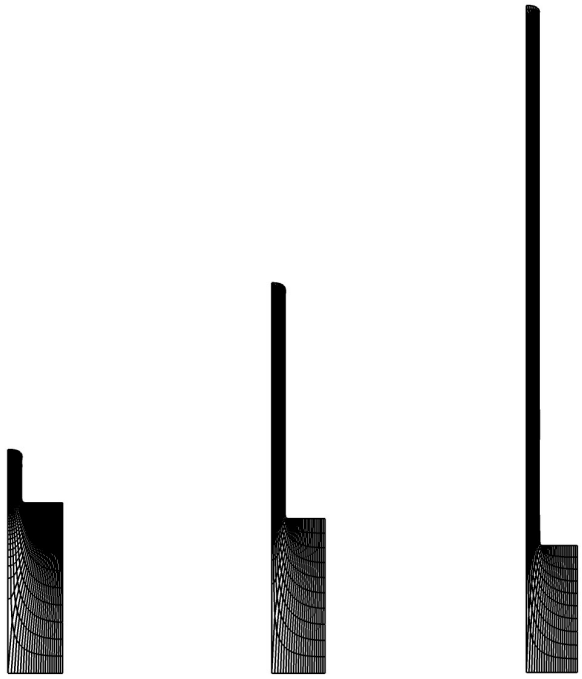


Figure 7: Deformed mesh at various times for Case 2.

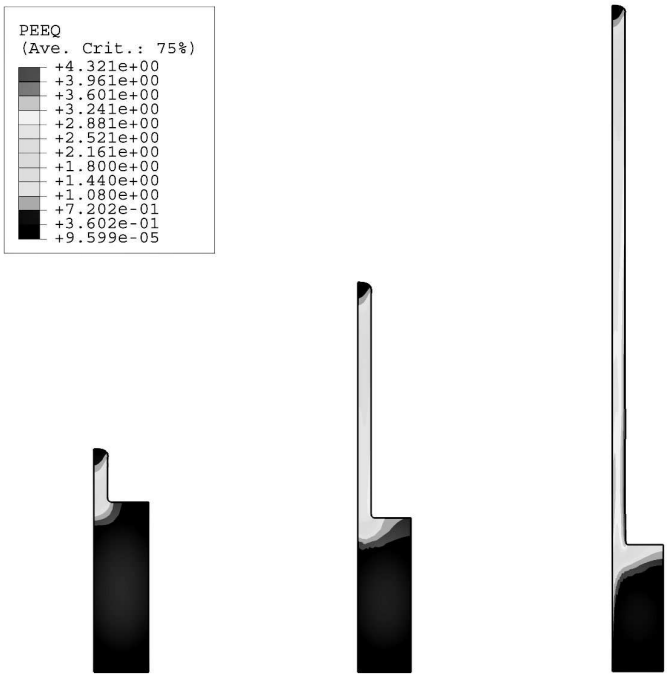


Figure 8: Contours of equivalent plastic strain at various times for Case 2.

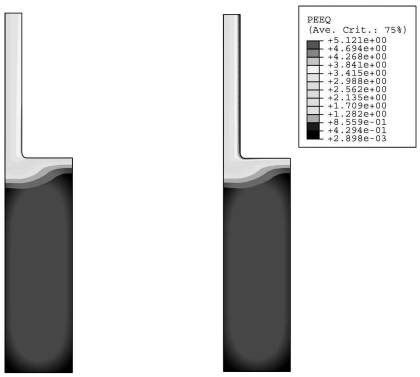


Figure 9: Contours of equivalent plastic strain at an intermediate stage and at the end of the analysis for Case 3.

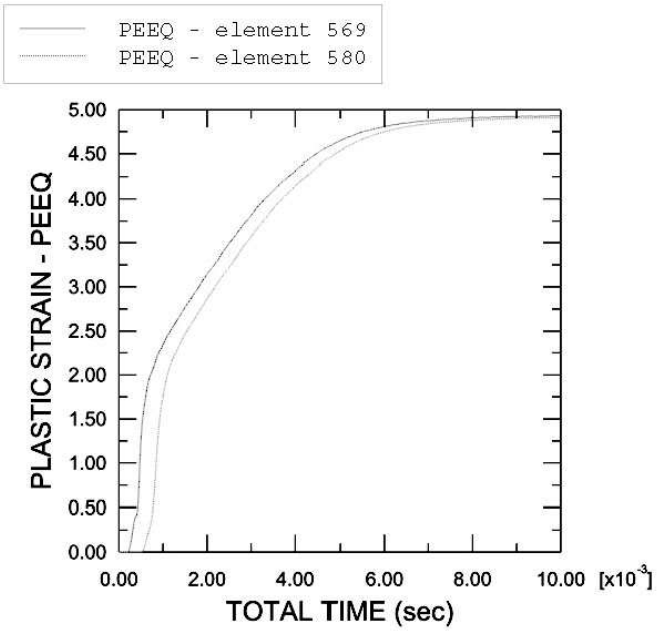


Figure 10: Time history of equivalent plastic strain along the outer edge of the extruded column for Case 3.



Figure 11: Final deformed mesh for Case 3.

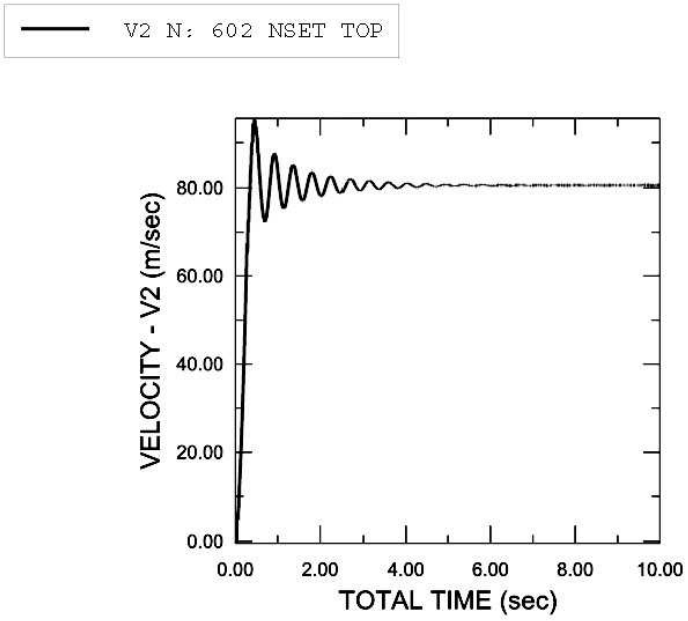


Figure 12: Time history of material velocity at the outflow boundary for Case 3.

Two-step forming simulation

Products: Abaqus/Explicit

Overview

This example illustrates the use of adaptive meshing in simulations of a two-step, bulk metal forming process. The problem is based on a benchmark problem presented at the Metal Forming Process Simulation in Industry conference.

Problem description

The model consists of two sets of rigid forming tools (one set for each forming step) and a deformable blank. The blank and forming die geometries used in the simulation are shown in *Figure 1*. The initial configurations of the blank and the tools for each step are shown in *Figure 2* and *Figure 4*. All forming tools are modeled as discrete rigid bodies and meshed with R3D4 and R3D3 elements. The blank, which is meshed with C3D8R elements, is cylindrical and measures 14.5 × 21 mm. A half model is constructed, so symmetry boundary conditions are prescribed at the y=0 plane.

The blank is made of a steel alloy that is assumed to satisfy the Ramberg-Osgood relation for true stress and logarithmic strain,

$$\epsilon = (\sigma/K)^{1/n},$$

with a reference stress value (K) of 763 MPa and a work-hardening exponent (n) of 0.245. Isotropic elasticity is assumed, with a Young's modulus of 211 GPa and a Poisson's ratio of 0.3. An initial yield stress of 200 MPa is obtained with these data. The stress-strain behavior is defined by piecewise linear segments matching the Ramberg-Osgood curve up to a total (logarithmic) strain level of 140%, with von Mises yield and isotropic hardening.

The analysis is conducted in two steps. For the first step the rigid tools consist of a planar punch, a planar base, and a forming die. The initial configuration for this step is shown in *Figure 2*. The base, which is not shown, is placed at the opening of the forming die to prevent material from passing through the die. The motion of the tools is fully constrained, with the exception of the prescribed displacement in the z-direction for the punch, which is moved 12.69 mm toward the blank at a constant velocity of 30 m/sec consistent with a quasi-static response. The deformed configuration of the blank at the completion of the first step is shown in *Figure 3*.

In the second step the original punch and die are removed from the model and replaced with a new punch and die, as shown in *Figure 4*. The removal of the tools is accomplished by deleting the contact pairs between them and the blank. Although not shown in the figure, the base is retained; both it and the new die are fully constrained. The punch is moved 10.5 mm toward the blank at a constant velocity of 30 m/sec consistent with a quasi-static response. The deformed configuration of the blank at the completion of the second step is shown in *Figure 5*.

Adaptive meshing

A single adaptive mesh domain that incorporates the entire blank is used for both steps. A Lagrangian boundary region type (the default) is used to define the constraints on the symmetry plane, and a sliding boundary region type (the default) is used to define all contact surfaces. The frequency of adaptive meshing is increased to 5 for this problem since material flows quickly near the end of the step.

Results and discussion

Figure 6 shows the deformed mesh at the completion of forming for an analysis in which a pure Lagrangian mesh is used. Comparing *Figure 5* and *Figure 6*, the resultant mesh for the simulation in which adaptive meshing is used is clearly better than that obtained with a pure Lagrangian mesh.

In *Figure 7* through *Figure 9* path plots of equivalent plastic strain in the blank are shown using the pure Lagrangian and adaptive mesh domains for locations in the y=0 symmetry plane at an elevation of z=10 mm. The paths are defined in the positive x-direction (from left to right in *Figure 4* to *Figure 6*). As shown in *Figure 7*, the results are in good agreement at the end of the first step. At the end of the second step the path is discontinuous. Two paths are considered: one that spans the left-hand side and another that spans the right-hand side of the U-shaped cross-section along the symmetry plane. The left- and right-hand paths are shown in *Figure 8* and *Figure 9*, respectively. The solutions from the second step compare qualitatively. Small differences can be attributed to the increased mesh resolution and reduced mesh distortion for the adaptive mesh domain.

Input files

ale_forging_steelpart.inp	Analysis with adaptive meshing.
ale_forging_steelpartnode1.inp	External file referenced by the adaptive mesh analysis.
ale_forging_steelpartnode2.inp	External file referenced by the adaptive mesh analysis.
ale_forging_steelpartnode3.inp	External file referenced by the adaptive mesh analysis.
ale_forging_steelpartnode4.inp	External file referenced by the adaptive mesh analysis.
ale_forging_steelpartelem1.inp	External file referenced by the adaptive mesh analysis.
ale_forging_steelpartelem2.inp	External file referenced by the adaptive mesh analysis.
ale_forging_steelpartelem3.inp	External file referenced by the adaptive mesh analysis.
ale_forging_steelpartelem4.inp	External file referenced by the adaptive mesh analysis.
ale_forging_steelpartelem5.inp	External file referenced by the adaptive mesh analysis.
ale_forging_steelpartsets.inp	External file referenced by the adaptive mesh analysis.
lag_forging_steelpart.inp	Pure Lagrangian analysis.
lag_forging_steelpart_gcont.inp	Pure Lagrangian general contact analysis.

References

- Hermann, M., and A. Ruf, “Forming of a Steel Part,” *Metal Forming Process Simulation in Industry, Stuttgart, Germany*, September 1994.

Figures

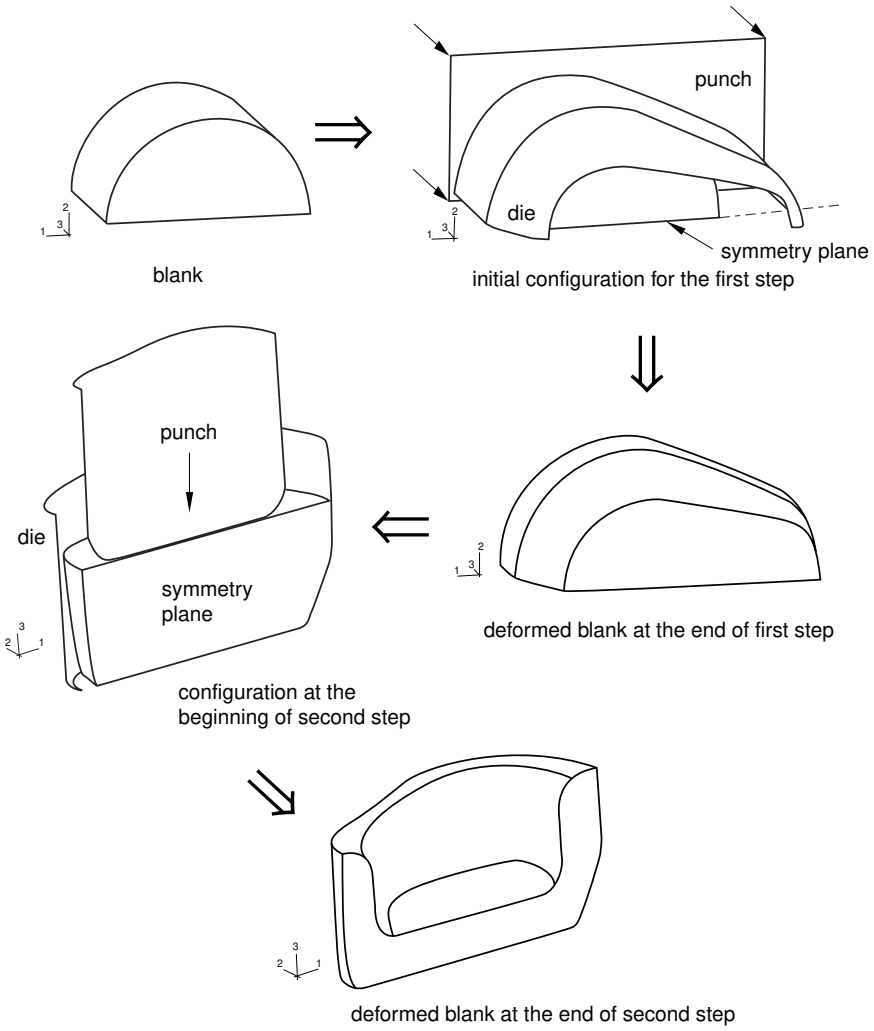


Figure 1: Two-step forging process.

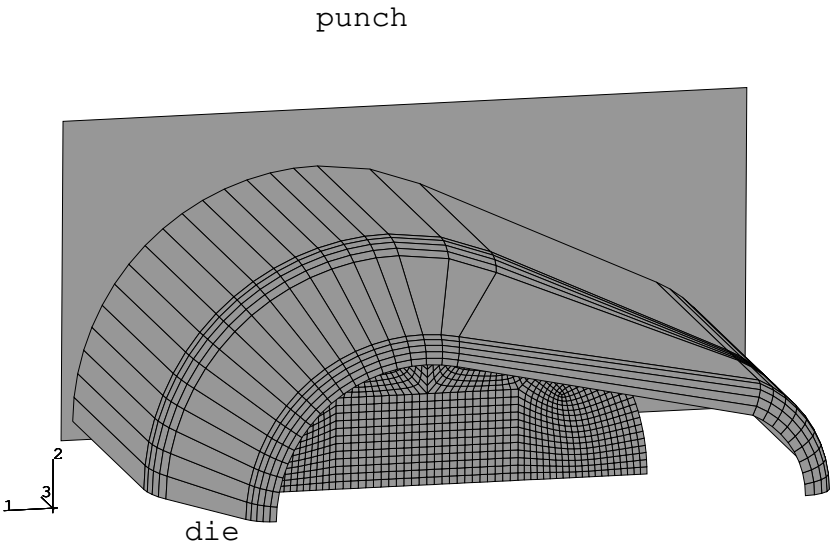


Figure 2: Initial configuration for the first step.

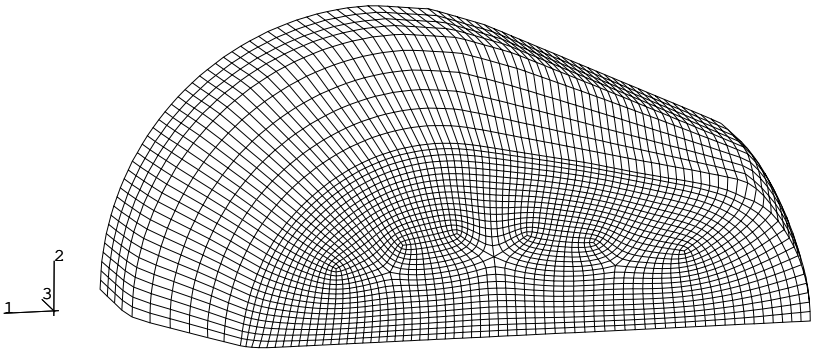


Figure 3: Deformed blank at the end of the first step.

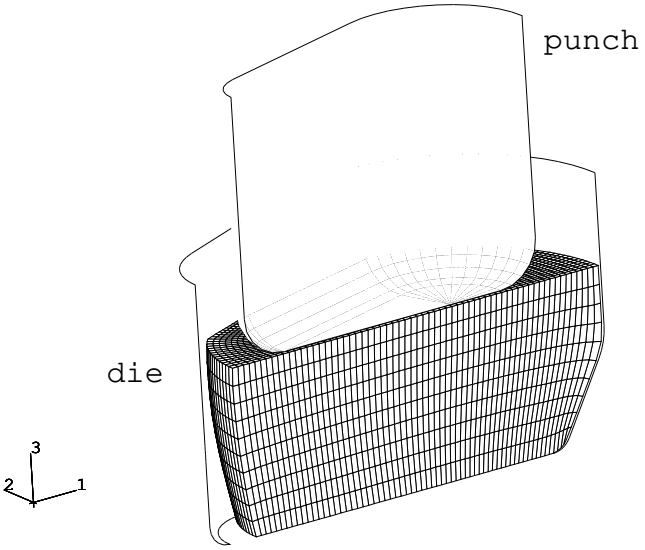


Figure 4: Configuration at the beginning of the second step.

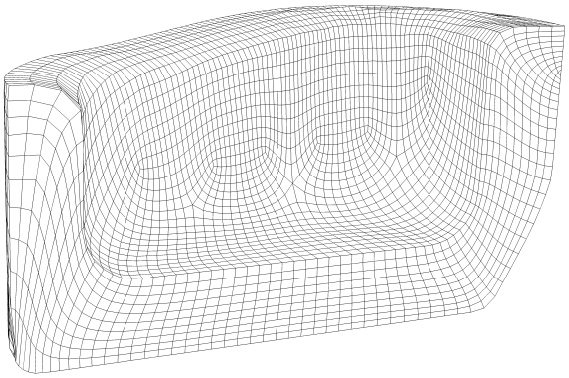


Figure 5: Deformed blank at the end of the second step for the adaptive mesh analysis.

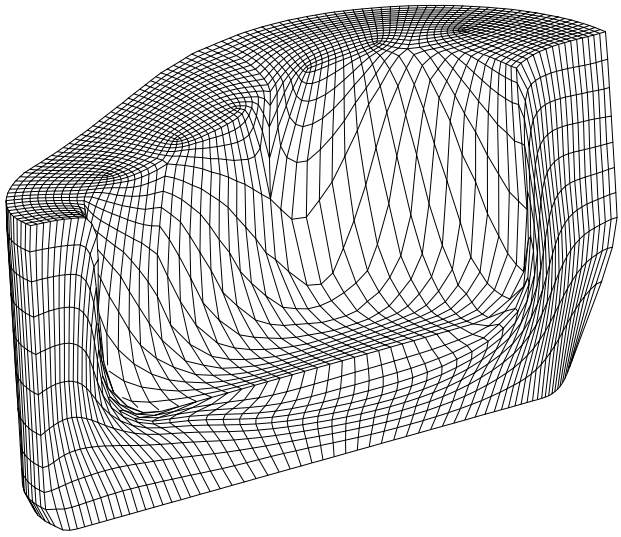


Figure 6: Deformed blank at the end of the second step for the pure Lagrangian analysis.

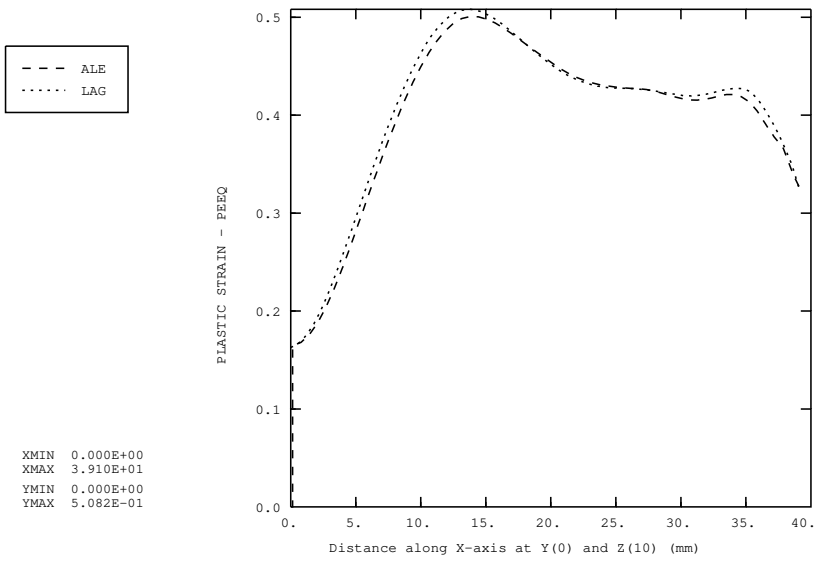


Figure 7: Path plot of equivalent plastic strain at the end of the first step.

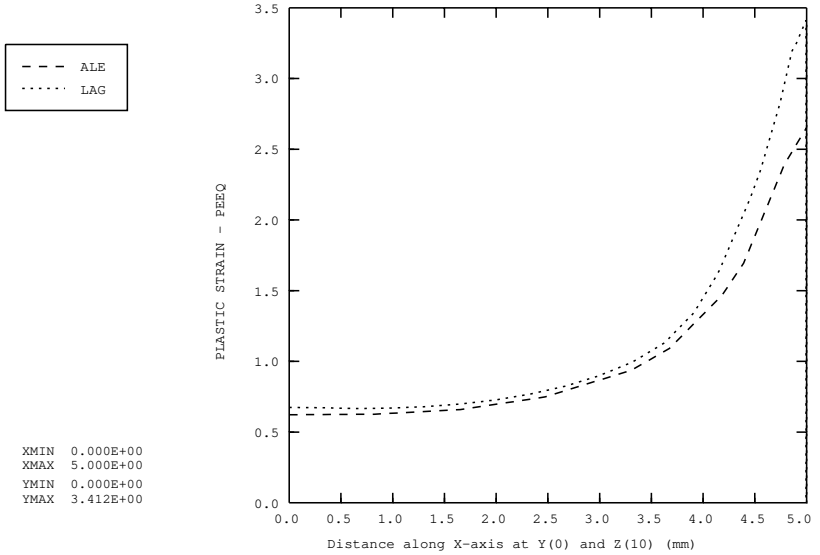


Figure 8: Path plot of equivalent plastic strain along the left side at the end of the second step.

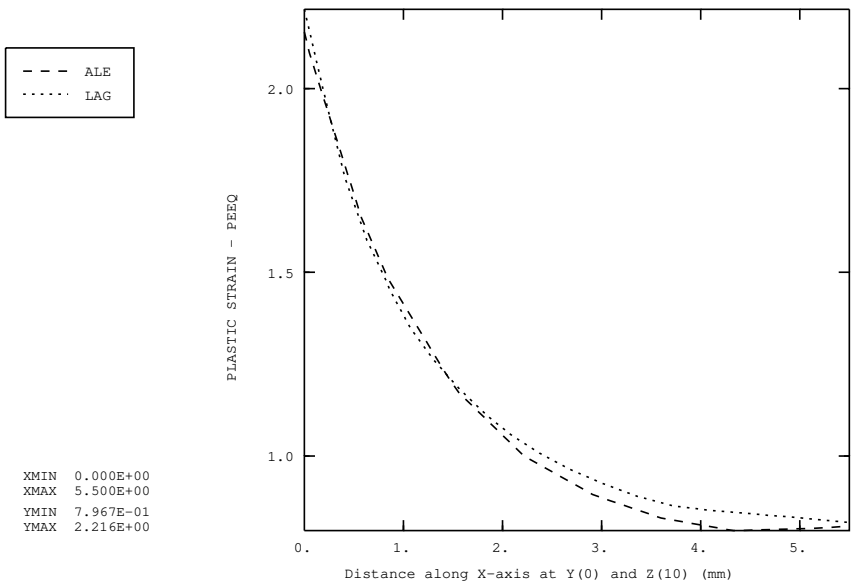


Figure 9: Path plot of equivalent plastic strain along the right side at the end of the second step.

Upsetting of a cylindrical billet: coupled temperature-displacement and adiabatic analysis

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example illustrates coupled temperature-displacement analysis in a metal forming application.

The case studied in this example is an extension of the standard test case that is defined in Lippmann (1979); thus, some verification of the results is available by comparison with the numerical results presented in that reference. The example is that of a small, circular billet of metal that is reduced in length by 60%. Here the problem is analyzed as a viscoplastic case, including heating of the billet by plastic work. Such analysis is often important in manufacturing processes, especially when significant temperature rises degrade the material. The problem is also analyzed in Abaqus/Standard using a porous metal material model. The same problem is used in *Upsetting of a cylindrical billet: quasi-static analysis with mesh-to-mesh solution mapping (Abaqus/Standard) and adaptive meshing (Abaqus/Explicit)* to illustrate mesh rezoning in Abaqus/Standard and adaptive meshing in Abaqus/Explicit.

Geometry and model

The specimen is shown in *Figure 1*: a circular billet, 30 mm long, with a radius of 10 mm, compressed between flat, rough, rigid dies. All surfaces of the billet are assumed to be fully insulated: this thermal boundary condition is chosen to maximize the temperature rise.

The finite element model is axisymmetric and includes the top half of the billet only since the middle surface of the billet is a plane of symmetry. In Abaqus/Standard elements of type CAX8RT, 8-node quadrilaterals with reduced integration that allow for fully coupled temperature-displacement analysis, are used. A regular mesh with six elements in each direction is used, as shown in *Figure 1*. In addition, the billet is modeled with CAX4RT elements in a 12 × 12 mesh for both Abaqus/Standard and Abaqus/Explicit analyses.

The contact between the top and the lateral exterior surfaces of the billet and the rigid die is modeled with a contact pair. The billet surface is defined by means of an element-based surface. The rigid die is modeled as an analytical rigid surface or as an element-based rigid surface. The mechanical interaction between the contact surfaces is assumed to be nonintermittent, rough frictional contact in Abaqus/Standard. Therefore, the contact property includes two additional specifications: a no-separation contact pressure-overclosure relationship to ensure that separation does not occur once contact has been established and rough friction to enforce a no-slip constraint once contact has been established. In Abaqus/Explicit the friction coefficient between the billet and the rigid die is 1.0.

The problem is also solved in Abaqus/Standard with the first-order fully coupled temperature-displacement CAX4T elements in a 12 × 12 mesh. Similarly, the problem is solved using CAX8RT elements and user subroutines *UMAT* and *UMATHT* to illustrate the use of these subroutines.

No mesh convergence studies have been performed, but the comparison with results given in Lippmann (1979) suggests that these meshes provide accuracy similar to the best of those analyses.

The Abaqus/Explicit simulations are performed both with and without adaptive meshing.

Material

The material definition is basically that given in Lippmann (1979), except that the metal is assumed to be rate dependent. The thermal properties are added, with values that correspond to a typical steel, as well as the data for the porous metal plasticity model. The material properties are then as follows:

Young's modulus:	200 GPa
Poisson's ratio:	0.3
Thermal expansion coefficient:	1.2×10^{-5} per °C
Initial static yield stress:	700 MPa
Work hardening rate:	300 MPa
Strain rate dependence:	$\dot{\varepsilon}^{pl} = D((\sigma/\sigma^{\circ}) - 1)^{p}$; $D = 40$/s, $p = 5$
Specific heat:	586 J/(kg°C)
Density:	7833 kg/m^3
Conductivity:	52 J/(m-s-°C)
Porous material parameters:	$q_1 = q_2 = q_3 = 1.0$
Initial relative density:	0.95 ($f_0 =$ 0.05)

Since the problem definition in Abaqus/Standard assumes that the dies are completely rough, no tangential slipping is allowed wherever the metal contacts the die.

Boundary conditions and loading

The kinematic boundary conditions are symmetry on the axis (nodes at $r = 0$, in node set `AXIS`, have $u_r = 0$ prescribed) and symmetry about $z = 0$ (all nodes at $z = 0$, in node set `MIDDLE`, have $u_z = 0$ prescribed). To avoid overconstraint, the node on the top surface of the billet that lies on the symmetry axis is not part of the node set `AXIS`: the radial motion of this node is already constrained by a no-slip frictional constraint (see *Common Difficulties Associated with Contact Modeling in Abaqus/Standard* and *Common Difficulties Associated with Contact Modeling Using Contact Pairs in Abaqus/Explicit*). The rigid body reference node for the rigid surface that defines the die is constrained to have no rotation or u_r-displacement, and its u_z-displacement is prescribed to move 9 mm down the axis at constant velocity. The reaction force at the rigid reference node corresponds to the total force applied by the die.

The thermal boundary conditions are that all external surfaces are insulated (no heat flux allowed). This condition is chosen because it is the most extreme case: it must provide the largest temperature rises possible, since no heat can be removed from the specimen.

One of the controls for the automatic time incrementation scheme in Abaqus/Standard is the limit on the maximum temperature change allowed to occur in any increment. It is set to 100°C, which is a large value and indicates that we are not restricting the time increments because of accuracy considerations in integrating the heat transfer equations. In fact, the automatic time incrementation scheme will choose fairly small increments because of the severe nonlinearity present in the problem and the resultant need for several iterations per increment even with a relatively large number of increments. The maximum allowable temperature change in an increment is set to a large value to obtain a reasonable solution at low cost.

In Abaqus/Explicit the automatic time incrementation scheme is used to ensure numerical stability and to advance the solution in time. Mass scaling is used to reduce the computational cost of the analysis.

The amplitude is applied linearly over the step because the default amplitude variation for a transient, coupled temperature-displacement analysis is a step function, but here we want the die to move down at a constant velocity.

Two versions of the analysis are run: a slow upsetting, where the upsetting occurs in 100 seconds, and a fast upsetting, where the event takes 0.1 second. Both versions are analyzed with the coupled temperature-displacement procedure. The fast upsetting is also run in Abaqus/Standard as an adiabatic static stress analysis. The time period values are specified with the respective procedure options. The adiabatic stress analysis is performed in the same time frame as the fast upsetting case. In all cases analyzed with Abaqus/Standard an initial time increment of

1.5% of the time period is used; that is, 1.5 seconds in the slow case and 0.0015 second in the fast case. This value is chosen because it will result in a nominal axial strain of about 1% per increment, and experience suggests that such increment sizes are generally suitable for cases like this.

Results and discussion

The results of the Abaqus/Standard simulations are discussed first, beginning with the results for the viscoplastic fully dense material. The results of the slow upsetting are illustrated in *Figure 2* to *Figure 4*. The results for the fast upsetting coupled temperature-displacement analysis are illustrated in *Figure 5* to *Figure 7*; those for the adiabatic static stress analysis are shown in *Figure 8* and *Figure 9*. *Figure 2* and *Figure 5* show the configuration that is predicted at 60% upsetting. The configuration for the adiabatic analysis is not shown since it is almost identical to the fast upsetting coupled case. Both the slow and the fast upsetting cases show the folding of the top outside surface of the billet onto the die, as well as the severe straining of the middle of the specimen. The second figure in each series (*Figure 3* for the slow case, *Figure 6* for the fast case, and *Figure 8* for the adiabatic case) shows the equivalent plastic strain in the billet. Peak strains of around 180% occur in the center of the specimen. The third figure in each series (*Figure 4* for the slow case, *Figure 7* for the fast case, and *Figure 9* for the adiabatic case) shows the temperature distributions, which are noticeably different between the slow and fast upsetting cases. In the slow case there is time for the heat to diffuse (the 60% upsetting takes place in 100 sec, on a specimen where a typical length is 10 mm), so the temperature distribution at 100 sec is quite uniform, varying only between 180°C and 185°C through the billet. In contrast, the fast upsetting occurs too quickly for the heat to diffuse. In this case the middle of the top surface of the specimen remains at 0°C at the end of the event, while the center of the specimen heats up to almost 600°C. There is no significant difference in temperatures between the fast coupled case and the adiabatic case. In the outer top section of the billet there are differences that are a result of the severe distortion of the elements in that region and the lack of dissipation of generated heat. The temperature in the rest of the billet compares well. This example illustrates the advantage of an adiabatic analysis, since a good representation of the results is obtained in about 60% of the computer time required for the fully coupled analysis.

The results of the slow and fast upsetting of the billet modeled with the porous metal plasticity model are shown in *Figure 10* to *Figure 15*. The deformed configuration is identical to that of *Figure 2* and *Figure 5*. The extent of growth/closure of the voids in the specimen at the end of the analysis is shown in *Figure 10* and *Figure 13*. The porous material is almost fully compacted near the center of the billet because of the compressive nature of the stress field in that region; on the other hand, the corner element is folded up and stretched out near the outer top portion of the billet, increasing the void volume fraction to almost 0.1 (or 10%) and indicating that tearing of the material is likely. The equivalent plastic strain is shown in *Figure 11* (slow upsetting) and *Figure 14* (fast upsetting) for the porous material; *Figure 12* and *Figure 15* show the temperature distribution for the slow and the fast upsetting of the porous metal. The porous metal needs less external work to achieve the same deformation compared to a fully dense metal. Consequently, there is less plastic work being dissipated as heat; hence, the temperature increase is not as much as that of fully dense metal. This effect is more pronounced in the fast upsetting problem, where the specimen heats up to only 510°C, compared to about 600°C for fully dense metal.

Figure 16 to *Figure 18* show predictions of total upsetting force versus displacement of the die. In *Figure 16* the slow upsetting viscoplastic and porous plasticity results are compared with several elastic-plastic and rigid-plastic results that were collected by Lippmann (1979) and slow viscoplastic results obtained by Taylor (1981). There is general agreement between all the rate-independent results, and these correspond to the slow viscoplastic results of the present example and of those found by Taylor (1981). In *Figure 17* rate dependence of the yield stress is investigated. The fast viscoplastic and porous plasticity results show significantly higher force values throughout the event than the slow results. This effect can be estimated easily. A nominal strain rate of 6 sec is maintained throughout the event. With the viscoplastic model that is used, this effect increases the yield stress by 68%. This factor is very close to the load amplification factor that appears in *Figure 17*. *Figure 18* shows that the force versus displacement prediction of the fast viscoplastic adiabatic analysis agrees well with the fully coupled results.

Two cases using an element-based rigid surface to model the die are also considered in Abaqus/Standard. To define the element-based rigid surface, the elements are assigned to rigid bodies using an isothermal rigid body constraint. The results agree very well with the case when the analytical rigid surface is used.

The automatic load incrementation results suggest that overall nominal strain increments of about 2% per increment were obtained, which is slightly better than what was anticipated in the initial time increment suggestion. These values are typical for problems of this class and are useful guidelines for estimating the computational effort required for such cases.

The results obtained with Abaqus/Explicit compare well with those obtained with Abaqus/Standard, as illustrated in *Figure 19*, which compares the results obtained with Abaqus/Explicit (without adaptive meshing) for the total upsetting force versus the displacement of the die against the same results obtained with Abaqus/Standard. The agreement between the two solutions is excellent. Similar agreement is obtained with the results obtained from the Abaqus/Explicit simulation using adaptive meshing. The mesh distortion is significantly reduced in this case, as illustrated in *Figure 20*.

Input files

Abaqus/Standard input files

cylbillet_cax4t_slow_dense.inp	Slow upsetting case with 144 CAX4T elements, using the fully dense material.
cylbillet_cax4t_fast_dense.inp	Fast upsetting case with 144 CAX4T elements, using the fully dense material.
cylbillet_cax4rt_slow_dense.inp	Slow upsetting case with 144 CAX4RT elements, using the fully dense material.
cylbillet_cax4rt_fast_dense.inp	Fast upsetting case with 144 CAX4RT elements, using the fully dense material.
cylbillet_cax8rt_slow_dense.inp	Slow upsetting case with CAX8RT elements, using the fully dense material.
cylbillet_cax8rt_rb_s_dense.inp	Slow upsetting case with CAX8RT elements, using the fully dense material and an element-based rigid surface for the die.
cylbillet_cax8rt_fast_dense.inp	Fast upsetting case with CAX8RT elements, using the fully dense material.
cylbillet_cax8rt_slow_por.inp	Slow upsetting case with CAX8RT elements, using the porous material.
cylbillet_cax8rt_fast_por.inp	Fast upsetting case with CAX8RT elements, using the porous material.
cylbillet_cgax4t_slow_dense.inp	Slow upsetting case with 144 CGAX4T elements, using the fully dense material.
cylbillet_cgax4t_fast_dense.inp	Fast upsetting case with 144 CGAX4T elements, using the fully dense material.
cylbillet_cgax4t_rb_f_dense.inp	Fast upsetting case with 144 CGAX4T elements, using the fully dense material and an element-based rigid surface for the die.
cylbillet_cgax4t_rb_f_dense_surf.inp	Fast upsetting case with 144 CGAX4T elements, using the fully dense material and an element-based rigid surface for the die with surface-to-surface contact.
cylbillet_cgax8rt_slow_dense.inp	Slow upsetting case with CGAX8RT elements, using the fully dense material.
cylbillet_cgax8rt_fast_dense.inp	Fast upsetting case with CGAX8RT elements, using the fully dense material.

cylbillet_c3d10m_adiab_dense.inp	Adiabatic static analysis with fully dense material modeled with C3D10M elements.
cylbillet_c3d10m_adiab_dense_surf.inp	Adiabatic static analysis with fully dense material modeled with C3D10M elements using surface-to-surface contact.
cylbillet_c3d10m_adiab_dense_po.inp	**POST OUTPUT* analysis of cylbillet_c3d10m_adiab_dense.inp.
cylbillet_cax6m_adiab_dense.inp	Adiabatic static analysis with fully dense material modeled with CAX6M elements.
cylbillet_cax8r_adiab_dense.inp	Adiabatic static analysis with fully dense material modeled with CAX8R elements.
cylbillet_postoutput.inp	**POST OUTPUT* analysis using the fully dense material.
cylbillet_slow_usr_umat_umatht.inp	Slow upsetting case with the material behavior defined in user subroutines *UMAT* and *UMATHT*.
cylbillet_slow_usr_umat_umatht.f	User subroutines *UMAT* and *UMATHT* used in cylbillet_slow_usr_umat_umatht.inp.

Abaqus/Explicit input files

cylbillet_x_cax4rt_slow.inp	Slow upsetting case with fully dense material modeled with CAX4RT elements and without adaptive meshing; kinematic mechanical contact.
cylbillet_x_cax4rt_fast.inp	Fast upsetting case with fully dense material modeled with CAX4RT elements and without adaptive meshing; kinematic mechanical contact.
cylbillet_x_cax4rt_slow_adap.inp	Slow upsetting case with fully dense material modeled with CAX4RT elements and with adaptive meshing; kinematic mechanical contact.
cylbillet_x_cax4rt_fast_adap.inp	Fast upsetting case with fully dense material modeled with CAX4RT elements and with adaptive meshing; kinematic mechanical contact.
cylbillet_xp_cax4rt_fast.inp	Fast upsetting case with fully dense material modeled with CAX4RT elements and without adaptive meshing; penalty mechanical contact.

References

- Lippmann, H., Metal Forming Plasticity, Springer-Verlag, Berlin, 1979.
- Taylor, L. M., "A Finite Element Analysis for Large Deformation Metal Forming Problems Involving Contact and Friction," *Ph.D. Thesis, U. of Texas at Austin*, 1981.

Figures

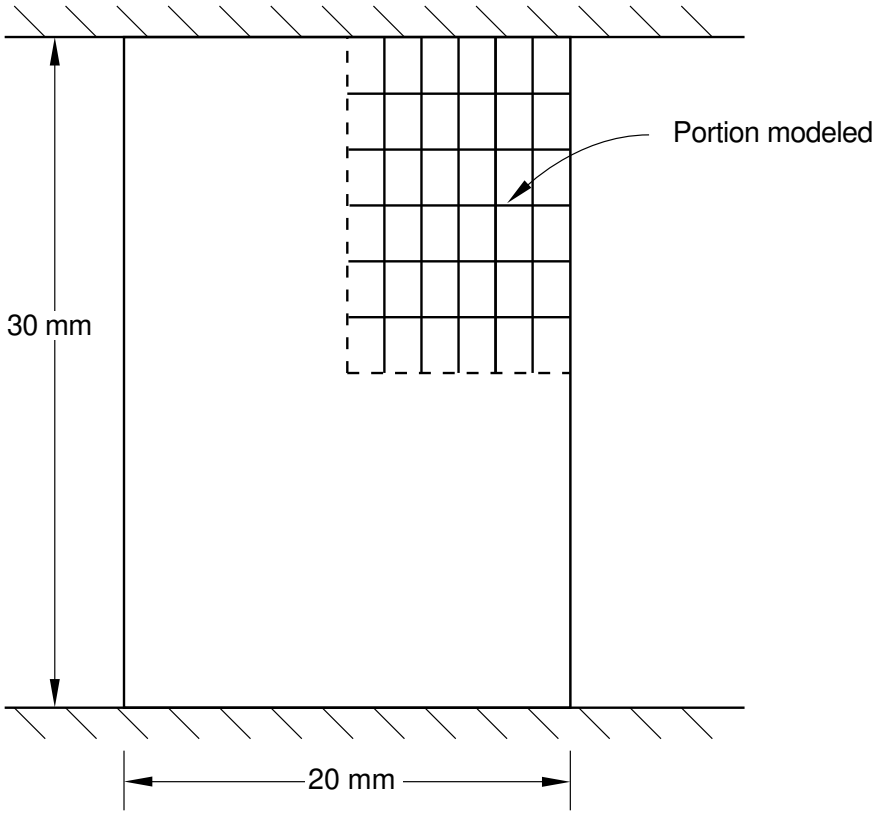


Figure 1: Axisymmetric upsetting example: geometry and mesh (element type CAX8RT).

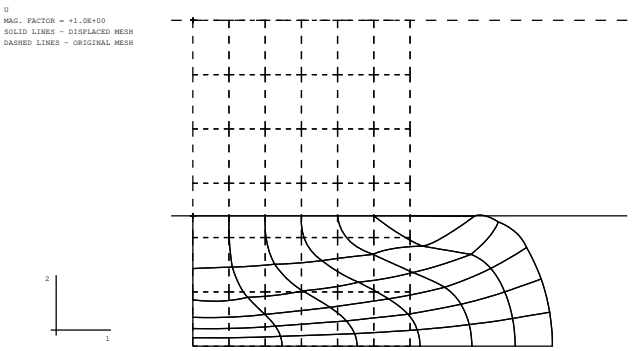


Figure 2: Deformed configuration at 60% upsetting: slow case, coupled temperature-displacement analysis, Abaqus/Standard.

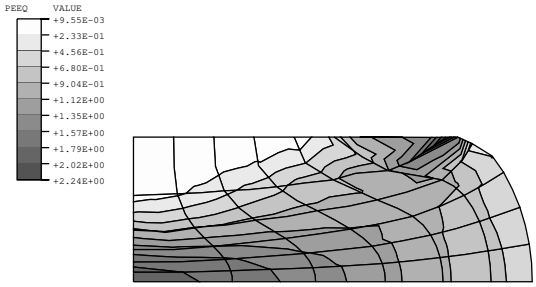


Figure 3: Plastic strain at 60% upsetting: slow case, coupled temperature-displacement analysis, Abaqus/Standard.

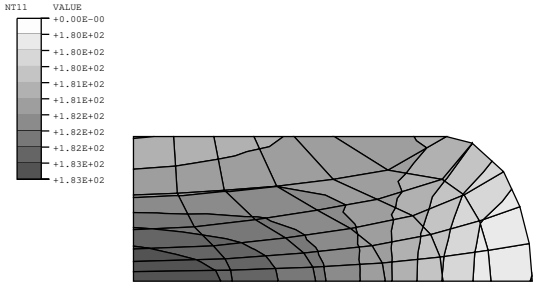


Figure 4: Temperature at 60% upsetting: slow case, coupled temperature-displacement analysis, Abaqus/Standard.

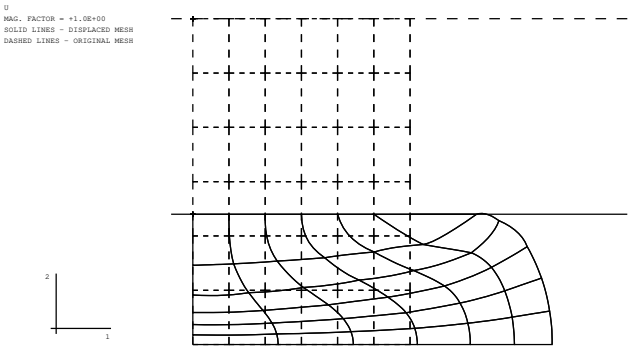


Figure 5: Deformed configuration at 60% upsetting: fast case, coupled temperature-displacement analysis, Abaqus/Standard.

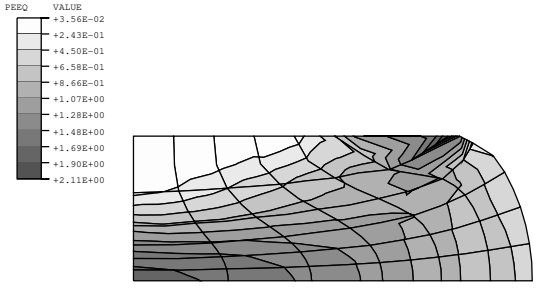


Figure 6: Plastic strain at 60% upsetting: fast case, coupled temperature-displacement analysis, Abaqus/Standard.

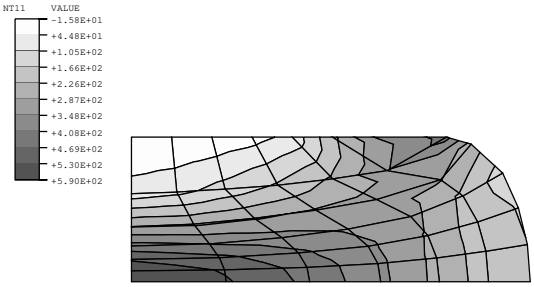


Figure 7: Temperature at 60% upsetting: fast case, coupled temperature-displacement analysis, Abaqus/Standard.

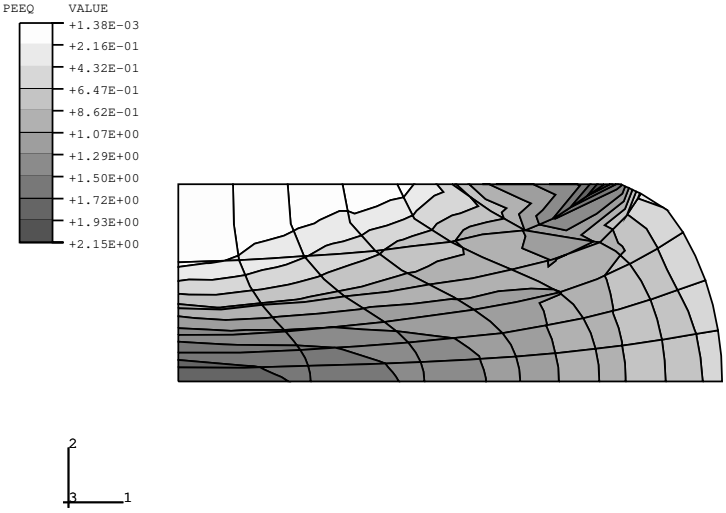


Figure 8: Plastic strain at 60% upsetting: fast case, adiabatic stress analysis, Abaqus/Standard.

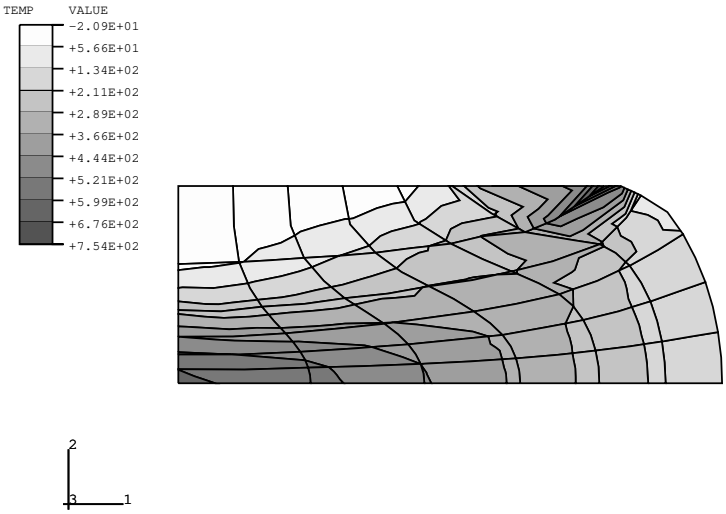


Figure 9: Temperature at 60% upsetting: fast case, adiabatic stress analysis, Abaqus/Standard.

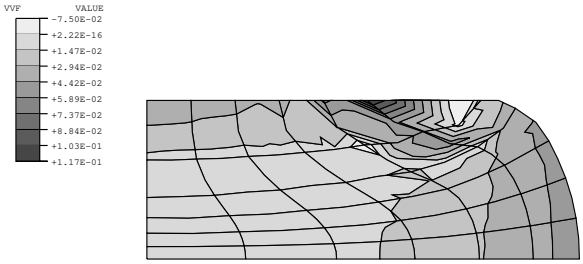


Figure 10: Void volume fraction at 60% upsetting: porous material, slow coupled temperature-displacement analysis, Abaqus/Standard.

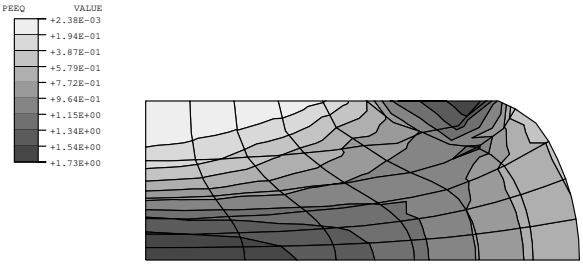


Figure 11: Plastic strain at 60% upsetting: porous material, slow coupled temperature-displacement analysis, Abaqus/Standard.

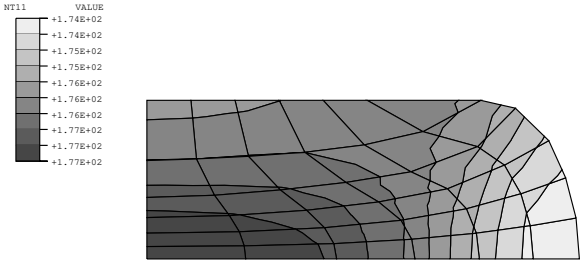


Figure 12: Temperature at 60% upsetting: porous material, slow coupled temperature-displacement analysis, Abaqus/Standard.

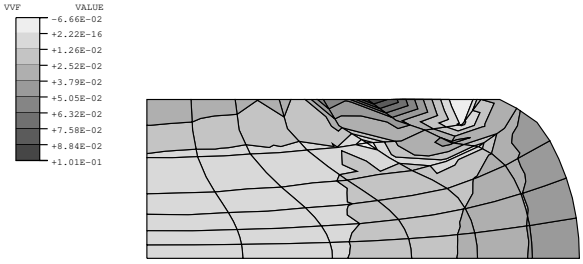


Figure 13: Void volume fraction at 60% upsetting: porous material, fast coupled temperature-displacement analysis, Abaqus/Standard.

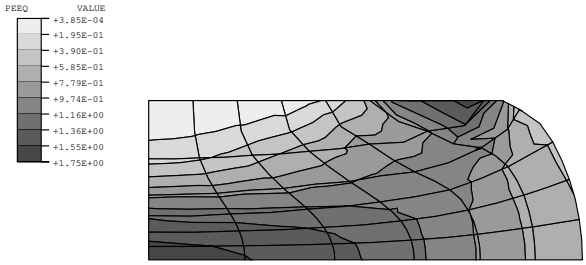


Figure 14: Plastic strain at 60% upsetting: porous material, fast coupled temperature-displacement analysis, Abaqus/Standard.

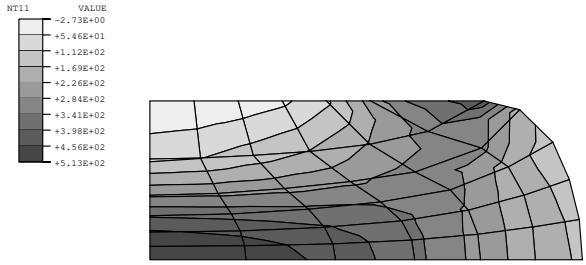


Figure 15: Temperature at 60% upsetting: porous material, fast coupled temperature-displacement analysis, Abaqus/Standard.

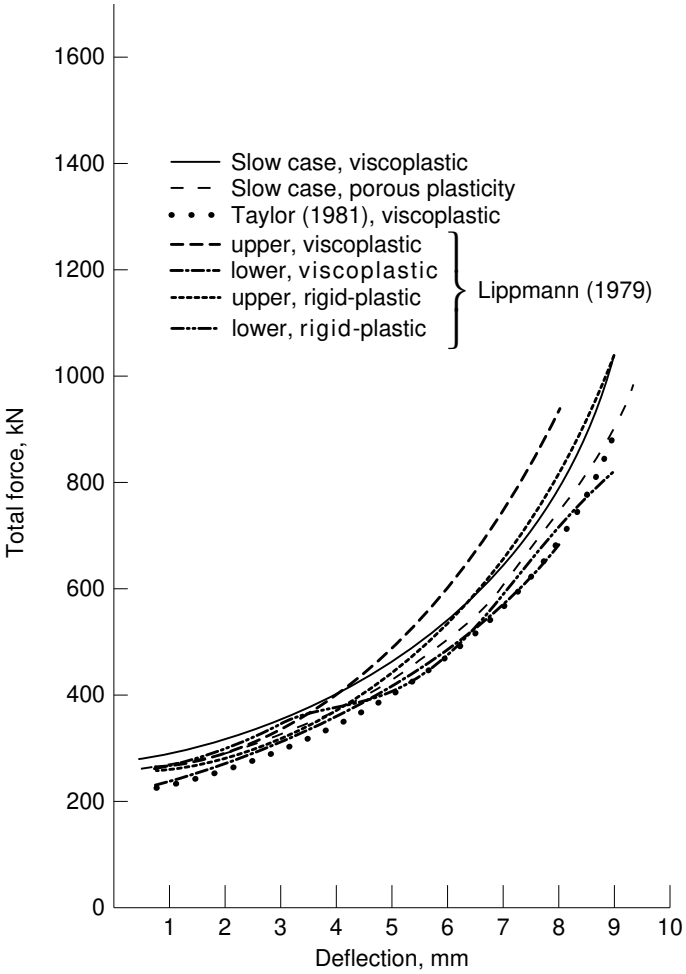


Figure 16: Force-deflection response for slow cylinder upsetting, Abaqus/Standard.

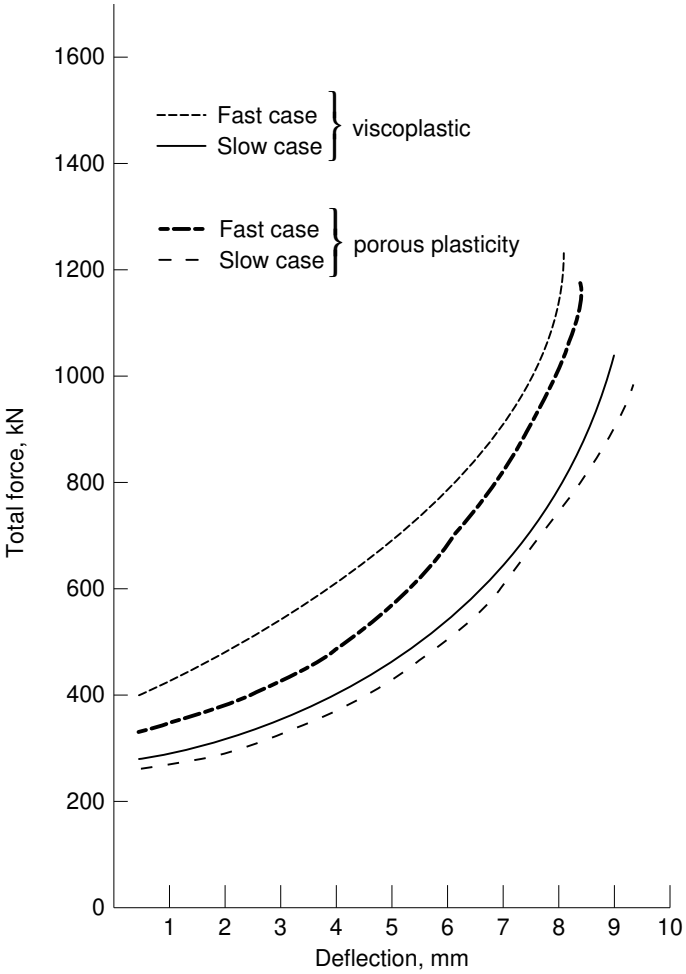


Figure 17: Rate dependence of the force-deflection response, Abaqus/Standard.

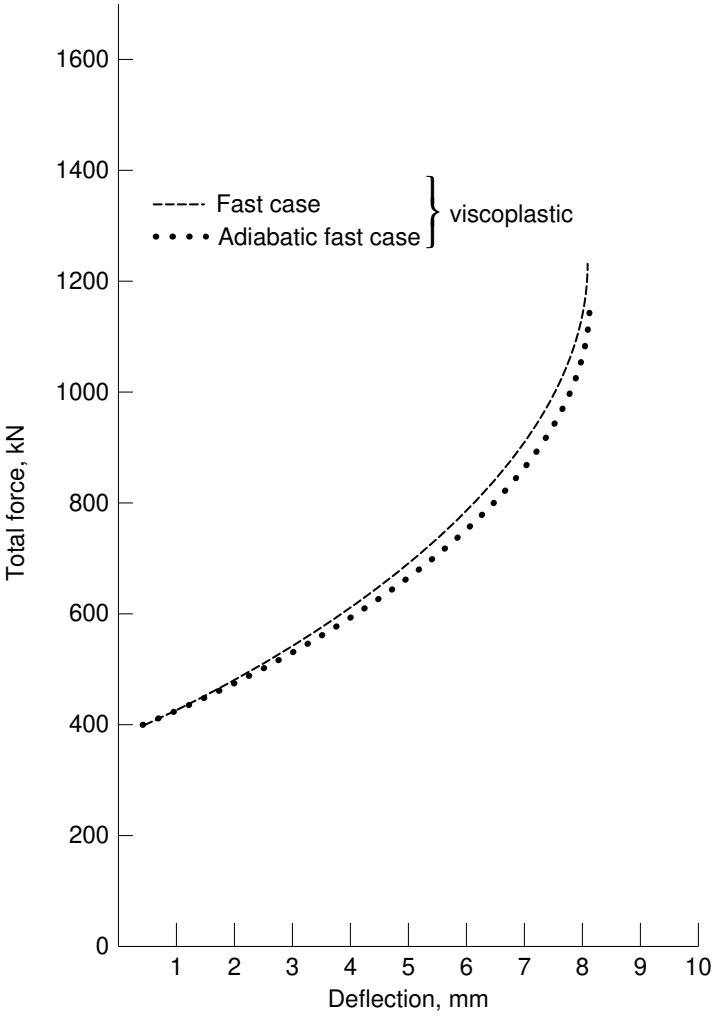


Figure 18: Force-deflection response: adiabatic versus fully coupled analysis, Abaqus/Standard.

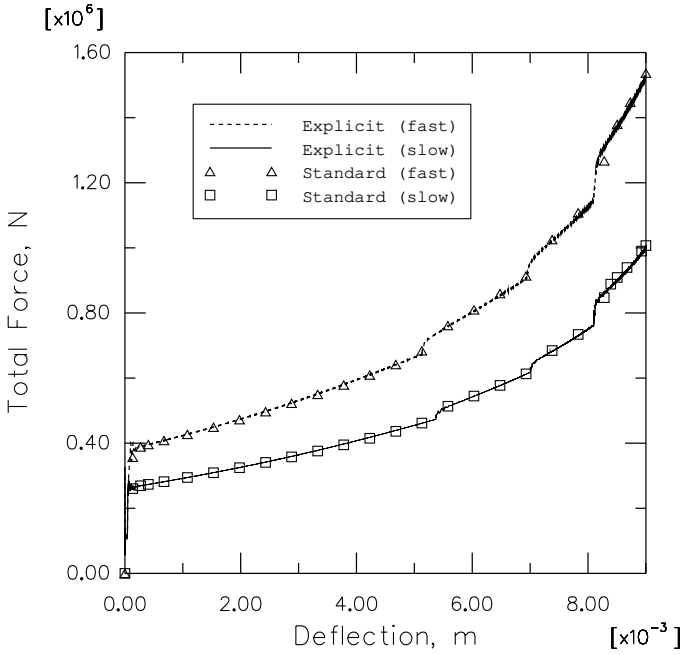


Figure 19: Force-deflection response: Abaqus/Explicit versus Abaqus/Standard.

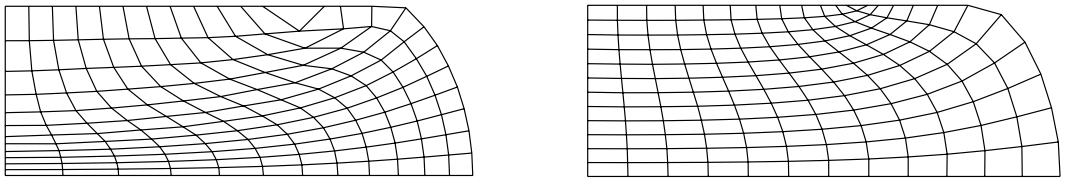


Figure 20: Deformed configuration at 60% upsetting: slow case, Abaqus/Explicit (without adaptive meshing, left; with adaptive meshing, right).

Unstable static problem: thermal forming of a metal sheet

Products: Abaqus/Standard

Overview

This example demonstrates the use of automatic contact stabilization to avoid unstable static problems.

Geometrically nonlinear static problems can become unstable for a variety of reasons. Instability may occur in contact problems, either because of chattering or because contact intended to prevent rigid body motions is not established initially. Localized instabilities can also occur; they can be either geometrical, such as local buckling, or material, such as material softening.

This problem models the thermal forming of a metal sheet; the shape of the die may make it difficult to place the undeformed sheet exactly in initial contact, in which case the initial rigid body motion prevention algorithm is useful. Metal forming problems are characterized by relatively simply shaped parts being deformed by relatively complex-shaped dies. The initial placement of the workpiece on a die or the initial placement of a second die may not be a trivial geometrical exercise for an engineer modeling the forming process. Abaqus accepts initial penetrations in contact pairs and instantaneously tries to resolve them; as long as the geometry allows for this to happen without excessive deformation, the misplacement of the workpiece usually does not cause problems. On the other hand, if the workpiece is initially placed away from the dies, serious convergence problems may arise. Unless there are enough boundary conditions applied or a stabilization method is used, singular finite element systems of equations result because one or more of the bodies has free rigid body motions. This typically arises when the deformation is applied through loads instead of boundary conditions. Contact stabilization can be helpful for avoiding convergence problems while contact is established without significantly influencing the results of interest (see *Automatic Stabilization of Rigid Body Motions in Contact Problems*).

This example looks at the thermal forming of an aluminum sheet. The deformation is produced by applying pressure and gravity loads to push the sheet against a sculptured die. The deformation is initially elastic. Through heating, the yield stress of the material is lowered until permanent plastic deformations are produced. Subsequently, the assembly is cooled and the pressure loads are removed, leaving a formed part with some springback. Although the sheet is initially flat, the geometrical nature of the die makes it difficult to determine the exact location of the sheet when it is placed on the die. Therefore, an initial gap between the two bodies is modeled, as shown in *Figure 1*.

Geometry and model

The model consists of a trapezoidal sheet 10.0 m (394.0 in) long, tapering from 2.0 m (78.75 in) to 3.0 m (118.0 in) wide, and 10.0 mm (0.4 in) thick. The die is a ruled surface controlled by two circles of radii 13.0 m (517.0 in) and 6.0 m (242.0 in) and dimensions slightly larger than the sheet. The sheet is initially placed over 0.2 m (7.9 in) apart from the die. The sheet has a longitudinal symmetry boundary condition, and one node prevents the remaining nodes from experiencing in-plane rigid body motion. The die is fixed throughout the analysis. The sheet mesh consists of 640 S4R shell elements, while the die is represented by 640 R3D4 rigid elements. The material is an aluminum alloy with a flow stress of 1.0×10^8 Pa (14.5 ksi) at room temperature. A flow stress of 1.0×10^3 Pa (0.15 psi) at 400°C is also provided, essentially declaring that at the higher temperature the material will flow plastically at any stress. A Coulomb friction coefficient of 0.1 is used to model the interaction between the sheet and die.

Results and discussion

The analysis consists of three steps. In the first step a gravity load and a pressure load of 1.0×10^5 Pa (14.5 psi) are applied, both pushing the sheet against the die. This step is aided by contact stabilization to prevent unrestrained motion of the sheet prior to establishing contact with the die. In this case, contact stabilization normal to the

nearby contact surfaces provides adequate stabilization. Avoiding tangential contact stabilization is recommended, if possible, because tangential contact stabilization is more likely to influence solution variables. The clearance distance range over which the contact stabilization is effective has been specified in this example such that contact stabilization is active for the initial separation distance between the sheet and die. Abaqus ramps down the contact stabilization such that no contact stabilization remains at the end of the first step. This guarantees that the viscous forces decrease to zero, thus avoiding any discontinuity in the forces at the start of the next step. The shape and relatively low curvatures of the die are such that the deformation at the end of the step is elastic (*Figure 2*). In the second step a two-hour heating (from room temperature to 360°C) and cooling (back to 50°C) cycle is applied to the loaded assembly. As a result of the decrease in flow stress permanent (plastic) deformation develops, as shown in *Figure 3*. Finally, in the third step the pressure load is removed and the springback of the deformed sheet is calculated, as depicted in *Figure 4*.

Acknowledgements

SIMULIA would like to thank British Aerospace Airbus, Ltd. for providing the basic data from which this example was derived.

Input files

unstablestatic_forming.inp	Thermal forming model.
unstablestatic_forming_surf.inp	Thermal forming model with surface-to-surface contact.

Figures

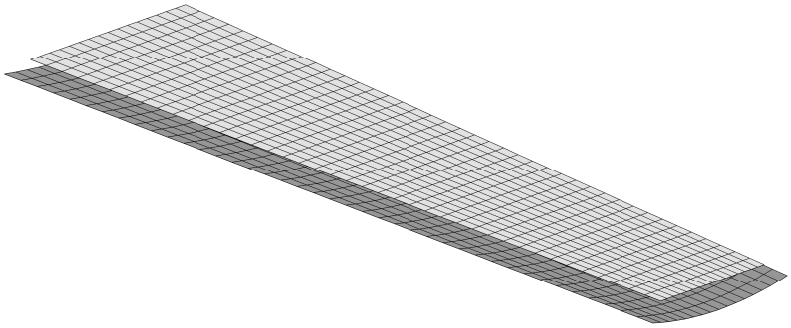


Figure 1: Initial placement of the sheet apart from the die.

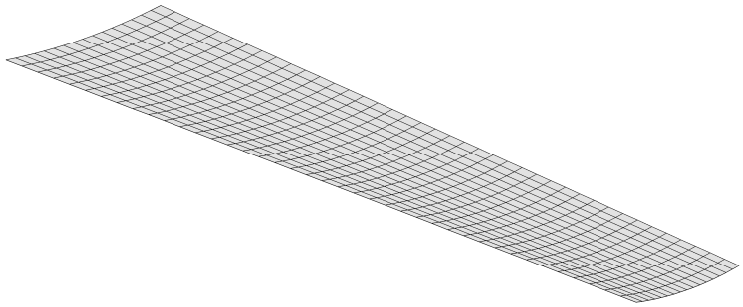


Figure 2: Elastic deformation after gravity and pressure loading.

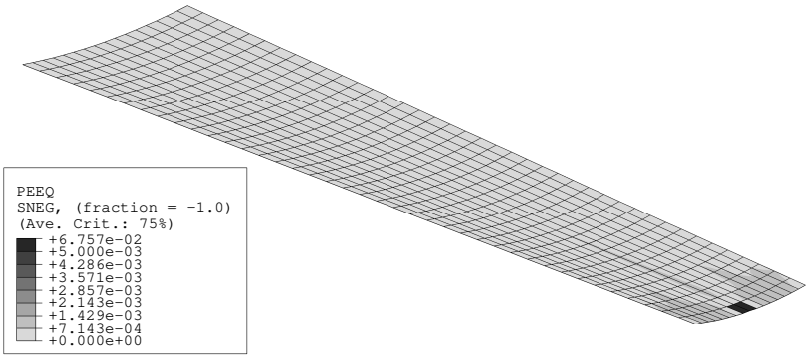


Figure 3: Permanent deformation produced by heating.

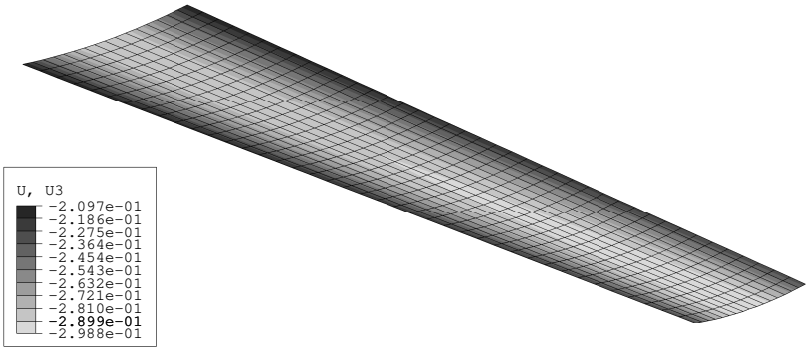


Figure 4: Springback.

Inertia welding simulation using Abaqus/Standard and Abaqus/CAE

Products: Abaqus/Standard Abaqus/CAE

References:

- *Mesh-to-Mesh Solution Mapping*
- **MAP SOLUTION*
- *FRIC*
- *UEL*
- *Terminating an Analysis*
- *Part object*

Overview

This example shows how simulation is an important tool in the inertia welding process design.

This example demonstrates the following Abaqus features:

- thermomechanical coupling for inertia welding simulation,
- semi-automatic remeshing using Python scripting and output database scripting methods for extracting deformed configurations,
- defining a complex friction law in a user subroutine,
- flywheel loading through user subroutine definitions, and
- combining and presenting results from a sequence of output database (`.odb`) files.

Application description

This example examines the inertia friction welding process of the pipes shown in *Figure 1*. The specific arrangement considered is the resulting as-welded configuration shown in *Figure 2*. In this weld process kinetic energy is converted rapidly to thermal energy at a frictional interface. The resulting rapid rise in interface temperature is exploited to produce high-quality welds. In this example the weld process is simulated, and the initial temperature rise and material plastic flow are observed. An important factor in the process design is control of the initial speed of the flywheel so that, when the flywheel stops, the temperature rises to just below the melting point, which in turn results in significant flow of material in the region of the weld joint. Understanding the friction, material properties, and heat transfer environment are important design aspects in an effective inertia welding process; therefore, simulation is a helpful tool in the process design.

Geometry

The weld process in this example is shown in *Figure 1*, where two pipes are positioned for girth-weld joining. The two pipes are identical, each with a length of 21.0 mm, an inside radius of 42.0 mm, and an outside radius of 48.0 mm. The pipes are adjacent, touching each other initially at the intended weld interface.

Materials

The pipes are made of Astroloy, a high-strength alloy used in gas turbine components. *Figure 3* shows flow stress curves as a function of temperature and plastic strain rate. At temperatures relevant to the welding process,

the material is highly sensitive to plastic strain rate and temperature. Specific heat is a function of temperature, as shown in *Figure 4*.

Other material properties are defined as follows:

Young's modulus:	180,000 MPa
Poisson's ratio:	0.3
Density:	7.8×10^{-9} Mg/mm^3
Conductivity:	14.7 W/m/°C at 20°C
	28 W/m/°C at 1200°C

Initial conditions

The pipes are initially set at 20°C, representing room temperature.

Boundary conditions and loading

A pressure of 360 MPa is applied to the top surface of the upper pipe. The initial rotational velocity of the flywheel is set at 48.17 rad/s, or 7.7 revolutions per second. The mass moment of inertia of the flywheel is 102,000 Mg mm^2.

Interactions

The principal interaction occurs at the weld interface between the pipes; however, a secondary concern is the possibility of contact of weld flash with the side of the pipes. The weld-interface friction behavior is assumed to follow that described by Moal and Massoni (1995), where the ratio of shear stress to the prescribed pressure is observed to be a complex function of interface slip rate. The heat generation from the frictional sliding, combined with plastic deformation, contributes to the temperature rise in the pipes.

Abaqus modeling approaches and simulation techniques

Abaqus/CAE and Abaqus/Standard are used together to affect the weld simulation in a way that permits extreme deformation of the pipes in the weld region. This process is automated through the use of Python scripts. Three cases are studied in this example.

Summary of analysis cases

Case 1	Initial flywheel velocity = 48.17 rad/s. This case produces a successful weld.
Case 2	Initial flywheel velocity = 20.0 rad/s. This case illustrates an unsuccessful weld scenario; the flywheel has insufficient energy to begin the weld process.
Case 3	Initial flywheel velocity = 70.0 rad/s. This case illustrates an unsuccessful weld scenario; the flywheel has excessive energy, resulting in a temperature rise into the liquidus regime of the pipe material.

The following sections discuss analysis considerations that are applicable to all the cases. Python scripts that generate the model databases and Abaqus/Standard input files are provided for Case 1, with instructions in the scripts for executing the Case 2 and Case 3 simulations.

Analysis types

The analysis is nonlinear, quasi-static with thermomechanical coupling. A fully coupled temperature-displacement procedure is used.

Analysis techniques

The key feature required for successful simulation of this process is remeshing. In this example, because of the large deformation near the weld region, multiple analyses are employed to limit element distortion. These analyses are executed in sequence, with remeshing performed between executions, and are automated through the use of Python scripts.

At each remesh point the current model configuration represents a significant change in the pipes' shape and in the current analysis mesh. Abaqus/CAE is used to extract the outer surface of the pipes, reseed the surface, and remesh the pipe regions. This process employs the Abaqus Scripting Interface*PartFromOdb(...)*command, which is used to extract orphan mesh parts representing the deformed pipes. These parts are then passed to the *Part2DGeomFrom2DMesh(...)* command. This command creates a geometric Part object from the orphan mesh imported earlier. Once the profile of the deformed part has been created, options in the Mesh module are used to remesh the part. The new mesh results in a new Abaqus/Standard analysis, and the map solution procedure maps state variables from the previous analysis (see *Mesh-to-Mesh Solution Mapping*).

Mesh design

The pipes are modeled as axisymmetric. The element formulation used is the fully coupled temperature-displacement axisymmetric elements with twist degrees of freedom (element types CGAX4HT and CGAX3HT), where the twist degree of freedom enables modeling of rotation and shear deformation in the out-of-plane direction. The hybrid formulation is required to handle the incompressible nature of the material during the plastic flow. The mesh is divided into two regions for each pipe. In the region near the weld interface, smaller elements are created (see *Figure 5*). During the remeshing process, the region near the weld surface is recalculated so that the new flash region is also meshed with smaller elements (see *Figure 6*).

Materials

The material model defined for this example approximates the high-temperature behavior of Astroloy, where it is reported by Soucail et al. (1992) using a Norton-Hoff constitutive law to describe the temperature and strain-rate viscoplastic behavior. A similar model is defined in Abaqus as a rate-dependent perfectly plastic material model. For the loading in this model, these material parameters result in the onset of local plastic flow only after the interface temperature has exceeded roughly 1200°C, near the material solidus temperature of 1250°C. Above this temperature the Mises flow stress is highly sensitive to variations in temperature and strain rate. A special adjustment in the flow stress at high strain rates is necessary to avoid divergence during the iteration procedure of the nonlinear solution. In the material model definition an extreme case of stress data is defined when the strain rate is 1.0×10^{6} s^{-1}. Stress data when the strain rate equals zero are also defined to be the same as the stress data at strain rate 1.0×10^{-5} s^{-1}. As a result of large deformation, thermal expansion is not considered in the material model.

It is assumed that 90% of the inelastic deformation energy contributes to the internal heat generation, which is the Abaqus default for inelastic heat fraction.

Initial conditions

An initial temperature of 20°C is specified for the entire model. The flywheel initial velocity is specified as a solution-dependent variable initial condition for the user element (see `inertiaweld_sub.f`).

Boundary conditions

The bottom pipe is fixed completely. The top pipe is free of boundary conditions.

Loads

A pressure of 360 MPa is applied as a distributed load on the top surface of the upper pipe.

The heat flux resulting from frictional heat generation is considered to be many times greater than any heat lost due to convection. Thus, all boundaries are assumed to be adiabatic.

Interactions

Five interactions are defined: four contact interactions and one actuator-sensor interaction.

The contact interactions include a pair of interactions that define the weld interface between the pipes, which is identified in *Figure 1*. This pair of interactions is symmetrical: one interaction defines the top pipe as the main surface with the bottom pipe as the secondary surface, and the second interaction reverses the main-secondary pairing. This "balanced main-secondary" arrangement is important for the analysis to obtain more uniform contact pressure in the weld interface and to avoid hourglass effects, and it is combined with a softened contact interaction property to promote a sharing of the local contact pressure among nodes on both sides of the interface.

To simulate the Moal and Massoni (1995) friction definition, the weld interface friction model is defined in user subroutine *FRIC* (see `inertiaweld_sub.f`). This model is nonlocal in the sense that the interface pressure for all contact nodes is the applied pressure of 360 MPa, and the sliding velocity is computed based on the rotational angular velocity of the flywheel. This treatment of the friction force helps stabilize the solution. Frictional heat generation is calculated based on the frictional traction and the sliding velocity. Nondefault contact controls definitions are used to improve convergence. These definitions include delaying the friction computation upon contact and automatic tolerance control to avoid contact chattering.

The two remaining contact interactions address the possibility of self-contact near the weld area in the pipes. Self-contact in the flash area can cause problems during remeshing: the *Part2DGeomFrom2DMesh(...)* command that was used to generate the new, current configuration geometry performs curve-fit operations that can result in self intersections of the boundary, which lead to invalid part topology and a meshing failure. To avoid this problem, a softened contact model that introduces a normal pressure with a small separation distance (−0.01 mm) is used. It is important to keep this separation distance as small as practical to avoid causing any nonphysical contact behavior.

The actuator-sensor interaction, which acts through user subroutine *UEL*, enables the simulation of a flywheel attached to the top pipe. The user element also has a sensor role in the analysis, measuring the weld upset, or axial shortening, of the weld assembly. When a critical user-defined upset distance is exceeded, the user element subroutine calls `XIT` to terminate the analysis and signal a remesh point. This parameter, the allowed "upset distance," correlates well with the extent of mesh deformation in the weld region.

Analysis steps

Each of the analyses run in the simulation sequence uses a single fully coupled temperature-displacement step. Since the duration of the step is controlled by the user element, as described above, it is not known a priori for each analysis. Therefore, the step duration is set in each analysis to be the total remaining time for the simulation, a time that is reached only in the final analysis in the simulation sequence.

Solution controls

Full Newton iteration is used for the nonlinear solution. Automatic time stepping is based on the maximum temperature change in each increment. Results from the previous increment are used as the initial guess in the current increment, with the parabolic extrapolation technique adopted to transfer the previous state to the current state.

Heat fluxes due to frictional heat generation can cause contact chattering. To overcome this difficulty, contact controls with the automatic tolerances parameter are used.

Output requests

Field output of temperature, stress, and equivalent plastic strain is requested. Various energy measures are requested for history output, allowing you to record the flywheel energy change during the simulation.

Convergence

In general, convergence is difficult in the initial increments, especially after remeshing. More iterations are also observed near the end of the analysis-run sequence when deformation increases and the mesh distortion becomes severe.

Run procedure

This example is run using the Abaqus/CAE Python scripts provided. Use the following command to run the example:

```
abaqus cae -script inertiaweld.py
```

The simulation results in a sequence of separate output database (`.odb`) files, each corresponding to one of the analyses.

For postprocessing convenience a number of scripts are provided for evaluating simulation results across the sequence of files. Use the following command to create the `inertia_weld_animation.avi` animation file:

```
abaqus cae -script inertiaweld_movie.py
```

Run the following Python scripts to create *X*–*Y* plots:

- *`inertiaweld_xy.py`*, for a history plot of energy results.
- *`inertiaweld_upset.py`*, for a history plot of assembly upset.
- *`inertiaweld_maxtemp.py`*, for a history plot of the maximum temperature.

Discussion of results and comparison of cases

This example focuses mainly on the successful weld case, Case 1. Case 2 and Case 3 are provided to show the precarious nature of the inertia weld process, where it is essential to select the flywheel size correctly, measured by the mass moment of inertia, and the initial velocity of the flywheel. Case 2 and Case 3 illustrate why simulation is a valuable tool in the weld process design as each case clearly shows that the selected process parameters do not result in a successful weld.

Welding success: Case 1

The example simulation creates multiple output database (`.odb`) files, requiring 22 remeshings to reach the simulation time of 5.0 seconds. The results in the first analysis before the first remeshing show that the temperature rises very fast near the weld interface. At about 1.6 seconds the temperature reaches 1172°C and the material starts to flow, squeezing out to form flash (see *Figure 7*). After 2.31 seconds and 5 remeshings the flash extends enough to fold back and contact the pipe (see *Figure 8*). At 3.51 seconds the flywheel velocity slows down to 3.51 rad/s, the temperature starts to drop, and the material flow slows down. At this point a considerable amount of flash build-up can be observed (see *Figure 9*). After 5.0 seconds the flywheel stops, the temperature drops below 1000°C, and the pipes are welded (see *Figure 10*).

The flash prediction in *Figure 11* is in good agreement with those shown by Moal and Massoni (1995). *Figure 12* plots the kinetic energy history (created with *`inertiaweld_xy.py`*) and shows how the flywheel slows down and stops. The “RemeshPoints” marks indicate the time at which the remeshing procedure occurred. The upset distance during the welding process is shown in *Figure 13* (created with *`inertiaweld_upset.py`*). The curve of the maximum temperature for Case 1 is plotted in *Figure 14* (created with *`inertiaweld_maxtemp.py`*).

Insufficient flywheel energy: Case 2

If a lower initial flywheel velocity is selected, the temperature may not reach a level high enough for the material to flow. For this case the initial velocity is reduced to 20 rad/s. *Figure 15* shows the configuration at 5.0 seconds, where not much deformation is observed and the temperature near the interface is about 250°C. The history plot of the maximum temperature for Case 2 in *Figure 14* shows the pipe temperature reaches only 700°C about 1.1 seconds before it cools down. In this case the material is not hot enough to initiate the material flow, and welding will not be successful.

Excessive flywheel energy: Case 3

If a higher initial velocity of the flywheel is selected, the material becomes so hot that it starts to melt. For this case the initial velocity of the flywheel is set at 70 rad/s. *Figure 16* shows that the temperature rapidly reaches 1360°C (at about 0.9 seconds), which is well beyond the melting temperature at 1250°C, before convergence failures stop the analysis. *Figure 14* shows the history plot of the maximum temperature in the pipes for Case 3. In this case the excessive energy results in melting and a failed weld.

Python scripts

inertiaweld.py	Main Python script to control model building, remeshing, and the analyses.
inertiaweld_solver_param.py	Script to initialize base options for an Abaqus/Standard analysis.
inertiaweld_utils.py	Script to define utility functions, such as applying seeds and replacing the part after deformation.
inertiaweld_original_build.py	Script to create the initial model.
inertiaweld_job_param.py	Script to provide control parameters and input data, such as mesh size and simulation time.
inertiaweld_remesh_build.py	Script to create a remeshed model and provide input files for restart.
inertiaweld_movie.py	Script to create an animation of this welding simulation.
inertiaweld_xy.py	Script to create an *X*–*Y* plot of flywheel kinetic energy history.
inertiaweld_upset.py	Script to create an *X*–*Y* plot of the upset history.
inertiaweld_maxtemp.py	Script to create an *X*–*Y* plot of the maximum temperature history during the welding process.

Input files

inertiaweld_sub.f	Fortran file containing user subroutines *UEL* and *FRIC*.

References

- Moal, A., and E. Massoni, “Finite Element Simulation of the Inertia Welding of Two Similar Parts,” Engineering Computations, vol. 12, pp. 497–512, 1995.
- Soucail, M., A. Moal, L. Naze, E. Massoni, C. Levaillant, and Y. Bienvenu, “Microstructural Study and Numerical Simulation of Inertia Friction Welding of Astroloy,” 7th International Symposium on Superalloys, Seven Springs, USA, 1992.

Figures

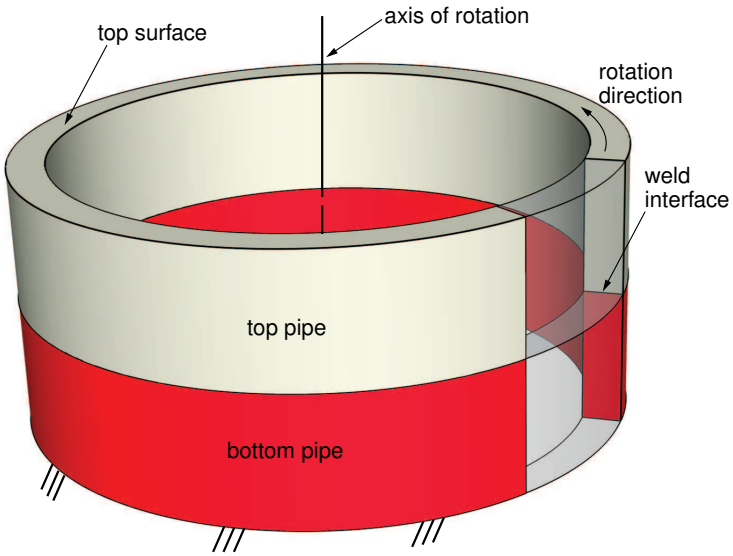


Figure 1: Inertia weld configuration for two pipes.

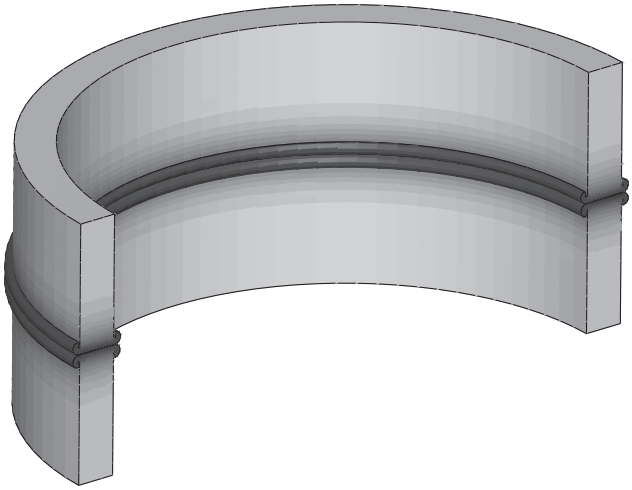


Figure 2: Half-section view of the inertia welding final as-welded configuration showing the temperature fields at 5 seconds after welding begins.

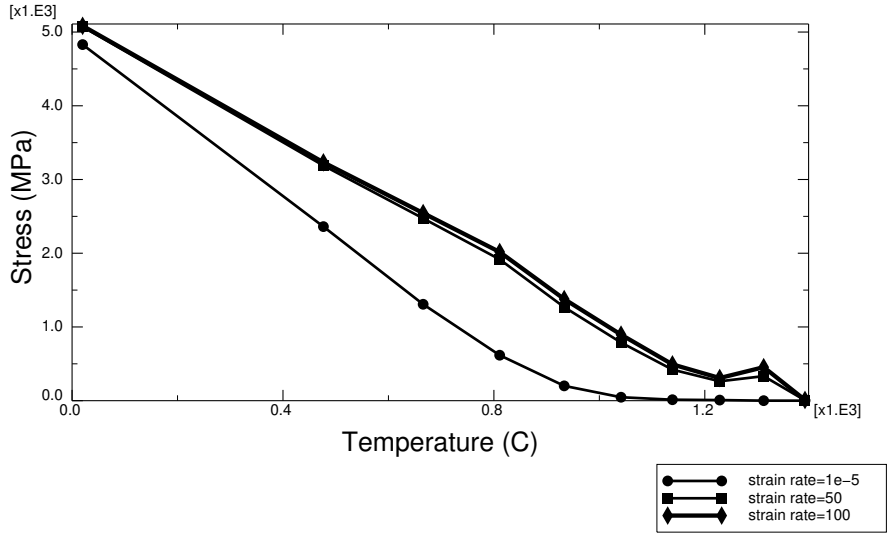


Figure 3: Flow stress curves.

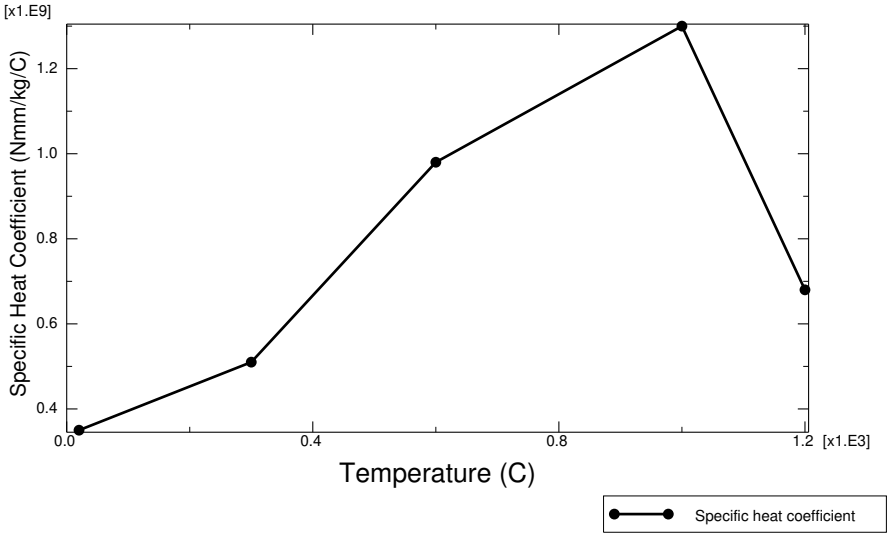


Figure 4: Specific heat curve (N mm/kg/°C).

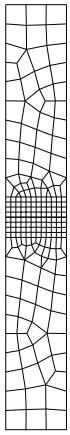


Figure 5: Initial mesh.

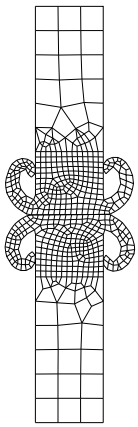


Figure 6: Typical mesh design on a deformed configuration pair of parts.

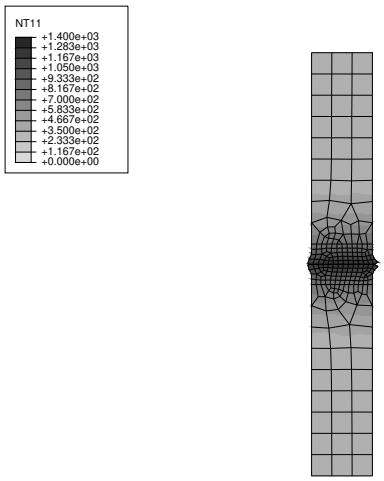


Figure 7: Temperature at 1.78 sec.

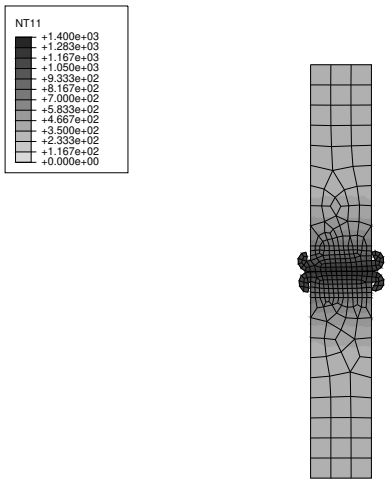


Figure 8: Temperature at 2.31 sec.

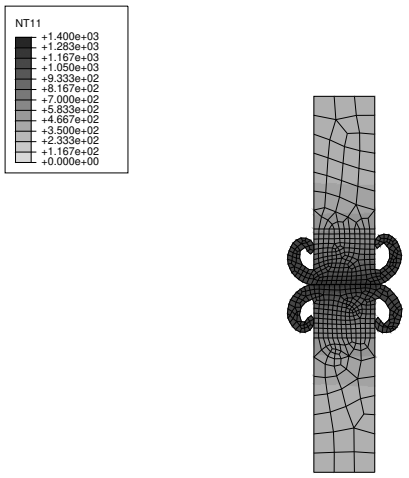


Figure 9: Temperature at 3.51 sec.

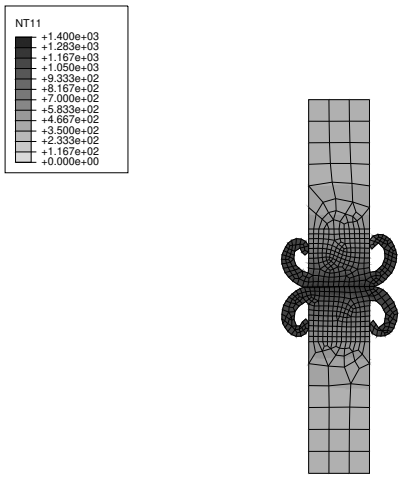


Figure 10: Temperature at 5.0 sec.

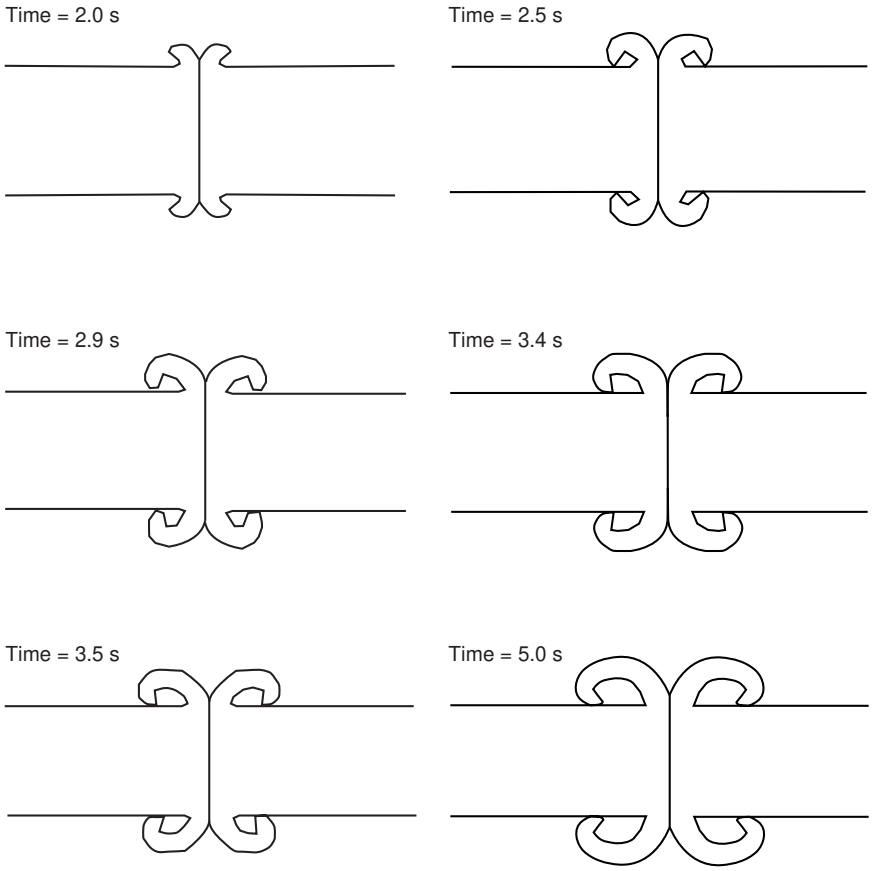


Figure 11: Flash prediction in current analysis.

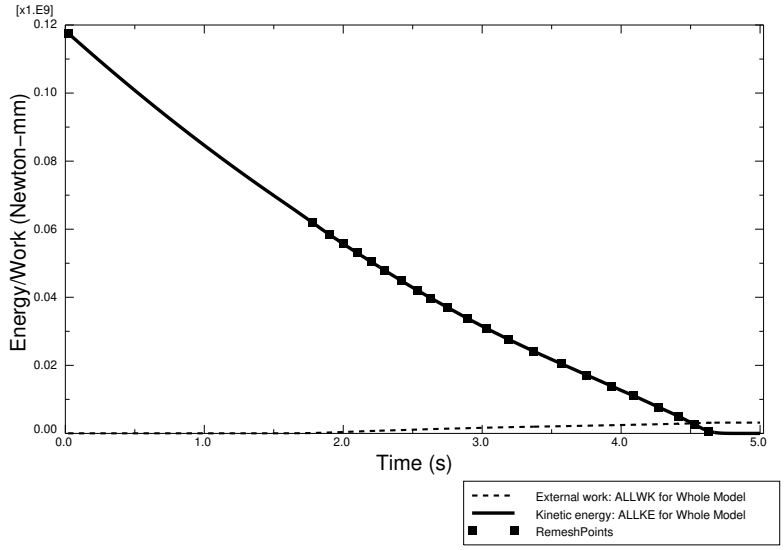


Figure 12: Kinetic energy history showing the flywheel slowdown.

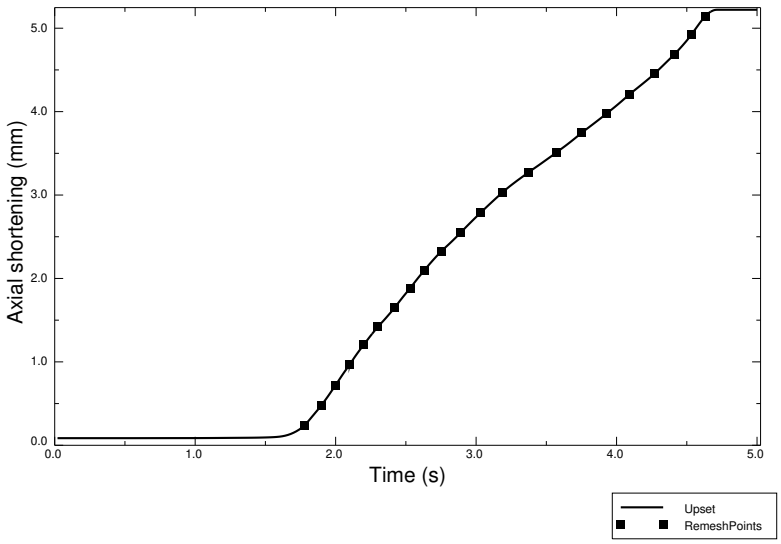


Figure 13: Assembly upset (shortening) history.

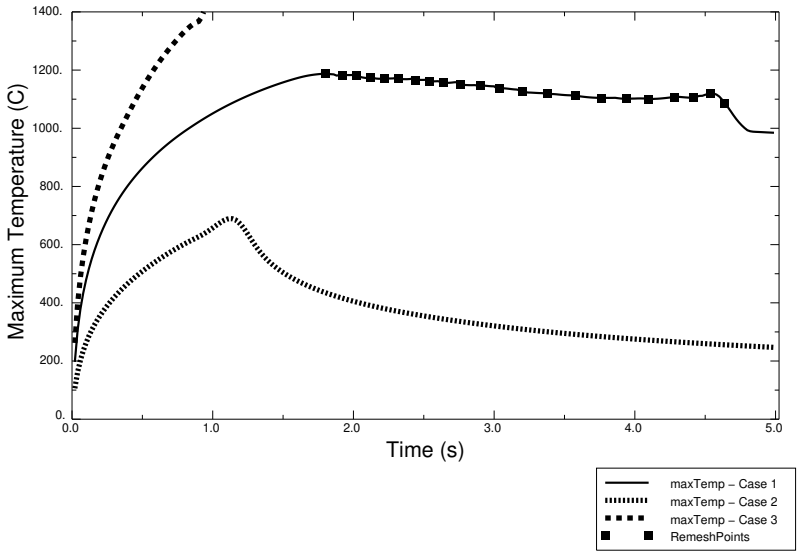


Figure 14: Maximum temperature history comparisons of the three cases. Case 1: Successful weld; Case 2: Failed weld with insufficient flywheel energy; Case 3: Failed weld with excessive flywheel energy.

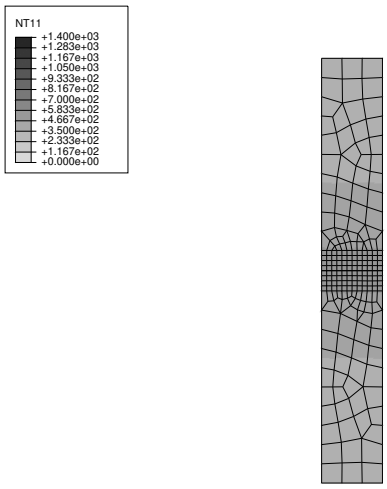


Figure 15: Temperature distribution at 5.0 sec for Case 2 (insufficient flywheel energy for welding).

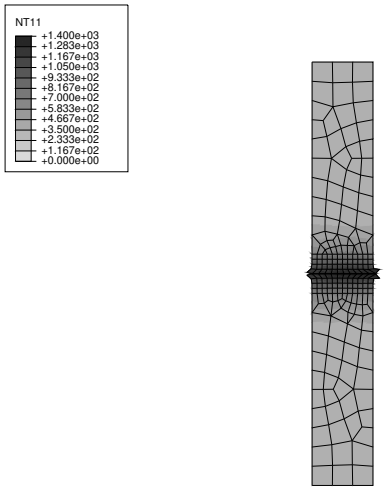


Figure 16: Temperature distribution at 0.9 sec for Case 3 (excessive initial flywheel energy results in a rapid temperature rise into the melting regime).

Moldflow translation examples

Products: Abaqus/Standard

References:

• *Translating Moldflow Data to Abaqus Input Files*

Overview

This example illustrates how you can effectively couple Moldflow and Abaqus.

The following features of the **abaqus moldflow** translator are demonstrated:

- transforming finite element model information from a Moldflow analysis into a partial Abaqus input file,
- adding appropriate data for the analysis,
- performing natural frequency analyses, and
- calculating deformation due to initial stresses.

Application description

The results of a Moldflow simulation, which models the plastics injection mold-filling process, include calculations of material properties and residual stresses in the plastic part. Two types of parts, a filled bracket and an unfilled bracket, are studied in this example to illustrate how you can transform finite element model information from a Moldflow simulation into a partial Abaqus input file, adding boundary conditions and step data, and requesting output.

Abaqus modeling approaches and simulation techniques

This example illustrates the use of the **abaqus moldflow** translator.

Summary of analysis cases

Case 1	Natural frequency analysis of a fiber-filled bracket.
Case 2	Natural frequency analysis of an unfilled bracket.
Case 3	Deformation due to initial stresses in a three-dimensional filled bracket.

Run procedure

The three cases share the same general approach. The following procedure summarizes the typical usage of the **abaqus moldflow** translator:

1. Export the data from Moldflow simulation as follows:
   - For a midplane Moldflow simulation, export the finite element mesh data to a file named *job-name*`.pat` and the results data (material properties and residual stresses) to a file named *job-name*`.osp`.
   - For a three-dimensional solid Moldflow simulation using Moldflow Version MPI 6, run the Visual Basic script `mpi2abq.vbs` to export the finite element mesh data to a file named *job-name*`_mesh.inp` and the results data to `.xml` files.

2. Run the **abaqus moldflow** translator to create a partial Abaqus input (`.inp`) file from the Moldflow interface files.
3. Edit the Abaqus input file to add appropriate data for the analysis (for example, add boundary conditions and step data).
4. Submit the Abaqus input file for analysis.

Extracting the files

All files associated with these analyses are included with the Abaqus release; you can use the Abaqus**fetch** utility to extract example problem files from the compressed archive files.

To extract all the relevant files for a particular example problem, enter the following commands:

Case 1	`abaqus fetch job=moldflow_ex1*`
Case 2	`abaqus fetch job=moldflow_ex2*`
Case 3	`abaqus fetch job=bracket3d_mpi6*`

For information on using wildcard expressions with the Abaqus**fetch** utility, see *Fetching Sample Input Files*.

Case 1: Natural frequency analysis of a fiber-filled bracket

The bracket in Case 1 consists of 926 nodes and 1719 S3R elements. The model contains seven different element sets. Each element set has a different thickness and is modeled as a laminated composite with 20 layers.

Ten unrestrained vibration modes are computed. The first six frequencies are approximately zero. The frequencies for the first four flexible modes are listed in *Table 1*.

The Abaqus finite element model is shown in *Figure 1*.

Case 2: Natural frequency analysis of an unfilled bracket

Case 2 uses the same Abaqus finite element model as Case 1, but the material properties are transversely isotropic. The shell section definition is homogeneous instead of composite. Twenty-one equally spaced Simpson integration points are used through the shell thickness.

The frequencies for the first four flexible vibration modes of the unfilled bracket are listed in *Table 2*. The unfilled material in this example is softer than the filled material in Case 1; consequently, the frequencies are lower.

Case 3: Deformation due to initial stresses in a three-dimensional filled bracket

Case 3 uses a solid Abaqus finite element model that is similar to the model used in Case 1.

To execute the **abaqus moldflow** translator, enter the following command:

```
abaqus moldflow job=bracket3d_mpi6 3D initial_stress=on
```

A contour plot of initial stresses is shown in *Figure 2*.

Input files

Case 1

moldflow_ex1.inp	Input file to analyze a fiber-filled bracket.

Case 2

moldflow_ex2.inp Input file to analyze an unfilled bracket.

Tables

Table 1: Frequencies for the first four flexible modes for the fiber-filled bracket.

Mode	Frequency, Hz
7	334
8	430
9	740
10	752

Table 2: Frequencies for the first four flexible modes for the unfilled bracket.

Mode	Frequency, Hz
7	146
8	217
9	363
10	371

Figures

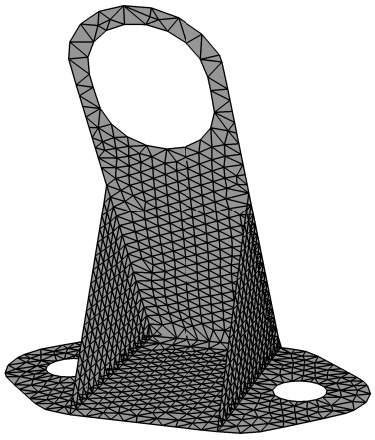


Figure 1: Finite element mesh of the fiber-filled bracket.

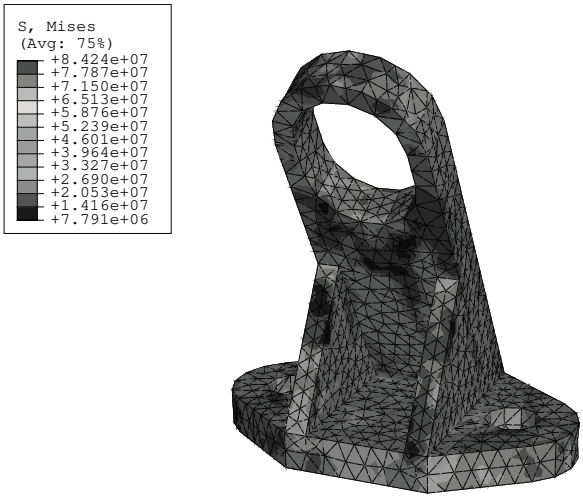


Figure 2: Contour plot of the initial stresses for the filled bracket using Moldflow Version MPI 6.

Fracture and damage

In this section:

- *A plate with a part-through crack: elastic line spring modeling*
- *Contour integrals for a conical crack in a linear elastic infinite half space*
- *Elastic-plastic line spring modeling of a finite length cylinder with a part-through axial flaw*
- *Crack growth in a three-point bend specimen*
- *Analysis of skin-stiffener debonding under tension*
- *Failure of blunt notched fiber metal laminates*
- *Debonding behavior of a double cantilever beam*
- *Debonding behavior of a single leg bending specimen*
- *Postbuckling and growth of delaminations in composite panels*

A plate with a part-through crack: elastic line spring modeling

Products: Abaqus/Standard

Overview

This example shows the use of line spring elements and the shell-to-solid submodeling technique to model a plate with a part-through crack.

The line spring elements in Abaqus allow inexpensive evaluation of the effects of surface flaws in shell structures, with sufficient accuracy for use in design studies. The basic concept of these elements is that they introduce the local solution, dominated by the singularity at the crack tip, into a shell model of the uncracked geometry. The relative displacements and rotations across the cracked section, calculated in the line spring elements, are then used to determine the magnitude of the local strain field and hence the *J*-integral and stress intensity factor values, as functions of position along the crack front. This example illustrates the use of these elements and provides some verification of the results they provide by comparison with a published solution and also by making use of the shell-to-solid submodeling technique.

Problem description

A large plate with a symmetric, centrally located, semi-elliptic, part-through crack is subjected to edge tension and bending. The objective is to estimate the Mode I stress intensity factor, K_I, as a function of position along the crack front. Symmetry allows one quarter of the plate to be modeled, as shown in *Figure 1*. The 8-node shell element, S8R, and the corresponding 3-node (symmetry plane) line spring element LS3S are used in the model.

A mesh using LS6 elements is also included. Only half-symmetry is used in this case. When LS6 elements are used, the shell elements on either side of an LS6 element must be numbered such that the normals to these shell elements point in approximately the same direction.

Geometry and model

For each load case (tension and bending) two plate thicknesses are studied: a “thick” case, for which the plate thickness is 76.2 mm (3.0 in); and a “thin” case, for which the plate thickness is 19.05 mm (0.75 in). For both thicknesses the semi-elliptic crack has a maximum depth (a_0 in *Figure 2*) of 15.24 mm (0.6 in) and a half-length, *c*, of 76.2 mm (3.0 in). The plate is assumed to be square, with dimensions 609.6 × 609.6 mm (24 × 24 in).

The material is assumed to be linear elastic, with Young's modulus 207 GPa (30×10^6 lb/in^2) and Poisson's ratio 0.3.

A quarter of the plate is modeled, with symmetry along the edges of the quarter-model at $x = 0$ and $y = 0$. On the edge containing the flaw ($y = 0$), the symmetry boundary conditions are imposed only on the unflawed segment of the edge, since they are built into the symmetry plane of the line spring element being used (LS3S).

The loading consists of a uniform edge tension (per unit length) of 52.44 kN/m (300 lb/in) or a uniform edge moment (per unit length) of 1335 N-m/m (300 lb-in/in).

Results and discussion

The stress intensity factors for the thick and thin plates are compared with the detailed solutions of Raju and Newman (1979) and Newman and Raju (1979) in *Figure 3* (tension load) and *Figure 4* (bending load). These plots show that the present results agree reasonably well with those of Raju and Newman over the middle portion

of the flaw ($\phi > 30°$), with better correlation being provided for the thick case, possibly because the crack is shallower in that geometry. The accuracy is probably adequate for basic assessment of the criticality of the flaw for design purposes. For values of ϕ less than about 30° (that is, at the ends of the flaw), the stress intensity values predicted by the line spring model lose accuracy. This accuracy loss arises from a combination of the relative coarseness of the mesh, (especially in this end region where the crack depth varies rapidly), as well as from theoretical considerations regarding the appropriateness of line spring modeling at the ends of the crack. These points are discussed in detail by Parks (1981) and Parks et al. (1981).

Shell-to-solid submodeling around the crack tip

An input file for the case $a_0/t = 0.2$, which uses the shell-to-solid submodeling capability, is included. This C3D20R element mesh allows the user to study the local crack area using the energy domain integral formulation for the *J*-integral. The submodel uses a focused mesh with four rows of elements around the crack tip. A $1/\sqrt{r}$ singularity is utilized at the crack tip, the correct singularity for a linear elastic solution. Symmetry boundary conditions are imposed on two edges of the submodel mesh, while results from the global shell analysis are interpolated to two edges by using the submodeling technique. The global shell mesh gives satisfactory *J*-integral results; hence, we assume that the displacements at the submodel boundary are sufficiently accurate to drive the deformation in the submodel. No attempt has been made to study the effect of making the submodel region larger or smaller. The submodel is shown superimposed on the global shell model in *Figure 5*.

The variations of the *J*-integral values along the crack in the submodeled analysis are compared to the line spring element analysis in *Figure 3* (tension load) and *Figure 4* (bending load). Excellent correlation is seen between the three solutions. A more refined mesh in the shell-to-solid submodel near the plate surface would be required to obtain *J*-integral values that more closely match the reference solution.

Input files

crackplate_ls3s.inp	LS3S elements.
crackplate_surfaceflaw.f	A small program used to create a data file containing the surface flaw depths.
crackplate_ls6_nosym.inp	LS6 elements without symmetry about $y = 0$.
crackplate_postoutput.inp	**POST OUTPUT* analysis.
crackplate_submodel.inp	Shell-to-solid submodel.

References

- Newman, J. C., Jr., and I. S. Raju, "Analysis of Surface Cracks in Finite Plates Under Tension or Bending Loads," *NASA Technical Paper 1578, National Aeronautics and Space Administration*, December 1979.
- Parks, D. M., "The Inelastic Line Spring: Estimates of Elastic-Plastic Fracture Mechanics Parameters for Surface-Cracked Plates and Shells," *Journal of Pressure Vessel Technology*, vol. 13, pp. 246–254, 1981.
- Parks, D. M., R. R. Lockett, and J. R. Brockenbrough, "Stress Intensity Factors for Surface-Cracked Plates and Cylindrical Shells Using Line Spring Finite Elements," *Advances in Aerospace Structures and Materials*, Edited by S. S. Wang and W. J. Renton, ASME, AD–01, pp. 279–286, 1981.
- Raju, I. S., and J. C. Newman, Jr., "Stress Intensity Factors for a Wide Range of Semi-Elliptic Surface Cracks in Finite Thickness Plates," *Journal of Engineering Fracture Mechanics*, vol. 11, pp. 817–829, 1979.

Figures

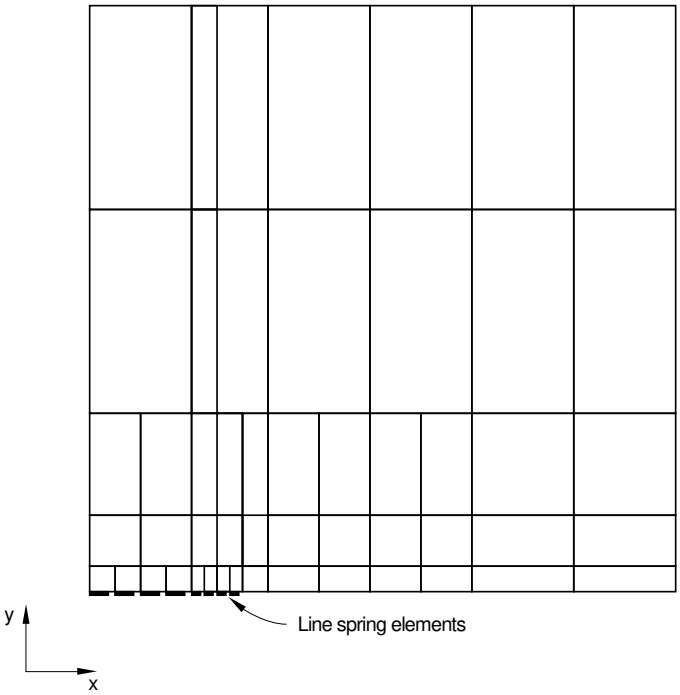
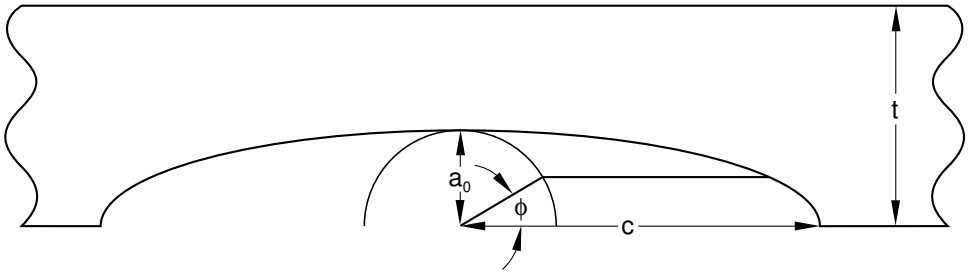


Figure 1: Quarter model of large plate with center surface crack.



a_0 = maximum flaw depth

2c = surface length of crack

t = shell thickness

ϕ = angle on an inscribed circle for locating a point on the crack

Figure 2: Schematic surface crack geometry for a semi-elliptical crack.

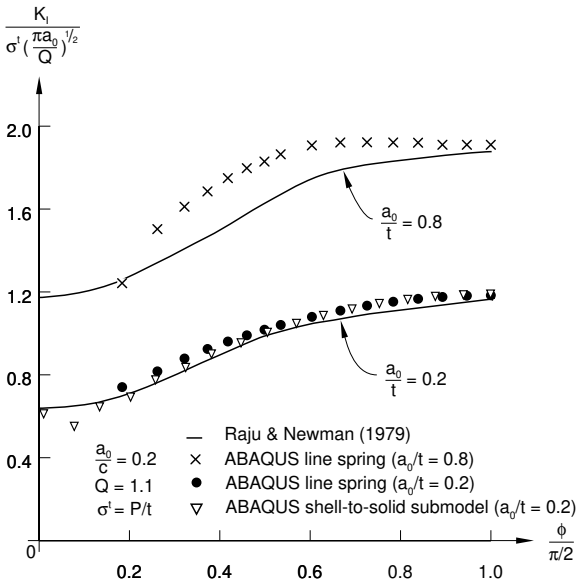


Figure 3: Stress intensity factor dependence on crack front position: tension loading.

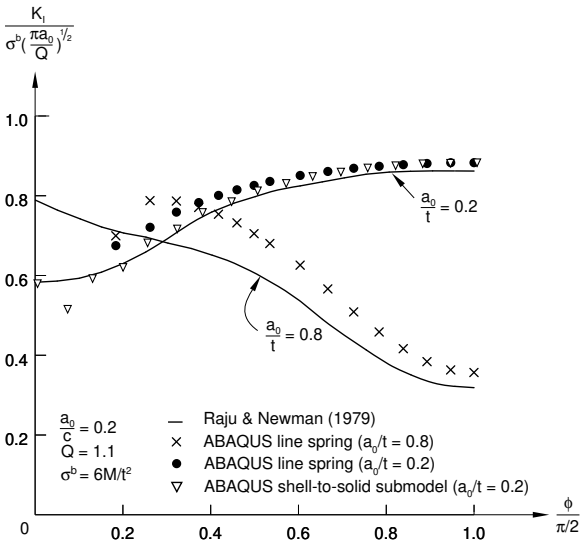


Figure 4: Stress intensity factor dependence on crack front position: moment loading.

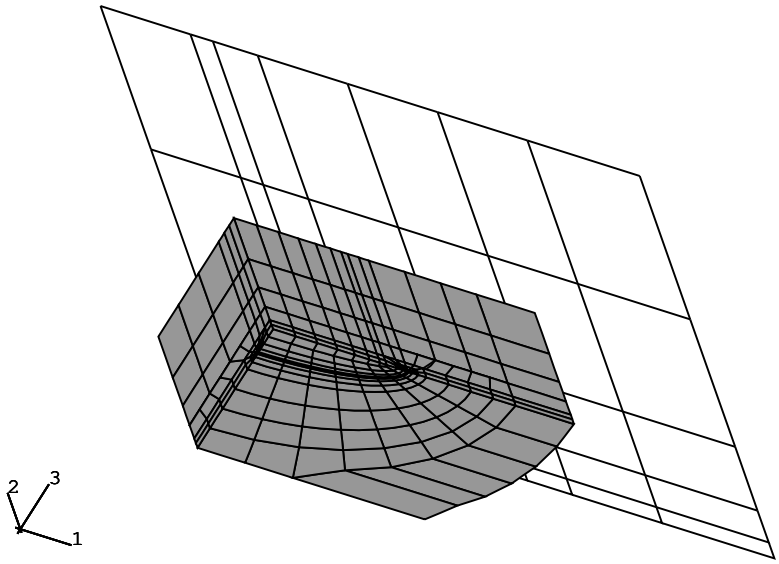


Figure 5: Solid submodel superimposed on shell global model.

Contour integrals for a conical crack in a linear elastic infinite half space

Products: Abaqus/Standard Abaqus/CAE

Overview

This example illustrates many of the capabilities available for linear elastic fracture mechanics analysis in Abaqus.

The following Abaqus features are demonstrated:

- evaluating contour integrals for axisymmetric and three-dimensional fracture mechanics based on linear static stress analysis;
- partitioning steps required to generate a mesh suitable for evaluating contour integrals in two- and three-dimensional analyses;
- evaluating three-dimensional contour integrals when the crack extension direction varies along the crack front;
- node-based submodeling in fracture problems (comparing results for a single refined analysis with the submodeling approach);
- surface-based submodeling based on global model stresses, with guidelines for obtaining adequate accuracy; and
- applying continuum infinite elements simulating an infinite domain.

Application description

This example examines the fracture behavior of a conical crack, which may result from a small hard object impacting a large brittle body. It shows how to evaluate the propensity of the crack to propagate under static loading but does not cover the event that formed the crack.

The *J*-integral is a widely applied fracture mechanics parameter that relates to energy release associated with crack growth and is a measure of the deformation intensity at a crack tip. In practice, the calculated *J*-integral can be compared with a critical value for the material under consideration to predict fracture. The *T*-stress represents stress parallel to the crack face. Together, the *T*-stress and the *J*-integral provide a two-parameter fracture model describing Mode I elastic-plastic crack-tip stresses and deformation in plane strain or three dimensions over a wide range of crack configurations and loadings. The stress intensity factors, K_i, relate to the energy release rate and measure the propensity for crack propagation.

This example uses axisymmetric and three-dimensional models to demonstrate the Abaqus fracture mechanics capability, where the crack extension direction varies along a curved crack front. Submodeling and the use of infinite elements to simulate far-field boundaries are also demonstrated.

Geometry

The problem domain contains a conical crack in an infinite solid half-space, as shown in *Figure 1*. The crack extension direction changes as the crack is swept around a circle. The units for this example are nonphysical; therefore, dimensions, loads, and material properties are described in terms of length and force units. The crack circumscribes a circle with a radius of 10 length units on the free surface. The crack intersects the free surface at 45° and extends 15 length units into the solid domain.

Materials

The material is a linear elastic solid.

Boundary conditions and loading

The semi-infinite domain is constrained from rigid body motion. The applied load is a static pressure with a magnitude of 10 force/length^2 applied on the circular free surface of the block circumscribed by the crack. The loading is illustrated in *Figure 1*.

Abaqus modeling approaches and simulation techniques

This example includes six cases demonstrating different modeling approaches using Abaqus/Standard. The crack is modeled as a seam since the crack surfaces in the unloaded state lie next to one another with no gap.

The geometry is axisymmetric and can be modeled as such. However, the three-dimensional cases demonstrate the Abaqus fracture mechanics capability, where the crack extension direction varies along a curved crack front. The infinite half-space is treated using multiple techniques. In Case 1 through Case 4, the domain is extended well beyond the region of interest. Far-field boundary conditions applied a significant distance from the region of interest have negligible influence on the response near the crack. Cases 5 and Case 6 demonstrate the use of continuum infinite elements. Axisymmetric and three-dimensional cases are provided with and without submodeling. *Fracture Mechanics* provides detailed information on fracture mechanics procedures.

Summary of analysis cases

Case 1	Full axisymmetric model using Abaqus/CAE.
Case 2	Full three-dimensional model using Abaqus/CAE.
Case 3	Axisymmetric approach with submodeling using Abaqus/CAE.
Case 4	Three-dimensional approach with submodeling using Abaqus/CAE.
Case 5	Axisymmetric approach with submodeling and infinite elements using input files.
Case 6	Three-dimensional approach with submodeling and infinite elements using input files.

The following sections discuss analysis considerations that are applicable to several or all the cases. More detailed descriptions are provided later including discussions of results and listings of files provided. The models for Case 1 through Case 4 were generated using Abaqus/CAE. In addition to the Python scripts that generate the model databases, Abaqus/Standard input files are also provided for those cases.

Mesh design

The mesh includes a seam along the crack with duplicate nodes, which allow the crack to open when loaded. The geometry is partitioned to map rings of elements around the crack tip for the contour integral calculations. The models use either quadrilateral or brick elements with a collapsed side to create triangular elements for two-dimensional cases or wedge-shaped elements for three-dimensional cases, which introduce a singularity at the crack tip. To be used for the evaluation of contour integrals, the mesh around the crack tip must be modeled as described in *Using contour integrals to model fracture mechanics*. In the axisymmetric cases a circular partition is created to mesh around the crack tip. In the three-dimensional cases the corresponding partition is a curved tubular volume enclosing the crack tip.

A refined mesh at the crack tip is required to obtain contour-independent results; i.e., there is no significant variation in the contour integral values calculated for successive rings of elements around the crack tip. In the circular partitioned region surrounding the crack tip where the contour integrals are calculated, the mesh should be biased moderately toward the crack tip. The accuracy of the contour integrals is not very sensitive to the biasing. Engineering judgment is required to establish adequate mesh refinement to produce contour-independent results while avoiding the possibility of creating elements at the crack tip that are so small in relation to other elements that they introduce numerical conditioning issues and associated round-off errors.

When the deformation and the material are linear as in this example, the diameter of the circular partition used to map the crack-tip mesh for contour integral calculations is not critical. (If the material is elastic-plastic, the size of the circular partition should generally contain the plastic zone and allow a number of the contours for the contour integrals to enclose the plastic zone while still remaining in the elastic region.) The remaining partitions are created so that the element shapes satisfy the element quality criteria in the regions away from the crack tip.

To understand the types of singularities created by collapsing the side of an element in two or three dimensions, see "Constructing a fracture mechanics mesh for small-strain analysis" in *Contour Integral Evaluation*. In this application we want to have a square root singularity in strain at the crack since the material is linear elastic and we will perform a small-strain analysis.

The stress intensity factors and the *T*-stresses are calculated using the interaction integral method, in which auxiliary plane strain crack-tip fields are employed. The crack front radius of curvature is significant for this problem. Therefore, to calculate the contour integrals accurately for the three-dimensional cases, a very refined mesh is used to approach the plane strain condition locally around the crack front. This refined mesh makes the contour integral domain sufficiently small to minimize the influence of curvature on the results.

Additional details of the meshing procedures for the axisymmetric and three-dimensional cases are discussed below within the descriptions of the individual cases.

Materials

The linear static structural analysis requires specification of Young's modulus, which is 30,000,000 units of force/length2, and Poisson's ratio, which is 0.3. One solid, homogenous section is used to assign material properties to the elements.

Loads

A uniform pressure load of 10 units of force/length2 is applied along the free top surface of the crack. In the axisymmetric models the load region, where the pressure is applied is represented by a line segment. For the three-dimensional cases the load region where the pressure is applied an area.

Analysis steps

Each analysis is performed using a single linear static step.

Output requests

Output requests are used to specify calculation of contour integrals, stress intensity factors, and *T*-stress. See *Requesting contour integral output* for more information regarding fracture mechanics output. The global models used in the submodeling cases include output requests necessary to write displacement and stress results to the output database (`.odb`) file; in the case where node-based submodeling is used, displacement results are used to establish boundary conditions on the corresponding submodels. In the case where surface-based submodeling is used, stress results are used to establish boundary tractions on the corresponding submodel.

Submodeling

Realistic fracture analyses tend to require significant computer resources. To obtain accurate results when analyzing the stress field around a crack tip, a refined mesh must be used to capture the strong gradients near the tip. The required mesh refinement can make fracture mechanics models large since a crack is normally a very small feature compared with the model dimensions. An alternative technique that reduces computational resources is to use submodeling to obtain accurate results by running two smaller models sequentially instead of performing a single global analysis with a refined mesh around the crack. The first step is to solve a less refined global model to obtain a solution that is accurate away from the crack tip but is not sufficiently refined to capture strong gradients near the region of interest. A refined submodel of the crack-tip region is then used to obtain a more accurate solution and, hence, more accurate contour integrals. The boundaries of the submodel

must be far enough from the region of interest that the less refined global model is able to provide accurate results at the submodel boundaries, particularly important when surface-based submodeling is used. This condition is verified during postprocessing by confirming that the stress contours at the boundaries of the submodel are similar to the stress contours at the same location in the global model.

Although the submodeling approach is not required for this example because the refined models for the entire domain analyzed in Case 1 and Case 2 are small enough to run on commonly available computing platforms, this application provides an opportunity to demonstrate submodeling techniques for both axisymmetric and three-dimensional fracture mechanics cases, as well as showing the differences between node-based submodeling based on displacements and surface-based submodeling based on stresses. Submodeling procedures are described in detail in *Node-Based Submodeling*, *Surface-Based Submodeling*, and *Submodeling*.

Modeling an infinite domain

Case 1 through Case 4 simulate the infinite extent of the domain with a continuum mesh that is large compared to the crack dimensions with appropriate far-field boundary conditions. In those cases the domain extends 20 times the crack length. Case 5 and Case 6 demonstrate the use of continuum infinite elements and represent the region of interest with reduced-integration continuum elements to a distance approximately 10 times the crack dimensions surrounded by a layer of continuum infinite elements. Far-field boundary conditions are not required in these cases.

Case 1 Full axisymmetric model with Abaqus/CAE

The axisymmetric domain is a solid with a radius equal to the height of 300 length units (see *Figure 2*). The top edge of the model represents the free surface containing the crack. The semi-infinite domain is simulated by extending the continuum model to a distance 20 times the length of the crack and applying appropriate far-field boundary conditions. This model uses continuum axisymmetric quadratic reduced-integration (CAX8R) elements.

Mesh design

When calculating contour integrals in two-dimensional problems, quadrilateral elements must be used around the crack tip where the contour integral calculations will be performed with triangular elements adjacent to the crack tip. These triangular elements are actually collapsed quadrilaterals, which introduce a singularity. The axisymmetric model must be partitioned as shown in *Figure 2* to define the crack, introduce a singularity by collapsing elements at the crack tip, and create rings of quadrilateral elements for contour integral calculations. A straight line partition is created where the seam crack is defined along with a circular partition, which enables mapping rings of elements around the crack tip. When structured meshing is used for this partition, triangular elements are created adjacent to the crack tip with quadrilaterals surrounding them (see *Using contour integrals to model fracture mechanics*). Abaqus/CAE automatically converts triangular elements at the crack tip to quadrilaterals with one side collapsed to introduce a singularity.

Creating a seam describes how to pick partition segments to define the seam (crack). Procedures to create a square root singularity in strain at the crack tip are described in *Controlling the singularity at the crack tip*. After defining the seam, pick the crack tip to specify the region defining the first contour integral and define the q vector to specify the crack extension direction as described in *Creating a contour integral crack*.

Boundary conditions

The right edge of the model shown in *Figure 3* is unconstrained to represent the far-field boundary. The bottom edge of the model is constrained to zero displacement (U2) to eliminate rigid body motion while simulating the far-field boundary. These edges are far enough away from the area of interest around the crack to represent an infinite domain with negligible influence on the area of interest.

Run procedure

The Case 1 model is generated using Abaqus/CAE to create and to mesh native geometry. Python scripts are provided to automate building the model and running the solution. The scripts can be run interactively or from the command line.

To create the model interactively, start Abaqus/CAE and select **Run Script** from the **Start Session** dialog box that appears. Select the first file for the case, `AxisymmConeCrack_model.py`. When the script completes, you can use Abaqus/CAE commands to display and to query the model. When you are ready to analyze the model, select **Run Script** from the **File** menu and choose the next script, `AxisymmConeCrack_job.py`. The Python scripts provided allow you to modify the model interactively with Abaqus/CAE to explore additional variations on the cases provided here.

Alternatively, the Python scripts can be run from the command line with the Abaqus/CAE`noGUI` option in the order listed:

```
abaqus cae noGUI=filename.py
```

where `abaqus` is the system command to run the program and *filename* is the name of the script to be run.

As an alternative to the Python scripts, an Abaqus/Standard input file `AxisymmConeCrack.inp` is also provided to run this case. You can submit the analysis using the input file with the following command:

```
abaqus input=AxisymmConeCrack.inp
```

Case 2 Full three-dimensional model with Abaqus/CAE

The three-dimensional domain is a cube with an edge length of 300 units, as shown in *Figure 4*. The mesh represents a quarter-symmetric segment of the problem domain. The top of the model represents the free surface containing the crack. The semi-infinite domain is represented by extending the continuum mesh to a distance 20 times the length of the crack with appropriate symmetry and far-field boundary conditions.

Mesh design

When calculating three-dimensional contour integrals, a model meshed with brick or tetrahedral elements is used. For the model with brick elements, rings of brick elements must be used around the crack tip where the contour integral calculations will be performed with wedge elements adjacent to the crack tip (these wedges are actually collapsed bricks). When the tetrahedral elements are used, refined mesh is required along the crack front. Concentric tubular partitions are created to map the mesh around the crack tip. The three-dimensional domain and partitioning of the geometry are illustrated in *Figure 5* and *Figure 6*. The seam crack is shown in *Figure 7*. When structured meshing is used for the inner tubular partition, wedge elements are created adjacent to the crack tip with rings of bricks surrounding them (see *Using contour integrals to model fracture mechanics* for details). A swept mesh used in the inner ring creates wedge elements at the crack tip. The outer ring is meshed with hexahedral elements using the structured meshing technique.

For details on how to define the crack propagation direction where the direction of the vectors varies along the crack front (referred to as the q vector), see *Defining the crack extension direction*. *Figure 8* illustrates the q vectors.

Boundary conditions

Symmetric displacement boundary conditions are applied normal to the symmetry planes. The far-field faces on the sides of the model, which are not symmetry planes, are unconstrained. The bottom face of the model is constrained from displacement in the direction of the pressure load (U2=0) to prevent rigid body motion while simulating a far-field boundary condition.

Run procedure

The Python scripts provided to generate the Abaqus/CAE model and to analyze the model are run following the same procedures as those described for Case 1.

As an alternative to the Python scripts, an Abaqus/Standard input file `SymmConeCrackOrphan.inp` is also provided to run this case. You can submit the analysis using the input file with the following command:

```
abaqus input= SymmConeCrackOrphan.inp
```

The files defining nodes and elements for this case, `SymmConeCrackOrphan_node.inp` and `SymmConeCrackOrphan_elem.inp`, must be available when the input file is submitted.

Figure 9 shows a deformed shape plot for the three-dimensional model of the crack from the full three-dimensional analysis of Case 2. The displacement is exaggerated using a scaling factor to visualize the crack opening.

Case 3 Axisymmetric approach with submodeling using Abaqus/CAE

Case 3 uses the submodeling approach and, hence, requires two sequential analyses, which are referred to as the global model and the submodel. First, a less refined global model is solved to obtain the displacement solution with sufficient accuracy away from the crack tip. A refined submodel of the area of interest driven by the displacement solution from the global model is then used to obtain an accurate solution in the crack-tip region. Each of these models is much smaller than the fully refined axisymmetric global model used in Case 1.

Mesh design

The axisymmetric global model has a relatively less refined mesh in the crack region. The global model used for Case 3 has two rings of elements where the mesh focuses on the crack tip, compared to 13 rings of elements around the crack tip in the full model used in Case 1.

The axisymmetric global model and the submodel meshes for Case 3 are shown in *Figure 10*. The axisymmetric submodel has a refined mesh around the crack tip with 12 rings of elements surrounding the crack tip. It is assumed that the global model's coarse mesh is sufficiently accurate to drive the submodel: the submodel can obtain accurate contour integral results if the global model's displacement field is accurate at the boundaries of the submodel, which lie far from the crack tip. You can verify this at the postprocessing stage by comparing stress contours at the boundaries of the submodel to the corresponding contours of the global model.

Boundary conditions

The boundary conditions applied to the axisymmetric global model are the same as those used in Case 1. The displacement solution from the global model is applied to the submodel boundaries when the submodeling technique is used.

Run procedure

The models used for Case 3 are generated using Abaqus/CAE to create and to mesh native geometry. The same procedures used to run the Python scripts for Cases 1 and 2 are used to create and to analyze the global model. The script that builds the submodel refers to the global model output database (`.odb`) file, which must be available when the submodel is analyzed. After analyzing the global model, run the scripts to build and to analyze the submodel using the same procedure used for the global model.

Abaqus/Standard input files are also provided to run this case. First, run the job to create and to analyze the global model; then run the submodel job. The results from the global model must be available to run the submodel. A typical execution procedure is as follows:

```
abaqus job=globalModelJobName  input=globalModelInputfile.inp
abaqus job=submodelJobname, input=subModelInputfile.inp,
globalmodel=globalModelJobNameOutputDatabase
```

Case 4 Three-dimensional approach with submodeling using Abaqus/CAE

Case 4 uses a three-dimensional submodeling approach and requires two sequential analyses, a global model and a submodel. First, a global model is solved with sufficient refinement to provide an accurate displacement and stress solution away from the crack tip.

Two versions of refined submodels of the area of interest are then used to obtain an accurate solution in the crack-tip region. In one submodel analysis the area of interest is driven by the displacement solution from the global model. In the other submodel analysis the area of interest is driven by the stress solution from the global model.

Each of the models used in this case is much smaller than the fully refined three-dimensional global model used in Case 2.

Mesh design

The three-dimensional global model, with a less refined mesh in the crack region, is first analyzed and then used to drive the submodel. For the three-dimensional global model only 18 elements are used along the crack line, whereas 38 elements are used along the crack line in the submodel. *Figure 11* shows the meshes for the three-dimensional global model and the submodel.

Boundary conditions

The boundary conditions applied to the global model in Case 4 are the same as those used in the full three-dimensional Case 2. The submodeling approach uses either the displacement or stress solution from the global model to drive the submodel boundaries.

Run procedure

The models used for Case 4 are generated using Abaqus/CAE to create and to mesh native geometry. The same procedures used to run Case 3 can be used with Case 4. The script that builds the submodel refers to the global model output database (`.odb`) file, which must be available when the submodel is analyzed.

As an alternative to the Python scripts, Abaqus/Standard input files are also provided to run this case. These are submitted using the same procedure described for the input files under Case 3.

Case 5 Axisymmetric submodeling approach with infinite elements using Abaqus/Standard input files

Case 5 uses the submodeling approach with an axisymmetric mesh utilizing continuum infinite elements to simulate the far-field boundary condition. The submodeling technique requires two sequential analyses, a less refined global model and a refined submodel at the crack tip.

The global model in Case 5 comprises an axisymmetric representation of the hemispherical domain with continuum elements to a radius of 170 length units. Eight-node biquadratic axisymmetric quadrilateral, reduced-integration elements (CAX8R) are used to model the solid domain in the region adjacent to the crack. The domain is further extended using a layer of continuum infinite elements to a radius of 340 length units. Five-node quadratic axisymmetric one-way infinite elements (CINAX5R) are used to simulate the far-field region of the solid. The

submodel used in Case 5 does not encompass the complete crack face but extends to a distance far enough from the crack tip that strong variations in the stress field are captured within the submodel. This result can be verified by comparing stress contours of the submodel with the corresponding stress contours in the global model.

Mesh design

The axisymmetric global model, with a relatively less refined mesh in the crack region, is first analyzed and then used to drive the submodel. The axisymmetric global model and the submodel meshes for Case 5 are shown in *Figure 12*.

Boundary conditions

The continuum infinite elements eliminate the need for far-field constraints, which were required in Case 1 through Case 4.

Run procedure

The models used for Case 5 are generated using Abaqus/Standard input files. First, run the job to create and to analyze the global model; then run the submodel job. The results from the global model must be available to run the submodel. A typical execution procedure is as follows:

```
abaqus job=globalModelJobName   input=globalModelInputfile.inp
abaqus job=submodelJobname, input=subModelInputfile.inp,
globalmodel=globalModalJobNameOutputDatabase
```

Case 6 Three-dimensional approach with infinite elements using submodeling with Abaqus/Standard input files

Case 6 uses the submodeling approach with a three-dimensional mesh utilizing continuum infinite elements to simulate the far-field boundary condition. The domain modeled for Case 6 encompasses an eighth of a sphere, representing one-quarter of the semi-infinite problem domain. Continuum elements are used to a radius of 170 length units. The domain is extended using a layer of infinite continuum elements to a radius of 340 length units.

The submodeling technique requires two separate analyses, a less refined global model and a refined submodel at the crack tip. Case 6 uses Abaqus/Standard input files to generate the models rather than Abaqus/CAE Python scripts. Twenty-node quadratic, reduced-integration solid elements (C3D20R) are used to model the solid domain in the region adjacent to the crack. The domain is further extended using a layer of 12-node quadratic one-way infinite brick elements (CIN3D12R) to a radius of 340 length units.

The submodel used in Case 6 does not encompass the complete crack face but extends to a distance far enough from the crack tip that strong variations in the stress field are captured within the submodel.

Mesh design

The three-dimensional global model, with a relatively less refined mesh in the crack region, is first analyzed and then used to drive the submodel. The three-dimensional global model and the submodel meshes for Case 6 are shown in *Figure 13*.

Boundary conditions

The continuum infinite elements eliminate the need for far-field constraints, which were required in Case 1 through Case 4.

Run procedure

The models of Case 6 are generated using Abaqus/Standard input files. The same procedure used in Case 5 is also used in Case 6. The files containing the node and element definitions for the three-dimensional global model must be available when the input file is run to create the global model . The output database (`.odb`) file from the global model must be available to run the submodel.

Discussion of results and comparison of cases

Contour integral results obtained from the data (`.dat`) file for each case are summarized in *Table 1* through *Table 4*. These results are also available from the output database (`.odb`) file by displaying history output in the Visualization module of Abaqus/CAE. While there is no analytical solution available for comparison, an additional axisymmetric analysis with extreme mesh refinement is used as the basis for a reference solution. Each table includes the reference solution value in the table title.

Abaqus calculates the *J*-integral using two methods. Values of the *J*-integral are based on the stress intensity factors, JK, and by evaluating the contour integral directly, JA. The stress intensity factors $\boldsymbol{K_I}$ and $\boldsymbol{K_{II}}$, and the *T*-stresses are given in *Table 2*, *Table 3*, and *Table 4*, respectively. When the stress intensity factors are requested, Abaqus automatically outputs the *J*-integrals based on the stress intensity factors, JK. Values of $\boldsymbol{K_{III}}$ are not tabulated because these values should equal zero based on the loading and are negligibly small relative to $\boldsymbol{K_I}$ and $\boldsymbol{K_{II}}$.

The tables list values for contour 1 through contour 5. Each contour corresponds to a successive ring of elements progressing outward radially from the crack tip. For the axisymmetric cases one set of results is available for each contour. For the three-dimensional cases Abaqus/Standard provides contour integral values at each crack-tip node. The values listed in *Table 1* through *Table 4* for the three-dimensional cases correspond to the location halfway along the circumference of the crack, which lies midway between the symmetry faces of the three-dimensional models. A detailed examination of the results for the three-dimensional cases confirms that the contour integral values are essentially constant at each node along the circumference of the crack tip. The exception is the value calculated for $\boldsymbol{K_{III}}$; which fluctuates but remains small relative to $\boldsymbol{K_I}$ and $\boldsymbol{K_{II}}$ over almost the full length of the crack; however $\boldsymbol{K_{III}}$ increases at the open end faces of the crack corresponding to the symmetry planes. A loss of accuracy occurring at the node corresponding to the open end of a three-dimensional crack is a known limitation that can be expected when applying this method.

Results from the first contour are generally not used when evaluating fracture problems because the first contour is influenced by the singularity associated with the crack tip. The average quantities reported in the tabular results exclude the first contour. Comparisons refer only to contour 2 through contour 5. The axisymmetric and the three-dimensional modeling approaches are in close agreement, with and without submodeling. For each case, values of the tabulated quantities for *J* calculated by evaluating the contour integrals directly, (JA), $\boldsymbol{K_I}$, $\boldsymbol{K_{II}}$, and *T*-stresses deviate by less than 2% of the average of the corresponding values for contour 2 through contour 5. The *J*-integral for each case, calculated from the stress intensity factors (JK) deviate by less than 3.5% of the average of corresponding values. The larger deviation for JK versus JA is expected because the method of calculating contour integrals from the stress intensity factors (JK) is more sensitive to numerical precision than calculating the contour integrals directly (JA). $\boldsymbol{K_{III}}$ is analytically equal to zero due to the geometry and loading symmetry in this example; the numerical results for $\boldsymbol{K_{III}}$ are negligibly small relative to $\boldsymbol{K_I}$ and $\boldsymbol{K_{II}}$.

Submodeling results

The global models used to calculate the deformation and stress fields that drive the submodels use crack-tip meshes that are too coarse to give accurate results for the contour integrals; therefore, the results for the global models are not tabulated. Results are tabulated for the submodels that refer to this global analysis. Generally these results verify that the submodeling approach provides adequate accuracy in fracture problems where it

may not be practical to use a sufficiently refined mesh in the crack-tip region of a global model. The node-based submodeling approach provides greater accuracy than the surface-based approach.

Node-based submodeling results

The *J*-integral values for the node-based submodel analyses match those for the full model analyses (analyses with adequate mesh refinement around the crack tip) to within less than 1%.

Surface-based submodeling results

The three-dimensional submodeling case also considers surface-based submodeling, where the submodel is driven by the global model stress field. Two different pairs of global models and surface-based submodels are considered: one that matches the mesh design used in the node-based analysis, and one where adjustments are made to improve accuracy. The *J*-integral values for the first analysis pair, with the same meshes as in node-based submodeling, match those for the full model only to within 6%. These inaccurate results arise from a modeling arrangement that violates guidelines established in *Surface-Based Submodeling*, namely that

- the submodel surface should intersect the global model in regions of relatively low stress gradients, and
- the submodel surface should intersect the global model in regions of uniform element size.

Adjusted global and submodel analyses that adhere to these guidelines are run. In this case the submodel driven surface is farther from the crack region and the high stress gradient, and the global model mesh is refined so that elements are more uniform in the region of the submodel surface. *Figure 14* shows a comparison of the submodel/global model pairs. The modeling arrangement on the left places the lower submodel boundary too near to the crack and high stress gradients and cuts through high aspect ratio elements. The arrangement on the right provides lower aspect ratio elements in the global model and positions the lower submodel boundary further from the crack. The adjusted analysis with the further boundary now matches the *J*-integral values for the full model to within 2%. This accuracy difference illustrates the importance of adhering to the guidelines for surface-based submodel design. In practice, in the absence of a reference global solution, you should use the following guidelines to ensure your surface-based solution is adequate:

- As with any submodel analysis, compare solution results between the global model and submodel on the submodel boundary. In this case a stress comparison is appropriate. *Figure 15* compares the 2-component of stress for the two surface-based submodel analyses and their corresponding global model. Results are plotted on a path lying in the lower submodel boundary and extending from the center radially outward. The near-boundary submodel has a significantly greater stress discrepancy with the global model.
- In cases where inertia relief is employed to address rigid body modes in surface-based submodeling, if the inertia relief force output variable (IRF) is small compared to the prevailing force level in the model, the surface-based stress distribution is equilibrated. In this model the prevailing force is the 10 units of pressure acting on the surface circumscribed on the crack (a radius of 6), or 786 units of force for the three-dimensional quarter symmetry model.

  In this analysis the inertia relief force in the 2-direction is similar in both cases (33 for the near-boundary model and 32 for the far-boundary model) and relatively small; hence, in this case, the inertia relief force would not suggest poorer results with the near-boundary submodel, and its small value is not a sufficient measure of the adequacy of the submodel design.

Files

You can use the Python scripts for Abaqus/CAE and input files for Abaqus/Standard to create and to run the cases.

Case 1 Full axisymmetric analysis

AxisymmConeCrack_model.py	Script to create the model, including instructions for creating the mesh used for the reference solution.
AxisymmConeCrack_job.py	Script to analyze the model.
AxisymmConeCrack.inp	Input file to create and to analyze the model.

Case 2 Full three-dimensional model

SymmConeCrack_model.py	Script to create the model.
SymmConeCrack_job.py	Script to analyze the model.
SymmConeCrackOrphan.inp	Input file to create and to analyze the model with brick elements.
SymmConeCrackOrphan_node.inp	Nodes for SymmConeCrackOrphan.inp.
SymmConeCrackOrphan_elem.inp	Elements for SymmConeCrackOrphan.inp.
SymmConeCrackOrphan_c3d10.inp	Input file to create and to analyze the model with tetrahedral elements.

Case 3 Axisymmetric submodel analysis

AxisymmConeCrackGl_model.py	Script to create the model.
AxisymmConeCrackGl_job.py	Script to analyze the model and to create the output database file that drives the submodel.
AxisymmConeCrackSub_model.py	Script to create the submodel.
AxisymmConeCrackSub_job.py	Script to analyze the submodel using the results from the global model output database file to drive it.
AxisymmConeCrackGl.inp	Input file to create and to analyze the global model.
AxisymmConeCrackSub.inp	Input file to create and to analyze the submodel.

Case 4 Three-dimensional submodel analysis

SymmConeCrackGl_model.py	Script to create the global model.
SymmConeCrackGl_job.py	Script to analyze the global model and to create the output database file that drives the submodel. Refer to parameter definitions in the script to create the adjusted global model referred to in *Submodeling results*.
SymmConeCrackSub_model.py	Script to create the node-based submodel.
SymmConeCrackSub_job.py	Script to analyze the node-based submodel using the results from the global model output database file to drive it.
SymmConeCrackSubSb_near_model.py	Script to create the surface-based submodel.
SymmConeCrackSubSb_near_job.py	Script to analyze the surface-based submodel using the stress results from the global model output database file to drive it.
SymmConeCrackSubSb_far_model.py	Script to create the surface-based submodel with a far-boundary submodel.
SymmConeCrackSubSb_far_job.py	Script to analyze the surface-based submodel with the far-boundary submodel using the stress results from the global model output database file to drive it.
SymmConeCrackGlOrphan.inp	Input file to create and to analyze the global model.
SymmConeCrackGlOrphan_node.inp	Nodes for SymmConeCrackGlOrphan.inp.

SymmConeCrackGlOrphan_elem.inp	Elements for SymmConeCrackGlOrphan.inp.
SymmConeCrackGlOrphanAdj.inp	Input file to create and to analyze the global model that is adjusted for improved surface-based submodel accuracy.
SymmConeCrackGlOrphanAdj_node.inp	Nodes for SymmConeCrackGlOrphanAdj.inp.
SymmConeCrackGlOrphanAdj_elem.inp	Elements for SymmConeCrackGlOrphanAdj.inp.
SymmConeCrackSubOr.inp	Input file to create and to analyze the node-based submodel.
SymmConeCrackSubOr_node.inp	Nodes for SymmConeCrackSubOr.inp.
SymmConeCrackSubOr_elem.inp	Elements for SymmConeCrackSubOr.inp.
SymmConeCrackSubOrSb_near.inp	Input file to create and to analyze the submodel using the surface-based submodel technique to drive the submodel stresses.
SymmConeCrackSubOrSb_near_node.inp	Nodes for SymmConeCrackSubOrSb_near.inp.
SymmConeCrackSubOrSb_near_elem.inp	Elements for SymmConeCrackSubOrSb_near.inp.
SymmConeCrackSubOrSb_far.inp	Input file to create and to analyze the submodel with the far-boundary submodel using the surface-based submodel technique to drive the submodel stresses.
SymmConeCrackSubOrSb_far_node.inp	Nodes for SymmConeCrackSubOrSb_far.inp.
SymmConeCrackSubOrSb_far_elem.inp	Elements for SymmConeCrackSubOrSb_far.inp.

Case 5 Axisymmetric submodel analysis using infinite continuum elements

conicalcrack_axiglobal.inp	Input file to analyze the axisymmetric global model and to create the output database file that drives the submodel.
conicalcrack_axisubmodel_rms.inp	Input file to analyze the axisymmetric submodel using the results from the global model output database file to drive it.

Case 6 Three-dimensional submodeling analysis using infinite continuum elements

conicalcrack_3dglobal.inp	Input file to analyze the three-dimensional global model and to create the output database file that drives the submodel.
conicalcrack_3dsubmodel_rms.inp	Input file to analyze the three-dimensional submodel using the results from the global model output database file to drive it.

References

- Shih, C. F., B. Moran, and T. Nakamura, “Energy Release Rate Along a Three-Dimensional Crack Front in a Thermally Stressed Body,” International Journal of Fracture, vol. 30, pp.79–102, 1986.

Tables

Table 1: J-integral estimates ($\times 10^{-7}$) for conical crack using Abaqus. JK denotes the J values estimated from stress intensity factors; JA denotes the J values estimated directly by Abaqus. The reference solution J-integral value is 1.33.

Solution	Contour						Average Value, Contours 2–5
	J estimate method	1	2	3	4	5	
Case 1: Full axisymmetric	JK	1.326	1.308	1.288	1.262	1.228	1.272

Solution	Contour						Average Value, Contours 2–5
	J estimate method	1	2	3	4	5	
	JA	1.334	1.333	1.334	1.334	1.334	1.334
Case 2: Full three-dimensional	JK	1.303	1.325	1.312	1.295	1.274	1.302
	JA	1.308	1.334	1.336	1.337	1.337	1.336
Case 3: Submodel axisymmetric	JK	1.327	1.319	1.311	1.300	1.287	1.304
	JA	1.330	1.329	1.330	1.330	1.330	1.330
Case 4: Node-based submodel three-dimensional	JK	1.314	1.316	1.303	1.285	1.264	1.292
	JA	1.318	1.326	1.328	1.328	1.328	1.328
Case 4: Surface-based submodel three-dimensional	JK	1.396	1.398	1.385	1.367	1.345	1.374
	JA	1.400	1.408	1.409	1.408	1.407	1.408
Case 4: Surface-based submodel with far boundary, three-dimensional	JK	1.345	1.347	1.335	1.317	1.296	1.324
	JA	1.349	1.357	1.359	1.358	1.358	1.358
Case 5: Submodel axisymmetric with infinite elements	JK	1.413	1.359	1.363	1.363	1.361	1.362
	JA	1.407	1.360	1.365	1.365	1.365	1.364
Case 6: Submodel three-dimensional with infinite elements	JK	1.329	1.363	1.367	1.368	1.368	1.367
	JA	1.336	1.361	1.366	1.366	1.366	1.365

Table 2: Stress intensity factor K_I estimates for conical crack using Abaqus. Contour 1 is omitted from the average value calculations. The reference solution K_I value is 0.491.

Solution	Contour					Average Value, Contours 2–5
	1	2	3	4	5	
Case 1: Full axisymmetric	0.495	0.497	0.499	0.500	0.499	0.499
Case 2: Full three-dimensional	0.492	0.501	0.503	0.502	0.500	0.502
Case 3: Submodel axisymmetric	0.491	0.493	0.494	0.495	0.496	0.494
Case 4: Node-based submodel three-dimensional	0.491	0.496	0.498	0.497	0.494	0.497
Case 4: Surface-based submodel three-dimensional	0.426	0.431	0.433	0.431	0.427	0.430
Case 4: Surface-based submodel with far boundary, three-dimensional	0.436	0.441	0.443	0.442	0.439	0.441
Case 5: Submodel axisymmetric with infinite elements	0.537	0.527	0.528	0.528	0.529	0.528

Solution	Contour					Average Value, Contours 2–5
	1	2	3	4	5	
Case 6: Submodel three dimensional with infinite elements	0.522	0.528	0.529	0.530	0.530	0.528

Table 3: Stress intensity factor K_{II} estimates for conical crack using Abaqus. Contour 1 is omitted from the average value calculations. The reference solution K_{II} value is –2.03.

Solution	Contour					Average Value, Contours 2–5
	1	2	3	4	5	
Case 1: Full axisymmetric	–2.032	–2.016	–2.000	–1.978	–1.949	–1.986
Case 2: Full three-dimensional	–2.013	–2.029	–2.018	–2.004	–1.987	–2.010
Case 3: Submodel axisymmetric	–2.033	–2.026	–2.019	–2.010	–1.999	–2.014
Case 4: Node-based submodel three-dimensional	–2.023	–2.023	–2.012	–1.997	–1.980	–2.003
Case 4: Surface-based submodel three-dimensional	–2.102	–2.103	–2.092	–2.078	–2.061	–2.084
Case 4: Surface-based submodel with far boundary, three-dimensional	–2.060	–2.060	–2.050	–2.036	–2.019	–2.041
Case 5: Submodel axisymmetric with infinite elements	–2.090	–2.050	–2.053	–2.052	–2.051	–2.051
Case 6: Submodel three dimensional with infinite elements	2.027	2.053	2.057	2.057	2.057	2.056

Table 4: T-stress estimates for conical crack using Abaqus. Contour 1 is omitted from the average value calculations. The reference solution T-stress value is 0.979.

Solution	Contour					Average Value, Contours 2–5
	1	2	3	4	5	
Case 1:Full axisymmetric	–0.982	–0.979	–0.976	–0.972	–0.967	–0.973
Case 2: Full three-dimensional	–0.942	–0.972	–0.966	–0.960	–0.954	–0.963
Case 3:Submodel Axisymmetric	–0.980	–0.978	–0.977	–0.975	–0.973	–0.976
Case 4: Node-based submodel three-dimensional	–0.947	–0.966	–0.959	–0.953	–0.947	–0.956
Case 4: Surface-based submodel three-dimensional	–0.981	–0.996	–0.989	–0.983	–0.976	–0.986
Case 4: Surface-based submodel with far boundary, three-dimensional	–0.958	–0.973	–0.966	–0.960	–0.954	–0.963

Solution	Contour					Average Value, Contours 2–5
	1	2	3	4	5	
Case 5: Submodel axisymmetric with infinite elements	–1.182	–0.983	–0.985	–0.984	–0.984	–0.984
Case 6: Submodel three-dimensional with infinite elements	–0.599	–0.982	–0.984	–0.983	–0.982	–0.982

Figures

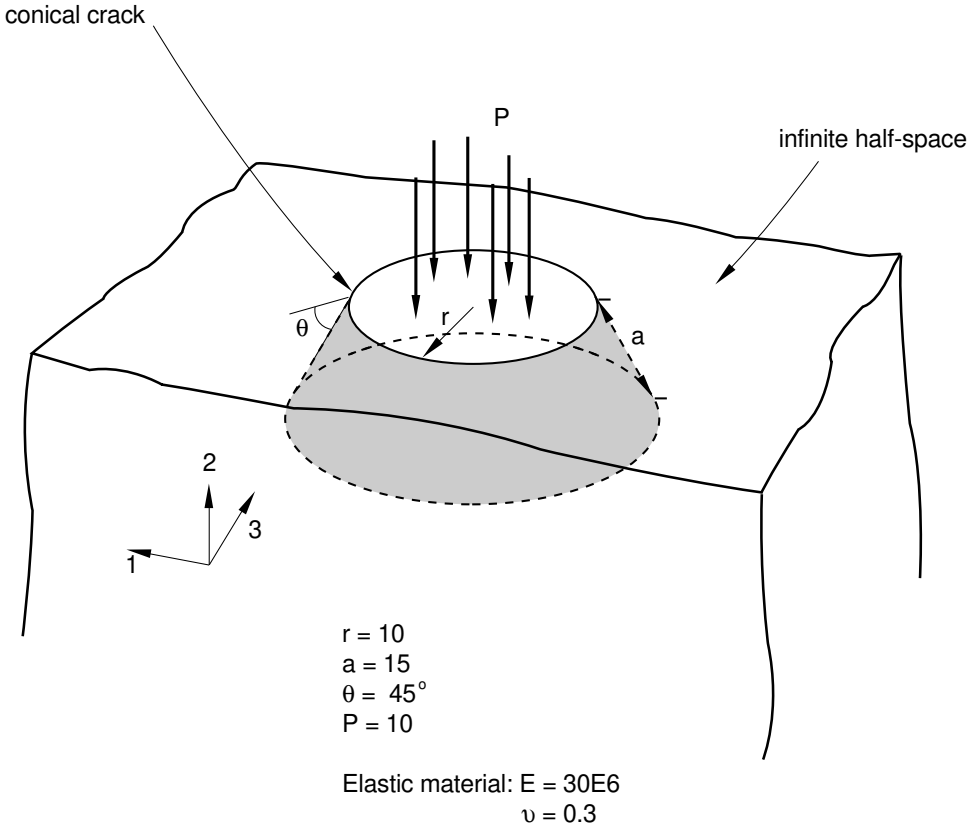


Figure 1: Conical crack in a half-space.

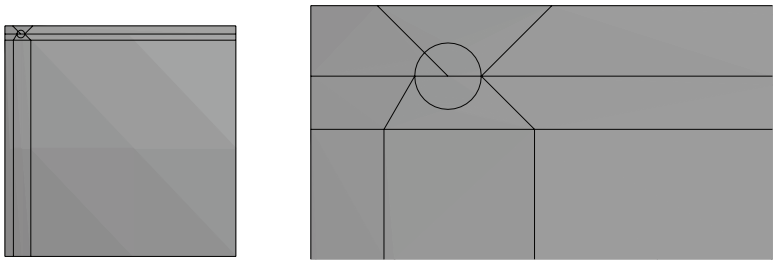


Figure 2: Case 1: Partitioning axisymmetric geometry.

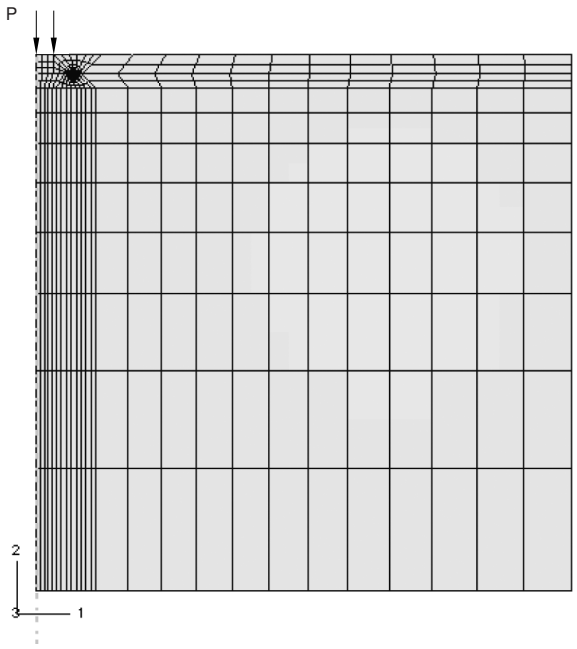


Figure 3: Case 1: Full axisymmetric mesh.

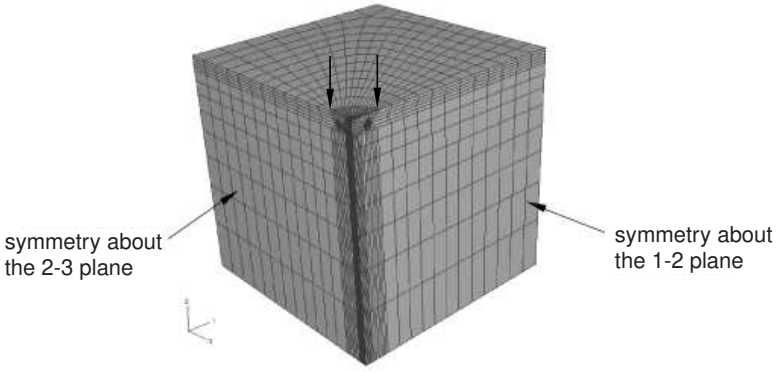


Figure 4: Case 2: Full three-dimensional mesh.

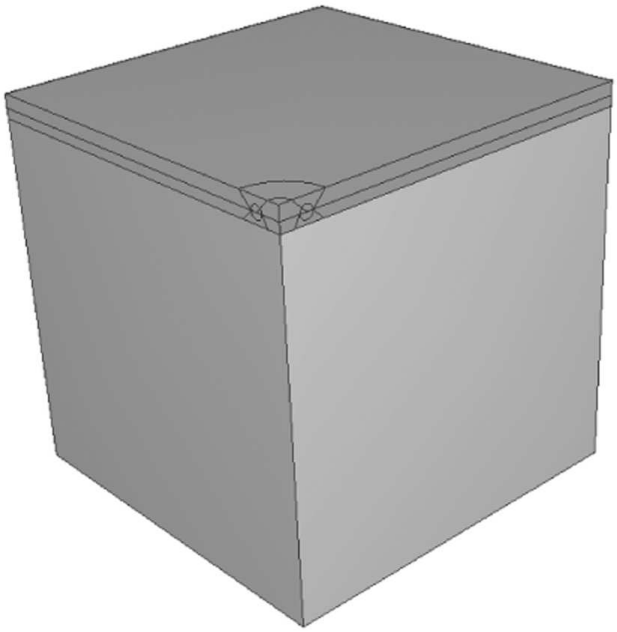


Figure 5: Case 2: Partitioned full three-dimensional model.

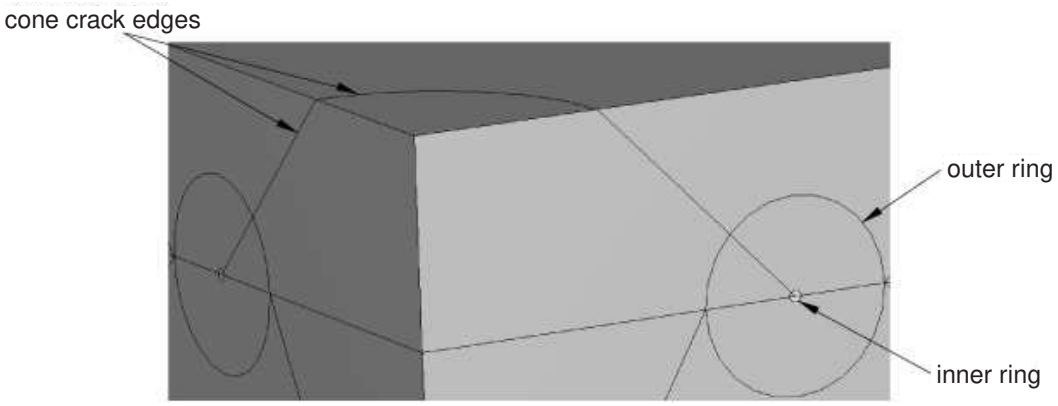


Figure 6: Case 2: Partitions around the crack line. The smaller inner ring is swept meshed using wedge elements. The outer ring is meshed using hexahedral elements and the structured meshing technique. The cone partitions are also visible.

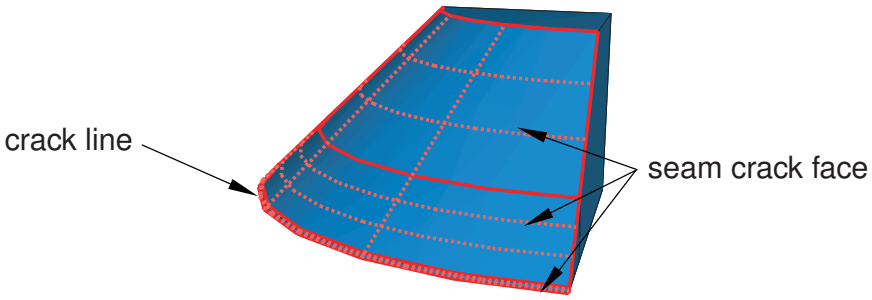


Figure 7: Case 2: Seam crack faces for the cone.

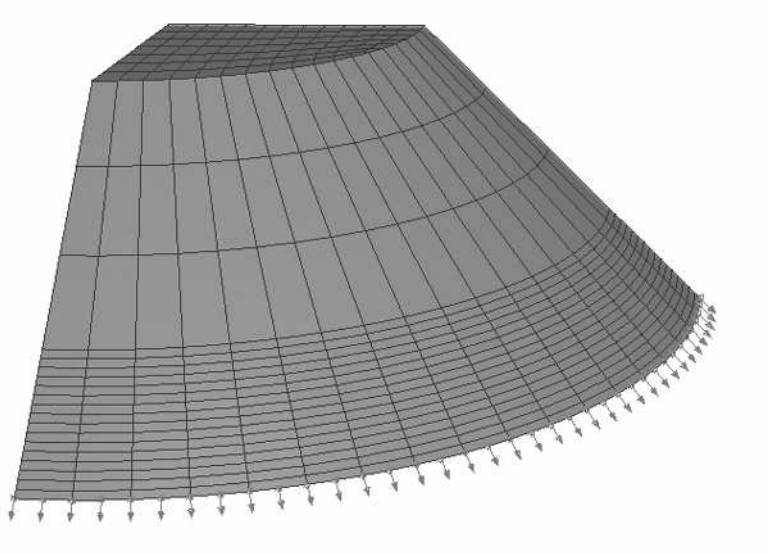


Figure 8: Case 2: q vectors defined along the entire crack line on an orphan mesh.

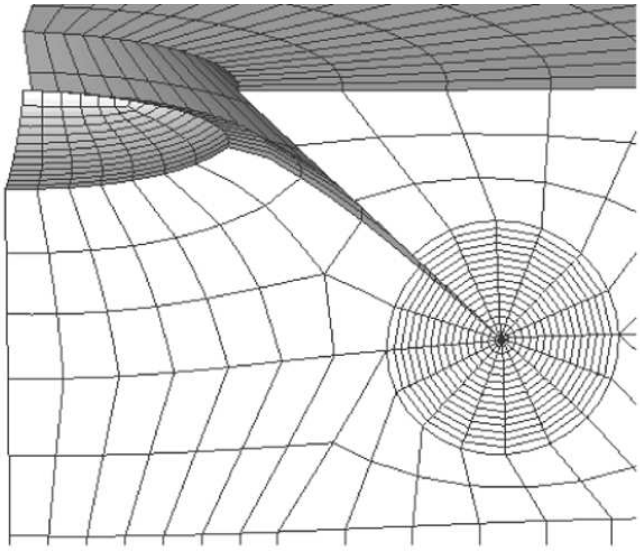


Figure 9: Case 2: Results three-dimensional displaced shape.

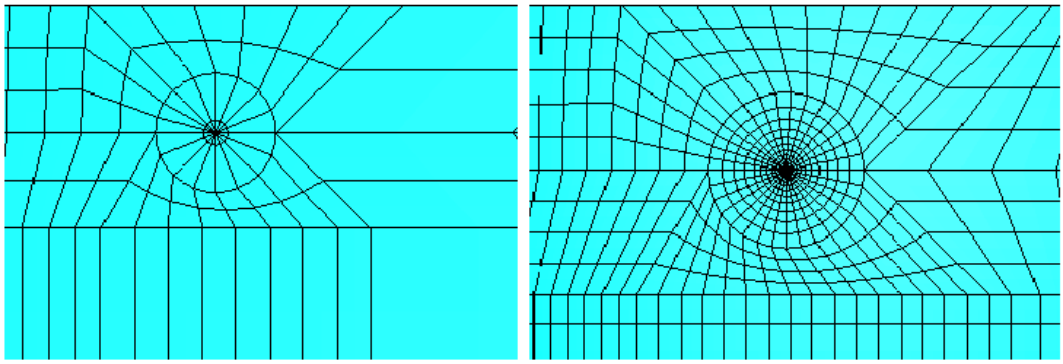


Figure 10: Case 3: Axisymmetric global and submodel meshes around the crack line.

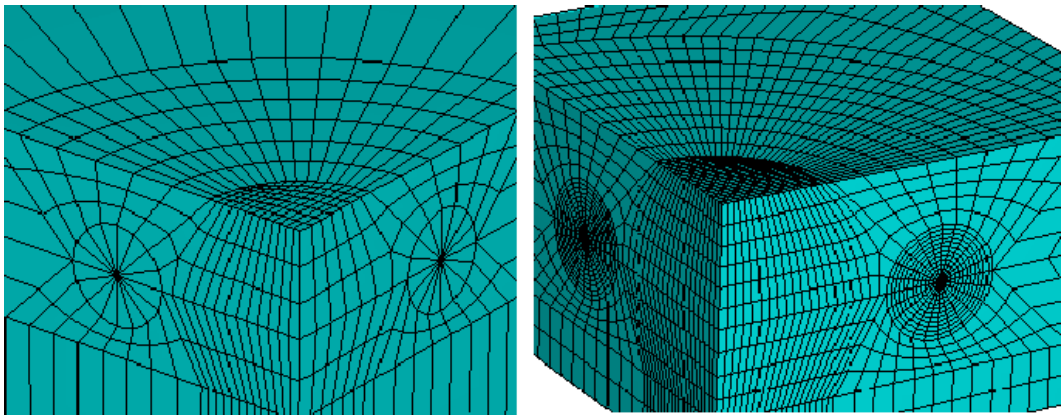


Figure 11: Case 4: Full three-dimensional global model and submodel meshes around the crack line.

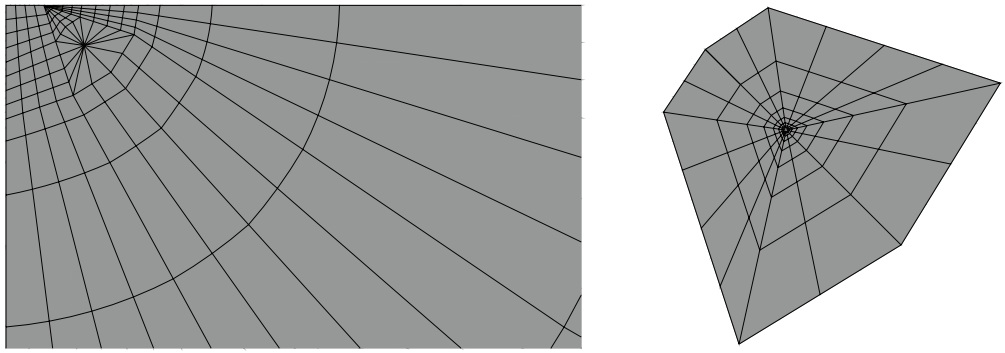


Figure 12: Case 5: Axisymmetric global model using infinite elements and submodel meshes.

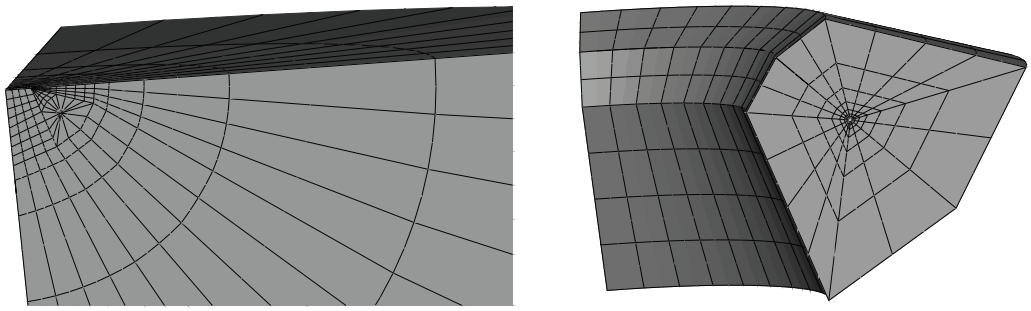


Figure 13: Case 6: Three-dimensional global model with infinite elements and submodel meshes.

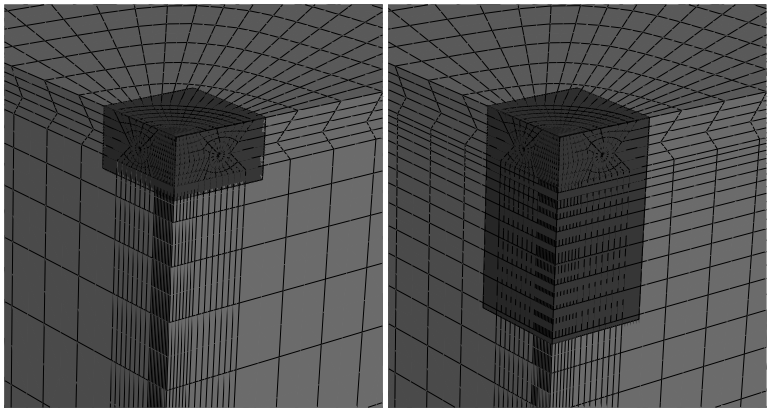


Figure 14: Case 4: Comparison of inadequate (left) and adequate (right) global and submodel designs for a surface-based submodel stress solution.

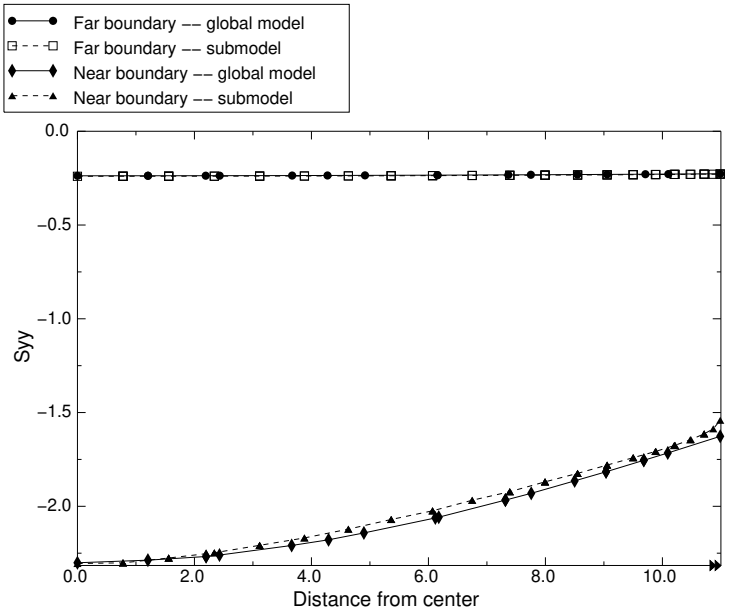


Figure 15: Case 4: Confirmation of stress agreement between the global model and submodel.

Elastic-plastic line spring modeling of a finite length cylinder with a part-through axial flaw

Products: Abaqus/Standard

Overview

This example illustrates the use of line spring elements to model a finite length cylinder with a part-through axial flaw.

The elastic-plastic line spring elements in Abaqus are intended to provide inexpensive solutions for problems involving part-through surface cracks in shell structures loaded predominantly in Mode I by combined membrane and bending action in cases where it is important to include the effects of inelastic deformation. When the line spring element model reaches theoretical limitations, the shell-to-solid submodeling technique is utilized to provide accurate J-integral results. The energy domain integral is used to evaluate the J-integral for this case.

The case considered is a long cylinder with an axial flaw in its inside surface, subjected to internal pressure. It is taken from the paper by Parks and White (1982).

Geometry and model

The cylinder has an inside radius of 254 mm (10 in), wall thickness of 25.4 mm (1 in), and is assumed to be very long. The mesh is shown in *Figure 1*. It is refined around the crack by using multi-point constraints (MPCs). There are 70 shell elements of type S8R in the symmetric quarter-model and eight symmetric line spring elements (type LS3S) along the crack. The mesh is taken from Parks and White, who suggest that this mesh is adequately convergent with respect to the fracture parameters (J-integral values) that are the primary objective of the analysis. No independent mesh studies have been done. The use of MPCs to refine a mesh of reduced integration shell elements (such as S8R) is generally satisfactory in relatively thick shells as in this case. However, it is not recommended for thin shells because it introduces constraints that "lock" the response in the finer mesh regions. In a thin shell case the finer mesh would have to be carried out well away from the region of high strain gradients.

Three different flaws are studied. All have the semi-elliptic geometry shown in *Figure 2*, with, in all cases, $c = 3a_0$. The three flaws have a_0/t ratios of 0.25 (a shallow crack), 0.5, and 0.8 (a deep crack). In all cases the axial length of the cylinder is taken as 14 times the crack half-length, c: this is assumed to be sufficient to approximate the infinite length.

An input data file for the case $a_0/t = .5$ without making the symmetry assumption about $z = 0$ is also included. This mesh uses the LS6 line spring elements and serves to check the elastic-plastic capability of the LS6 elements. The results are the same as for the corresponding mesh using LS3S elements and symmetry about $z = 0$. The formulation of the LS6 elements assumes that the plasticity is predominately due to Mode I deformation around the flaw and neglects the effect of the Mode II and Mode III deformation around the flaw. In the global mesh the displacement in the z-direction is constrained to be zero at the node at the end of the flaw where the flaw depth goes to zero. To duplicate this constraint in the mesh using LS6 elements, the two nodes at the end of the flaw (flaw depth = 0) are constrained to have the same displacements.

Material

The cylinder is assumed to be made of an elastic-plastic metal, with a Young's modulus of 206.8 GPa (30×10^6 lb/in^2), a Poisson's ratio of 0.3, an initial yield stress of 482.5 MPa (70000 lb/in^2, and constant work hardening to an ultimate stress of 689.4 MPa (10^5 lb/in^2) at 10% plastic strain, with perfectly plastic behavior at higher strains.

Loading

The loading consists of uniform internal pressure applied to all of the shell elements, with edge loads applied to the far end of the cylinder to provide the axial stress corresponding to a closed-end condition. Even though the flaw is on the inside surface of the cylinder, the pressure is not applied on the exposed crack face. Since pressure loads on the flaw surface of line spring elements are implemented using linear superposition in Abaqus, there is no theoretical basis for applying these loads when nonlinearities are present. We assume that this is not a large effect in this problem. For consistency with the line spring element models, pressure loading of the crack face is not applied to the shell-to-solid submodel.

Results and discussion

The line spring elements provide *J*-integral values directly. *Figure 3* shows the J-integral values at the center of the crack as functions of applied pressure for the three flaws. In the input data the maximum time increment size has been limited so that adequately smooth graphs can be obtained. *Figure 4* shows the variations of the J-integral values along the crack for the half-thickness crack ($a_0/t = 0.5$), at several different pressure levels (a normalized pressure, $\hat{p} = p\overline{R}/\sigma_y t$, is used, where $\overline{R}$ is the mean radius of the cylinder). These results all agree closely with those reported by Parks and White (1982), where the authors state that these results are also confirmed by other work. In the region $\phi < 30°$ the results are inaccurate for two reasons. First, the depth of the flaw is changing very rapidly in this region, which makes the line spring approximation quite inaccurate. Second, J^{el} is of the same order of magnitude as J^{pl}, but the line spring plasticity model is only valid when $J^{el} \ll J^{pl}$. The results toward the center of the crack ($\phi > 30°$) are more accurate than those at the ends of the crack since the flaw depth changes less rapidly with position in this region and J^{pl} is much larger than J^{el}. For this reason only *J* values for $\phi > 30°$ are shown in *Figure 4*.

Shell-to-solid submodeling around the crack tip

An input file for the case $a_0/t = 0.25$, which uses the shell-to-solid submodeling capability, is included. This C3D20R element mesh allows the user to study the local crack area using the energy domain integral formulation for the J-integral. The submodel uses a focused mesh with four rows of elements around the crack tip. A $1/r$ singularity is utilized at the crack tip, the correct singularity for a fully developed perfectly plastic solution. Symmetry boundary conditions are imposed on two edges of the submodel mesh, while results from the global shell analysis are interpolated to two surfaces via the submodeling technique. The global shell mesh gives satisfactory *J*-integral results; hence, we assume that the displacements at the submodel boundary are sufficiently accurate to drive the deformation in the submodel. No attempt has been made to study the effect of making the submodel region larger or smaller. The submodel is shown superimposed on the global shell model in *Figure 5*.

In addition, an input file for the case $a_0/t = 0.25$, which consists of a full three-dimensional C3D20R solid element model, is included for use as a reference solution. This model has the same general characteristics as the submodel mesh. See *inelasticlinespring_c3d20r_ful.inp* for further details about this mesh. One important difference exists in performing this analysis with shell elements as opposed to continuum elements. The pressure loading is applied to the midsurface of the shell elements as opposed to the continuum elements, where the pressure is accurately applied along the inside surface of the cylinder. For this analysis this discrepancy results in about 10% higher *J*-integral values for the line spring shell element analysis as compared to the full three-dimensional solid element model.

Results from the submodeled analyses are compared to the LS3S line spring element analysis and full solid element mesh for variations of the *J*-integral values along the crack at the a normalized pressure loading of $p\overline{R}/(\sigma_y t) =$ 0.898, where $\overline{R}$ is the mean radius of the cylinder. As seen in *Figure 6*, the line spring elements underestimate the J-integral values for $\phi < 50°$ for reasons described previously. Note that at $\phi = 0°$ the *J*-integral should be zero due to the lack of crack-tip constraint at the cylinder surface. A more refined mesh would be required to model this phenomenon properly. It is quite obvious that the use of shell-to-solid submodeling is required to augment a line spring element model analysis to obtain accurate J-integral values near the surface of the cylinder.

Input files

inelasticlinespring_05.inp	$a_0/t =$ 0.5.
inelasticlinespring_05_nosym.inp	$a_0/t =$ 0.5 without the symmetry assumption across $z = 0$, using line spring element type LS6.
inelasticlinespring_progcrack.f	A program used to create a data file giving the flaw depths as a function of position along the crack.
inelasticlinespring_025.inp	Shallow crack case, $a_0/t = 0.25$.
inelasticlinespring_08.inp	Deep crack case, $a_0/t = 0.8$.
inelasticlinespring_c3d20r_sub.inp	C3D20R ($a_0/t = 0.25$) submodel.
inelasticlinespring_c3d20r_ful.inp	C3D20R ($a_0/t = 0.25$) full model.

References

- Parks, D. M., and C. S. White, “Elastic-Plastic Line-Spring Finite Elements for Surface-Cracked Plates and Shells,” Transactions of the ASME, Journal of Pressure Vessel Technology, vol. 104, pp. 287–292, November 1982.

Figures

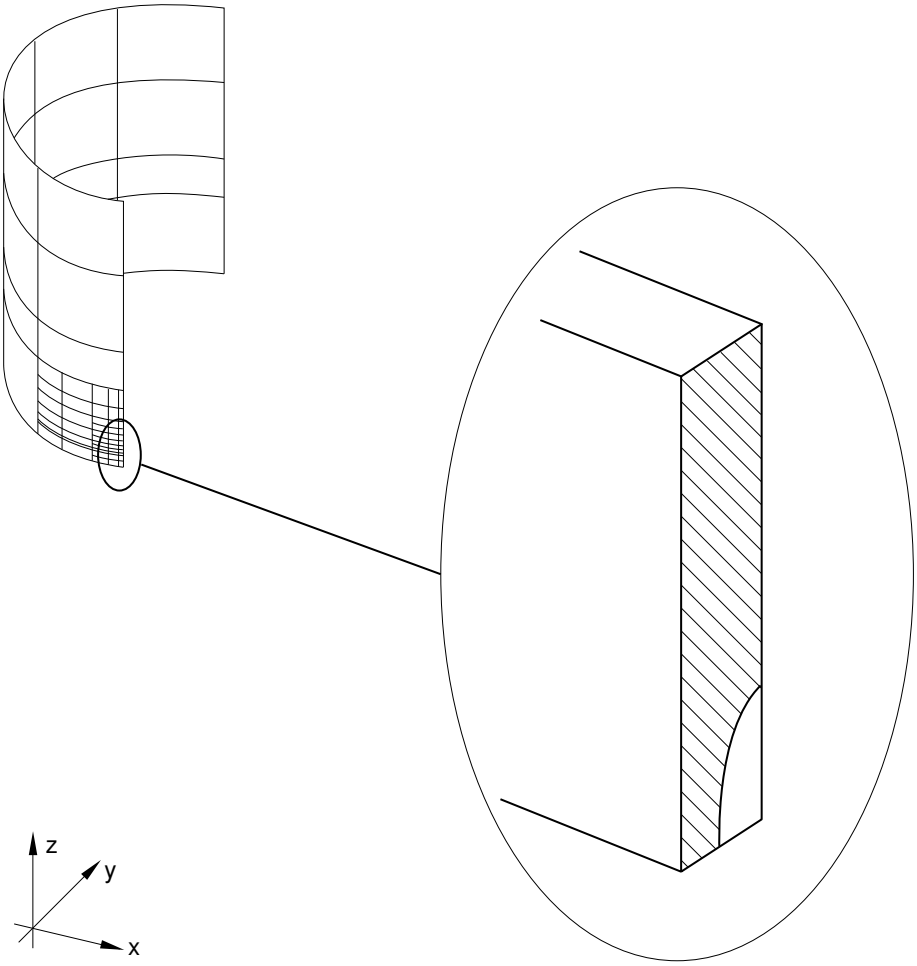
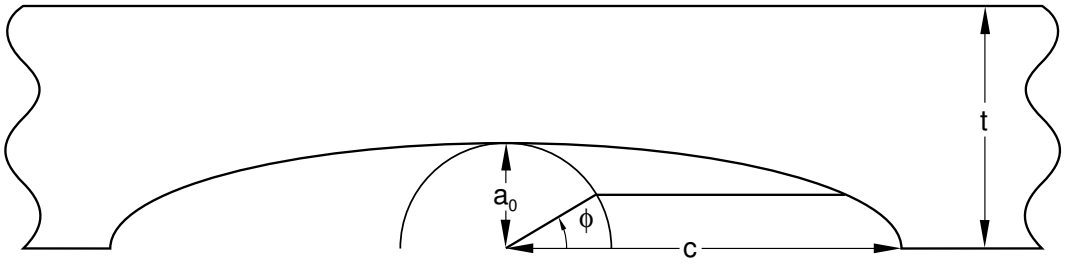


Figure 1: Finite element model for an axial flaw in a pressurized cylinder.



a_0 = maximum flaw depth

2c = surface length of crack

t = shell thickness

ϕ = angle on an inscribed circle for locating a point on the crack

Figure 2: Schematic of a semi-elliptical surface crack.

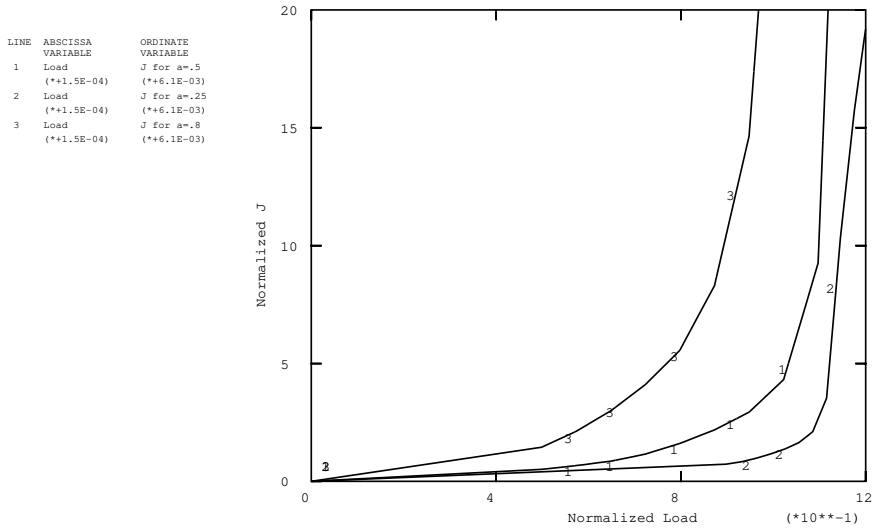


Figure 3: Normalized J-integral values $EJ/\left(\sigma_y^2 t\right)$ versus normalized applied pressure $p\overline{R}/\left(\sigma_y t\right)$, where $\overline{R}$ is the mean radius of the cylinder.

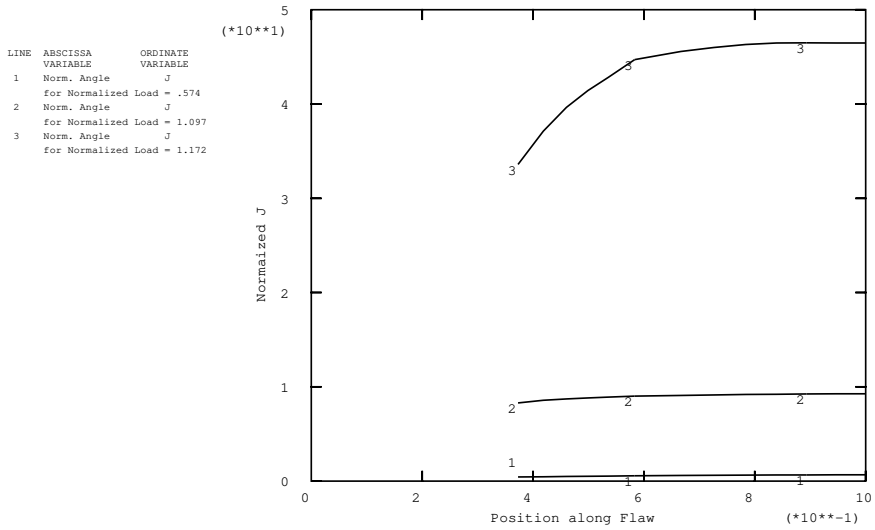


Figure 4: Normalized J-integral values $EJ/\left(\sigma_y^2 t\right)$ versus position along the flaw surface given by $2\phi/\pi$, for $a_0/t = 0.5$, and normalized applied pressures $p\overline{R}/\left(\sigma_y t\right)$= .574, 1.097, and 1.172. $\overline{R}$ is the mean radius of the cylinder.

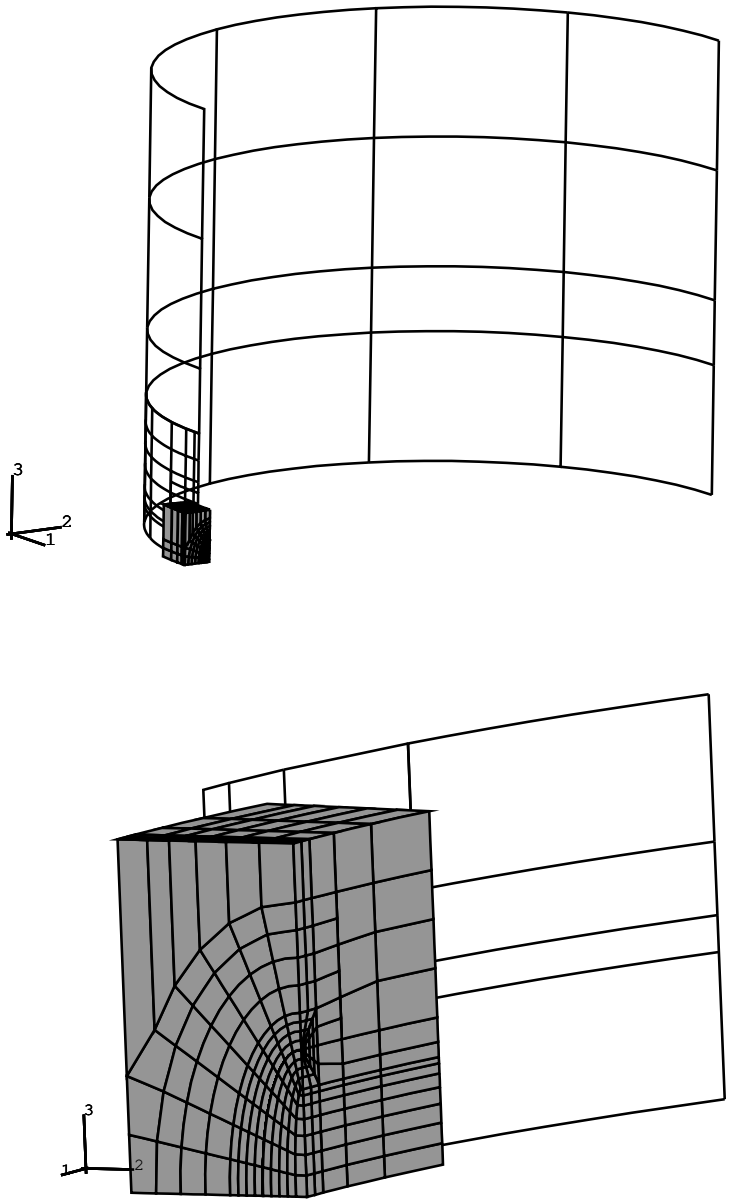


Figure 5: Solid submodel superimposed on shell global model.

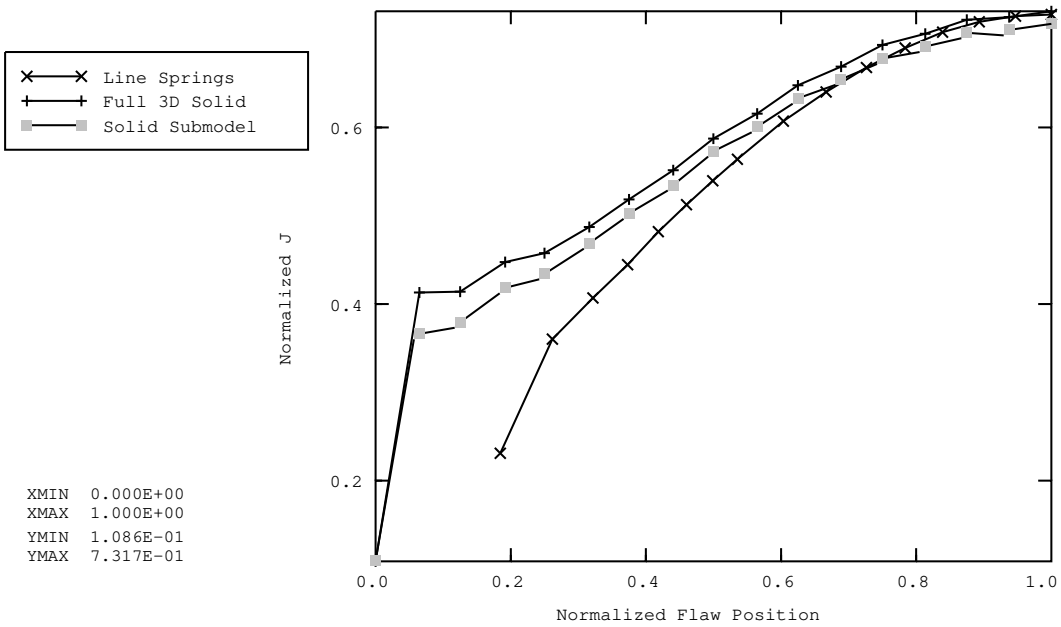


Figure 6: Normalized J-integral values $EJ/\left(\sigma_y^2 t\right)$ versus position along the flaw surface given by $2\phi/\pi$ for $a_0/t = 0.25$ and at the normalized pressure. $p\overline{R}/\left(\sigma_y t\right) = 0.898$. $\overline{R}$ is the mean radius of the cylinder.

Crack growth in a three-point bend specimen

Products: Abaqus/Standard

Overview

This example illustrates the modeling of crack length versus time to simulate crack propagation and the use of crack opening displacement as a crack propagation criterion.

For stable crack growth in ductile materials, experimental evidence indicates that the value of the crack opening displacement (COD) at a specified distance behind the crack tip associated with ongoing crack extension is usually a constant. Abaqus provides the critical crack opening displacement, at a specified distance behind the crack tip, as a crack propagation criterion. The other crack propagation model used in this example—prescribed crack length versus time—is usually used to verify the results obtained from experiments. Abaqus also provides the critical stress criterion for crack propagation in brittle materials.

In this example an edge crack in a three-point bend specimen is allowed to grow based on the crack opening displacement criterion. Crack propagation is first modeled by giving the crack length as a function of time. The data for the crack length are taken from Kunecke, Klingbeil, and Schicker (1993). The data for the crack propagation analysis using the COD criterion are taken from the first analysis. This example demonstrates how the COD criterion can be used in stable crack growth analysis.

Problem description

An edge crack in a three-point bend specimen in plane strain, subjected to Mode I loading, is considered (see *Figure 1*). The crack length to specimen width ratio is 0.2. The length of the specimen is 55 mm, and its width is 10 mm. The specimen is subjected to bending loads such that initially a well-contained plastic zone develops for the stationary crack. Subsequently, the crack is allowed to grow.

Geometry and model

Due to symmetry only one-half of the specimen is analyzed. The crack tip is modeled as initially blunted so that finite deformation effects near the crack tip can be taken into account (geometric nonlinearities considered in the step). The mesh is composed of 1737 CPE4 elements (*Figure 2*). A reasonably fine mesh, necessary to obtain a smooth load versus crack length relation, is used to model the area in which the plastic zone grows and crack propagation occurs. The loading point and the support points for the specimen are simulated by analytical rigid surfaces, as shown in *Figure 2*.

Material

The material is assumed to be elastic-plastic, with a Young's modulus of $E =$ 200 GPa and Poisson's ratio of 0.3. The plastic work hardening data are given in *Table 1*.

Loading and solution control

The analysis is carried out in two stages. The first stage consists of pushing the rigid surface 1.0 mm into the specimen. No crack growth is specified during this stage.

In the second stage the crack is allowed to propagate while the rigid surface is moved an additional 1.951 mm.

Once a crack-tip node debonds, the traction at the tip is initially carried as a reaction force at that node. This force is ramped down to zero according to the amplitude curve specified in the crack propagation analysis. The manner in which the forces at the debonded nodes are ramped down greatly influences the convergence of the solution. The convergence of the solution is also affected by reversals in plastic flow due to crack propagation. In such circumstances, very small time increments are required to continue the analysis. In the present analysis solution controls are defined on the displacement field and on the warping degree of freedom equilibrium equations to relax the tolerances so that more rapid convergence is achieved. Because of the localized nature of the nonlinearity in this problem, the resulting loss of accuracy is not significant. The definition of solution controls is generally not recommended.

Crack length versus time

In the case when the crack length is given as a function of time, the second step in the analysis consists of letting the crack grow according to a prescribed crack length versus time relationship, using the data taken from Kunecke, Klingbeil, and Schicker.

COD criterion

The loading of the specimen and the specification of the COD criterion for crack growth demonstrates the flexibility of the critical crack opening displacement criterion. Frequently, the crack opening displacement is measured at the mouth of the crack tip: this is called the crack mouth opening displacement (CMOD). The crack opening displacement can also be measured at the position where the initial crack tip was located. Alternatively, the crack-tip opening angle (CTOA), defined as the angle between the two surfaces at the tip of the crack, is measured. The crack-tip opening angle can be easily reinterpreted as the crack opening at a distance behind the crack tip. In this example the COD specification required to use both the CMOD and the CTOA criteria is demonstrated.

For the purposes of demonstration the crack opening displacement at the mouth of the crack is used as the initial debond criterion. The first three nodes along the crack propagation surface are allowed to debond when the crack opening displacement at the mouth of the crack reaches a critical value. To achieve this, the following loading sequence is adopted: in Step 1, the specimen is loaded to a particular value (crack propagation analysis is not used), and in Step 2 the first crack-tip node is allowed to debond (crack propagation analysis is used). Steps 3 and 4 and Steps 5 and 6 follow the same sequence as Steps 1 and 2 so that the two successive nodes can debond. Since, the crack opening displacement is measured at the mouth of the crack, the value of the distance behind the crack tip along the secondary surface is different in Steps 2, 4, and 6.

The loading sequence adopted above outlines a way in which the CMOD measurements can be simulated without encountering the situation in which the COD is measured beyond the bound of the specimen, which would lead to an error message. In this example, the loads at which the crack-tip nodes debonded were known *a priori*. In general, such information may not be available, and the restart capabilities in Abaqus can be used to determine the load at which the fracture criterion is satisfied.

The remaining bonded nodes along the crack propagation surface are allowed to debond based on averaged values of the crack-tip opening angles for different accumulated crack lengths. The data prescribed under the crack propagation criteria in Step 7 are the crack opening displacement values that were computed from the crack-tip opening angles observed in the analysis that uses the prescribed crack length versus time criterion. These crack-tip opening angles are converted to critical crack opening displacements at a fixed distance of 0.04 mm behind the crack tip. Hence, the crack opening displacement is measured very close to the current crack tip.

Results and discussion

Figure 3 shows a plot of the accumulated incremental crack length versus time. The user-specified data, as well as the results obtained from the finite element analysis based on the two criteria, are plotted. Good agreement is observed between the user input values and the results from the analysis. The curve based on the COD criterion

does not correspond with the user-specified data toward the end of the analysis because an average crack opening displacement was assumed.

Figure 4 shows the reaction force at the node where the displacements are applied as a function of the accumulated incremental crack length, obtained from the analysis in which the crack length was specified as a function of time. The curve obtained when the COD criterion is used is almost identical and is not shown in this figure.

Figure 5 depicts the variation of the reaction force as a function of the displacement at the rigid body reference node.

The contours of equivalent plastic strain in the near crack-tip region for two different crack advance positions are shown in *Figure 6* and *Figure 7*. Contours of the Mises equivalent stress at the final stage of the analysis are shown in *Figure 8*.

Input files

crackgrowth_lengthvtime.inp	Analysis with the crack length versus time criterion.
crackgrowth_cod.inp	Analysis with the COD criterion.
crackgrowth_model.inp	Model data for the two analysis files.

References

- G. Kunecke, D. Klingbeil, and J. Schicker, “Rißfortschrittssimulation mit der Abaqus-option DEBOND am Beispiel einer statisch belasteten Kerbschlagbiegeprobe,” presented at the Abaqus German Fracture Mechanics group meeting in Stuttgart, November 1993.

Tables

Table 1: Stress-strain data for isotropic plastic behavior.

True Stress (MPa)	True Strain
461.000	0.0
472.810	0.0187
521.390	0.0280
628.960	0.0590
736.306	0.1245
837.413	0.2970
905.831	0.5756
1208.000	1.9942

Figures

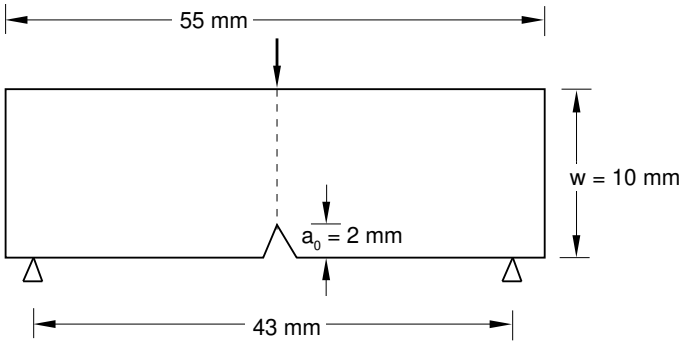


Figure 1: Schematic of the three-point bend specimen.

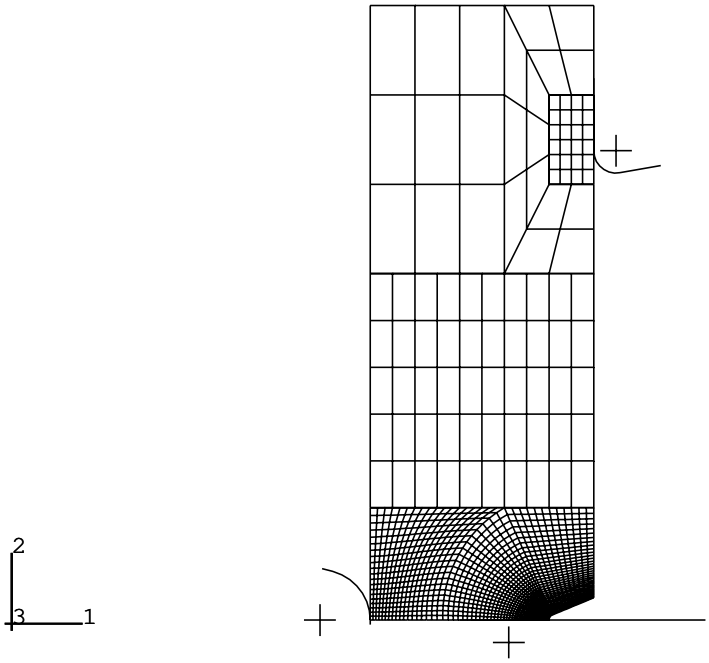


Figure 2: Finite element mesh for the three-point bend specimen.

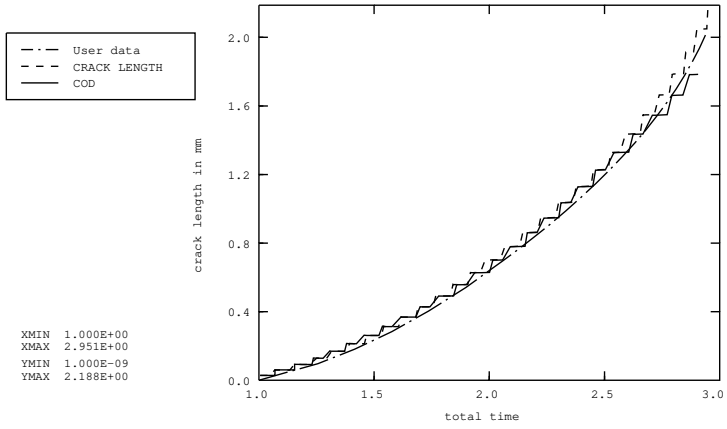


Figure 3: Accumulated incremental crack length versus time.

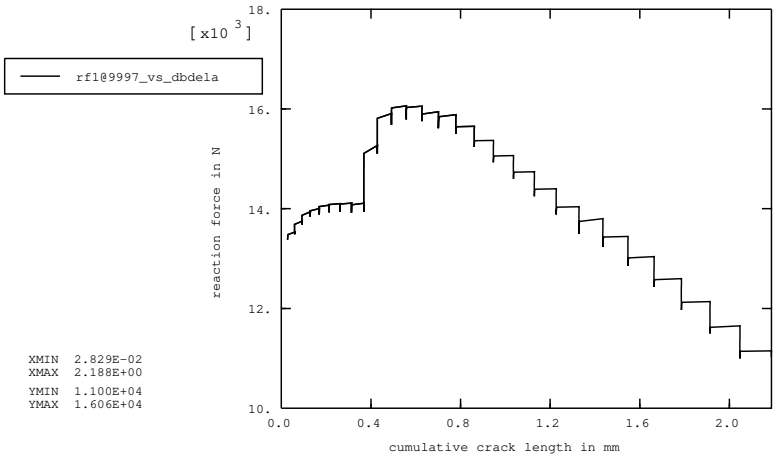


Figure 4: Variation of the reaction force as a function of the cumulative crack length.

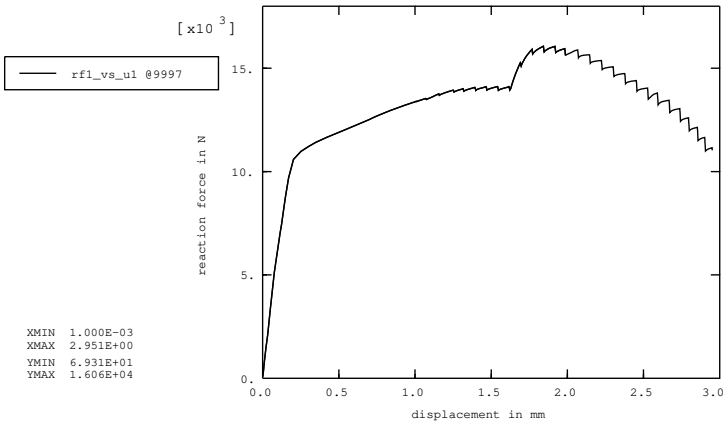


Figure 5: Variation of the reaction force as a function of displacement.

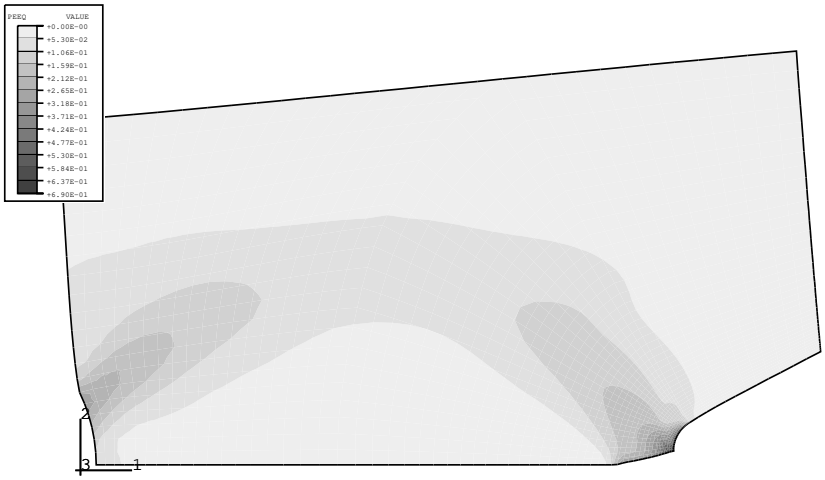


Figure 6: Plastic zone for an accumulated crack length of 1.03 mm.

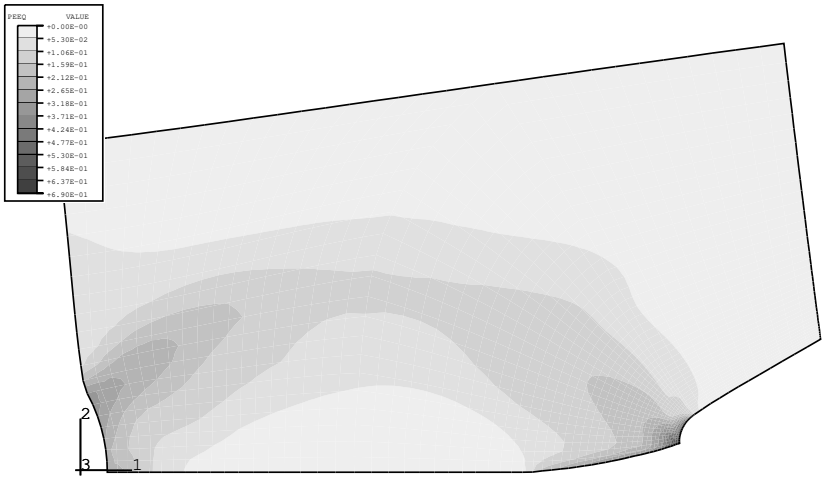


Figure 7: Plastic zone for an accumulated crack length of 2.18 mm.

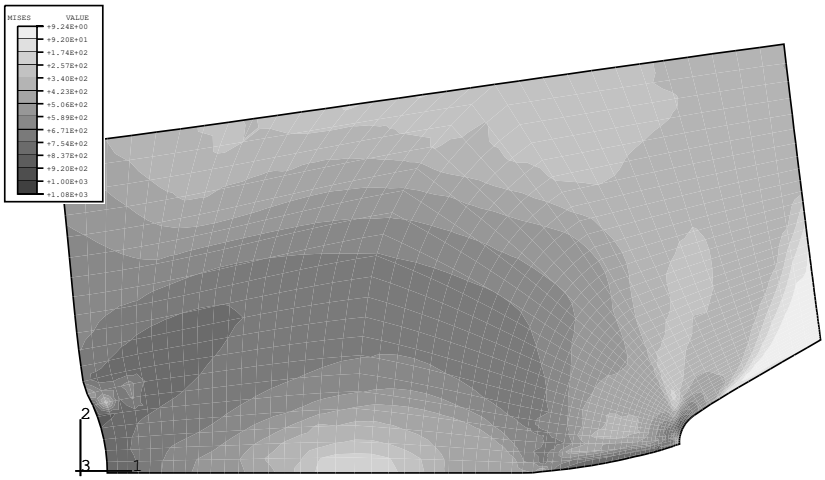


Figure 8: Contours of von Mises stress for an accumulated crack length of 2.18 mm.

Analysis of skin-stiffener debonding under tension

Products: Abaqus/Standard

Overview

This example illustrates the application of cohesive elements in Abaqus to predict the initiation and progression of debonding at the skin-stiffener interface in stiffened panels, which is a common failure mode for this type of structure.

The particular problem considered here is described in Davila (2003); it consists of a stringer flange bonded onto a skin, originally developed by Krueger (2000). The results presented are compared against the experimental results presented in Davila (2003). The problem is analyzed in Abaqus/Standard using a damaged, linear elastic constitutive model for the skin/stiffener interface.

Geometry and model

The problem geometry and loading are depicted in *Figure 1*: a 203-mm-long and 25.4-mm-wide specimen with a total skin thickness of 2.632 mm and maximum flange thickness of 1.88 mm, loaded in tension along the length direction. For the model in which the loading is simulated through prescribed displacements, the free gauge length is 127 mm. The skin thickness direction is comprised of 14 composite plies; while the flange is made up of 10 plies, each having a uniform thickness of 0.188 mm.

The finite element mesh for the three-dimensional model of the debonding problem is identical to that used in Davila (2003) except that the “decohesion” elements utilized in that reference to represent the skin/flange interface are replaced with Abaqus cohesive elements. Both the skin and the flange are modeled by two layers each of C3D8I elements. The interface between them is represented by COH3D8 elements, with the cohesive element mesh sharing nodes with the matching C3D8I meshes of the flange and the skin on either side. The model has a total of 828 solid elements and 174 cohesive elements. As stated in Davila, the two tapered ends of the flange are discretized differently to eliminate model symmetry and to prevent simultaneous delamination from occurring at both ends. The analysis includes a thermal loading step prior to the mechanical loading to simulate the residual stresses in the specimen due to a difference of 157°C between the curing temperature and room temperature. The temperature difference causes residual stresses at the skin/flange interface due to the fact that the thermal expansion coefficients of both the skin and flange material are orthotropic (even though they are specified to be the same) and the ply layups in the skin and flange are different.

Material

The material data, as given in Davila (2003), are reproduced below.

Composite material properties:

Engineering constants	
E_1	144.7 GPa
E_2	9.65 GPa
E_3	9.65 GPa
ν_{12}	0.30
ν_{13}	0.30

Engineering constants	
ν_{23}	0.45
G_{12}	5.2 GPa
G_{13}	5.2 GPa
G_{23}	3.4 GPa

The elastic properties of the interface material are defined using uncoupled traction-separation behavior (see *Defining Elasticity in Terms of Tractions and Separations for Cohesive Elements*), with stiffness values of E= 1.0×10^6 MPa, G_1= 1.0×10^6 MPa, and G_2= 1.0×10^6 MPa. The quadratic traction-interaction failure criterion is chosen for damage initiation in the cohesive elements; and a mixed-mode, energy-based damage evolution law based on the Benzeggagh-Kenane criterion is used for damage propagation. The relevant material data are as follows: N_0=61 MPa, T_0= 68 MPa, S_0= 68 MPa, G_{1C}= 0.075 N/mm, G_{2C}= 0.547 N/mm, G_{3C}= 0.547 N/mm, and η= 1.45.

Results and discussion

The deformed geometry is given in *Figure 2*, which clearly shows the flange separation from the skin. In *Figure 3* the load-extension predictions are compared with the experimental data presented by Davila (2003). The initiation of delamination is marked by the sharp slope change of the curves. The results presented here are obtained using three different viscous regularizations; i.e., $\mu = 1.0e^{-5}$, $1.0e^{-4}$, and $1.0e^{-3}$. Higher viscosity provides better convergence but also affects the results more than lower viscosity. The results using $\mu = 1.0e^{-5}$ are close to the ones without viscosity, while the results using $\mu = 1.0e^{-3}$ agree best with the experimental results in terms of debonding initiation.

Input files

skinflangetension.inp Input data for the three-dimensional skin/flange delamination model.

References

- Davila, C. G., and P. P. Camanho, “Analysis of the Effects of Residual Strains and Defects on Skin/Stiffener Debonding using Decohesion Elements,” SDM Conference, Norfolk, VA, April 7–10, 2003.
- Krueger, R., M. K. Cvitkovich, T. K. O'Brien, and P. J. Minguet, “Testing and Analysis of Composite Skin/Stringer Debonding under Multi-Axial Loading,” *Journal of Composite Materials*, vol. 34, no. 15, pp. 1263–1300, 2000.

Figures

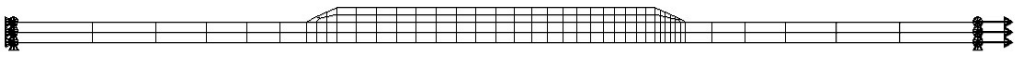


Figure 1: Model geometry for the skin/flange debond problem.

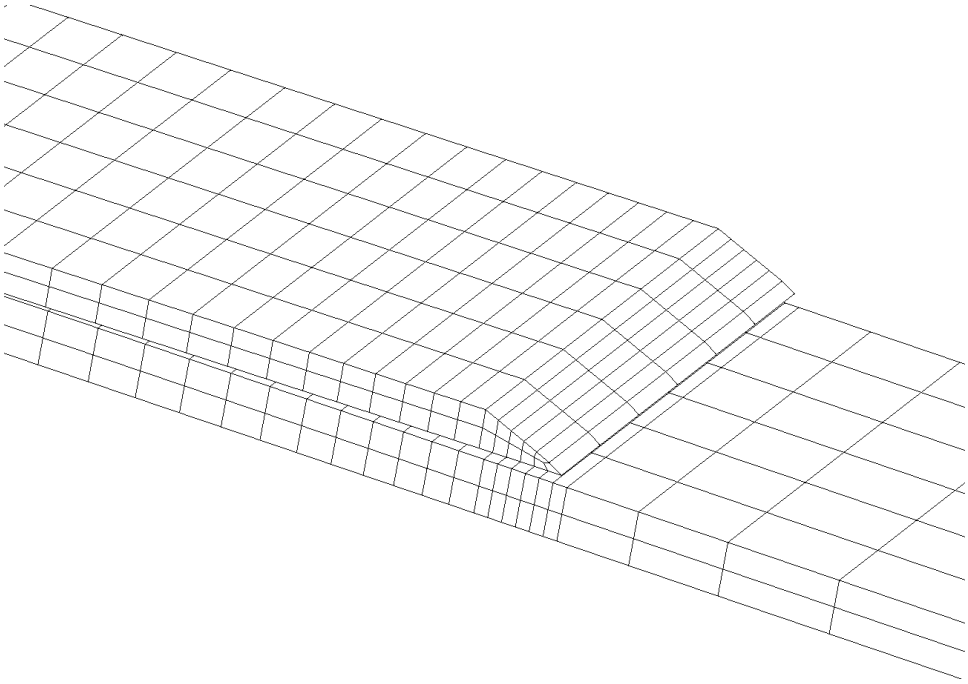


Figure 2: Deformed geometry after skin/flange debond.

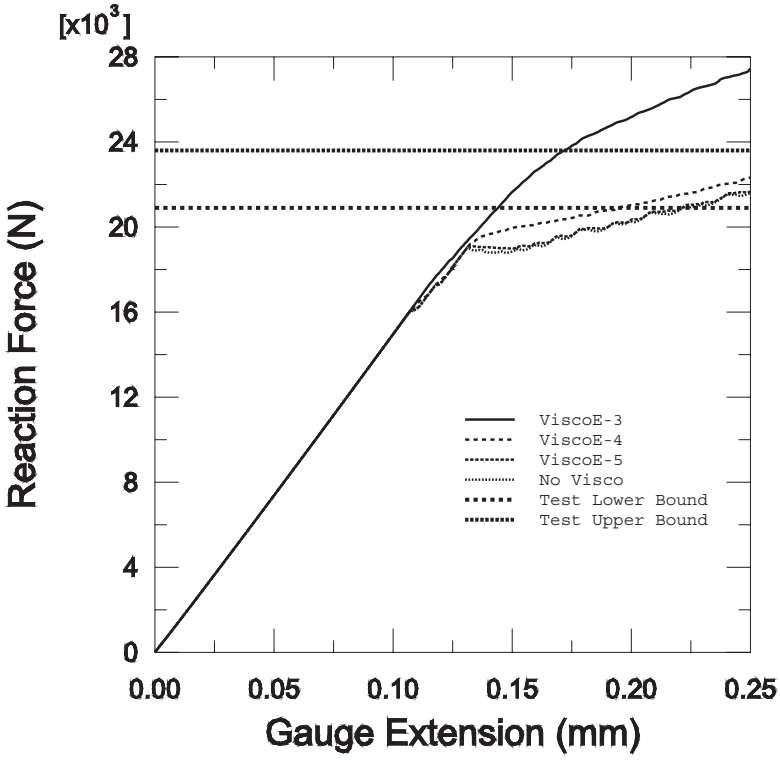


Figure 3: Predicted and experimental debond loads.

Failure of blunt notched fiber metal laminates

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example simulates failure and damage in a fiber metal laminate containing a blunt notch subjected to quasi-static loading conditions.

Fiber metal laminates (FMLs) are composed of laminated thin aluminum layers bonded with intermediate glass fiber-reinforced epoxy layers. FMLs are of great interest in the aerospace industry due to their superior properties, such as high fracture toughness and low-density when compared to solid aluminum sheets.

Cohesive elements are used to model the interlaminar delamination, and three different material models are used to predict the behavior of the fiber-reinforced layer:

- Hashin damage model for unidirectional fiber-reinforced materials (*Hashin Criterion*).
- Damage model proposed by Linde et al. (2004), which is implemented in user subroutine *UMAT*.
- Multiscale damage model for unidirectional fiber-reinforced materials (*Damage Initiation*).

Both Abaqus/Standard and Abaqus/Explicit are used for simulation when the Hashin damage model is used for the fiber-reinforced epoxy layers.

This type of problem is important in the aerospace industry since blunt notches (for example, fastener holes) commonly occur in airplane structures; the strength of the structure containing a blunt notch is a crucial design parameter. The models presented in this example demonstrate how to predict the blunt notch strength, the failure patterns of the fiber and matrix within the fiber-reinforced epoxy layer, and the delamination between different layers of FMLs.

Problem description and material characteristics

Figure 1 shows the geometry of the laminate containing the blunt notch for this example. The laminate is subjected to uniaxial tension in the longitudinal direction. The laminate is made of three layers of aluminum and two layers of 0°/90° glass fiber-reinforced epoxy. Only 1/8 of the laminate needs to be modeled, with appropriate symmetric boundary conditions applied as shown in *Figure 2*. *Figure 2* also shows the through-thickness lay-up of the 1/8 model.

The material behavior of aluminum is assumed to be isotropic elastic-plastic with isotropic hardening. The Young's modulus is 73,800 MPa, and the Poisson's ratio is 0.33; the isotropic hardening data are listed in *Table 1*.

The material behavior of the glass fiber-reinforced epoxy layers is assumed to be orthotropic, with stiffer response along the fiber direction and softer behavior in the matrix. The elastic properties—longitudinal modulus, E_L; transverse modulus, E_T; shear moduli, G_{LT} and G_{TT}; and Poisson's ratios, ν_{TT} and ν_{LT}—are listed in *Table 2*. The subscript "L" refers to the longitudinal direction (or fiber direction), and the subscript "T" refers to the two transverse directions orthogonal to the fiber direction.

The damage initiation and evolution behavior is also assumed to be orthotropic. *Table 3* lists the ultimate values of the longitudinal failure stresses, $\sigma_L^{f,t}$ and $\sigma_L^{f,c}$; transverse failure stresses, $\sigma_T^{f,t}$ and $\sigma_T^{f,c}$; and in-plane shear failure stress, τ_{LT}^{f}. The superscripts "t" and "c" refer to tension and compression, respectively. The fracture energies of the fiber and matrix are assumed to be G_f=12.5 N/mm and G_m=1.0 N/mm, respectively.

Three material models that use the parameters listed above are considered, as follows:

1. The material is modeled based on the Hashin model for damage in unidirectional fiber-reinforced composites available in Abaqus (see *About Damage and Failure for Fiber-Reinforced Composites*).
2. The material is modeled using the damage model proposed by Linde et al. (2004). This damage model is implemented in user subroutine *UMAT* and is referred to in this discussion as the *UMAT* model. Details of the *UMAT* model are provided below.
3. The material is modeled using the multiscale material model in Abaqus (see *Mean-Field Homogenization*), and the damage model is implemented in user subroutine *USDFLD* at the constituent level. Details of the multiscale material and the damage model are provided below.

The adhesive used to bond neighboring layers is modeled using interface layers with a thickness of t=0.001 mm. To simulate the interlaminar delamination, these interface layers are modeled with cohesive elements. The initial elastic properties of each interface are assumed to be isotropic with Young's modulus E=2000 MPa and Poisson's ratio ν=0.33. The failure stresses of the interface layers are assumed to be $t_n^f=t_s^f=t_t^f$=50 MPa; the fracture energies are $G_n=G_s=G_t$=4.0 N/mm. The subscripts "n," "s," and "t" refer to the normal direction and the first and second shear directions (for further discussion of the constitutive modeling methods used for the adhesive layers, see *Defining the Constitutive Response of Cohesive Elements Using a Traction-Separation Description*).

The plate is loaded with displacement boundary conditions applied at the right edge. To simplify the postprocessing, the displacement loading is applied at a reference point and an equation constraint is used to constrain the displacement along the loading direction between the right edge and the reference point. Except for those files designed exclusively to study the effect of the loading direction on the strength, the loading direction (along the global X-direction) aligns with the fiber direction of the 0° fiber-reinforced epoxy layer.

`UMAT` model for fiber-reinforced epoxy layers

For fiber-reinforced epoxy layers, the primary model considered is based on the Hashin damage model for unidirectional fiber-reinforced composites available in both Abaqus/Standard and Abaqus/Explicit. Alternatively, in Abaqus/Standard, the damage in the fiber-reinforced epoxy is also simulated using the model proposed by Linde et al. (2004), which is implemented in user subroutine *UMAT* and is discussed below.

In the *UMAT* model, the damage initiation criteria are expressed in terms of strains. Unlike the Hashin damage model in Abaqus, which uses four internal (damage) variables, the *UMAT* model uses two damage variables to describe damage in the fiber and matrix without distinguishing between tension and compression. Although the performance of the two models is expected to be similar for monotonic loads, such as in this example problem, the results obtained might differ considerably for more complex loads in which, for example, tension is followed by compression. For the *UMAT* model, if the material is subjected to tensile stresses that are large enough to cause partial or full damage (the damage variable corresponding to this damage mode is greater than zero), both tensile and compressive responses of the material are affected. However, in the case of the Hashin damage model, only the tensile response is degraded while the material compressive response is not affected. In many cases the latter behavior is more suitable for modeling fiber-reinforced composites. In this section the governing equations for damage initiation and evolution as proposed by Linde et al. (2004) are discussed, followed by a description of the user subroutine *UMAT* implementation.

Damage in the fiber is initiated when the following criterion is reached:

$$f_f = \sqrt{\frac{\epsilon_{11}^{f,t}}{\epsilon_{11}^{f,c}}(\epsilon_{11})^2 + \left(\epsilon_{11}^{f,t} - \frac{\left(\epsilon_{11}^{f,t}\right)^2}{\epsilon_{11}^{f,c}}\right)\epsilon_{11}} > \epsilon_{11}^{f,t},$$

where $\epsilon_{11}^{f,t} = \sigma_L^{f,t}/C_{11}$, $\epsilon_{11}^{f,c} = \sigma_L^{f,c}/C_{11}$, and C_{ij} are the components of the elasticity matrix in the undamaged state. Once the above criterion is satisfied, the fiber damage variable, d_f, evolves according to the equation

$$d_f = 1 - \frac{\epsilon_{11}^{f,t}}{f_f} e^{\left(-C_{11}\epsilon_{11}^{f,t}\left(f_f - \epsilon_{11}^{f,t}\right)L^c/G_f\right)},$$

where L^c is the characteristic length associated with the material point. Similarly, damage initiation in the matrix is governed by the criterion

$$f_m = \sqrt{\frac{\epsilon_{22}^{f,t}}{\epsilon_{22}^{f,c}}(\epsilon_{22})^2 + \left(\epsilon_{22}^{f,t} - \frac{\left(\epsilon_{22}^{f,t}\right)^2}{\epsilon_{22}^{f,c}}\right)\epsilon_{22} + \left(\frac{\epsilon_{22}^{f,t}}{\epsilon_{12}^{f}}\right)^2 (\epsilon_{12})^2} > \epsilon_{22}^{f,t},$$

where $\epsilon_{22}^{f,t} = \sigma_T^{f,t}/C_{22}$, $\epsilon_{22}^{f,c} = \sigma_T^{f,c}/C_{22}$, and $\epsilon_{12}^{f} = \tau_{LT}^{f}/C_{44}$. The evolution law of the matrix damage variable, d_m, is

$$d_m = 1 - \frac{\epsilon_{22}^{f,t}}{f_m} e^{\left(-C_{22}\epsilon_{22}^{f,t}\left(f_m - \epsilon_{22}^{f,t}\right)L^c/G_m\right)}.$$

During progressive damage the effective elasticity matrix is reduced by the two damage variables d_f and d_m, as follows:

$$\mathbf{C}_d = \begin{bmatrix} (1-d_f)\,C_{11} & (1-d_f)\,(1-d_m)\,C_{12} & (1-d_f)\,C_{13} & 0 & 0 & 0 \\ & (1-d_m)\,C_{22} & (1-d_m)\,C_{23} & 0 & 0 & 0 \\ & & C_{33} & 0 & 0 & 0 \\ & \textit{symmetric} & & (1-d_f)\,(1-d_m)\,C_{44} & 0 & 0 \\ & & & & C_{55} & 0 \\ & & & & & C_{66} \end{bmatrix}.$$

The use of the fracture energy-based damage evolution law and the introduction of the characteristic length L^c in the damage evolution law help to minimize the mesh sensitivity of the numerical results, which is a common problem of constitutive models with strain softening response. However, since the characteristic length calculation is based only on the element geometry without taking into account the real cracking direction, some level of mesh sensitivity remains. Therefore, elements with an aspect ratio close to one are recommended (for a discussion of mesh sensitivity, see *Concrete Damaged Plasticity*).

In user subroutine *UMAT* the stresses are updated according to the following equation:

$$\boldsymbol{\sigma} = \mathbf{C}_d : \boldsymbol{\epsilon}.$$

The Jacobian matrix can be obtained by differentiating the above equation:

$$\begin{aligned} \frac{\partial \boldsymbol{\sigma}}{\partial \boldsymbol{\epsilon}} &= \mathbf{C}_d + \frac{\partial \mathbf{C}_d}{\partial \boldsymbol{\epsilon}} : \boldsymbol{\epsilon} \\ &= \mathbf{C}_d + \left(\frac{\partial \mathbf{C}_d}{\partial d_m} : \boldsymbol{\epsilon}\right)\left(\frac{\partial d_m}{\partial f_m}\frac{\partial f_m}{\partial \boldsymbol{\epsilon}}\right) + \left(\frac{\partial \mathbf{C}_d}{\partial d_f} : \boldsymbol{\epsilon}\right)\left(\frac{\partial d_f}{\partial f_f}\frac{\partial f_f}{\partial \boldsymbol{\epsilon}}\right). \end{aligned}$$

The above Jacobian matrix is not symmetric; therefore, the unsymmetric equation solution technique is recommended if the convergence rate is slow.

To improve convergence, a technique based on viscous regularization (a generalization of the Duvaut-Lions regularization) of the damage variables is implemented in the user subroutine. In this technique we do not use the damage variables calculated from the aforementioned damage evolution equations directly; instead, the damage variables are "regularized" using the following equations:

$$\dot{d}_m^v = \frac{1}{\eta}\left(d_m - d_m^v\right),$$

$$\dot{d}_f^v = \frac{1}{\eta}\left(d_f - d_f^v\right),$$

where d_m and d_f are the matrix and fiber damage variables calculated according to the damage evolution laws presented above, d_m^v and d_f^v are the "regularized" damage variables used in the real calculations of the damaged elasticity matrix and the Jacobian matrix, and η is the viscosity parameter controlling the rate at which the regularized damage variables d_m^v and d_f^v approach the true damage variables d_m and d_f.

To update the "regularized" damage variables at time $t_0 + \Delta t$, the above equations are discretized in time as follows:

$$d_m^v|_{t_0+\Delta t} = \frac{\Delta t}{\eta+\Delta t} d_m\Big|_{t_0+\Delta t} + \frac{\eta}{\eta+\Delta t} d_m^v\Big|_{t_0},$$

$$d_f^v\Big|_{t_0+\Delta t} = \frac{\Delta t}{\eta+\Delta t} d_f\Big|_{t_0+\Delta t} + \frac{\eta}{\eta+\Delta t} d_f^v\Big|_{t_0}.$$

From the above expressions, it can be seen that

$$\frac{\partial d_m^v}{\partial d_m} = \frac{\partial d_f^v}{\partial d_f} = \frac{\Delta t}{\eta+\Delta t}.$$

Therefore, the Jacobian matrix can be further formulated as follows:

$$\frac{\partial \Delta\sigma}{\partial \Delta\epsilon} = \mathbf{C}_d + \left[\left(\frac{\partial \mathbf{C}_d}{\partial d_m^v} : \epsilon\right)\left(\frac{\partial d_m}{\partial f_m}\frac{\partial f_m}{\partial \epsilon}\right) + \left(\frac{\partial \mathbf{C}_d}{\partial d_f^v} : \epsilon\right)\left(\frac{\partial d_f}{\partial f_f}\frac{\partial f_f}{\partial \epsilon}\right)\right]\frac{\Delta t}{\eta+\Delta t}.$$

Care must be exercised to choose an appropriate value for η since a large value of viscosity might cause a noticeable delay in the degradation of the stiffness. To estimate the effect of viscous regularization, the approximate amount of energy associated with viscous regularization is integrated incrementally in user subroutine *UMAT* by updating the variable `SCD` as follows:

$$\Delta E_{SCD} = \frac{1}{2}\left[\left(\mathbf{C}_d : \epsilon\right)|_{t_0} + \left(\mathbf{C}_d : \epsilon\right)|_{t_0+\Delta t}\right] : d\epsilon - \frac{1}{2}\left[\left(\mathbf{C}_d^0 : \epsilon\right)\Big|_{t_0} + \left(\mathbf{C}_d^0 : \epsilon\right)\Big|_{t_0+\Delta t}\right] : d\epsilon,$$

where $\mathbf{C}_d^0$ is the damaged elasticity matrix calculated using the damage variables, d_m and d_f; and $\mathbf{C}_d$ is the damaged elasticity matrix calculated using the regularized damage variables, d_m^v and d_f^v. To avoid unrealistic results due to viscous regularization, the above calculated energy (available as output variable ALLCD) should be small compared to the other real energies in the system, such as the strain energy ALLSE.

This user subroutine can be used with either three-dimensional solid elements or elements with plane stress formulations. In the user subroutine the fiber direction is assumed to be along the local 1 material direction. Therefore, when solid elements are used or when shell elements are used and the fiber direction does not align with the global *X*-direction, a local material orientation should be specified. The damage variables (d_m, d_f, d_m^v, and d_f^v) are stored as solution-dependent variables, which can be viewed in the Visualization module of Abaqus/CAE.

Multiscale model for fiber-reinforced epoxy layers

For fiber-reinforced epoxy layers, the Hashin damage model and the *UMAT* model simulate the damage process at the composite level. Alternatively, in Abaqus/Standard, you can model the damage in the fiber-reinforced epoxy layers using multiscale modeling (*About Damage and Failure for Fiber-Reinforced Composites Using Multiscale Modeling*)

In the multiscale model, the Mori-Tanaka homogenization method is used, and the volume fraction of the fiber is set to 68%. Both the matrix material and fiber material are modeled with a linear elastic material model. The elastic moduli are calibrated to match the composite properties listed in *Table 2*. The matrix material is assumed to be isotropic. The calibrated Poisson's ratio is 0.416, and the calibrated Young's modulus is 3552.32 MPa. The fiber material is modeled with transversely isotropic linear elastic material with the calibrated properties listed in *Table 4*. The shape of the fiber is assumed to be cylindrical with infinite length.

The HSNFIBER criterion is used for damage initiation in the glass fiber, and the HSNMATRIX criterion is used for damage initiation in the epoxy. The composite strength listed in *Table 3* is specified directly as the allowable stress for the multiscale material (*Allowable Stress*). The fracture energies of the constituents (fiber and matrix) are calibrated to match the stress/strain response of the composite in a uniaxial tension test in both the longitudinal (fiber) direction and the transverse direction orthogonal to the fiber direction. The calibrated fiber fracture energy is G_f = 15.8 N/mm, and the calibrated matrix fracture energy is G_m = 1.4 N/mm.

Finite element model

The finite element model uses a separate mesh for each of the respective layers shown in *Figure 2*: two aluminum layers, two fiber-reinforced epoxy layers, and three adhesive layers. While not required, a similar finite element discretization in the plane of the laminate, such as that shown in *Figure 3*, can be used for all layers.

Modeling considerations for aluminum layers

Due to the interactions with the fiber-reinforced epoxy layers, the stress state within the aluminum layers (especially surrounding the notch tip) cannot be approximated using the plane stress assumption. To model this three-dimensional plasticity stress state accurately, solid elements must be used for the aluminum layers. In Abaqus/Standard incompatible mode elements (C3D8I) are used since local bending might exist in the post-failure region surrounding the notch. For the Abaqus/Explicit analysis, reduced-integration elements (C3D8R) are used for modeling the aluminum layers.

Modeling considerations for glass fiber-reinforced epoxy layers

The plane stress assumption can be used safely within the fiber-reinforced epoxy layers; therefore, either solid elements or shell elements can be adopted for these layers. However, it is important to have an accurate representation of the through-thickness geometry to model the interface between the adhesive and the fiber-reinforced epoxy realistically. This is achieved most conveniently with solid elements or continuum shell elements instead of conventional shell elements. The Hashin damage model for unidirectional fiber-reinforced materials is available only for elements with a plane stress formulation. Therefore, continuum shell elements are used with this model. Models are also included in which continuum elements (C3D8R or C3D8) are used along with user subroutine *UMAT* to model the fiber-reinforced epoxy layers. Same elements used with the *UMAT* model are also used with the multiscale model to model the fiber-reinforced epoxy layers.

Modeling considerations for adhesive layers

Cohesive elements (COH3D8) are used for the interface layers. The elastic response is defined in terms of a traction-separation law with uncoupled behavior between the normal and shear components. For convenience, a constitutive thickness of 1.0 mm is used so that we do not need to distinguish between the separation displacement and the nominal strain (NE). However, since the actual thickness is 0.001 mm, the diagonal terms in the elasticity matrix need to be scaled by the inverse of the actual thickness as follows:

$$K_{nn} = \frac{E}{t}, \quad K_{ss} = \frac{G}{t}, \quad K_{tt} = \frac{G}{t}.$$

The quadratic nominal strain criterion is used for the damage initiation:

$$\sqrt{\left(\frac{\epsilon_n}{\epsilon_n^f}\right)^2+\left(\frac{\epsilon_s}{\epsilon_s^f}\right)^2+\left(\frac{\epsilon_t}{\epsilon_t^f}\right)^2}>1.0\,,\quad \epsilon_n^f=\frac{t_n^f}{E/t}\,,\quad \epsilon_s^f=\frac{t_s^f}{G/t}\,,\quad \epsilon_t^f=\frac{t_t^f}{G/t}.$$

The damage evolution is based on fracture energy with the quadratic power law for the mixed mode behavior and exponential softening behavior (see *Defining the Constitutive Response of Cohesive Elements Using a Traction-Separation Description*).

Results and discussion

Results for each analysis are discussed in the following sections.

Abaqus/Standard results

Damage to the fiber-reinforced epoxy plays a key role in the response for the loading considered. *Figure 4* shows the load-displacement curve for the 0° loading direction for the Linde and Hashin (built-in) the damage models considered for the fiber-reinforced epoxy. The response shows a "bilinear" shape before the sudden loss of loading capacity; that is, an initial linear curve representing the initial elastic region, a smoothly deflecting nonlinear curve representing the local plasticity, and a second linear curve representing the net section yielding. The effect of the element type was studied using the *UMAT* model and C3D8R, C3D8, and SC8R elements; and the results are summarized in *Figure 5* and *Table 6*. The effect of the element type was also studied using the multiscale model, and the results are summarized in *Figure 6* and *Table 7*.The numerical results obtained using different element types and different damage models are similar and show a good agreement with the experimental results of De Vries (2001).

The fiber and matrix damage patterns in the 0° fiber-reinforced epoxy layer at the failure load are shown in

- *Figure 7* and *Figure 8*, for the Hashin damage model for fiber-reinforced materials;
- *Figure 9* and *Figure 10*, for the *UMAT* model; and
- *Figure 11* and *Figure 12*, for the multiscale material model.

It can be seen that the fiber damage in the 0° fiber-reinforced epoxy layer propagates along the ligament above the blunt notch tip (that is, orthogonal to the loading direction).

Figure 13 shows the matrix damage in the 90° layer for the damage model of Linde et al. (2004). There is no fiber damage in the 90° fiber-reinforced epoxy layer prior to the sudden fracture. Interlaminar damage is most severe between the 0° fiber-reinforced epoxy layer and the aluminum layer. These observations are in agreement with the experimental results of De Vries (2001).

The load-displacement results for different values of the viscosity parameter, η are given in

- *Figure 14* and *Table 5*, for the Hashin damage model for fiber-reinforced materials;
- *Figure 15* and *Table 8*, for the *UMAT* model; and
- *Figure 16* and *Table 9* for the multiscale material model.

The smaller the viscosity, the more abrupt the failure and the smaller the failure strength. For the Hashin damage model and the *UMAT* model, although a viscosity of 0.001 seems to overestimate the failure strength by a few percent (*Table 5* and *Table 8*), the convergence is noticeably improved. Therefore, a viscosity of 0.001 is used for all the other studies in this example. For the multiscale model, the comparison of failure strength with different values of viscosity is given in *Table 9*. The predicted strengths are very close for viscosity values smaller than 0.0005; therefore, a viscosity of 0.0005 is chosen for all the other studies using a multiscale material in this example.

For the Hashin damage model and the multiscale damage model for fiber-reinforced materials, only the viscosity in the fiber direction was varied while the viscosity in the matrix direction was kept constant at 0.005. This improved convergence and did not markedly affect the results.

The effect of the loading direction on the blunt notch strength is studied using the three-dimensional element, C3D8R, with the *UMAT* model. Three tests are performed in which the local material orientations in the 0°/90° fiber-reinforced epoxy are rotated by an angle of 15°, 30°, and 45°, respectively. For example, for a loading angle of 15° the fiber orientation in the 0° fiber-reinforced epoxy layer would be at a 15° angle with respect to the *X*-direction, while the fiber orientation in the 90° fiber-reinforced epoxy layer would be at an angle of −75° with respect to the *X*-direction (*Figure 17*). As can be seen in *Figure 18*, strain hardening is smaller for the larger loading angles. As can be seen in *Figure 19*, the failure strength decreases with the increasing loading angle and reaches the minimum at the 45° loading angle (the response for even larger loading angles is expected to be approximately symmetric with respect to the 45° angle due to the symmetric nature of the 0°/90° fiber-reinforced epoxy layer). As stated by De Vries (2001), this is expected and reflects the poor shear properties of the fiber-reinforced epoxy layer.

In the above discussions the net blunt notch strength is defined as $P_{max}/l_1 t$, where l_1 is the length of the ligament above the notch and t is the total thickness of the laminate. This example demonstrates that the approach employed in the study can be used to predict the blunt notch strength of the fiber metal laminates.

Abaqus/Explicit results

In the Abaqus/Explicit simulation, we only consider loading along the 0° ply. The simulation is conducted without damage stabilization, and no mass scaling is used. However, in order to reduce the computational time, the total loading is applied in a short interval of time (0.001 s). The overall load-displacement curve obtained from the explicit dynamic simulation is compared with the Abaqus/Standard result (with viscosity of 0.001) in *Figure 20*. The results from the explicit dynamic simulation are presented using an antialiasing filter to remove high frequency noise (see *Filtering Output and Operating on Output in Abaqus/Explicit*). The overall response compares well with the Abaqus/Standard results with some differences in the peak value of the load and in the post-peak response. Damage stabilization is used in the Abaqus/Standard simulation to achieve convergence and is likely to change the overall response (especially in the post-peak portion of the load-displacement curve). On the other hand, the Abaqus/Explicit simulation does not use damage stabilization and is better able to capture the dynamic behavior inherent in the damage and failure processes. The contour plots of various damage variables in the 0° and 90° plies agree qualitatively with the corresponding plots obtained from the Abaqus/Standard simulation using the Hashin damage model.

Python scripts

fml_c3d8r_deg0_vis1_std.py	C3D8R used in the fiber-reinforced epoxy layer, a loading angle of 0°, and a viscosity of 0.001.
fml_c3d8r_deg0_vis2_std.py	C3D8R used in the fiber-reinforced epoxy layer, a loading angle of 0°, and a viscosity of 0.0004.
fml_c3d8r_deg0_vis3_std.py	C3D8R used in the fiber-reinforced epoxy layer, a loading angle of 0°, and a viscosity of 0.00016.
fml_c3d8r_deg0_vis4_std.py	C3D8R used in the fiber-reinforced epoxy layer, a loading angle of 0°, and a viscosity of 0.000064.
fml_c3d8_deg0_vis1_std.py	C3D8 used in the fiber-reinforced epoxy layer, a loading angle of 0°, and a viscosity of 0.001.
fml_sc8r_deg0_vis1_std.py	SC8R used in the fiber-reinforced epoxy layer, a loading angle of 0°, and a viscosity of 0.001.
fml_c3d8r_deg15_vis1_std.py	C3D8R used in the fiber-reinforced epoxy layer, a loading angle of 15°, and a viscosity of 0.001.
fml_c3d8r_deg30_vis1_std.py	C3D8R used in the fiber-reinforced epoxy layer, a loading angle of 30°, and a viscosity of 0.001.
fml_c3d8r_deg45_vis1_std.py	C3D8R used in the fiber-reinforced epoxy layer, a loading angle of 45°, and a viscosity of 0.001.

mfh_calib_ud.py	Script that calibrates the constituent properties for the multiscale model.

Input files

Abaqus/Standard input files

fml_frm_sc8r_deg0_vis001_std.inp	SC8R used in the fiber-reinforced epoxy layer, a loading angle of 0°, and a viscosity of 0.001 in the fiber direction (using Hashin damage model).
fml_frm_sc8r_deg0_vis0005_std.inp	SC8R used in the fiber-reinforced epoxy layer, a loading angle of 0°, and a viscosity of 0.0005 in the fiber direction (using Hashin damage model).
fml_frm_sc8r_deg0_vis00025_std.inp	SC8R used in the fiber-reinforced epoxy layer, a loading angle of 0°, and a viscosity of 0.00025 in the fiber direction (using Hashin damage model).
fml_c3d8r_deg0_vis1_std.inp	C3D8R used in the fiber-reinforced epoxy layer, a loading angle of 0°, and a viscosity of 0.001 (using *UMAT* model).
fml_c3d8r_deg0_vis2_std.inp	C3D8R used in the fiber-reinforced epoxy layer, a loading angle of 0°, and a viscosity of 0.0004 (using *UMAT* model).
fml_c3d8r_deg0_vis3_std.inp	C3D8R used in the fiber-reinforced epoxy layer, a loading angle of 0°, and a viscosity of 0.00016 (using *UMAT* model).
fml_c3d8r_deg0_vis4_std.inp	C3D8R used in the fiber-reinforced epoxy layer, a loading angle of 0°, and a viscosity of 0.000064 (using *UMAT* model).
fml_c3d8_deg0_vis1_std.inp	C3D8 used in the fiber-reinforced epoxy layer, a loading angle of 0°, and a viscosity of 0.001 (using *UMAT* model).
fml_sc8r_deg0_vis1_std.inp	SC8R used in the fiber-reinforced epoxy layer, a loading angle of 0°, and a viscosity of 0.001 (using *UMAT* model).
fml_c3d8r_deg15_vis1_std.inp	C3D8R used in the fiber-reinforced epoxy layer, a loading angle of 15°, and a viscosity of 0.001 (using *UMAT* model).
fml_c3d8r_deg30_vis1_std.inp	C3D8R used in the fiber-reinforced epoxy layer, a loading angle of 30°, and a viscosity of 0.001 (using *UMAT* model).
fml_c3d8r_deg45_vis1_std.inp	C3D8R used in the fiber-reinforced epoxy layer, a loading angle of 45°, and a viscosity of 0.001 (using *UMAT* model).
exa_fml_ortho_damage_umat.f	User subroutine *UMAT* for modeling the damage initiation and evolution in the fiber-reinforced epoxy layers.
fml_mfh_c3d8r_deg0_vis1_std.inp	C3D8R used in the fiber-reinforced epoxy layer, a loading angle of 0°, and a viscosity of 0.001 in the fiber material (using multiscale material model).
fml_mfh_c3d8r_deg0_vis2_std.inp	C3D8R used in the fiber-reinforced epoxy layer, a loading angle of 0°, and a viscosity of 0.0005 in the fiber material (using multiscale material model).
fml_mfh_c3d8r_deg0_vis3_std.inp	C3D8R used in the fiber-reinforced epoxy layer, a loading angle of 0°, and a viscosity of 0.00025 in the fiber material (using multiscale material model).
fml_mfh_c3d8r_deg0_vis4_std.inp	C3D8R used in the fiber-reinforced epoxy layer, a loading angle of 0°, and a viscosity of 0.000125 in the fiber material (using multiscale material model).
fml_mfh_c3d8_deg0_vis2_std.inp	C3D8 used in the fiber-reinforced epoxy layer, a loading angle of 0°, and a viscosity of 0.0005 in the fiber material (using multiscale material model).

fml_mfh_sc8r_deg0_vis2_std.inp	SC8R used in the fiber-reinforced epoxy layer, a loading angle of 0°, and a viscosity of 0.0005 in the fiber material (using multiscale material model).

Abaqus/Explicit input file

fml_frm_sc8r_deg0_exp.inp	SC8R elements used in the fiber-reinforced epoxy layer and a loading angle of 0° (using Hashin damage model).

References

- De Vries, T. J., "Blunt and Sharp Notch Behavior of Glare Laminates," *Ph.D dissertation, Delft University Press*, 2001.
- Hagenbeek, M., C. Van Hengel, O. J. Bosker, and C. A. J. R. Vermeeren, "Static Properties of Fibre Metal Laminates," *Applied Composite Materials*, vol. 10 207–222, 2003.
- Linde, P., J. Pleitner, H. De Boer, and C. Carmone, "Modelling and Simulation of Fiber Metal Laminates," *Abaqus Users' Conference*, 2004.
- Mayes, J.S.., and A.C.. Hansen, "Multicontinuum Failure Analysis of Composite Structural Laminates," *Mechanics of Composite Materials and Structures*, vol. 8, no. 4 249-262, 2001.

Tables

Table 1: Isotropic hardening data for aluminum.

Yield stress (MPa)	Plastic strain (%)
300	0.000
320	0.016
340	0.047
355	0.119
375	0.449
390	1.036
410	2.130
430	3.439
450	5.133
470	8.000
484	14.710

Table 2: Orthotropic elastic properties of fiber-reinforced epoxy.

E_L (MPa)	E_T (MPa)	G_{LT} (MPa)	G_{TT} (MPa)	ν_{TT}	ν_{LT}
55000	9500	5500	3000	0.45	0.33

Table 3: Orthotropic damage initiation properties of fiber-reinforced epoxy.

$\sigma_L^{f,t}$ (MPa)	$\sigma_L^{f,c}$ (MPa)	$\sigma_T^{f,t}$ (MPa)	$\sigma_T^{f,c}$ (MPa)	τ_{LT}^{f} (MPa)
2500	2000	50	150	50

Table 4: Calibrated elastic constants of the glass fiber material.

E_L (MPa)	E_T (MPa)	G_{LT} (MPa)	ν_{TT}	ν_{LT}
79189	15753	31883	0.29	0.21

Table 5: Net blunt notch strength (MPa) for different values of the viscosity parameter in fiber direction (using Hashin damage model, viscosity in the matrix direction η_m=0.005).

Numerical results (SC8R, 0° loading angle)			Experimental results (De Vries, 2001)
η_f=0.001	η_f=0.0005	η_f=0.00025	
462.1	456.4	453.2	446

Table 6: Net blunt notch strength (MPa) for different element types used in the fiber-reinforced epoxy layers (using `UMAT` model).

Numerical results (η=0.001, 0° loading angle)			Experimental results (De Vries, 2001)
C3D8R	C3D8	SC8R	
463.7	467.1	458.7	446

Table 7: Net blunt notch strength (MPa) for different element types used in the fiber-reinforced epoxy layers (using multiscale model).

Numerical results (η=0.0005, 0° loading angle)			Experimental results (De Vries, 2001)
C3D8R	C3D8	SC8R	
448.1	479.8	482.3	446

Table 8: Net blunt notch strength (MPa) for different values of the viscosity parameter (using `UMAT` model).

Numerical results (C3D8R, 0° loading angle)				Experimental results (De Vries, 2001)
η=0.001	η=0.0004	η=0.00016	η=0.000064	
463.7	453.8	449.2	448.2	446

Table 9: Net blunt notch strength (MPa) for different values of the viscosity parameter (using multiscale model).

Numerical results (C3D8R, 0^o loading angle)				Experimental results (De Vries, 2001)
η=0.001	η=0.0005	η=0.00025	η=0.000125	
458.6	448.1	444.5	442.8	446

Figures

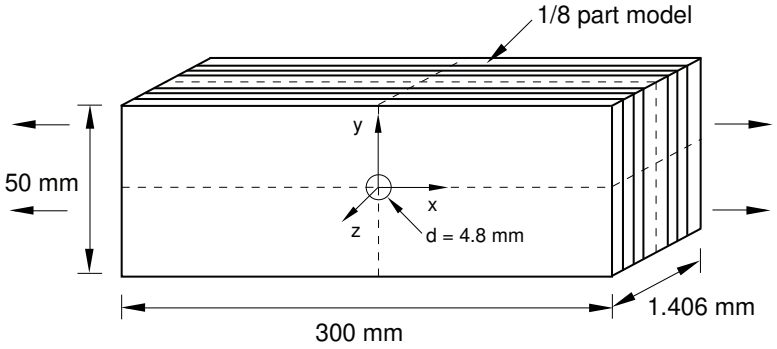


Figure 1: Plate geometry.

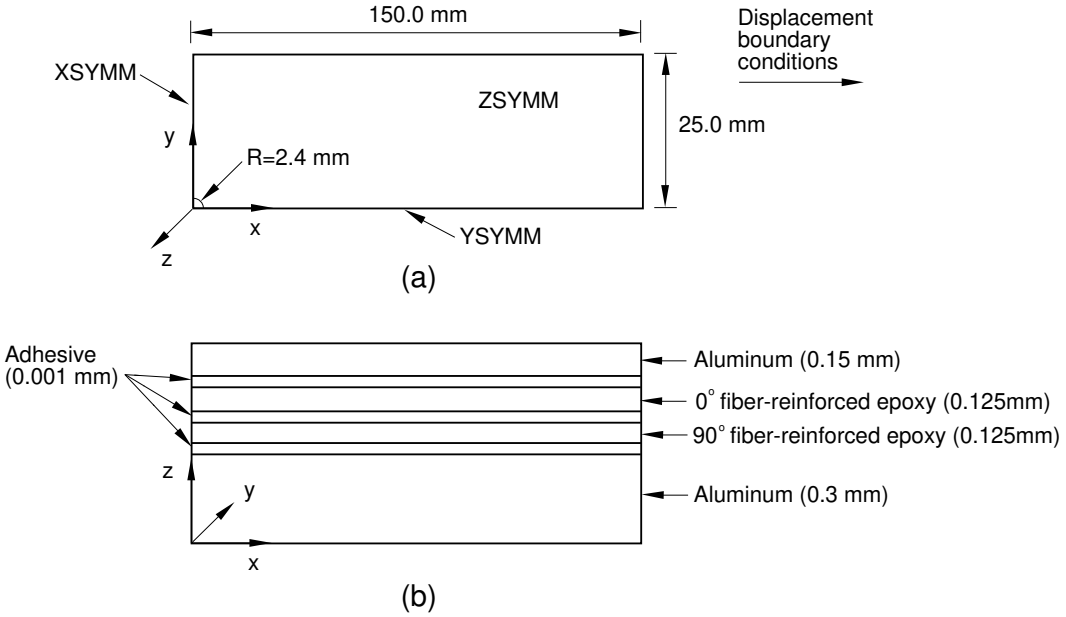


Figure 2: (a) In-plane view of the 1/8 plate; (b) through-thickness lay-up of the 1/8 plate.

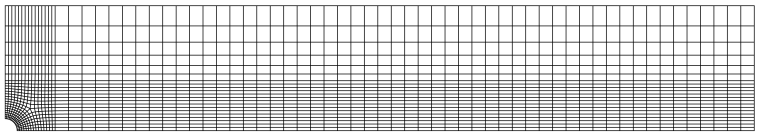


Figure 3: Finite element mesh.

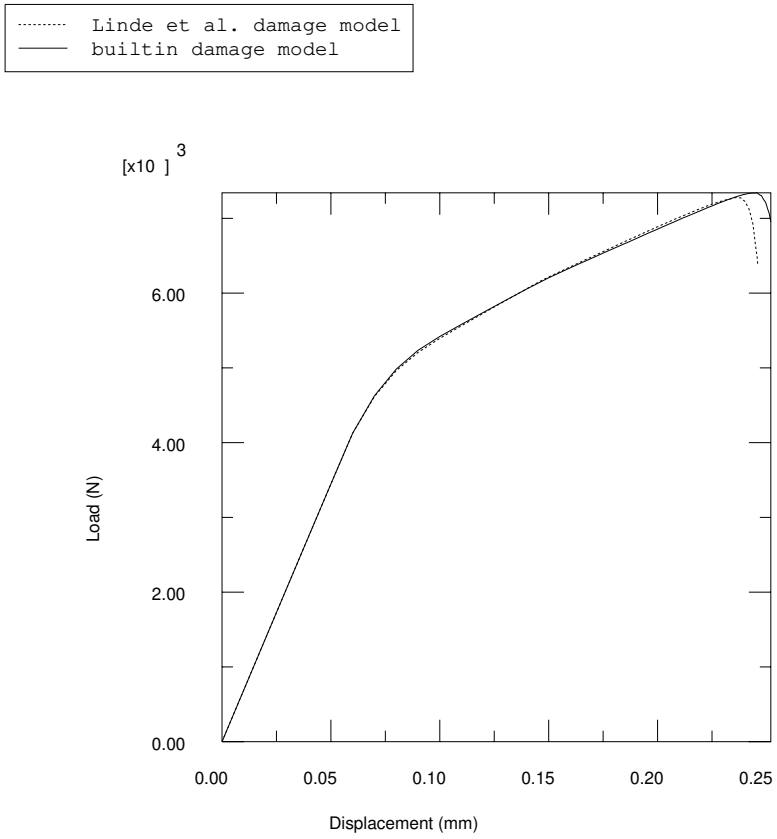


Figure 4: Load-displacement curves for different damage models in the fiber-reinforced epoxy layer for the 0° loading direction, η=0.001.

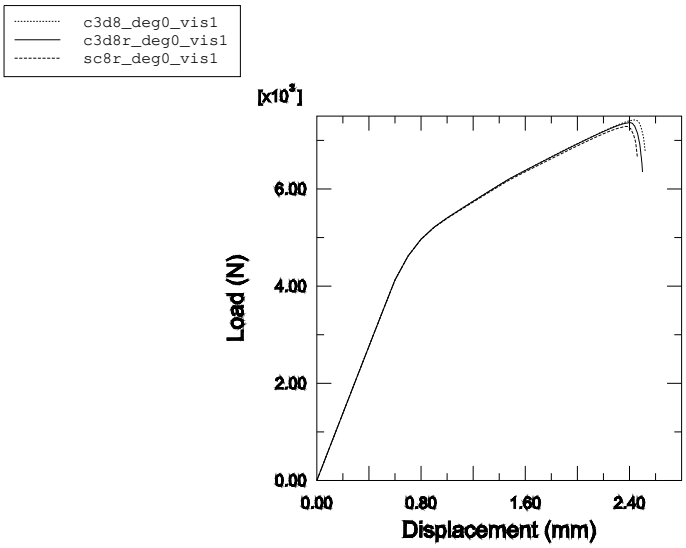


Figure 5: Load-displacement curves for different element types in the fiber-reinforced epoxy layer for the 0° loading direction (using `UMAT` model).

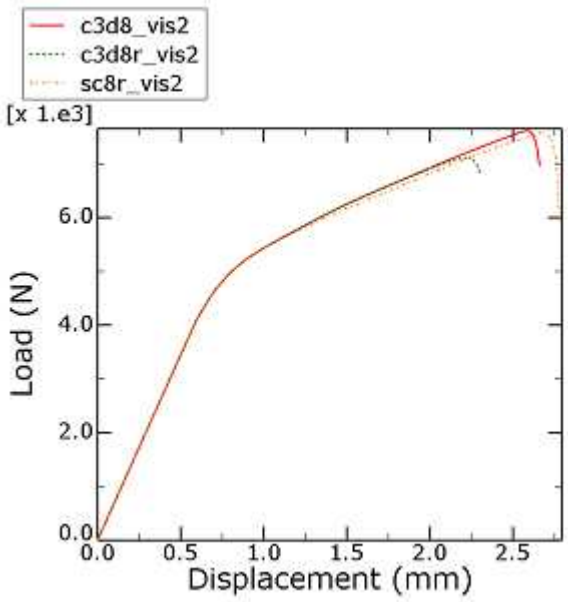


Figure 6: Load-displacement curves for different element types in the fiber-reinforced epoxy layer for the 0° loading direction (using multiscale model)

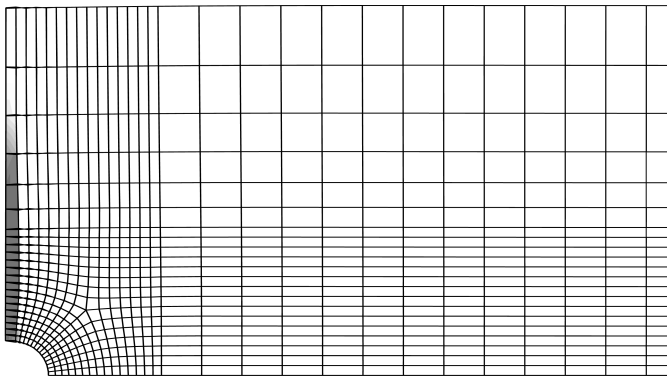


Figure 7: Fiber damage pattern in the 0° fiber-reinforced epoxy layer for the 0° loading direction (using the Hashin damage model, DAMAGEFT contour plot).

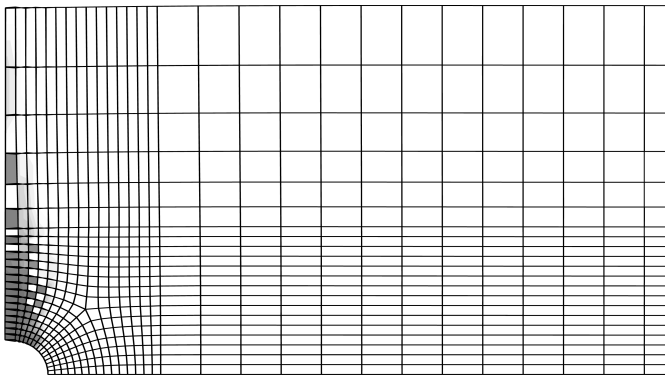


Figure 8: Matrix damage pattern in the 0° fiber-reinforced epoxy layer for the 0° loading direction (using the Hashin damage model, DAMAGEMT contour plot).

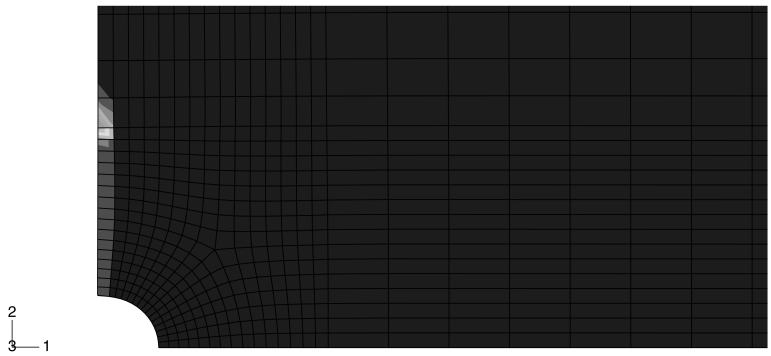


Figure 9: Fiber damage pattern in the 0° fiber-reinforced epoxy layer for the 0° loading direction (using `UMAT` model, SDV3 contour plot).

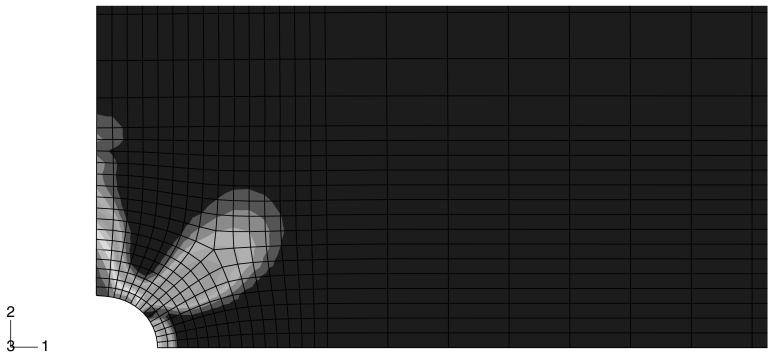


Figure 10: Matrix damage pattern in the 0° fiber-reinforced epoxy layer for the 0° loading direction (using `UMAT` model, SDV4 contour plot).

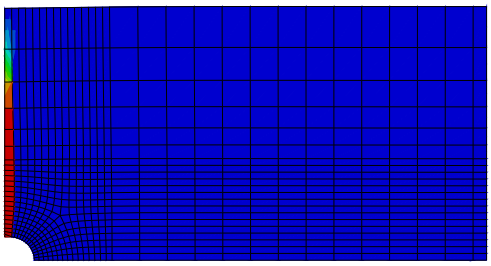


Figure 11: Fiber damage pattern in the 0° fiber-reinforced epoxy layer for the 0° loading direction (using multiscale material model, DAMAGEFT contour plot).

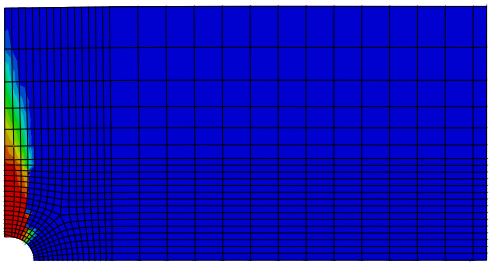


Figure 12: Matrix damage pattern in the 0° fiber-reinforced epoxy layer for the 0° loading direction (using multiscale material model, DAMAGEMT contour plot).

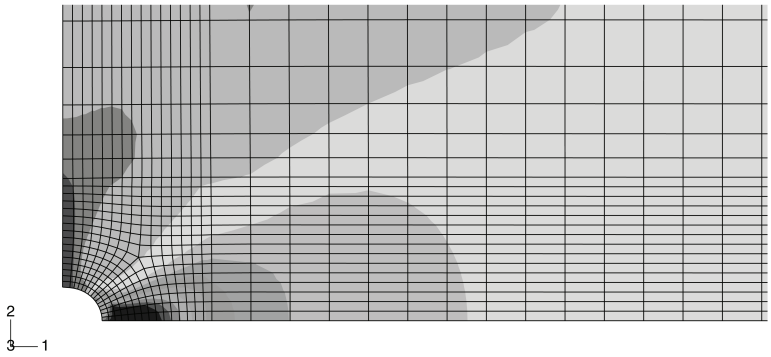


Figure 13: Matrix damage pattern in the 90° fiber-reinforced epoxy layer for the 0° loading direction (using `UMAT` model, SDV4 contour plot).

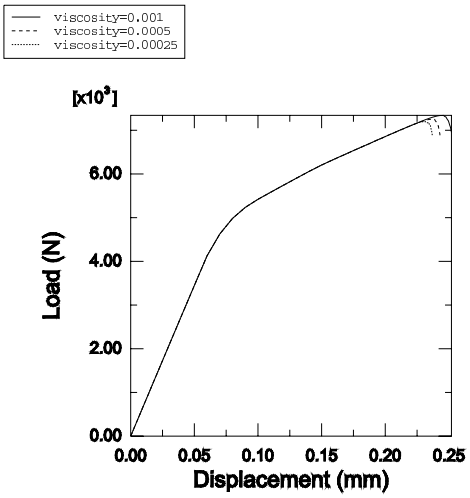


Figure 14: Load-displacement curves for different values of the viscosity parameter for the 0° loading direction (using the Hashin damage model).

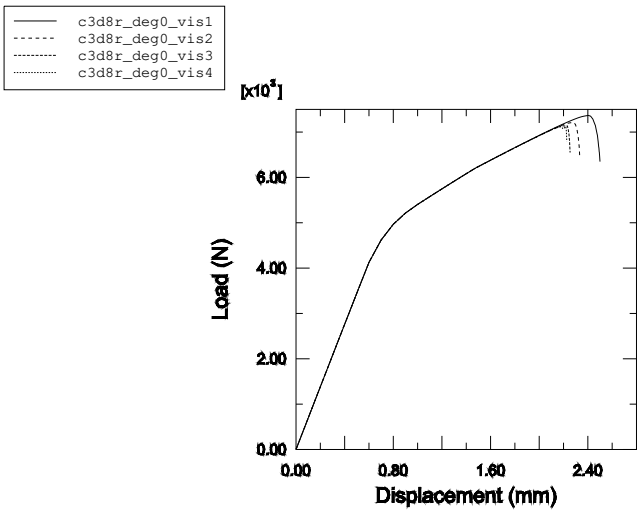


Figure 15: Load-displacement curves for different values of the viscosity parameter for the 0° loading direction (using `UMAT`).

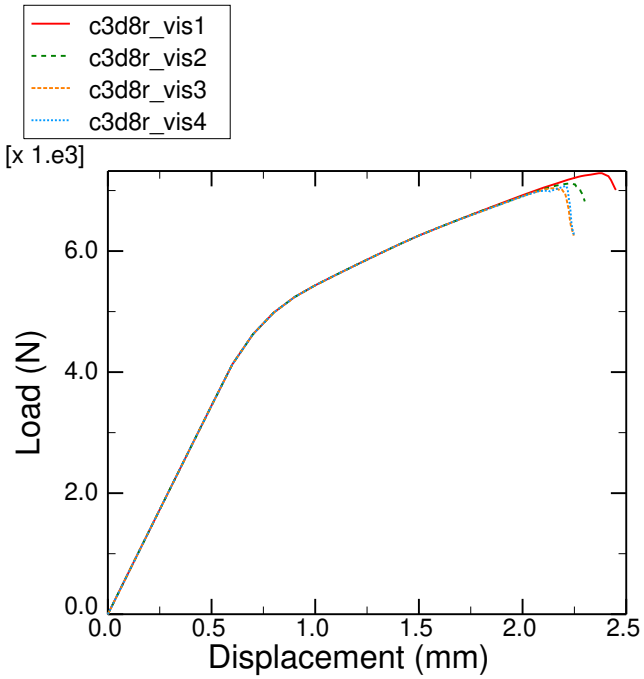


Figure 16: Load-displacement curves for different values of the viscosity parameter for the 0° loading direction (using multiscale material model).

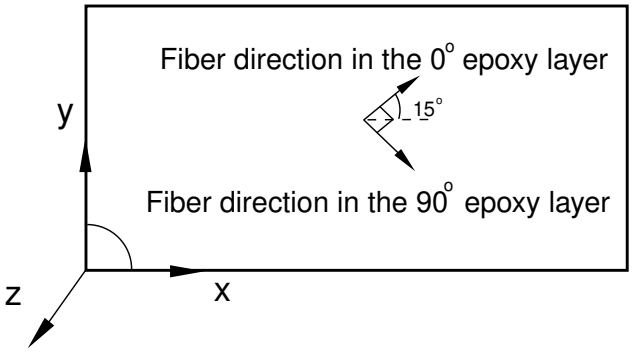


Figure 17: Local material orientations in the fiber-reinforced epoxy layers for the 15° loading direction.

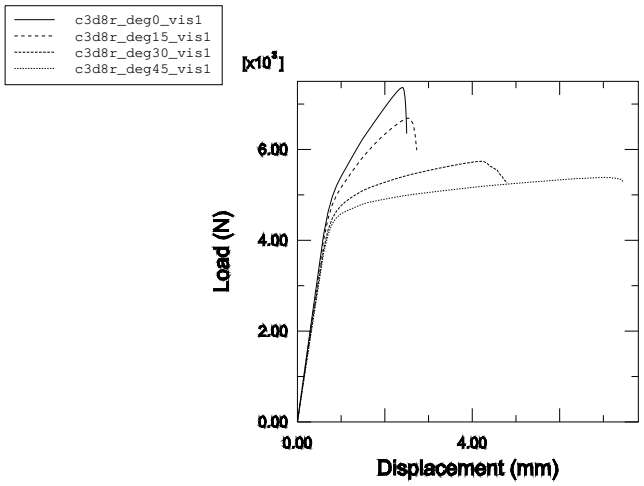


Figure 18: Load-displacement curves for different loading directions (using `UMAT` model).

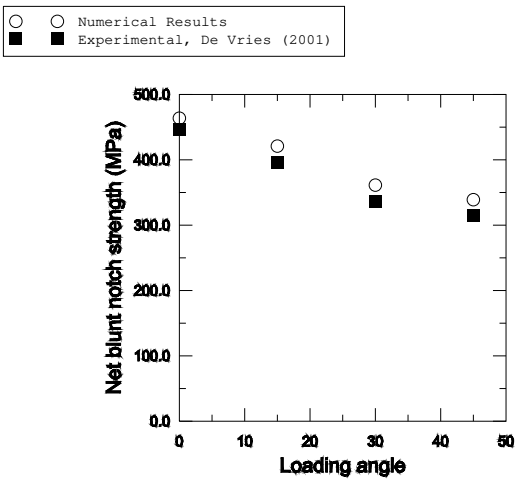


Figure 19: Calculated blunt notch strength for different loading angles in comparison with the experimental results (using `UMAT` model).

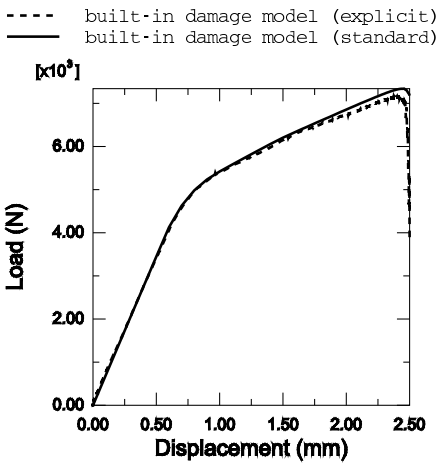


Figure 20: Load-displacement curves for the 0° loading direction: Abaqus/Explicit versus Abaqus/Standard.

Debonding behavior of a double cantilever beam

Products: Abaqus/Standard Abaqus/Explicit

References:

- *Linear Elastic Fatigue Crack Growth Analysis*
- *Low-Cycle Fatigue Analysis Using the Direct Cyclic Approach*
- *Crack Propagation Analysis*
- **COHESIVE BEHAVIOR*
- **CONTACT CLEARANCE ASSIGNMENT*
- **DEBOND*
- **DIRECT CYCLIC*
- **FATIGUE*
- **FRACTURE CRITERION*

Overview

This example examines the debonding behavior of a double cantilever beam.

The following Abaqus features are demonstrated:

- predicting debond growth in a double cantilever beam (DCB) using crack propagation analysis with the VCCT fracture criterion, the PRX-VCCT fracture criterion, cohesive elements, and surface-based cohesive behavior in Abaqus/Standard;
- demonstrating the use of contact clearance assignment between surfaces with the VCCT fracture criterion and surface-based cohesive behavior in Abaqus/Explicit to predict debond growth;
- restricting delamination growth and predicting the onset of debonding in models without a predefined crack tip using cohesive elements to supplement the VCCT fracture criterion for modeling the Z-pins and stitches that are commonly used in aerospace applications; and
- predicting progressive delamination growth at the interface in a double cantilever beam subjected to subcritical cyclic loading using the fatigue crack growth criterion based on the Paris law.

Application description

This example examines the debonding behavior of a double cantilever beam. Debond onset and growth are predicted for matched meshes in both Abaqus/Standard and Abaqus/Explicit and mismatched meshes in Abaqus/Standard. Different mesh discretizations are also used to investigate their effects on the debonding behavior. The results from Abaqus/Standard are compared with the results obtained using the VCCT-based fracture interface elements in Mabson (2003), as well as the results predicted by theory. The results predicted using VCCT, cohesive elements, and surface-based cohesive behavior in Abaqus/Standard are also compared. In addition, the results predicted by using the PRX-VCCT fracture criterion are compared with the results using the traditional VCCT fracture criterion.

The debonding behavior can also be studied by using the VCCT capability in Abaqus/Explicit. The model used in Abaqus/Explicit is constructed to achieve quasi-static behavior that allows the results obtained to be comparable with those generated using VCCT in Abaqus/Standard.

The same model is analyzed in Abaqus/Standard using the fatigue crack growth criterion to assess the fatigue life when the model is subjected to subcritical cyclic loading. The onset and delamination growth are characterized

using the Paris law, which relates the relative fracture energy release rate to the crack growth rate. The fracture energy release rate at the crack tip is calculated based on the VCCT technique. The results from Abaqus are compared with those predicted by the theory in Tada (1985).

Geometry

The double cantilever beam in this example has a span of 9.0 in (228.6 mm) with a rectangular cross-section of 1.0 in (25.4 mm) wide × 0.4 in (10.2 mm) deep, as shown in *Figure 1*.

Boundary conditions and loading

One end of the beam is fixed, and the displacements are applied at the other end, as shown in *Figure 1*. The maximum displacement is set equal to 0.16 in (4.1 mm) in the monotonic loading cases. In the fatigue crack growth analysis a cyclic displacement loading with a peak value of 0.05 in (1.3 mm) is specified.

Abaqus modeling approaches and simulation techniques

This example includes several two-dimensional models and one three-dimensional model.

Summary of analysis cases

Case 1	Prediction using matched and mismatched meshes for the two-dimensional DCB model.
Case 2	Comparison using different mesh discretizations for the two-dimensional DCB model.
Case 3	Theoretical and VCCT response prediction for the three-dimensional DCB model.
Case 4	Using cohesive elements with VCCT debond to model crack initiation.
Case 5	Using cohesive elements with VCCT debond to model Z-pins and stitches.
Case 6	Fatigue crack growth prediction with the direct cyclic approach, general fatigue crack growth approach, and simplified fatigue crack growth approach using the same model as in Case 1.
Case 7	Fatigue crack growth prediction with the direct cyclic approach using the same model as in Case 3.
Case 8	Comparison of the results obtained using VCCT, PRX-VCCT, cohesive elements, and cohesive behavior.
Case 9	Using cohesive behavior with VCCT in Abaqus/Explicit to model crack initiation. The model is identical to the model in Case 3.

Analysis Types

Static stress analyses are performed for Cases 1–8. Dynamic analysis is used for Case 9.

Case 1 Prediction using matched and mismatched meshes for the two-dimensional DCB model

This case verifies that the simulation results agree with the experimental results and compares the accuracy of the results obtained using matched and mismatched meshes.

Mesh design

Four CPE4 elements are used to model the thickness of each half of the beam. The top and bottom figures in *Figure 2* show the configurations of undeformed meshes used here with the initially bonded nodes. The model at the top has matched meshes with 90 × 4 meshes for each half of the DCB, while the model at the bottom has mismatched meshes with 90 × 4 meshes for the lower half and 85 × 4 meshes for the upper half of the DCB.

Results and discussion

The VCCT debond approach in Abaqus predicts the onset of debonding for both the matched and mismatched meshes accurately within 3% of the prediction by the fracture interface elements, as shown in *Figure 3*. *Figure 3* also shows the growth prediction calculated by the VCCT debond approach.

Case 2 Comparison using different mesh discretizations for the two-dimensional DCB model

This case shows the effect of mesh refinement on the response.

Mesh design

Four CPE4 elements are used to model the thickness of each half of the beam, the same as in Case 1. The following mesh discretizations are used along the span of the beam:

1. Matched mesh with 90 × 4 mesh for each half of the DCB.
2. Matched mesh with 180 × 4 mesh for each half of the DCB.
3. Matched mesh with 360 × 4 mesh for each half of the DCB.

The meshes are displayed in *Figure 2* (first three models).

Results and discussion

Figure 4 shows the response for different mesh discretizations of the two-dimensional model of the double cantilever beam and clearly illustrates the convergence of the response to the same solution with mesh refinement. The maximum energy release rate cutback tolerance is set to the default value of 0.2 in the input file provided for the two-dimensional model (`dcb_vcct_2d_1.inp`). However, to eliminate factors other than mesh size from the results shown in *Figure 4*, the comparisons were run with the cutback tolerance reduced to 0.1.

Case 3 Theoretical and VCCT response prediction for the three-dimensional DCB model

This case compares the three-dimensional results of Abaqus with the theoretical prediction.

Mesh design

The three-dimensional undeformed mesh and the initially bonded nodes are shown in *Figure 5*.

Results and discussion

Figure 6 shows a contour plot of the solution-dependent state variable BDSTAT for the three-dimensional model of the double cantilever beam. The debond growth is shown on the right side of the model.

The response predicted by the VCCT debond approach in Abaqus after the onset of debonding closely matches the theoretical results for the three-dimensional model of the double cantilever beam, as shown in *Figure 7*.

Case 4 Using cohesive elements with VCCT debond to model crack initiation

This case demonstrates how cohesive elements are used to initiate debond growth in a two-dimensional VCCT model.

Mesh design

A zero-thickness cohesive element (shown in *Figure 8*) is added at $x = 0$ at the interface between the two halves of the DCB. The cohesive element bonds the nodes at $x = 0$ at the contact interface. All the remaining nodes along the contact interface are initially bonded (see *Defining Initially Bonded Crack Surfaces in Abaqus/Standard*).

Materials

The cohesive element properties are chosen such that the energy required for its complete failure equals the fracture toughness of the interface. A damage initiation criterion is specified for the cohesive element.

Case 5 Using cohesive elements with VCCT debond to model Z-pins and stitches

Z-pins and stitches are additional reinforcements used at the bonded interface to retard the delamination growth. These can be modeled using zero-thickness cohesive elements at the debond interface with the appropriate material and damage initiation and damage evolution characteristics of the reinforcement. In the DCB model the nodes at the interface (shown in *Figure 9*) are initially bonded (see *Defining Initially Bonded Crack Surfaces in Abaqus/Standard*). In addition, a layer of cohesive elements is defined at the interface using the initially bonded nodes to represent the Z-pins and stitches.

Case 6 Fatigue prediction using the same model as in Case 1

This case verifies that delamination growth in a two-dimensional DCB model subjected to subcritical cyclic loading can be predicted using the fatigue crack growth criterion. Three different approaches are used in this case: the direct cyclic approach, the general fatigue crack growth approach, and the simplified fatigue crack growth approach. The simulation results are compared with the theoretical results.

Mesh design

The model has matched meshes the same as in Case 1 (model at the top of *Figure 2*).

Results and discussion

The results in terms of crack length versus the cycle number obtained using the direct cyclic approach in Abaqus are compared with the theoretical results in *Figure 10*. Reasonably good agreement is obtained. Similarly, good results are obtained when using the general and the simplified fatigue crack growth approaches.

Case 7 Fatigue prediction using the same model as in Case 3

This case verifies that delamination growth in a three-dimensional DCB model subjected to subcritical cyclic loading can be predicted using the fatigue crack growth criterion with the direct cyclic approach. The simulation results are compared with the theoretical results.

Mesh design

The mesh design is the same as in Case 3 (*Figure 5*).

Results and discussion

The results in terms of crack length versus the cycle number obtained using the fatigue crack growth criterion in Abaqus are compared with the theoretical results in *Figure 11*. Reasonably good agreement is obtained.

Case 8 Comparison of the results obtained using VCCT, PRX-VCCT, cohesive elements, and cohesive behavior

This case compares the results obtained using VCCT, PRX-VCCT, cohesive elements, and surface-based cohesive behavior for both the two- and three-dimensional DCB models.

Mesh design

For the two-dimensional model, four CPE4 elements are used to model the thickness of each half of the beam. The model at the top of *Figure 2* shows the configurations of undeformed meshes used here with the initially bonded nodes. The model has matched meshes with 90 × 4 meshes for each half of the DCB.

The three-dimensional undeformed mesh and the initially bonded nodes are shown in *Figure 5*.

Results and discussion

Figure 12 compares the results obtained using VCCT, cohesive elements, and surface-based cohesive behavior for the two-dimensional DCB model. *Figure 13* shows a similar comparison for the three-dimensional DCB model. All of the methods predicted nearly the same debond growth after debond onset. But the responses predicted using cohesive elements or surface-based cohesive behavior are not strictly linear before debond onset, as compared with the linear behavior predicted by the VCCT debond approach. This is not surprising since the secondary nodes for the model using the VCCT debond approach are released one after another, while all the secondary nodes for the models using cohesive elements or surface-based cohesive behavior debond simultaneously.

Similar comparisons have been made between VCCT and PRX-VCCT. *Figure 14* shows the results for the two-dimensional model, while *Figure 15* shows the results for the three-dimensional model. The loading curves using PRX-VCCT are much smoother than the ones using VCCT.

Case 9 Using VCCT in Abaqus/Explicit to model crack initiation

This case verifies that delamination growth in a three-dimensional DCB model can be predicted using the VCCT capability in Abaqus/Explicit. The simulation results are compared with Abaqus/StandardVCCT results.

Mesh design

The mesh design is the same as in Case 3 (*Figure 5*).

Results and discussion

The debond states obtained from the Abaqus/Explicit and Abaqus/Standard analyses match very well, as depicted in *Figure 16*. The measured reaction forces from the Abaqus/Explicit analysis compare extremely well with those from Abaqus/Standard, as shown in *Figure 17*. The debonding time, displacement, and energy release rate are highly consistent between both analyses.

Input files

dcb_vcct_2d_1.inp	Two-dimensional model with matched mesh; 90 × 4 mesh for each half of the DCB using VCCT.

dcb_vcct_2d_prx.inp	Two-dimensional model with matched mesh; 90 × 4 mesh for each half of the DCB using PRX-VCCT.
dcb_mismatch_vcct_2d_1.inp	Two-dimensional model with mismatched meshes.
dcb_coh_init_vcct_2d_1.inp	Two-dimensional model with 4-node two-dimensional cohesive elements (COH2D4) to predict the onset of debonding.
dcb_coh_stitch_vcct_2d_1.inp	Two-dimensional model with 4-node two-dimensional cohesive elements (COH2D4) representing Z-pins and stitches.
dcb_vcct_3d_1.inp	Three-dimensional model with matched meshes using VCCT.
dcb_vcct_3d_prx.inp	Three-dimensional model with matched meshes using PRX-VCCT.
dcb_vcct_fatigue_2d.inp	Same as dcb_vcct_2d_1.inp but subjected to cyclic displacement loading using the direct cyclic approach.
dcb_vcct_fatigue_2d_2.inp	Same as dcb_vcct_2d_1.inp but subjected to cyclic displacement loading using the general fatigue crack growth approach.
dcb_vcct_fatigue_2d_2_simp.inp	Same as dcb_vcct_2d_1.inp but subjected to cyclic displacement loading using the simplified fatigue crack growth approach.
dcb_vcct_fatigue_3d.inp	Same as dcb_vcct_3d_1.inp but subjected to cyclic displacement loading.
dcb_cohelm_2d.inp	Two-dimensional model using cohesive elements with matched mesh; 90 × 4 mesh for each half of the DCB.
dcb_surcoh_2d.inp	Two-dimensional model using surface-based cohesive behavior with matched mesh; 90 × 4 mesh for each half of the DCB.
dcb_cohelm_3d.inp	Three-dimensional model using cohesive elements with matched meshes.
dcb_surcoh_3d.inp	Three-dimensional model using surface-based cohesive behavior with matched meshes.
dcb_vcct_xpl_3d.inp	Abaqus/Explicit three-dimensional model with matched meshes.

References

- Mabson, G, “Fracture Interface Elements,” 46th PMC General Session of Mil-17 (Composites Materials Handbook) Organization, Charleston, SC, 2003.
- Tada, H, “The Stress Analysis of Cracks Handbook,” Paris Productions Incorporated, 1985.

Figures

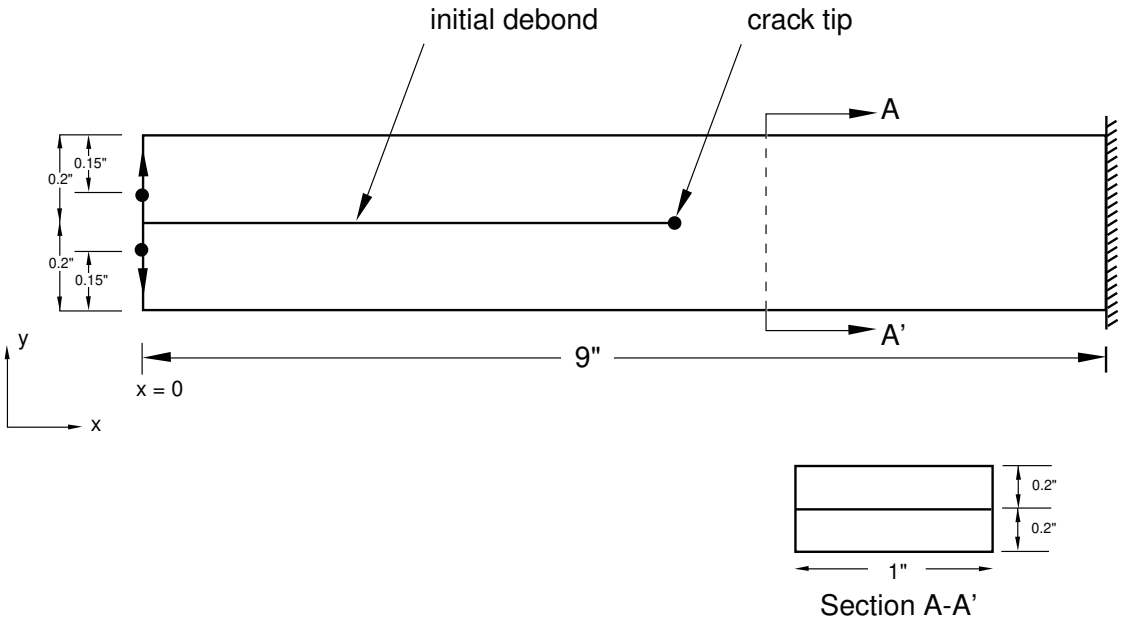


Figure 1: The double cantilever beam (DCB) model.

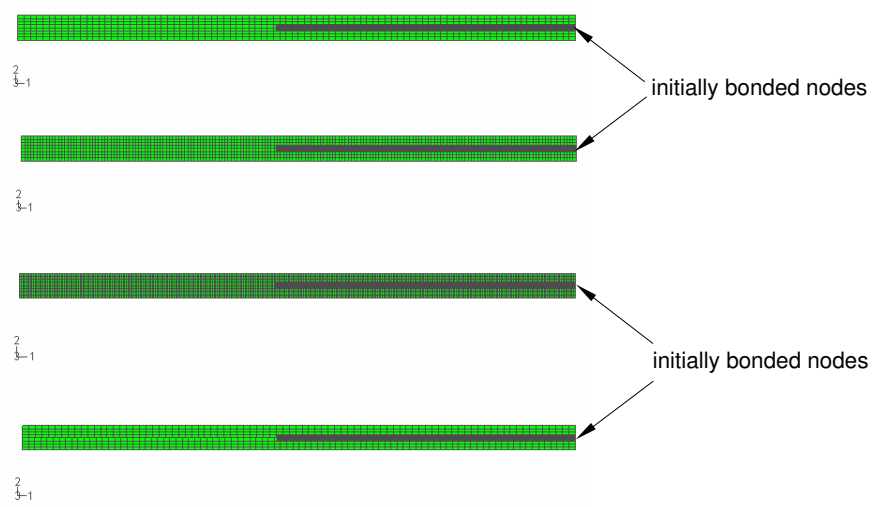


Figure 2: Mesh configurations for the two-dimensional DCB model.

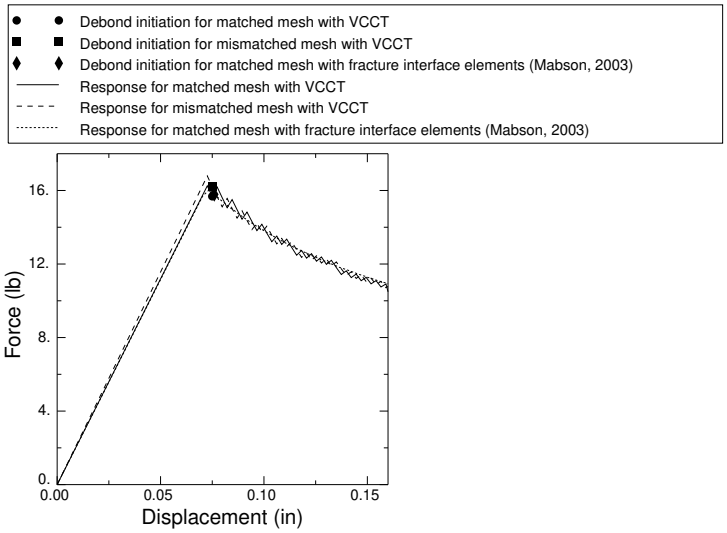


Figure 3: Debond onset and growth prediction for matched and mismatched meshes for the two-dimensional DCB model.

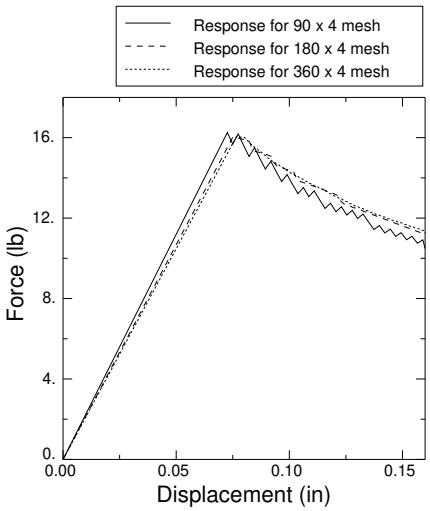


Figure 4: Response for different mesh discretizations of the two-dimensional DCB model.

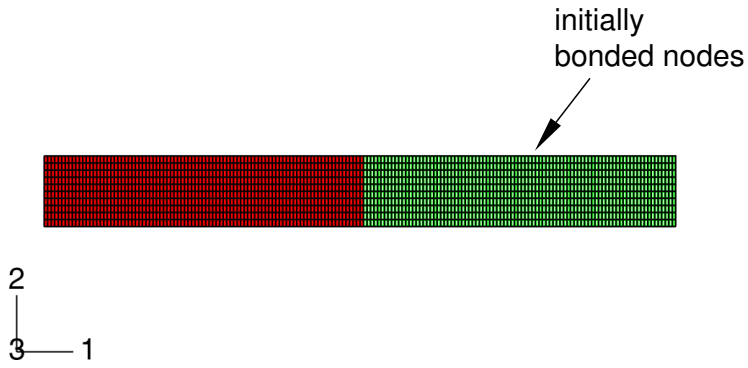


Figure 5: Mesh configuration for the three-dimensional DCB model.

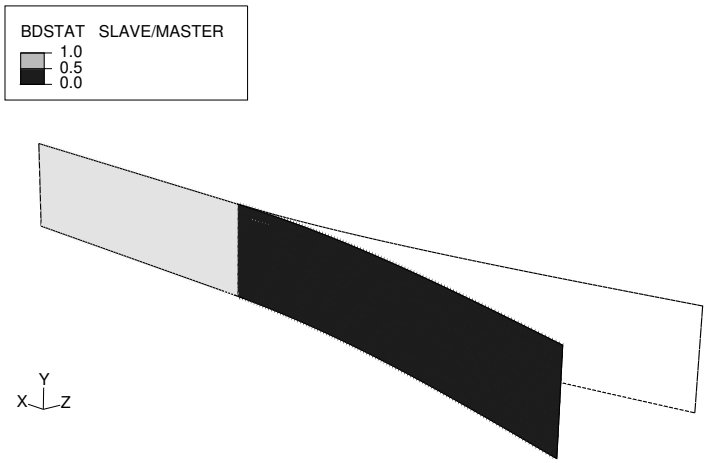


Figure 6: Debond growth for the three-dimensional DCB model.

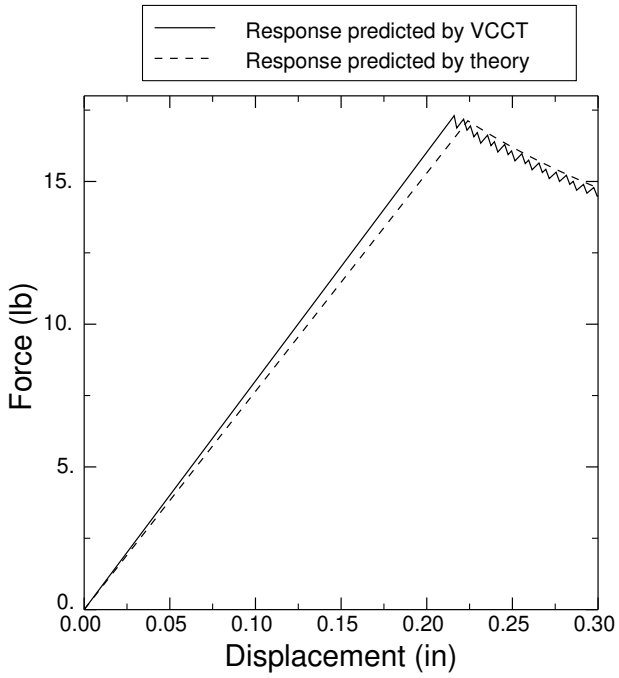


Figure 7: Theoretical and VCCT in Abaqus response prediction for the three-dimensional DCB model.



Figure 8: Using zero-thickness cohesive elements to model crack initiation.



Figure 9: Modeling Z-pins and stitches using cohesive elements.

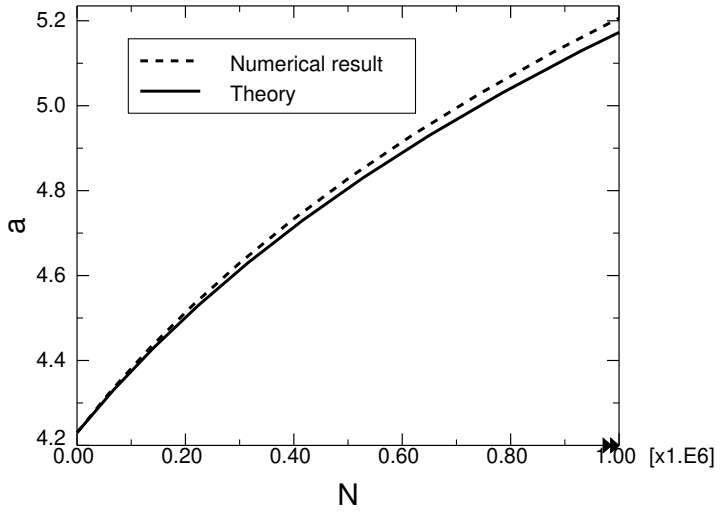


Figure 10: Crack length versus cycle number for the two-dimensional DCB model.

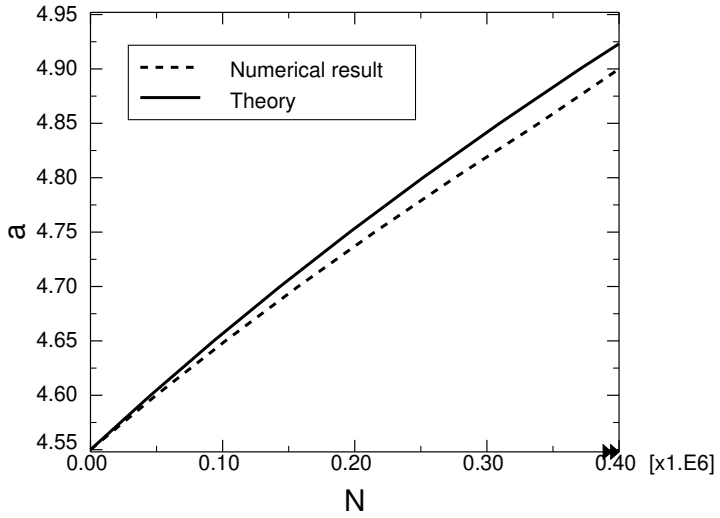


Figure 11: Crack length versus cycle number for the three-dimensional DCB model.

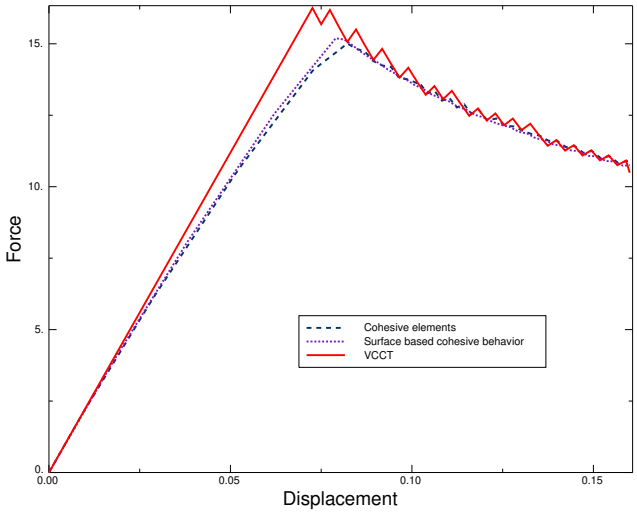


Figure 12: Comparison of the results using VCCT, cohesive elements, and surface-based cohesive behavior for the two-dimensional DCB model.

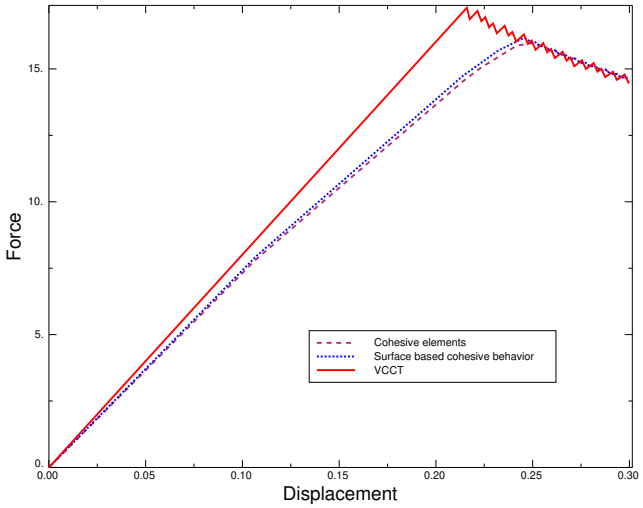


Figure 13: Comparison of the results using VCCT, cohesive elements, and surface-based cohesive behavior for the three-dimensional DCB model.

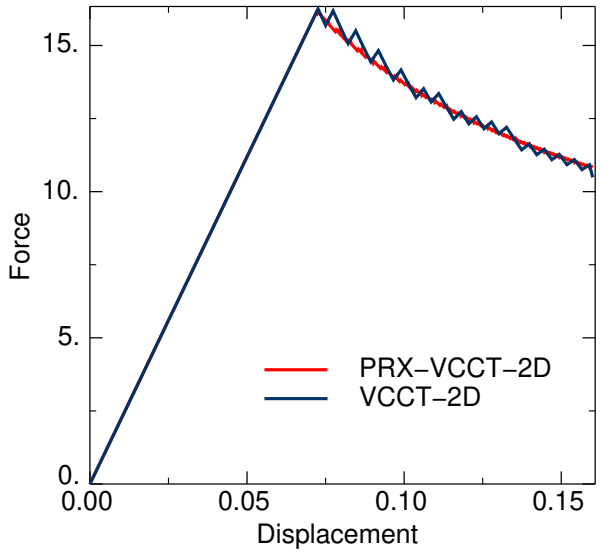


Figure 14: Comparison of the results using VCCT and PRX-VCCT for the two-dimensional DCB model.

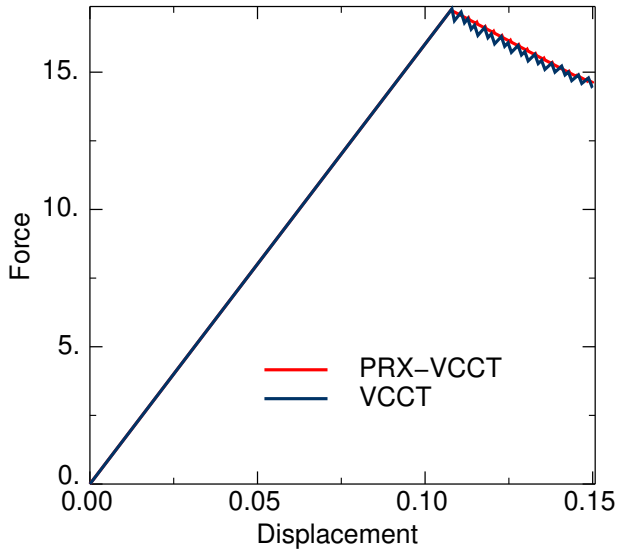


Figure 15: Comparison of the results using VCCT and PRX-VCCT for the three-dimensional DCB model.

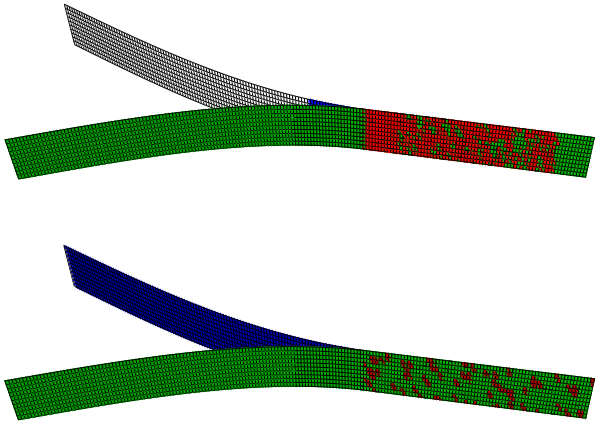


Figure 16: Debond state comparison between Abaqus/Explicit (top) and Abaqus/Standard (bottom).

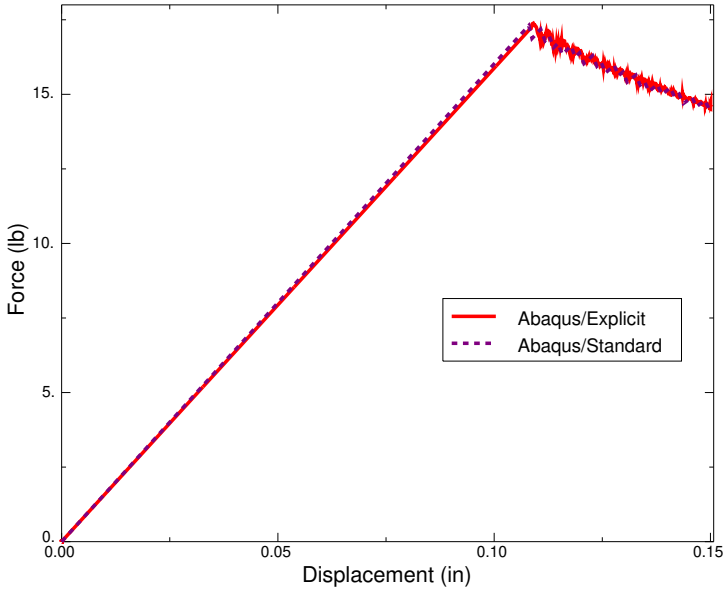


Figure 17: Comparison of the results between Abaqus/Explicit and Abaqus/Standard.

Debonding behavior of a single leg bending specimen

Products: Abaqus/Standard Abaqus/Explicit

References:

- *Linear Elastic Fatigue Crack Growth Analysis*
- *Low-Cycle Fatigue Analysis Using the Direct Cyclic Approach*
- *Crack Propagation Analysis*
- **COHESIVE BEHAVIOR*
- **CONTACT CLEARANCE ASSIGNMENT*
- **DEBOND*
- **DIRECT CYCLIC*
- **FATIGUE*
- **FRACTURE CRITERION*
- **NODAL ENERGY RATE*

Overview

This example examines the debonding behavior of a single leg bending specimen.

The following Abaqus features are demonstrated:

- predicting debond growth in a single leg bending (SLB) specimen using crack propagation analysis with the VCCT fracture criterion in Abaqus/Standard;
- predicting debond growth using the VCCT fracture criterion and surface-based cohesive behavior in Abaqus/Explicit; and
- predicting progressive delamination growth at the interface in this specimen subjected to subcritical cyclic loading using the traditional fatigue crack growth criterion and the PRX-Fatigue crack growth criterion, both based on the Paris law.

Application description

The example examines the debonding behavior of a single leg bending specimen and compares the simulation results with the results of the analysis performed using the VCCT-based fracture interface elements discussed in Mabson (2003). Damping is also used in a two-dimensional model to demonstrate how it can stabilize the crack growth.

The same model is analyzed using the fatigue crack growth criterion to assess the fatigue life when the model is subjected to subcritical cyclic loading. The onset and delamination growth are characterized using the Paris law, which relates the relative fracture energy release rate to the crack growth rate. The fracture energy release rate at the crack tip is calculated based on the VCCT technique. The results from Abaqus are compared with those predicted by the theory in Davidson (1995).

Geometry

The geometries of the two-dimensional and three-dimensional single leg bending specimens with their corresponding initial crack locations are shown in *Figure 1* and *Figure 2*, respectively.

Boundary conditions and loading

A displacement is applied on the top beam at the location shown in *Figure 1* for the two-dimensional model and in *Figure 2* for the three-dimensional model. The displacement results in a mixture of opening (mode I) and shearing (mode II) modes. The maximum displacements are set equal to 0.32 in (8.13 mm) in the two-dimensional model and 0.15 in (3.81 mm) in the three-dimensional model for the monotonic loading cases. In the fatigue crack growth analyses, cyclic displacement loadings with peak values of 0.12 in (3.05 mm) in the two-dimensional model and 0.035 in (0.89 mm) in the three-dimensional model are specified.

Abaqus modeling approaches and simulation techniques

This example includes one two-dimensional model and one three-dimensional model.

Summary of analysis cases

Case 1	Two-dimensional single leg bending model.
Case 2	Three-dimensional single leg bending model.
Case 3	Fatigue prediction with the direct cyclic approach, the linear elastic fatigue crack growth approach, and the PRX-Fatigue approach using the same model as in Case 1.
Case 4	Fatigue prediction with the direct cyclic approach, the linear elastic fatigue crack growth approach, and the PRX-Fatigue approach using the same model as in Case 2.
Case 5	Abaqus/Explicit with VCCT using the same model as in Case 2.

Analysis types

Static analyses are performed for Cases 1–4. Dynamic analysis is used for Case 5.

Case 1 Two-dimensional single leg bending model

This case compares the AbaqusVCCT results for the two-dimensional model with the results of Mabson (2003). The damping effects are also examined.

Mesh design

The model uses a finite element mesh of 4-node bilinear plane strain quadrilateral, incompatible mode elements (CPE4I) for both the long and short beams.

Results and discussion

Figure 3 shows the deformed configuration of the two-dimensional model. *Figure 4* shows a contour plot of bond status variable BDSTAT that illustrates the debonding growth in a two-dimensional model. The region of debonding is shown on the right side of the model. *Figure 5* compares the results from the two-dimensional analysis with the results of the analysis performed using the VCCT-based fracture interface elements discussed in Mabson (2003).

The damping effect to a two-dimensional simulation is also examined in this case by adding damping at the debond interface (see *Automatic Stabilization of Rigid Body Motions in Contact Problems*). The damping stabilizes the crack growth and allows the solution to converge. It is important to assess how much energy has been used for numerical damping by comparing the stabilization energy to the strain energy of the model. *Figure 6* shows the comparison between static stabilization energy (ALLSD) and the strain energy of the model (ALLSE). The comparison indicates that the maximum static stabilization energy is less than 3% of the maximum strain energy of the model during the analysis. This value is reasonable and indicates that the solution has not been affected by the addition of artificial numerical damping.

Case 2 Three-dimensional single leg bending model

This case compares the AbaqusVCCT results for the three-dimensional model with a theoretical prediction.

Mesh design

The model uses fully integrated first-order shell elements (S4) for both the long and short beams.

Results and discussion

Figure 7 shows a contour plot of bond status variable BDSTAT for the three-dimensional model. *Figure 8* shows a comparison between the results from the three-dimensional analysis and the results presented in Mabson (2003).

Case 3 Fatigue prediction using the same model as in Case 1

This case verifies that delamination growth in a two-dimensional single leg bending model subjected to subcritical cyclic loading can be predicted using the fatigue crack growth criterion. The direct cyclic approach, the linear elastic fatigue crack growth approach, and the PRX-Fatigue approach are used for this case study

Mesh design

The mesh design is the same as in Case 1.

Results and discussion

The simulation results for the direct cyclic approach are compared with the theoretical results in *Figure 9*, which shows the crack length versus the number of cycles. Reasonably good agreement is obtained between the simulation and the theoretical results. The corresponding simulation results using the linear elastic fatigue crack growth and the PRX-Fatigue approaches are close to the results obtained using the direct cyclic approach and are not shown here.

Case 4 Fatigue prediction using the same model as in Case 2

This case verifies that delamination growth in a three-dimensional single leg bending model subjected to subcritical cyclic loading can be predicted using the fatigue crack growth criterion with the debonding force removed immediately or ramped down gradually. The direct cyclic approach, the linear elastic fatigue crack growth approach, and the PRX-Fatigue approach are used in this case study.

Mesh design

The mesh design is the same as in Case 2.

Results and discussion

The simulation results using the direct cyclic approach are compared with the theoretical results in *Figure 10*, which shows the crack length versus the number of cycles. Reasonably good agreement is obtained between the simulation and the theoretical results. The corresponding simulation results using the linear elastic fatigue crack growth and the PRX-Fatigue approaches are close to the results obtained using the direct cyclic approach and are not shown here.

Case 5 Using VCCT in Abaqus/Explicit to model crack initiation

This case verifies that delamination growth can be predicted using Abaqus/Explicit. The simulation results are compared with results obtained from Abaqus/Standard. To reduce inertia effects and allow a better comparison between Abaqus/Standard and Abaqus/Explicit results, the material density was lowered and the loading was ramped on with a smooth step definition.

Mesh design

The mesh design is the same as in Case 2.

Results and discussion

The results obtained using Abaqus/Explicit with VCCT show reasonably good agreement with those obtained from Abaqus/Standard, as depicted in *Figure 11* and *Figure 12*. Due to the thin layer of elements and the specified boundary conditions in the model, inertia effects are clearly observed in the measured reaction forces. However, the peak forces at debond onset, debond time, and other VCCT output quantities are consistent between the two analyses. The reaction forces obtained in Abaqus/Explicit were filtered with a Butterworth filter with a cutoff frequency of 500 Hz to reduce high-frequency oscillations from the response curve.

Input files

slb_vcct_2d_1.inp	Two-dimensional model of the SLB specimen.
slb_vcct_3d_1.inp	Three-dimensional model of the SLB specimen.
slb_vcct_fatigue_2d.inp	Same as `slb_vcct_2d_1.inp` but subjected to cyclic displacement loading using the direct cyclic approach.
slb_vcct_fatigue_2d_2_simp.inp	Same as `slb_vcct_2d_1.inp` but subjected to cyclic displacement loading using the simplified fatigue crack growth approach.
slb_vcct_fatigue_2d_2_simp_prx.inp	Same as `slb_vcct_2d_1.inp` but subjected to cyclic displacement loading using the simplified PRX-Fatigue approach.
slb_vcct_fatigue_2d_3_simp_ tabular.inp	Same as `slb_vcct_2d_1.inp` but subjected to cyclic displacement loading using the simplified fatigue crack growth approach with the tabular form.
slb_vcct_fatigue_2d_3_tabular_prx.inp	Same as `slb_vcct_2d_1.inp` but subjected to cyclic displacement loading using the general PRX-Fatigue approach with the tabular form.
slb_vcct_fatigue_2d_3_simp_user.inp	Same as `slb_vcct_2d_1.inp` but subjected to cyclic displacement loading using the simplified fatigue crack growth approach with the user-defined subroutine.
slb_vcct_fatigue_3d_2_simp_prx.inp	Same as `slb_vcct_3d_1.inp` but subjected to cyclic displacement loading using the simplified PRX-Fatigue approach.
slb_vcct_fatigue_2d_3_tabular.inp	Same as `slb_vcct_2d_1.inp` but subjected to cyclic displacement loading using the general fatigue crack growth approach with the tabular form.
slb_vcct_fatigue_3d_3_tabular_prx.inp	Same as `slb_vcct_3d_1.inp` but subjected to cyclic displacement loading using the general PRX-Fatigue approach with the tabular form.
slb_vcct_fatigue_2d_3_user.inp	Same as `slb_vcct_2d_1.inp` but subjected to cyclic displacement loading using the general fatigue crack growth approach with the user-defined subroutine.
slb_vcct_fatigue_3d.inp	Same as `slb_vcct_3d_1.inp` but subjected to cyclic displacement loading using the direct cyclic approach.

slb_vcct_fatigue_3d_2_simp.inp	Same as `slb_vcct_3d_1.inp` but subjected to cyclic displacement loading using the simplified fatigue crack growth approach.
slb_vcct_fatigue_3d_3_simp_ tabular.inp	Same as `slb_vcct_3d_1.inp` but subjected to cyclic displacement loading using the simplified fatigue crack growth approach with the tabular form.
slb_vcct_fatigue_3d_3_simp_user.inp	Same as `slb_vcct_3d_1.inp` but subjected to cyclic displacement loading using the simplified fatigue crack growth approach with the user-defined subroutine.
slb_vcct_fatigue_3d_3_tabular.inp	Same as `slb_vcct_3d_1.inp` but subjected to cyclic displacement loading using the general fatigue crack growth approach with the tabular form.
slb_vcct_fatigue_3d_3_user.inp	Same as `slb_vcct_3d_1.inp` but subjected to cyclic displacement loading using the general fatigue crack growth approach with the user-defined subroutine.
slb_vcct_fatigue_3d_3_simp_tabular_ ramp.inp	Same as `slb_vcct_fatigue_3d_3_simp_tabular.inp` but with the debonding force ramped down gradually.
slb_vcct_fatigue_3d_3_simp_user_ ramp.inp	Same as `slb_vcct_fatigue_3d_3_simp_user.inp` but with the debonding force ramped down gradually.
slb_vcct_fatigue_3d_3_tabular_ ramp.inp	Same as `slb_vcct_fatigue_3d_3_tabular.inp` but with the debonding force ramped down gradually.
slb_vcct_fatigue_3d_3_user_ramp.inp	Same as `slb_vcct_fatigue_3d_3_user.inp` but with the debonding force ramped down gradually.
slb_vcct_xpl_3d.inp	Three-dimensional model of the SLB specimen using Abaqus/Explicit with VCCT.

References

- Mabson, G, “Fracture Interface Elements,” 46th PMC General Session of Mil-17 (Composites Materials Handbook) Organization, Charleston, SC, 2003.
- Davidson, B. D., R. Kruger, and M. Konig, “Three-Dimensional Analysis of Center-Delaminated Unidirectional and Multidirectional Single-Leg Bending Specimens,” Composites Science and Technology, vol. 54, pp. 385–394, 1995.

Figures

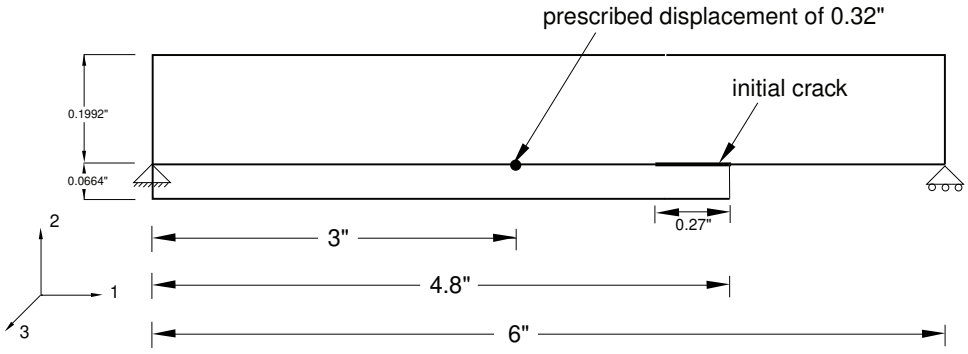


Figure 1: The two-dimensional single leg bending (SLB) specimen.

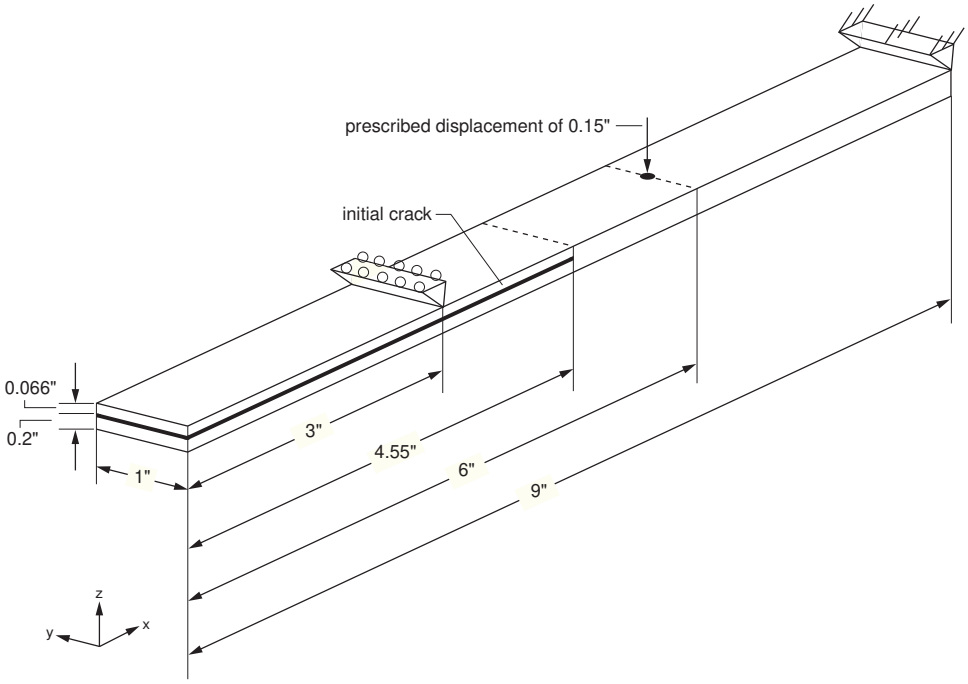


Figure 2: The three-dimensional single leg bending (SLB) specimen.

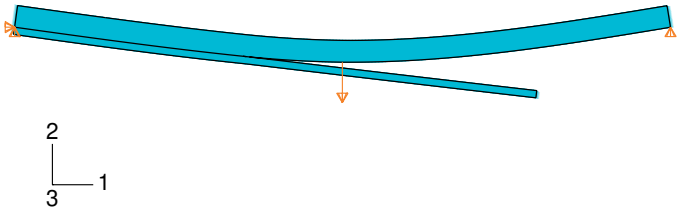


Figure 3: The deformed shape of the two-dimensional model showing boundary conditions and prescribed displacements.

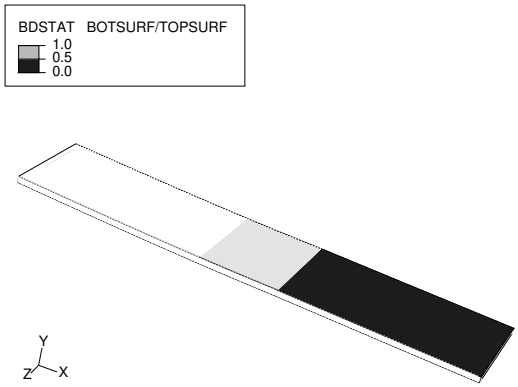


Figure 4: The prediction of debonding growth for the two-dimensional SLB model.

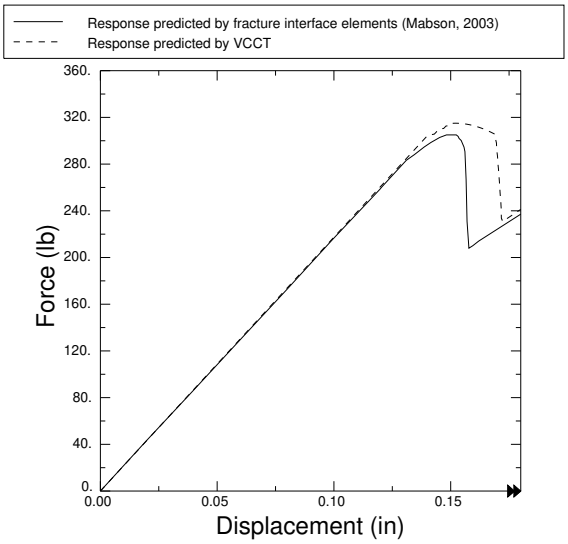


Figure 5: Response prediction for the two-dimensional SLB model.

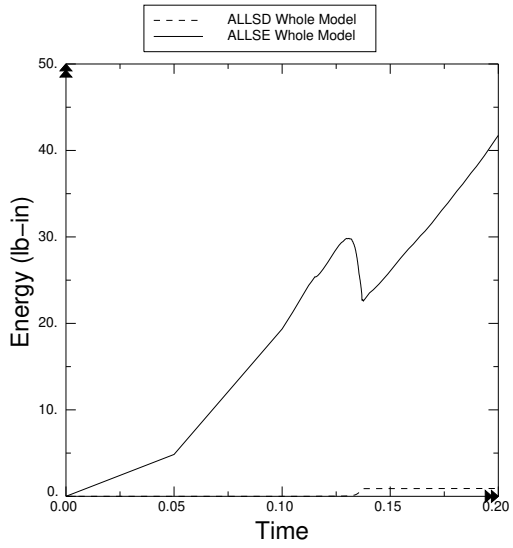


Figure 6: A comparison between ALLSD and ALLSE for the two-dimensional model.

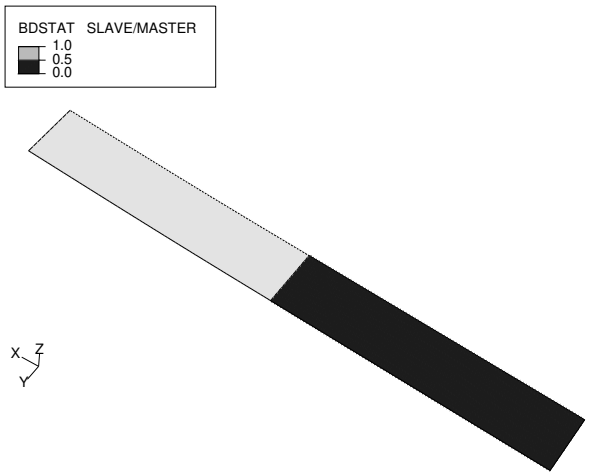


Figure 7: The prediction of debonding growth for the three-dimensional SLB model.

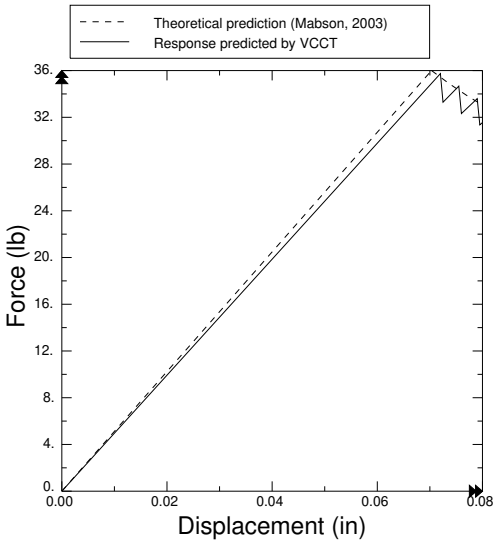


Figure 8: Response prediction for the three-dimensional SLB model.

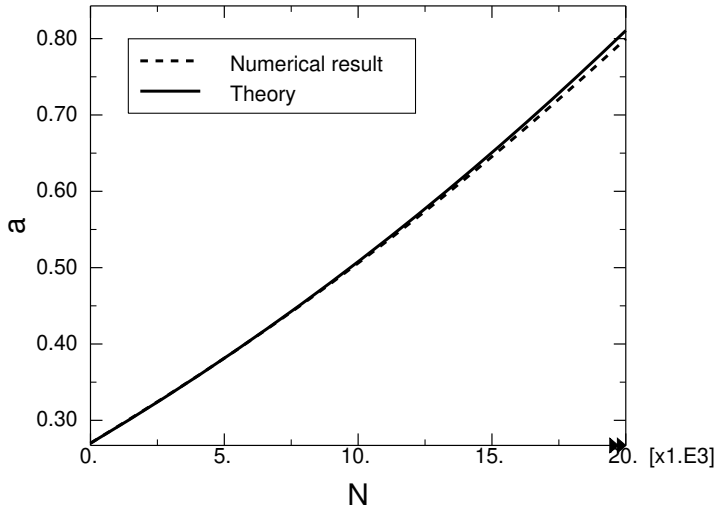


Figure 9: Crack length versus cycle number for the two-dimensional SLB model.

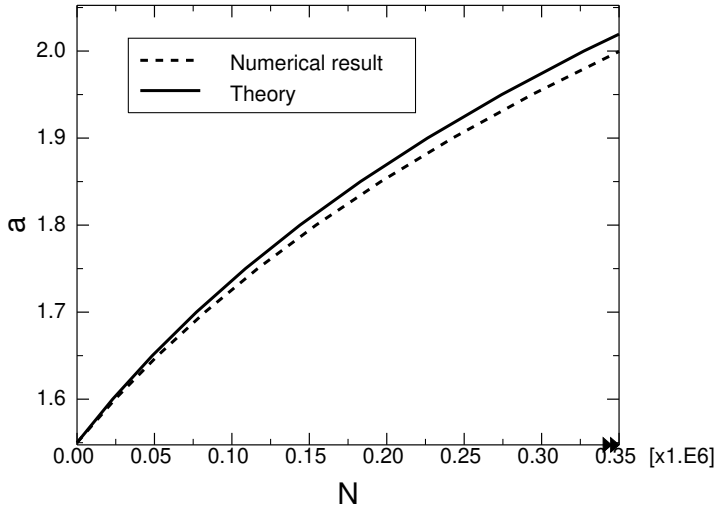


Figure 10: Crack length versus cycle number for the three-dimensional SLB model.

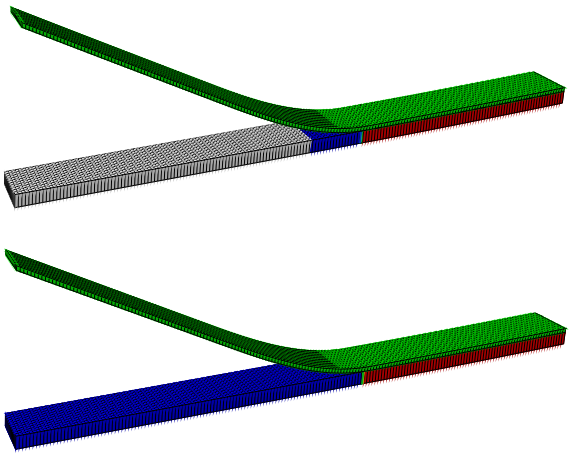


Figure 11: Debond state comparison between Abaqus/Explicit (top) and Abaqus/Standard (bottom).

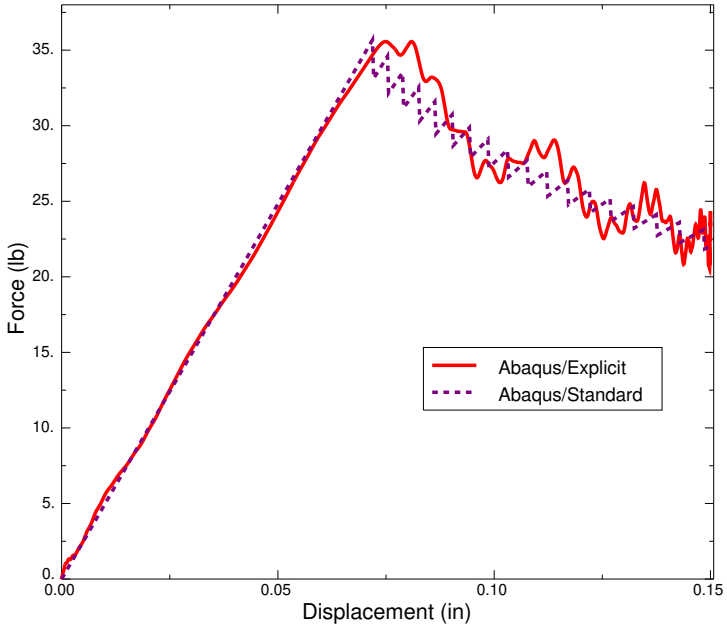


Figure 12: Comparison of the results between Abaqus/Explicit and Abaqus/Standard.

Postbuckling and growth of delaminations in composite panels

Products: Abaqus/Standard Abaqus/Explicit

References:

- *Crack Propagation Analysis*
- **COHESIVE BEHAVIOR*
- **CONTACT CLEARANCE ASSIGNMENT*
- **DEBOND*
- **FRACTURE CRITERION*

Overview

This example illustrates the application of the use of the VCCT and the PRX-VCCT fracture criteria in Abaqus/Standard and the VCCT fracture criterion in Abaqus/Explicit to predict the postbuckling response, onset, and growth of delaminations in laminated composite panels.

Application description

Delaminations are a primary failure mode for laminated composite materials. The delamination growth is more prominent under compressive loading since it results in buckling of a sublaminate leading to the delamination growth. The particular problem considered here is described in Reeder (2002). The results from the VCCT debond approach in Abaqus are compared to the experimental results. In addition, the results using the PRX-VCCT approach are compared to the ones using the VCCT approach.

Geometry

A flat 9.0 in (228.6 mm) × 4.5 in (114.3 mm) composite panel with a centrally located 2.5 in (63.5 mm) diameter delamination is studied in this example, as shown in *Figure 1*.

Materials

The panel is made of a AS4/3501-6 graphite/epoxy composite material system for which the typical lamina properties are given in *Table 1*. The laminate stacking sequence for the panel is $[(\pm 45/90/0)_2/\pm 60/\pm 15]_S$. The critical fracture toughness for Modes I, II, and III at the delamination interface are also given in *Table 1*.

Boundary conditions and loading

The panel is subjected to compressive loading along its long axis. The overall dimensions of the model with boundary conditions and loading can be seen in *Figure 1*.

Abaqus modeling approaches and simulation techniques

The delamination is placed at the interface between the 5th and 6th ply (between 45° and –45°). The delamination region is modeled using two superimposed shell elements with contact constraints defined to prevent penetration of elements. The Benzeggagh-Kenane mixed-mode failure criterion (Benzeggagh and Kenane, 1996) is used to determine the growth of delamination based on the strain energy release rate computed using VCCT.

Analysis Types

Both static and dynamic analyses are performed.

Mesh design

The finite element model is created with fully integrated first-order shell elements (S4). The finite element mesh for the model is shown in *Figure 2* with the circular delamination at the center.

Loads

The loading consists of a prescribed displacement of 0.03 in (0.76 mm) at the top edge of the panel.

Solution controls

To achieve a stable delamination growth in Abaqus/Standard, a small amount of damping is specified for the interface where delamination growth occurs (see *Automatic Stabilization of Rigid Body Motions in Contact Problems*). There are no solution controls specified in the Abaqus/Explicit analysis.

Results and discussion

The final deformed configuration of the composite panel obtained from Abaqus/Standard is shown in *Figure 3*. The postbuckling in the panel section cut along the long axis is illustrated in *Figure 4*. A contour plot of the bond status variable BDSTAT obtained from Abaqus/Standard indicating the growth of delamination is shown in *Figure 5*. The load-strain predictions are compared with the experimental data presented by Reeder (2002) in *Figure 6*. The predictions from VCCT are in agreement with the experimental results. The onset of delamination growth predicted by the VCCT debond approach in Abaqus is within 10% of the experimental data. The energy dissipated to stabilize the delamination growth is less than 4% of the total strain energy, as shown in *Figure 7*. In addition, the load-strain prediction using PRX-VCCT is compared to the one using VCCT in *Figure 8*. Using PRX-VCCT produced a clearer delamination initiation point than using VCCT.

The deformed configurations obtained from Abaqus/Explicit are shown in *Figure 9* and *Figure 10*. Delamination growth obtained from the Abaqus/Explicit analysis is shown in *Figure 11*. The force-displacement responses differ slightly from the Abaqus/Standard results, as shown in *Figure 12*, but show reasonable agreement.

Input files

nasa_postbuckle_vcct_1.inp	Postbuckling analysis of a composite plate using Abaqus/Standard.
nasa_postbuckle_xpl_vcct.inp	Postbuckling analysis of a composite plate using Abaqus/Explicit.
nasa_postbuckle_prx-vcct.inp	Postbuckling analysis of a composite plate using the PRX-VCCT fracture criterion in Abaqus/Standard.

References

- Benzeggagh, M., and M. Kenane, “Measurement of Mixed-Mode Delamination Fracture Toughness of Unidirectional Glass/Epoxy Composites with Mixed-Mode Bending Apparatus,” Composite Science and Technology, vol. 56, p. 439, 1996.
- Reeder, J., S. Kyongchan, P. B. Chunchu, and D. R. Ambur, “Postbuckling and Growth of Delaminations in Composite Plates Subjected to Axial Compression,” 43rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Denver, Colorado, vol. 1746, p. 10, 2002.

Tables

Table 1: Properties for AS4/3501-6 graphite/epoxy material.

Property	Value
E_1	18.500×10^6 lb/in^2 (127.554 kN/mm^2)
E_2	1.640×10^6 lb/in^2 (11.307 kN/mm^2)
G_{12}	0.871×10^6 lb/in^2 (6.005 kN/mm^2)
G_{13}	0.871×10^6 lb/in^2 (6.005 kN/mm^2)
G_{23}	0.522×10^6 lb/in^2 (3.599 kN/mm^2)
ν_{12}	0.30
G_{IC}	0.46863 lb/in (0.08207 N/mm)
G_{IIC}	3.171825 lb/in (0.55546 N/mm)
G_{IIIC}	3.171825 lb/in (0.55546 N/mm)

Figures

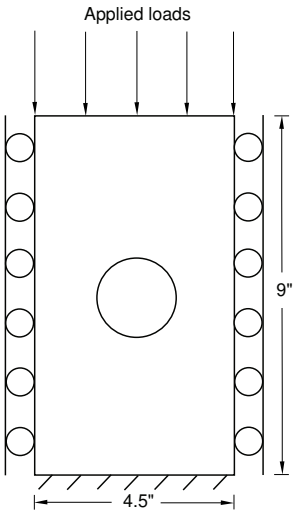


Figure 1: The flat composite panel.

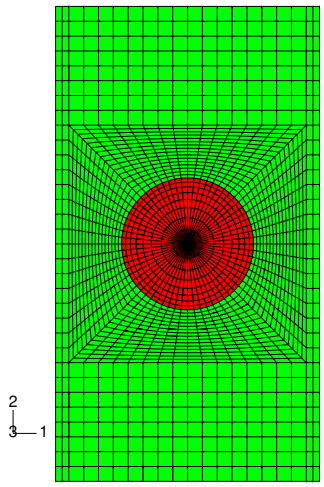


Figure 2: The meshed NASA panel model.

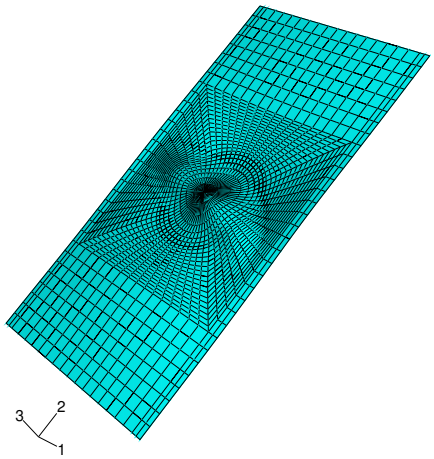


Figure 3: The final deformed configuration (Abaqus/Standard).

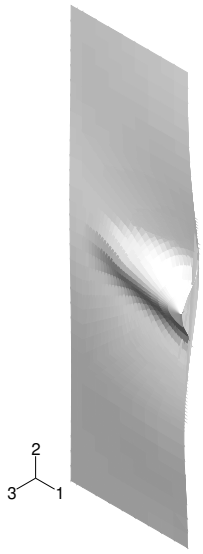


Figure 4: The postbuckling in the panel section (Abaqus/Standard).

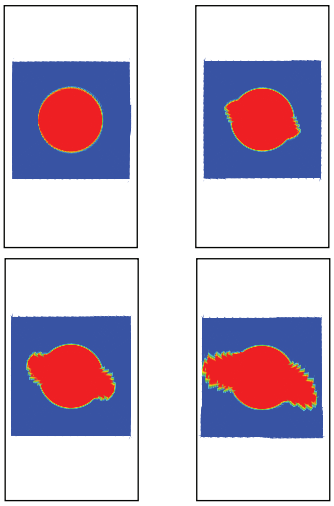


Figure 5: The growth of delamination (Abaqus/Standard).

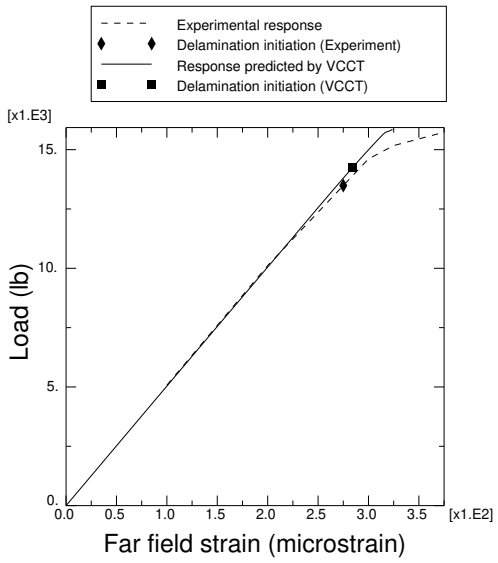


Figure 6: The load-strain predictions compared with experimental data.

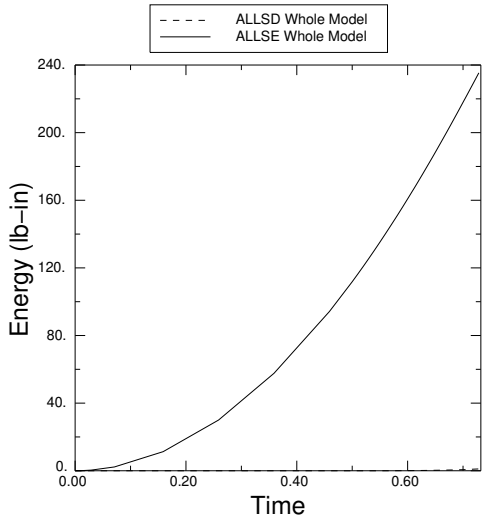


Figure 7: The energy dissipated to stabilize the delamination growth.

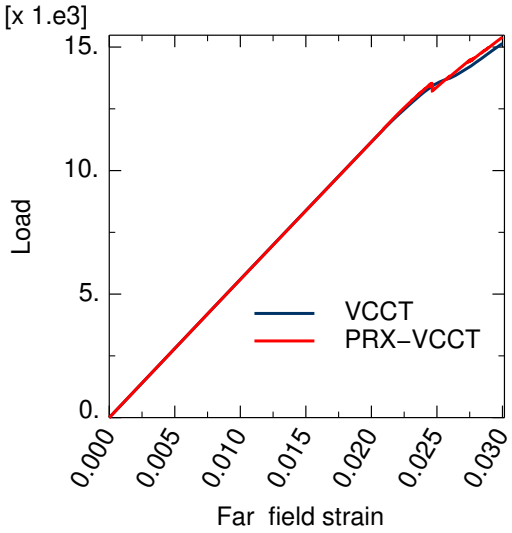


Figure 8: Comparison of the load-strain predictions using PRX-VCCT and VCCT.

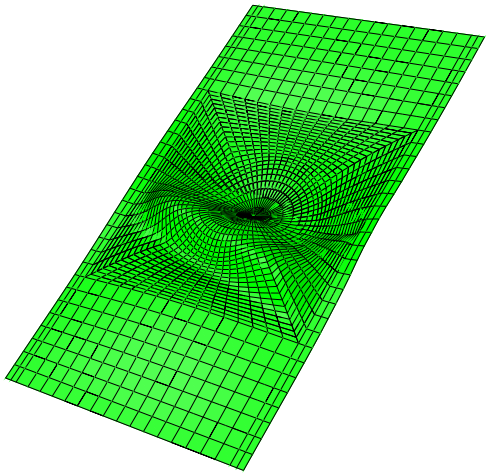


Figure 9: The final deformed configuration (Abaqus/Explicit).

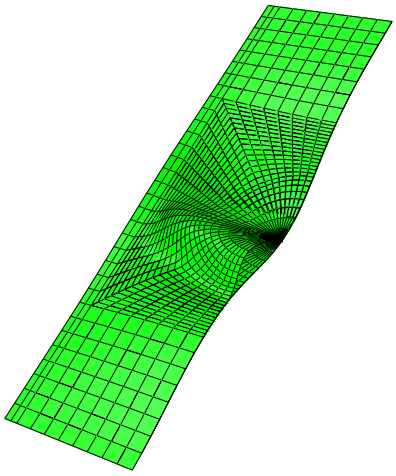


Figure 10: The postbuckling in the panel section (Abaqus/Explicit).

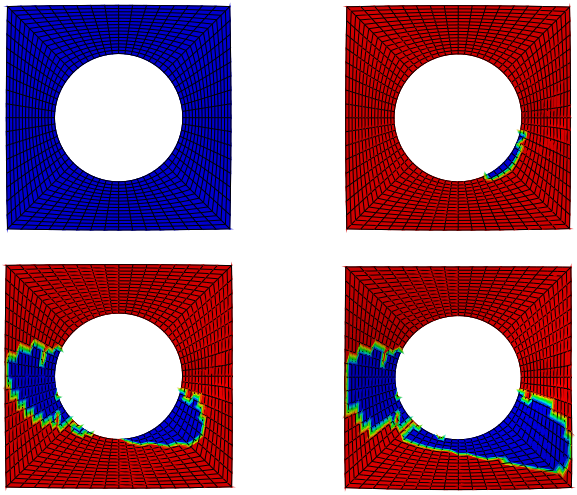


Figure 11: The growth of delamination (Abaqus/Explicit).

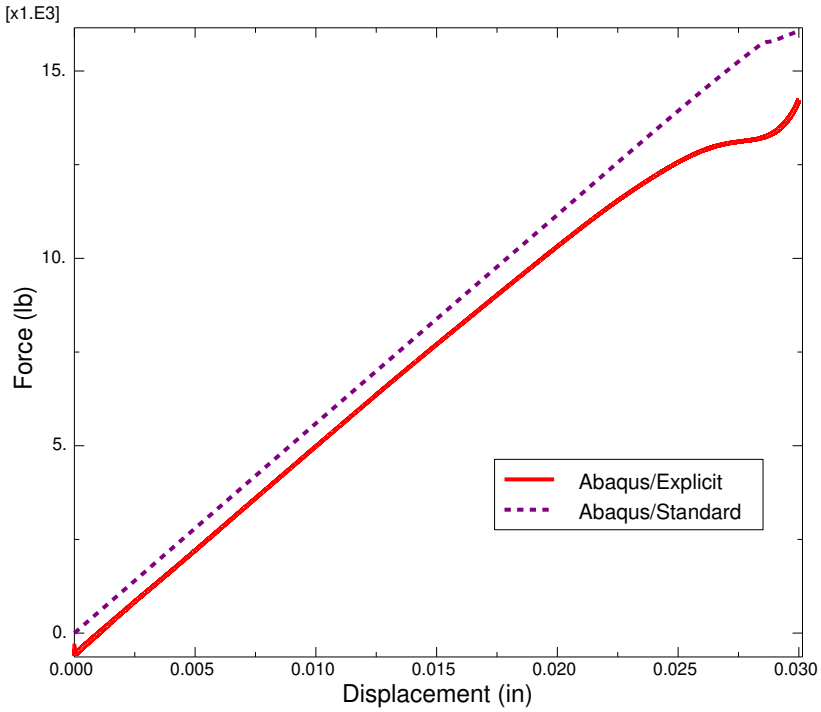


Figure 12: The force-displacement response comparison between Abaqus/Explicit and Abaqus/Standard.

Import analyses

In this section:

- *Springback of two-dimensional draw bending*
- *Deep drawing of a square box*

Springback of two-dimensional draw bending

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example illustrates the forming and springback analysis of a two-dimensional draw bending process. The forming analysis is performed using Abaqus/Explicit, and the springback analysis is run with Abaqus/Standard using an import analysis.

Problem description

The example described here is one of the benchmark tests reported at the Numisheet '93 Conference. The benchmark contains a series of six problems performed with three different materials and two different blank holder forces. One of the six problems is described here. The simulations for all the problems are described in the paper by Taylor et al. (1993).

The blank initially measures 350 mm by 35 mm and is 0.78 mm thick. The problem is essentially a plane strain problem (the out-of-plane dimension for the blank is 35 mm). A cross-section of the geometry of the die, the punch, the blank holder, and the blank is shown in *Figure 1*. The total blank holder force is 2.45 kN, and a mass of 5 kg is attached to the blank holder. A coefficient of friction of 0.144 is used for all interacting surfaces.

The blank is made of mild steel. The material is modeled as an elastic-plastic material with isotropic elasticity, using the Hill anisotropic yield criterion for the plasticity. The following material properties are used:

Young's modulus = 206.0 GPa

Poisson's ratio = 0.3

Density = 7800.

Yield stress σ_0 = 167.0 MPa

Anisotropic yield criterion: R_{11}=1.0, R_{22}=1.0402, R_{33}=1.24897, R_{12}=1.07895,

R_{13}=1.0, R_{23}=1.0

The problem is symmetric about a plane through the center of the punch, and only half of the problem is modeled. The blank is modeled with a single row of 175 first-order shell elements. Symmetry boundary conditions are applied on the plane of symmetry, and boundary conditions are applied on all the nodes of the blank to simulate the plane strain conditions. The out-of-plane dimension for the blank in the model is 5 mm; thus, the blank holder force is scaled appropriately.

The forming process is simulated in two steps with Abaqus/Explicit. The blank holder force is applied in the first step of the analysis. The force is ramped on with a smooth step definition to minimize inertia effects. In the second step of the analysis the punch is moved down 70 mm by prescribing the velocity of the rigid body reference node for the punch. The velocity is applied with a triangular smooth step amplitude function, starting and ending with zero velocity, and with a peak velocity occurring at the middle of the time period.

A significant amount of springback occurs in this case. Because the blank is very flexible and the fundamental mode of vibration is low, it would take a long simulation to obtain a quasi-static solution of the springback analysis in Abaqus/Explicit.

The springback analysis is performed with Abaqus/Standard using an import analysis. The results from the forming simulation in Abaqus/Explicit are imported into Abaqus/Standard, and a static analysis calculates the springback. During this step an artificial stress state that equilibrates the imported stress state is applied

automatically by Abaqus/Standard and gradually removed during the step. The displacement obtained at the end of the step is the springback, and the stresses give the residual stress state.

Settings in the import analysis determine the reference configuration. When you choose to update the reference configuration in an import analysis, the deformed sheet with its material state at the end of the Abaqus/Explicit analysis is imported into Abaqus/Standard and the deformed configuration becomes the reference configuration. This procedure is most convenient if, during postprocessing, the displacements due to springback need to be displayed. When you do not update the reference configuration, the material state, displacements, and strains of the deformed sheet at the end of the Abaqus/Explicit analysis are imported into Abaqus/Standard, and the original configuration remains as the reference configuration. This procedure should be used if it is desirable to obtain a continuous displacement solution.

In this two-dimensional draw bending problem significant springback occurs, and large-displacement effects are included in the calculations by considering geometric nonlinearities in the step definition.

Further details of the import capability are discussed in *Transferring Results between Abaqus/Explicit and Abaqus/Standard*.

Results and discussion

The optimum peak velocity for the punch (the value that gives quasi-static results at least cost) is determined by running the explicit analysis with peak velocities of 30 m/s, 15 m/s, and 5 m/s. The energy histories are shown in *Figure 2*, *Figure 3*, and *Figure 4*, respectively. From these results it is evident that the amount of kinetic energy in the model is too large at a peak velocity of 30 m/s for the analysis to simulate the quasi-static forming process, while at a peak velocity of 5 m/s the kinetic energy is virtually zero. A peak velocity for the punch of 15 m/s is chosen for the forming analysis, as the kinetic energy for this case is considered low enough not to affect the results significantly. For accurate springback analysis it is important that stresses are not influenced by inertia effects.

The blank at the end of the Abaqus/Explicit forming analysis is shown in *Figure 5*. The shape after springback is shown in *Figure 6*. The results compare well with the reported experimental data. In the numerical results the angle between the outside flange and the horizontal axis is 22° in the Abaqus/Explicit analysis and 17.1° in the Abaqus/Standard analysis. The differences in the results are due to differences in the contact calculations. In Abaqus/Explicit the change in shell thickness is accounted for during contact calculations, and the surface-to-surface contact formulation in Abaqus/Standard for contact pairs and general contact does the same. However, for the node-to-surface contact formulation in Abaqus/Standard for contact pairs when a shell is pinched between two surfaces, it is necessary to use “softened” contact to account for the shell thickness and, consequently, an approximation is made. A modified Abaqus/Explicit analysis that uses softened contact and zero shell thickness (with a zero-thickness surface) has been set up to compare the results from Abaqus/Explicit and Abaqus/Standard directly. The predicted results match closely. The average angle measured in the experiments is 17.1°, with a range from 9° to 23° in the experimental results. The results of the springback analysis when the reference configuration is updated are nearly identical to the results when the reference configuration is not updated.

Input files

springback_exp_form.inp	Forming analysis in Abaqus/Explicit with a punch velocity of 15 m/s using contact pairs.
springback_exp_form_gcont.inp	Forming analysis in Abaqus/Explicit with a punch velocity of 15 m/s using general contact.
springback_std_importyes.inp	Springback analysis in Abaqus/Standard with the **IMPORT*, UPDATE=YES option.
springback_std_importno.inp	Springback analysis in Abaqus/Standard with the **IMPORT*, UPDATE=NO option.

springback_std_both.inp	Input data used with Abaqus/Standard for both the forming and the springback analyses.
springback_std_both_surf.inp	Input data used with Abaqus/Standard for both the forming and the springback analyses using the surface-to-surface contact formulation.
springback_std_both_gcontsd.inp	Input data used with Abaqus/Standard for both the forming and the springback analyses using general contact.
springback_exp_form_soft.inp	Modified forming analysis in Abaqus/Explicit with a punch velocity of 15 m/s using softened contact and contact pairs as in springback_std_both.inp.
springback_exp_form_soft_gcont.inp	Modified forming analysis in Abaqus/Explicit with a punch velocity of 15 m/s using softened general contact.
springback_exp_punchv30.inp	Forming analysis in Abaqus/Explicit with a punch velocity of 30 m/s using contact pairs.
springback_exp_punchv5.inp	Forming analysis in Abaqus/Explicit with a punch velocity of 5 m/s using contact pairs.
springback_exp_punchv30_gcont.inp	Forming analysis in Abaqus/Explicit with a punch velocity of 30 m/s using general contact.
springback_exp_punchv5_gcont.inp	Forming analysis in Abaqus/Explicit with a punch velocity of 5 m/s using general contact.

References

- Taylor, L. M., J. Cao, A. P. Karafillis, and M. C. Boyce, "Numerical Simulations of Sheet Metal Forming," *Proceedings of 2nd International Conference, NUMISHEET 93, Isehara, Japan, Ed. A. Makinovchi, et al.*

Figures

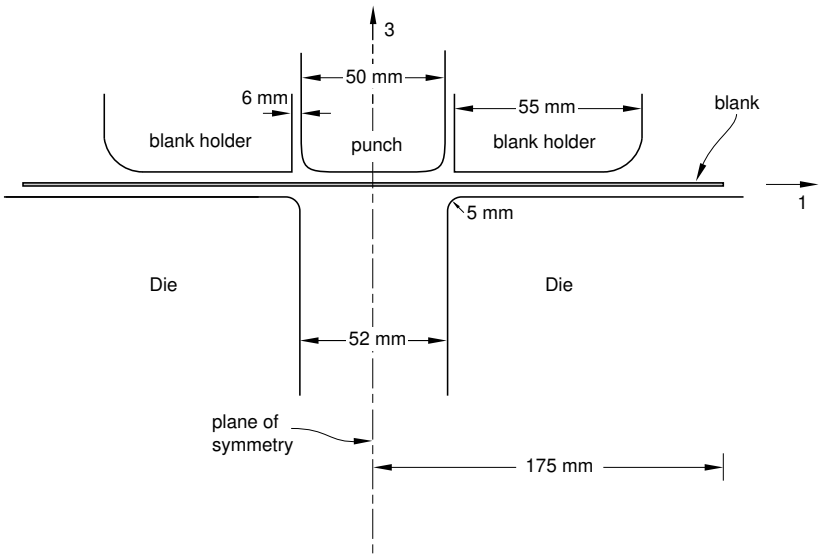


Figure 1: Cross-section showing the geometry of the die, the punch, the blank holder, and the blank.

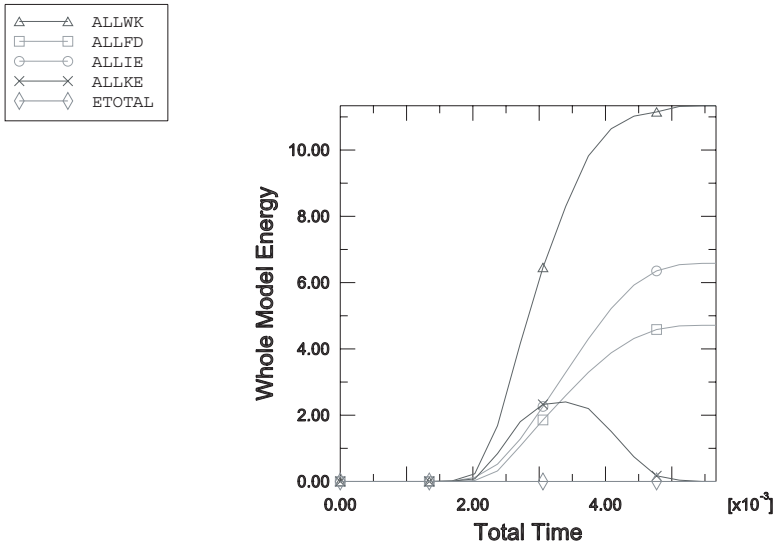


Figure 2: Energy history for forming analysis: 30 m/s peak velocity of punch.

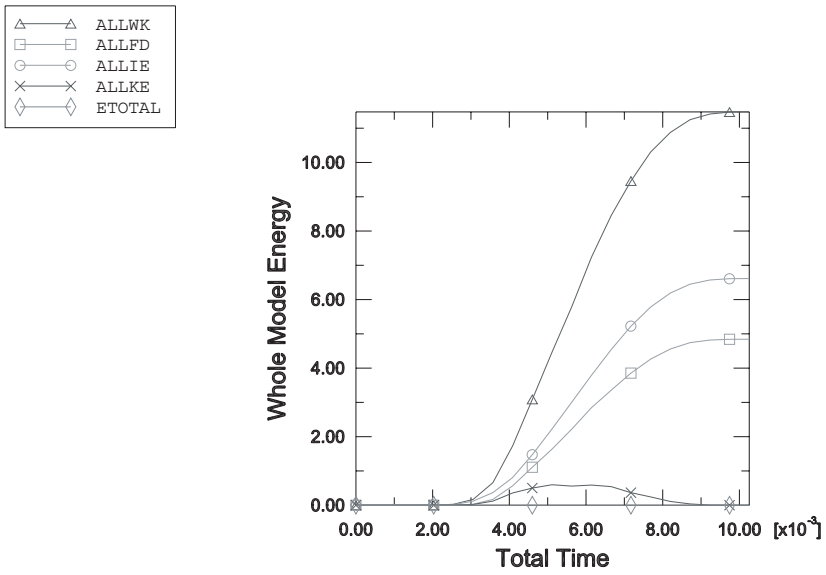


Figure 3: Energy history for forming analysis: 15 m/s peak velocity of punch.

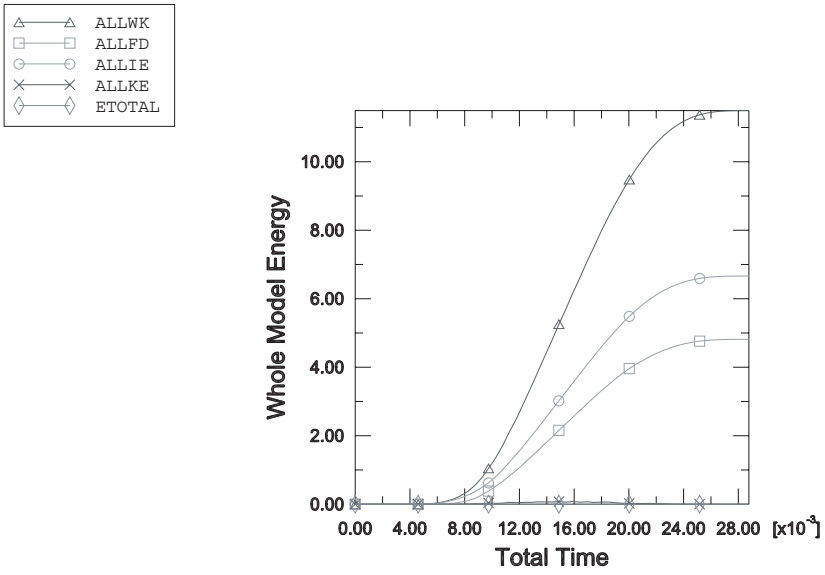


Figure 4: Energy history for forming analysis: 5 m/s peak velocity of punch.

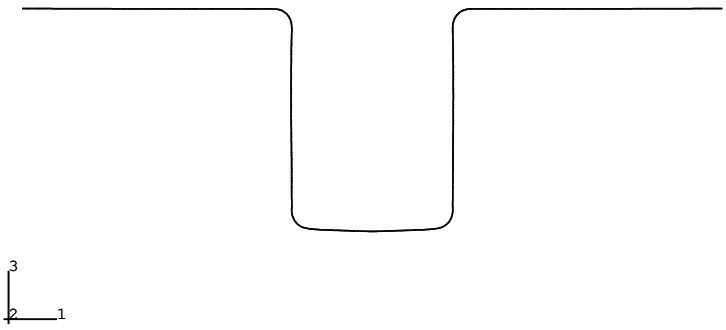


Figure 5: Blank at the end of the forming analysis in Abaqus/Explicit.

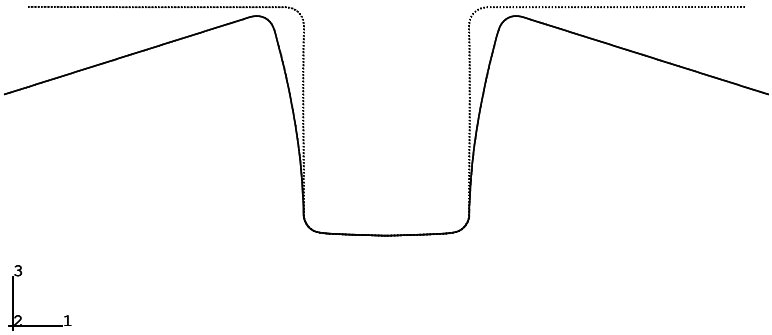


Figure 6: Blank after springback in Abaqus/Standard.

Deep drawing of a square box

Products: Abaqus/Standard Abaqus/Explicit

References:

- *About General Contact in Abaqus/Explicit*
- *About Contact Pairs in Abaqus/Standard*
- **CONTACT*
- **CONTACT PAIR*

Overview

This example illustrates the forming of a three-dimensional shape by a deep drawing process.

The following Abaqus features are demonstrated:

- transferring results from Abaqus/Explicit to Abaqus/Standard using the import analysis technique;
- comparing results from an analysis sequence that uses Abaqus/Explicit for a forming step and Abaqus/Standard for a springback analysis with results obtained using Abaqus/Standard for both the forming and springback steps; and
- comparing characteristics of different contact formulations with finite sliding, especially with regard to the treatment of surface thickness.

Application description

In general, the forming procedure involves a forming step followed by a springback that occurs after the blank is removed from the tool. The goal of analyzing the forming procedure is to determine the final deformed shape after springback.

Geometry

The blank is initially square, 200 mm by 200 mm, and is 0.82 mm thick. The rigid die is a flat surface with a square hole 102.5 mm by 102.5 mm, rounded at the edges with a radius of 10 mm. The rigid square punch measures 100 mm by 100 mm and is rounded at the edges with the same 10 mm radius. The rigid blank holder can be considered a flat plate, since the blank never comes close to its edges. The geometry of these rigid parts is illustrated in *Figure 1*.

Materials

The blank is made of aluminum-killed steel, which is assumed to satisfy the Ramberg-Osgood relation between true stress and logarithmic strain,

$$\epsilon = (\sigma/K)^{1/n},$$

with a reference stress value (K) of 513 MPa and a work-hardening exponent (n) of 0.223. Isotropic elasticity is assumed, with a Young's modulus of 211 GPa and a Poisson's ratio of 0.3. An initial yield stress of 91.3 MPa is obtained from these data. The stress-strain behavior is defined by piecewise linear segments matching the Ramberg-Osgood curve up to a total (logarithmic) strain level of 107%, with Mises yield, isotropic hardening, and no rate dependence.

Boundary conditions and loading

Given the symmetry of the problem, it is sufficient to model only a one-eighth sector of the box. However, for easier visualization we have employed a one-quarter model. Symmetry boundary conditions are applied at the quarter edges of the blank. The punch and the blank holders are allowed to move only in the vertical direction. Allowing vertical motion of the blank holders accommodates changes in the blank thickness during forming.

Interactions

Contact interaction is considered between the blank and the punch with a friction coefficient of 0.25 and between the blank and the die with a friction coefficient of 0.125. The contact interaction between the blank and the blank holders is assumed to be frictionless.

Abaqus modeling approaches and simulation techniques

The most efficient way to analyze this type of problem is to analyze the forming step using Abaqus/Explicit and to import the results in Abaqus/Standard to analyze the springback. For verification purposes the complete analysis is also carried out with Abaqus/Standard. However, this is computationally more expensive and will be prohibitively more expensive for simulation of the forming of realistic, complex components.

This problem is used in Nagtegaal and Taylor (1991) where implicit and explicit finite element techniques for forming problems are compared. The computer time involved in running the simulation using explicit time integration with a given mesh is directly proportional to the time period of the event, since the stable time increment size is a function of the mesh size (length) and the material stiffness. Thus, it is usually desirable to run the simulation at an artificially high speed compared to the physical process. If the speed in the simulation is increased too much, the solution does not correspond to the low-speed physical problem; i.e., inertial effects begin to dominate. In a typical forming process the punch may move at speeds on the order of 1 m/sec, which is extremely slow compared to typical wave speeds in the materials to be formed (the wave speed in steel is approximately 5000 m/sec). In general, inertia forces will not play a dominant role for forming rates that are considerably higher than the nominal 1 m/sec rates found in the physical problem. Therefore, explicit solutions are obtained with punch speeds of 10, 30, and 100 m/sec for comparison with the static solution obtained with Abaqus/Standard. In the results presented here, the drawing process is simulated by moving the reference node for the punch downward through a total distance of 36 mm in 0.0036 seconds. A detailed comparison of analyses of various metal forming problems using explicit dynamic and static procedures is discussed in the paper by Nagtegaal and Taylor (1991).

Although this example does not contain rate-dependent material properties, it is common in sheet metal forming applications for this to be a consideration. If the material is rate-dependent, the velocities cannot be artificially increased without affecting the material response. Instead, the analyst can use the technique of mass scaling to adjust the effective punch velocity without altering the material properties. *Rolling of thick plates* contains an explanation and an example of the mass scaling technique.

Summary of analysis cases

Forming analysis with Abaqus/Explicit.	Case 1a	Using the general contact capability.
	Case 1b	Using the kinematic contact pairs.
	Case 1c	Using penalty contact pairs.
	Case 1d	Forming analysis of a fine mesh case using the general contact capability (included for the sole purpose of testing the performance of the Abaqus/Explicit code).

	Case 1e	Forming analysis of a fine mesh case using kinematic contact pairs (included for the sole purpose of testing the performance of the Abaqus/Explicit code).
Springback analysis with Abaqus/Standard.	Case 2a	Abaqus/Standard springback analysis using an import analysis with no update of the reference configuration.
	Case 2b	Abaqus/Standard springback analysis using an import analysis with update of the reference configuration.
	Case 2c	Springback analysis of a fine mesh case (included for the sole purpose of testing the performance of the Abaqus/Standard code) using an import analysis with update of the reference configuration.
Forming and springback analysis with Abaqus/Standard.	Case 3a	Using the surface-to-surface contact formulation.
	Case 3b	Using the node-to-surface contact formulation.

Analysis types

As described earlier, the import capability in Abaqus is utilized to run the forming step as an explicit dynamic analysis followed by a static stress analysis using Abaqus/Standard for calculating the springback. For comparison, results from a complete static stress analysis using Abaqus/Standard for both the forming and the springback steps are presented.

Analysis techniques

The import feature in Abaqus is used for transferring results from Abaqus/Explicit to Abaqus/Standard.

Mesh design

The blank is modeled with 4-node, bilinear finite-strain elements (type S4R); while the punch, die, and the blank holder are meshed using 4-node, three-dimensional rigid surface elements (type R3D4). The mesh design for the various parts is shown in *Figure 1* and *Figure 2*.

Loads

The blank is held between the blank holders by applying a concentrated load of 22.87 kN. Further loading on the blank is applied by contact forces with the punch in the forming step.

Analysis steps

Using Abaqus/Explicit for the forming procedure involves a single forming step where the rigid punch is pushed against the blank while the blank is held by the blank holders by applying a concentrated load. This description applies to Cases 1a–1e. For the import analysis in Abaqus/Standard a single step is used to calculate the springback as in Cases 2a–2c. For the complete analysis in Abaqus/Standard as in Cases 3a and 3b, the following steps are adopted:

- First step: the blank holders are brought in contact with the blank by applying a small displacement to the reference point of one of the rigid blank holders.
- Second step: a concentrated load is applied to the reference point of the blank holder to hold the blank in place while maintaining contact.
- Following steps: the forming is effected by pushing the rigid punch against the blank.
- Final two steps: the springback is analyzed by deactivating the contact pairs.

Output requests

The output variables STH for shell thickness and PEEQ for equivalent plastic strain are specifically requested along with preselected variables. Further, the history of reaction force and displacement for the punch is also requested.

Case 1a: Explicit forming analysis using general contact

This analysis pertains only to the forming step. For the complete analysis the forming step in this case needs to be followed by a springback analysis (either Case 2a or Case 2b).

Interactions

General contact is used (see the general contact specification) to define contact interactions in this case. This allows very simple definitions of contact with very few restrictions on the types of surfaces involved (see *About General Contact in Abaqus/Explicit*). However, general contact does not account for changes in shell thickness by default. Consequently, the general contact surface property assignment must account for thinning of the blank.

Case 1b: Explicit forming analysis using kinematic contact pairs

This analysis again pertains only to the forming step. For the complete analysis the forming step needs to be followed by a springback analysis (either Case 2a or Case 2b).

Interactions

Contact pairs are defined to include blank interaction with the punch, die, and the blank holder separately with appropriate friction behavior as previously specified. The contact pair algorithm, which is specified in the contact pair definition, has more restrictions on the types of surfaces involved and often requires more careful definition of contact (see *About Contact Pairs in Abaqus/Explicit*). Contact interactions are defined between all element-based surfaces in the model.

Case 1c: Explicit forming analysis using penalty contact pairs

This analysis pertains only to the forming step. The springback calculations have to be done separately (Case 2a or Case 2b).

Interactions

Penalty contact is specified for contact pairs to include blank interaction with the punch, die, and the blank holder separately with appropriate friction behavior.

Case 1d: Explicit forming analysis using general contact with a refined mesh

In this case the mesh for the blank is uniformly refined so that the number of elements in each direction is twice the number in the previous cases. This case is run to purely benchmark the efficiency of performing an explicit analysis.

Interactions

The contact interactions are exactly the same as in Case 1a.

Case 1e: Explicit forming analysis using kinematic contact pairs with a refined mesh

In this case the refined mesh defined in Case 1d is utilized for performing the explicit forming analysis.

Interactions

The contact interactions are exactly the same as in Case 1b.

Case 2a: Static springback analysis with no update of the reference configuration during import

For running this case, a prior explicit forming analysis (Case 1a, Case 1b, or Case 1c) should have been completed for importing results into Abaqus/Standard. By specifying an import analysis with no update of the reference configuration, the displacements are the total values relative to the original reference configuration before the forming analysis. This makes it easy to compare the results with the analysis in which both the forming and springback are analyzed with Abaqus/Standard.

Boundary conditions

Boundary conditions are imposed in the Abaqus/Standard analysis to prevent rigid body motion and for symmetry. The node at the center of the box is fixed in the z-direction.

Interactions

No contact interactions are used in this analysis once the deformed sheet with its material state at the end of Abaqus/Explicit is imported.

Case 2b: Static springback analysis with update of the reference configuration during import

Similar to Case 2a, a prior explicit forming analysis (Case 1a, Case 1b, or Case 1c) should have been completed for importing results into Abaqus/Standard. However, specifying an import analysis with update of the reference configuration implies that the displacements are relative to the deformed configuration at the end of the forming analysis. The boundary conditions and interactions are exactly the same as Case 2a.

Case 2c: Static springback analysis using a refined mesh with update of the reference configuration during import

For running this case, Case 1d or Case 1e for explicit forming analysis should have been completed for importing results into Abaqus/Standard. Here again, specifying an import analysis with update of the reference configuration implies that the displacements are relative to the deformed configuration at the end of the forming analysis. The boundary conditions and interactions are exactly the same as Case 2a.

Case 3a: Static analysis of forming and springback using surface-to-surface contact

In this analysis both the forming and the springback steps are analyzed in Abaqus/Standard.

Interactions

In this case the surface-to-surface contact formulation is invoked. Since double-sided surfaces are not available in Abaqus/Standard, two single-sided surfaces are used to model the blank when the forming step is modeled in Abaqus/Standard: one surface to model the top of the blank and one to model the bottom of the blank. The surface-to-surface contact formulation considers the original shell thickness by default throughout the analysis. There is no option to consider the current shell thickness instead of the original shell thickness.

Solution controls

Contact stabilization is used to avoid chattering between the blank and the rigid surfaces it is in contact with. In addition, the adaptive automatic stabilization scheme is applied to improve the robustness of the static analysis.

Case 3b: Static analysis of forming and springback using node-to-surface contact

As in Case 3a, both the forming and the springback steps are analyzed in Abaqus/Standard.

Interactions

In this case the node-to-surface contact formulation is used. Since, shell thickness cannot be considered by node-to-surface finite-sliding contact, "softened" contact is used to approximate the thickness (see the modified contact pressure-overclosure relationship).

Discussion of results and comparison of cases

Figure 3, *Figure 5*, and *Figure 4* show contours of shell thickness in the blank at the end of the forming step before springback in Abaqus/Explicit (Case 1a) and Abaqus/Standard analyses (Case 3a and Case 3b), respectively. *Figure 6*, *Figure 7*, and *Figure 8* show contours of equivalent plastic strain in the blank in the final deformed shape for the Abaqus/Explicit and the two Abaqus/Standard analyses, respectively. The predicted results are very similar. The Abaqus/Explicit results match the surface-to-surface contact formulation in Abaqus/Standard more closely than the node-to-surface results in Abaqus/Standard. This observation is true for both the equivalent plastic strain contours and shell thickness contours and is a consequence of the intrinsic differences between the various contact formulations. The node-to-surface formulation in Abaqus/Standard accounts for the shell thickness indirectly by using carefully specified pressure-overclosure relationships (soft contact). The other analyses use contact formulations that account for shell thickness directly. Despite the fact that the surface-to-surface formulation in Abaqus/Standard uses the original shell thickness throughout the analysis, the results correlate well.

Closer inspection of the results reveals that the corners of the box are formed by stretching, whereas the sides are formed by drawing action. This effect leads to the formation of shear bands that run diagonally across the sides of the box, resulting in a nonhomogeneous wall thickness. The material draws unevenly from the originally straight sides of the blank. Applying a more localized restraint near the midedges of the box (for example, by applying drawbeads) and relaxing the restraint near the corners of the box is expected to increase the quality of the formed product.

Figure 9 shows the reaction force on the punch, and *Figure 10* shows the thinning of an element at the corner of the box. Here again, the results from the surface-to-surface formulation in Abaqus/Standard match those from Abaqus/Explicit better than the node-to-surface contact formulation in Abaqus/Standard. Despite the approximate treatment of surface thickness via the pressure-overclosure relationship for the node-to-surface formulation, the shell thicknesses predicted by Abaqus/Explicit and the node-to-surface formulation in Abaqus/Standard differ only by about 4%, reflecting the overall quality of the results.

The springback analysis runs in 6 increments for both of the contact formulations in Abaqus/Standard. Most of the springback occurs in the z-direction, and the springback is not significant. The corner of the outside edge of the formed box drops approximately 0.35 mm, while the vertical side of the box rises by approximately 0.26 mm. *Figure 11* shows a contour plot of the displacements in the z-direction obtained from the springback analysis using the node-to-surface formulation.

The analysis with no reference configuration update yields similar results. However, in this case the displacements are interpreted as total values relative to the original configuration.

Files

Case 1a: Explicit forming analysis using general contact

deepdrawbox_exp_form.inp Input file for the explicit forming step.

Case 1b: Explicit forming analysis using kinematic contact pairs

deepdrawbox_exp_form_cpair.inp Input file for the explicit forming step.

Case 1c: Explicit forming analysis using penalty contact pairs

deepdrawbox_exp_form_plty_cpair.inp Input file for the explicit forming step.

Case 1d: Explicit forming analysis using general contact with a refined mesh

deepdrawbox_exp_finemesh.inp Input file for the explicit forming step.

Case 1e: Explicit forming analysis using kinematic contact pairs with a refined mesh

deepdrawbox_exp_finemesh_cpair.inp Input file for the explicit forming step.

Case 2a: Static springback analysis with UPDATE=NO during import

deepdrawbox_std_importno.inp Input file for the static springback step.

Case 2b: Static springback analysis with UPDATE=YES during import

deepdrawbox_std_importyes.inp Input file for the static springback step.

Case 2c: Static springback analysis using a refined mesh with UPDATE=YES during import

deepdrawbox_std_finesprngback.inp Input file for the static springback step with a refined mesh for the blank.

Case 3a: Static analysis of forming and springback using surface-to-surface contact

deepdrawbox_std_both_surf.inp Input file for the complete static analysis.

deepdrawbox_std_both_surf_stabil_adap.inp Input file for the complete static analysis with adaptive stabilization.

Case 3b: Static analysis of forming and springback using node-to-surface contact

deepdrawbox_std_both.inp Input file for the complete static analysis.

References

- Nagtegaal J. C. and L. M. Taylor, “Comparison of Implicit and Explicit Finite Element Methods for Analysis of Sheet Forming Problems,”VDI Berichte No. 894, 1991.

Figures

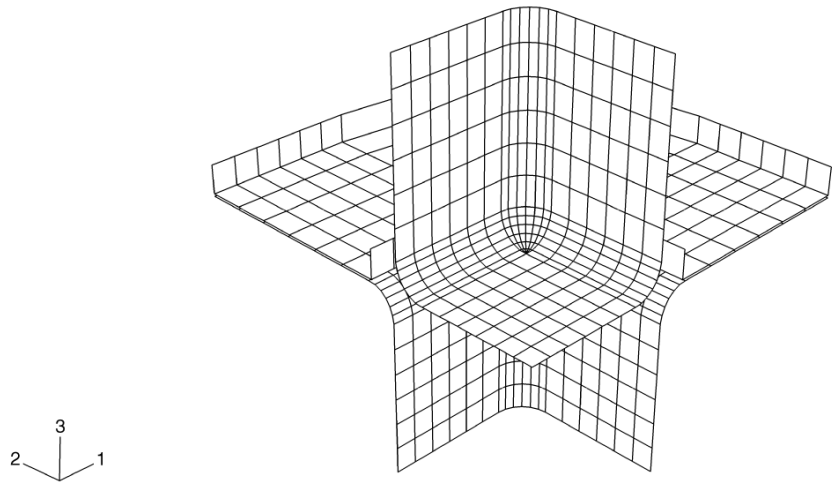


Figure 1: Meshes for the die, punch, and blank holder.

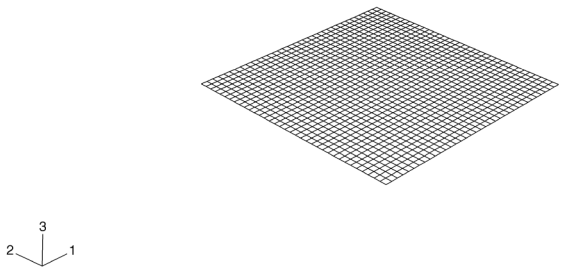


Figure 2: Undeformed mesh for the blank.

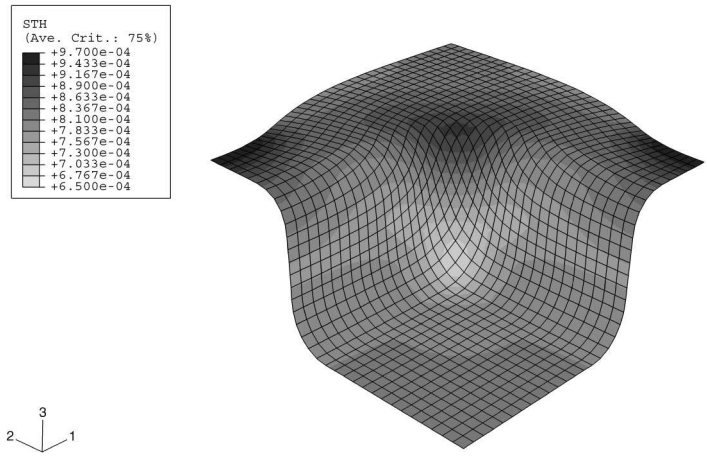


Figure 3: Contours of shell thickness with Abaqus/Explicit.

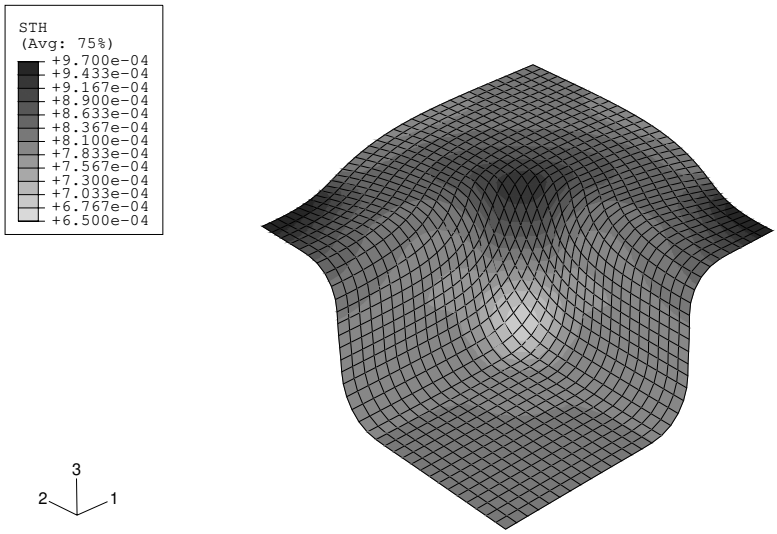


Figure 4: Contours of shell thickness with Abaqus/Standard using surface-to-surface contact formulation.

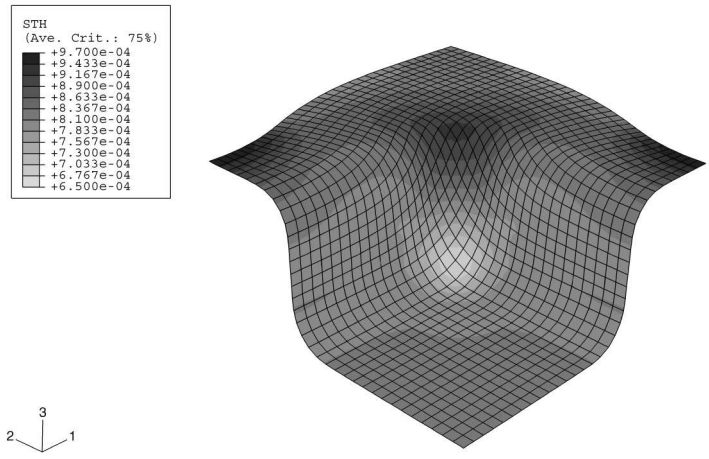


Figure 5: Contours of shell thickness with Abaqus/Standard using node-to-surface contact formulation.

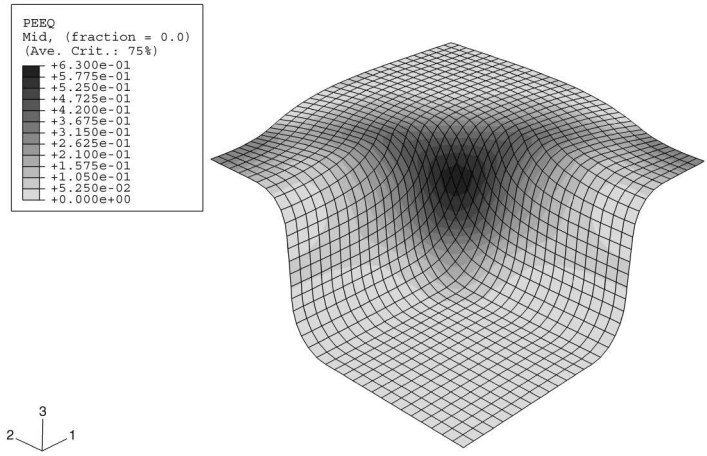


Figure 6: Contours of equivalent plastic strain with Abaqus/Explicit.

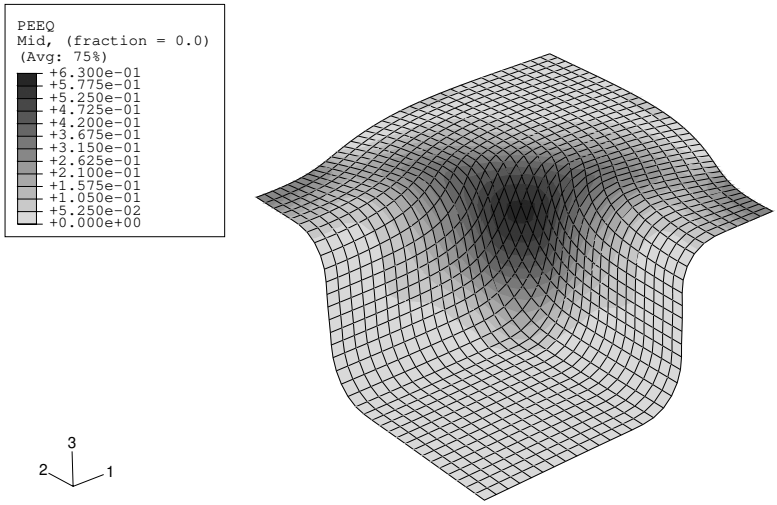


Figure 7: Contours of equivalent plastic strain with Abaqus/Standard using surface-to-surface contact formulation.

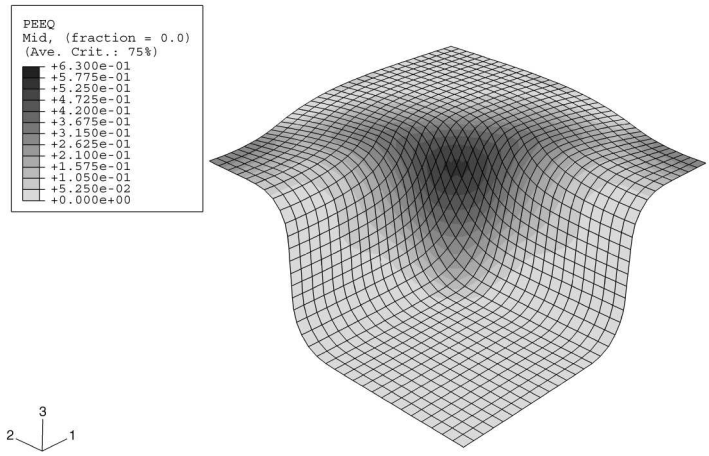


Figure 8: Contours of equivalent plastic strain with Abaqus/Standard using node-to-surface contact formulation.

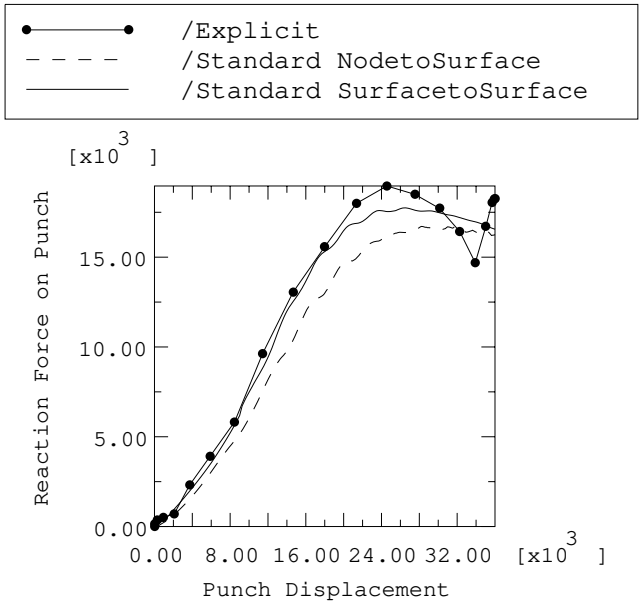


Figure 9: Reaction force on the punch versus punch displacement.

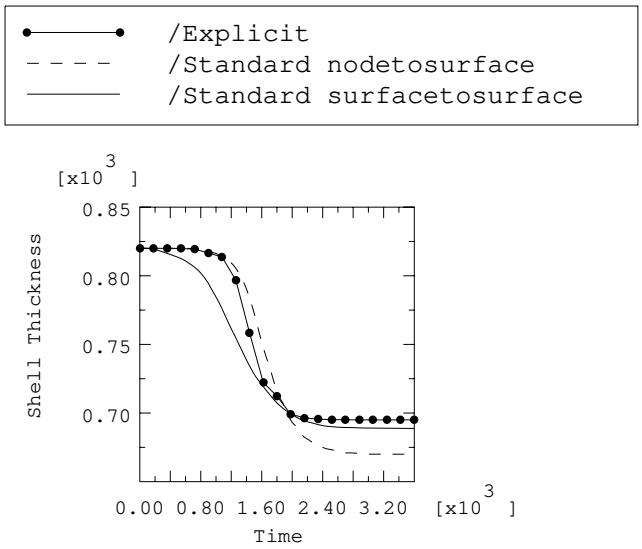


Figure 10: Shell thickness of the thinnest part of the blank versus time.

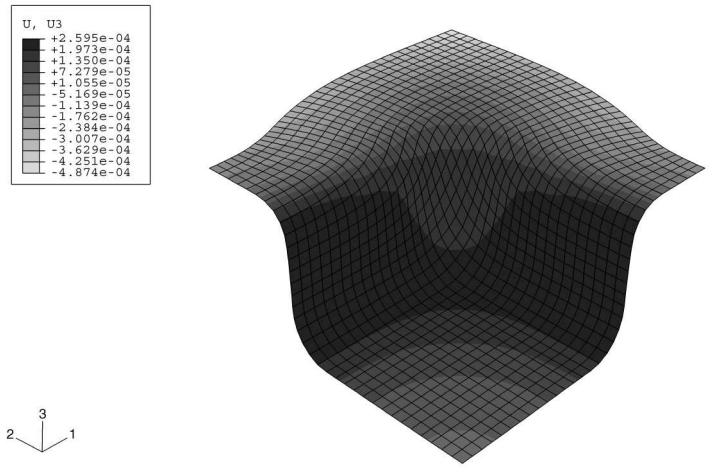


Figure 11: Contour plot showing the springback in the z-direction.

Dynamic Stress/Displacement Analyses

In this section:

- *Dynamic stress analyses*
- *Mode-based dynamic analyses*
- *Eulerian and co-simulation analyses*

Dynamic stress analyses

In this section:

- *Nonlinear dynamic analysis of a structure with local inelastic collapse*
- *Detroit Edison pipe whip experiment*
- *Rigid projectile impacting eroding plate*
- *Eroding projectile impacting eroding plate*
- *Tennis racket and ball*
- *Pressurized fuel tank with variable shell thickness*
- *Modeling of an automobile suspension*
- *Explosive pipe closure*
- *Knee bolster impact with general contact*
- *Crimp forming with general contact*
- *Collapse of a stack of blocks with general contact*
- *Cask drop with foam impact limiter*
- *Oblique impact of a copper rod*
- *Water sloshing in a baffled tank*
- *Seismic analysis of a concrete gravity dam*
- *Progressive failure analysis of thin-wall aluminum extrusion under quasi-static and dynamic loads*
- *Impact analysis of a pawl-ratchet device*
- *High-velocity impact of a ceramic target*
- *Confined compression of multiple two-dimensional cylinders with different material properties*

Nonlinear dynamic analysis of a structure with local inelastic collapse

Products: Abaqus/Standard

Overview

This example illustrates an inexpensive approach to the prediction of the overall response of a structure that exhibits complex local behavior.

The case studied in this example is an unrestrained pipe whip example, where an initially straight pipe undergoes so much motion that the pipe section collapses. A two-stage technique is used to predict the response. First, the collapse of the section is studied under static conditions using a generalized plane strain model. This analysis defines the moment-curvature relationship for the section under conditions of pure bending. It also shows how the section deforms as it collapses. This information can be used to judge whether the deformation is reasonable with respect to possible failure (fracture) of the section. In addition, this first stage analysis can be used to calculate the change in the cross-sectional area enclosed by the pipe as a function of the curvature of the pipe. In a pipe whip case the driving force is caused by fluid jetting from a break in the pipe; and, if the pipe does undergo such large motion, a section may be deformed sufficiently to choke the flow. The second stage of the analysis is to predict the overall dynamic response of the pipe, using the moment-curvature response of the section that has been obtained in the first analysis to define the inelastic bending behavior of the beam. This two-stage approach provides a straightforward, inexpensive method of evaluating the event. The method is approximate and may give rise to significant errors. That aspect of the approach is discussed in the last section below.

Modeling

The problem is shown in *Figure 1*. To investigate the static collapse of the section, we consider a unit length of an initially straight pipe subjected to a pure bending moment and assume that plane sections remain plane. We can think of this unit length of pipe as being bounded at its ends by rigid walls and imagine the bending to be achieved by rotation of the walls relative to each other, the end sections being allowed to distort only in the plane of the walls (see *Figure 2*). With this idealization the pipe section can be modeled and discretized using generalized plane strain elements, as shown in *Figure 2*. Bending occurs about the x-axis, and symmetry conditions are prescribed along the y-axis. There will not be symmetry about the x-axis because of the Poisson's effect. To remove rigid body motion in the y-direction, point A is fixed in that direction. Symmetry implies no x-displacement at $\boldsymbol{x} = 0$ and no rotation of the section about the y-axis. 4-node and 8-node generalized plane strain elements are used. In addition to the four or eight regular nodes used for interpolation, these elements require one extra reference node that is common to all elements in the model. Degree of freedom 3 at the reference node is the relative displacement between the boundary planes, while degrees of freedom 4 and 5 are the relative rotations of these planes.

Since the problem involves bending the pipe cross-section, regular fully integrated 4-node elements will not provide accurate results, especially when the pipe is fairly thin, because they will suffer from “shear locking”—they will not provide the bending deformation because to do so requires that they shear at their integration points and this shearing requires an unrealistically large amount of strain energy. This problem is avoided by integrating the elements only at their centroids but the elements then exhibit singular modes—modes of deformation that do not cause strain. Abaqus uses orthogonal hourglass generalized strains and associated stiffness to avoid such spurious singular mode behavior. Although these techniques are not always reliable, they can work well and do so in this example. The problem is also modeled with the fully integrated incompatible mode element CPEG4I and the modified triangular element CPEG6M. CPEG4I elements do not have any hourglass modes and perform well in bending. For additional discussion of these points see *Performance of continuum and shell elements for linear analysis of bending problems*.

For the dynamic analysis of the pipe whip event the pipeline is modeled with 10 beam elements of type B21. These are planar beam elements that use linear interpolation of displacement and rotation. The moment-curvature relation obtained from the static analysis (shown in *Figure 5*) is used in the nonlinear general cross-section to define the bending behavior of the beams. A definition for the axial force versus strain behavior of the beams is also required and is provided by conversion of the uniaxial stress-strain relation given in *Figure 1* into force versus strain by multiplying the stress by the current area, A, of the cross-section. This current area is computed from the original cross-sectional area A_0 by assuming that the material is incompressible, so $A = A_0 l_0 / l$, where l is the current length and l_0 is the original length.

This definition of the beam section behavior provides for no interaction between the bending and axial stretching, although in most real cases there will actually be some interaction. However, this approximation is probably reasonable in this particular problem since the response is predominantly bending and no appreciable error is introduced by the little stretching that does occur.

Loading and solution control

In the large-displacement static analysis of the inelastic collapse of the section, rotation of the boundary planes about the x-axis is prescribed at degree of freedom 4 of the generalized plane strain reference node. The Riks procedure is used: this method usually provides rapid convergence in such cases, especially when unstable response occurs.

In the large-displacement dynamic analysis the blowdown force is treated as a follower force. During the first 0.06 seconds of the event it has a constant magnitude of 30 kN (this is about three times the load required to produce maximum moment in the static response of the section). After that time the load is zero. The response is computed for a time period of 0.4 seconds, using automatic time incrementation. A half-increment residual tolerance (HAFTOL) of 30 kN (which is the magnitude of the applied load) is used. Since we expect considerable plastic deformation, high frequency response should be damped quickly in the actual event, so that this value of HAFTOL should be adequate to give reasonably accurate results.

Results and discussion

Figure 3 shows a series of contour plots of equivalent plastic strain (plotted on the deformed configuration) from the static analysis using element type CPEG8R, and *Figure 4* shows the same plots for the analysis using element type CPEG4R. *Figure 3* clearly shows that the discretization is too coarse or should be rezoned later in the deformation, but it is judged that this is not critical to the overall moment-rotation response prediction. *Figure 5* shows the moment-curvature responses predicted by the analyses. The unstable nature of the response is clearly illustrated.

Figure 6 shows a series of deformed configuration plots from the dynamic analysis. After the shutdown of the force at 0.06 seconds, the momentum of the pipe is enough to cause localization of the deformation at the root of the cantilever as the section collapses there: the pipe whips around this hinge in a full circle and beyond its initial configuration. As well as this major hinge at the root, permanent plastic deformation develops throughout most of the pipe, leaving it bent into an arc. Time history plots of the tip displacement are shown in *Figure 7* and of the curvature strain at the localized hinge in *Figure 8*. *Figure 9* shows the moment-curvature response for the element at the support and shows the elastic unloading and reloading that takes place during and at the end of the event. *Figure 10* shows the history of the energy content during the dynamic analysis and clearly shows the initial build-up of kinetic energy, which is then converted almost entirely to plastic dissipation.

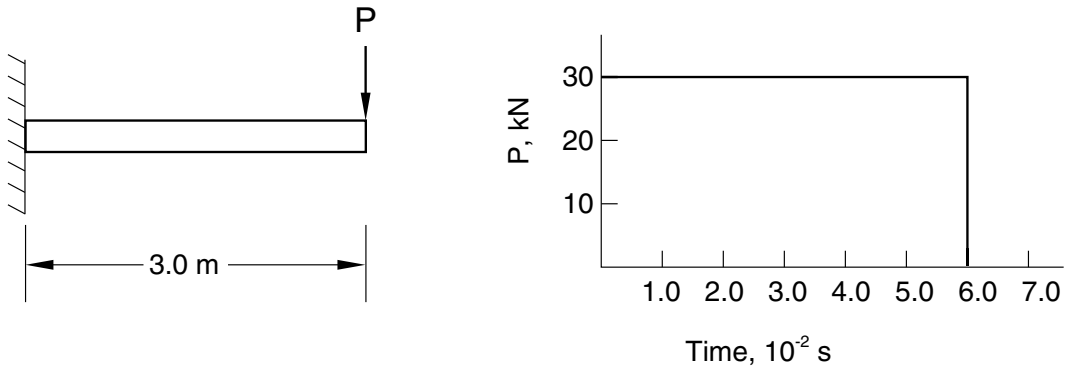
This two-stage approach to the problem has the advantages of being simple and computationally inexpensive. It contains some obvious approximations. One is that interaction effects between bending, axial, and torsional behavior are neglected. This lack of interaction between the various modes of cross-sectional response is a basic approximation of the nonlinear beam general section option. In reality, axial or torsional strain will have the effect of reducing the strength of the section in bending. This effect is unlikely to be significant in a case that is dominated by bending, but it can be important if large axial or torsional loadings occur. The approach also

neglects the effect of the axial gradient of the cross-sectional behavior on the response. This may be a significant error, but its evaluation would require a detailed, three-dimensional analysis for comparison; and that exercise is beyond the scope of this example. Another possibly significant error is the neglect of rate effects on the response. The cross-sectional collapse involves large strains, which occur in a very short time in the dynamic loadings, so high strain rates arise. It is likely that the material will exhibit strain rate dependence in its yield behavior and will, therefore, be rather stiffer than the static analysis predicts it to be. This should have the effect of spreading the hinge along the pipe and reducing the localization (because the strain rates increase at the section where most deformation is occurring, and that increased strain rate increases the resistance of the section). The magnitude of this effect can be estimated from the solution we have obtained. From *Figure 3* we see that typical strains in the section are about 10–20% when the section is far into collapse; and *Figure 8* shows that, in the dynamic event, it takes about 0.2 seconds for this to occur. This implies average gross strain rates of about 1.0 per second in that period of the response. In typical piping steels such a strain rate might raise the yield stress 5–10% above its static value. This is not a large effect, so the mitigation of localization by rate effects is probably not a major aspect of this event. Again, a more precise assessment of this error would require a fully three-dimensional analysis. Overall it seems likely that this simple and computationally inexpensive two-stage approach to the problem is providing results that are sufficiently realistic to be used in design, although it would be most desirable to compare these results with physical experimental data or data from a full, detailed, three-dimensional analysis to support that statement. Finally, it should be noted that the section considered here is relatively thick ($R/t = 3.5$). In pipes with thin walls ($R/t > 20$) it is to be expected that the behavior will be affected strongly by internal fluid pressure in the pipe and by the interaction between axial and bending forces. Such thin-walled pipes could be modeled at relatively low cost by using ELBOW elements directly in the dynamic analysis instead of this two-stage approach. An additional concern with very thin pipes is that they are more likely to tear and leak, rather than choke the flow.

Input files

nonlindyncollapse_cpeg8r.inp	Static analysis of the elastic-plastic collapse of the pipe section using CPEG8R elements.
nonlindyncollapse_nonlingsect.inp	Dynamic analysis of the inelastic pipe whip response using nonlinear beam general section definitions for the axial and bending behaviors of the pipe.
nonlindyncollapse_cpeg4i.inp	Static analysis using element type CPEG4I.
nonlindyncollapse_cpeg4r.inp	Static analysis using element type CPEG4R.
nonlindyncollapse_cpeg4r_eh.inp	Static analysis using element type CPEG4R with enhanced hourglass control.
nonlindyncollapse_cpeg6m.inp	Static analysis using element type CPEG6M.
nonlindyncollapse_postoutput1.inp	**POST OUTPUT* analysis.
nonlindyncollapse_postoutput2.inp	**POST OUTPUT* analysis.

Figures



Geometry:

Initial outside radius = 38.05 mm
Initial thickness = 9.5 mm

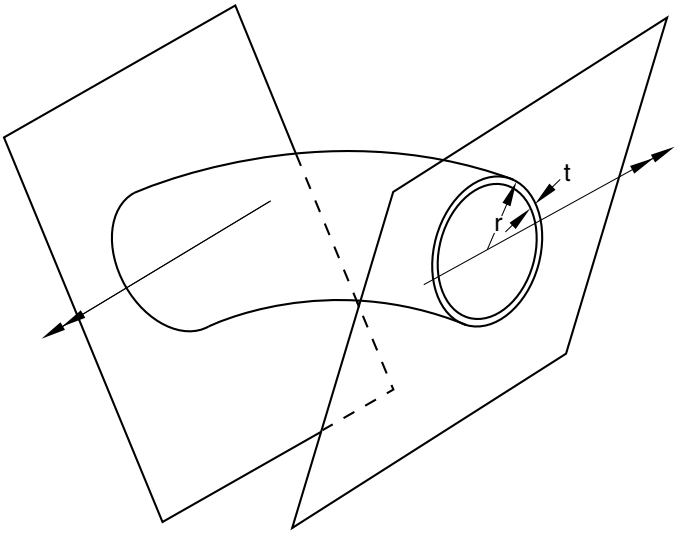
Material:

Young's modulus = 208 GPa
Poissons ratio = 0.3

Yield behavior:

$\sigma = 895\ \varepsilon_p^{0.26}$ MPa for $\sigma > 316$ MPa
ε_p = plastic strain

Figure 1: Elastic-plastic pipe subjected to rupture force.



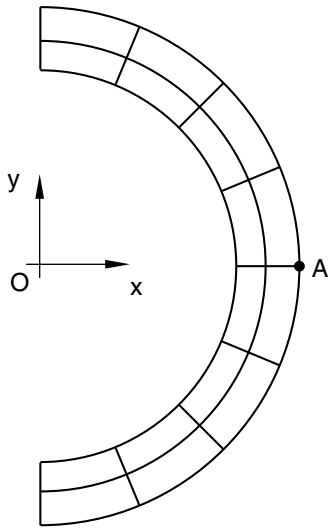


Figure 2: Initially straight pipe collapsing under pure bending; generalized plane strain model.

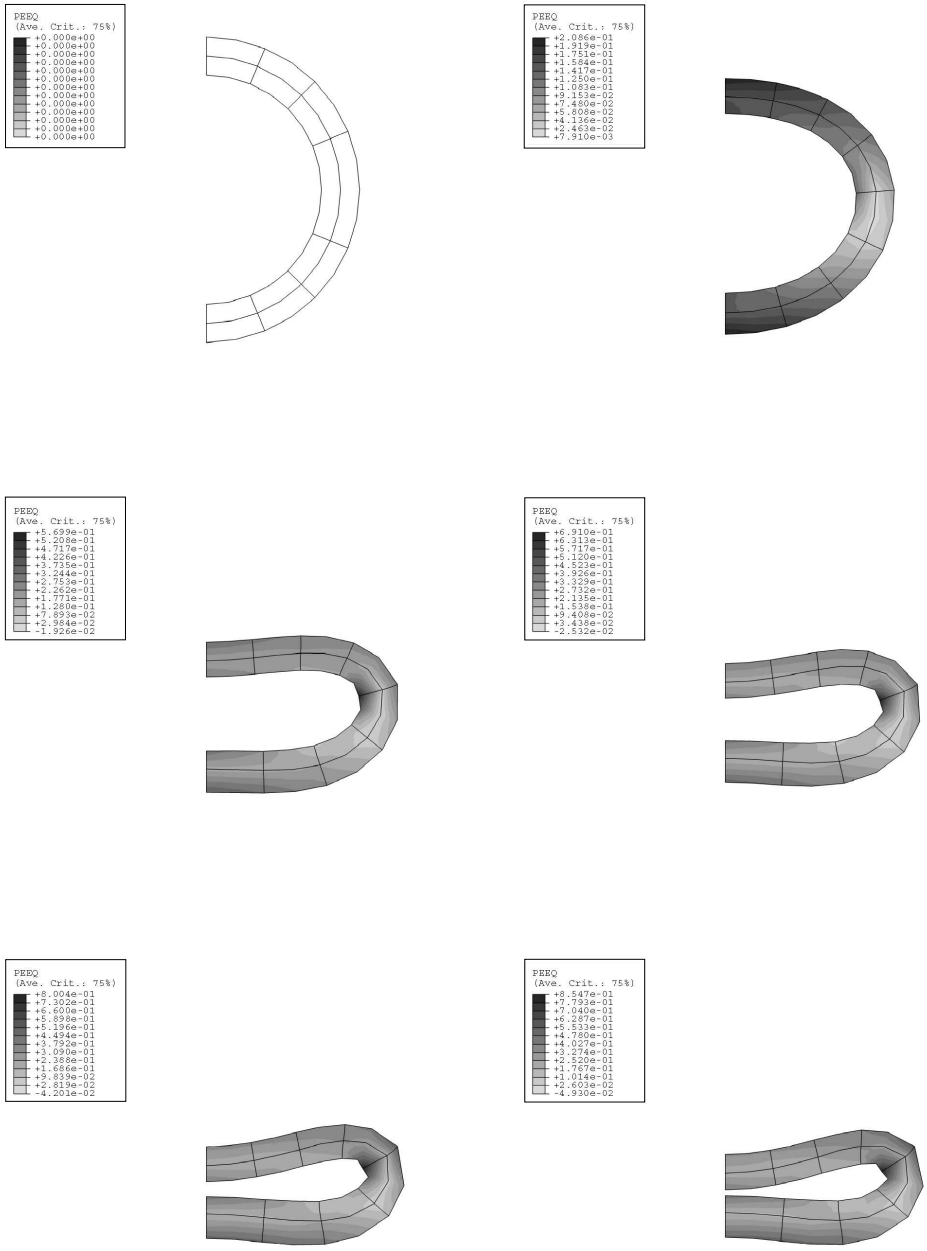


Figure 3: Equivalent plastic strain contours in collapsing pipe section, element type CPEG8R.

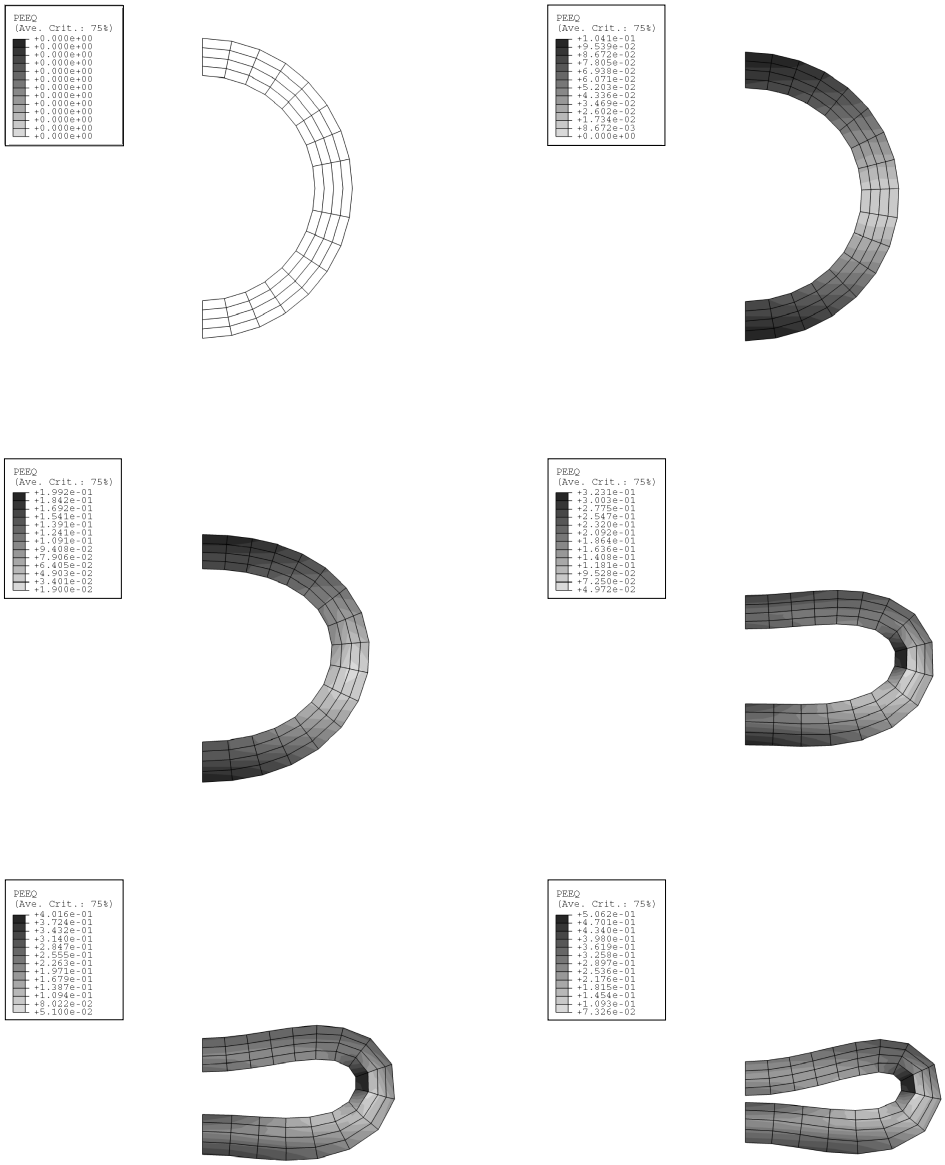


Figure 4: Equivalent plastic strain contours in collapsing pipe section, element type CPEG4R.

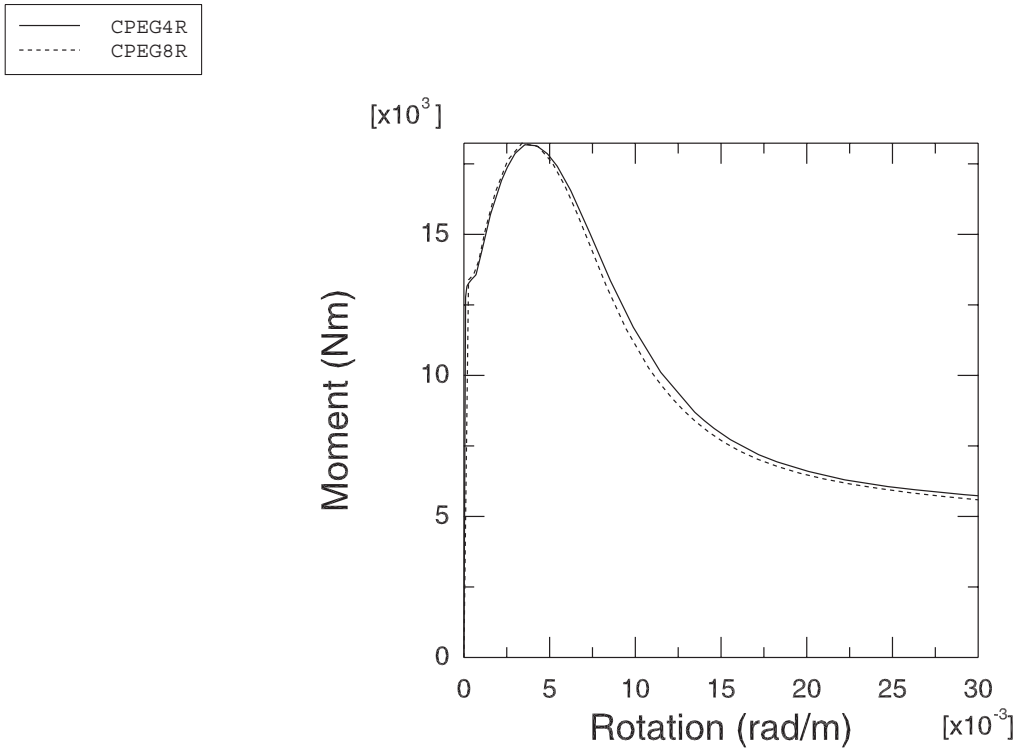


Figure 5: Moment-curvature response predicted for collapsing section under pure bending.

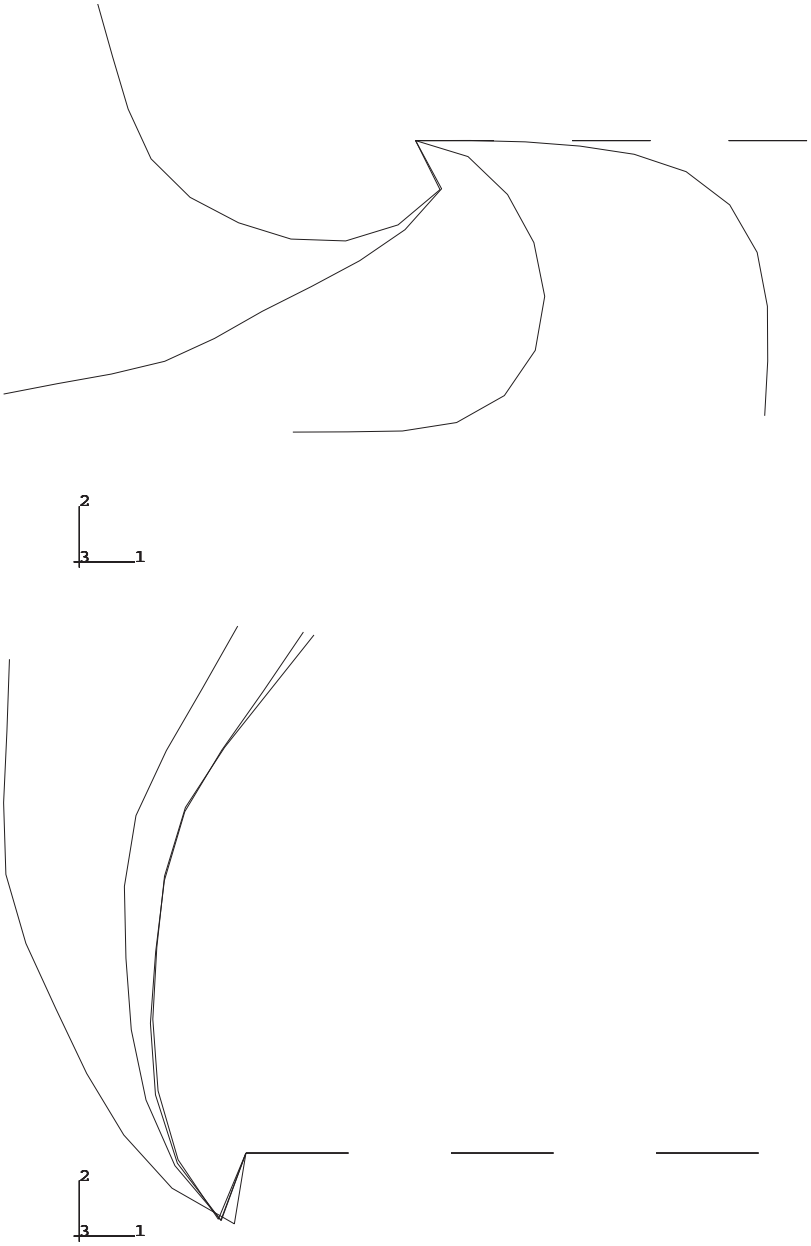


Figure 6: Displaced positions of pipe, every 20 increments. Initial increments in top figure, final increments in bottom figure.

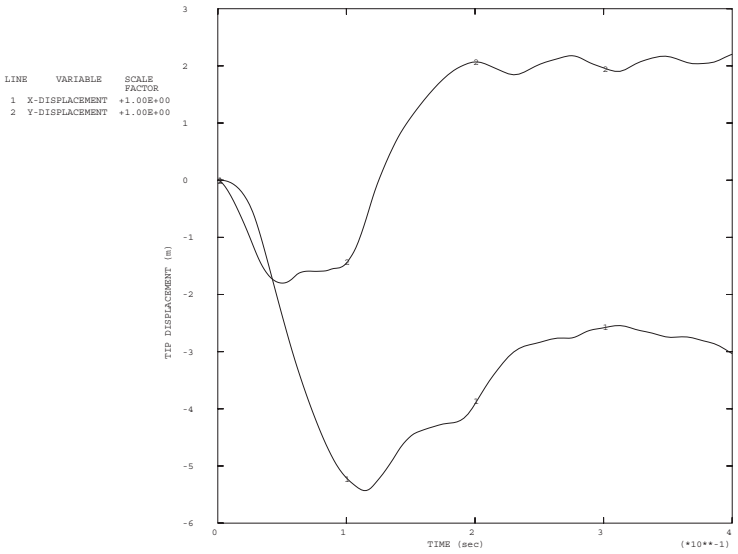


Figure 7: Tip displacement history.

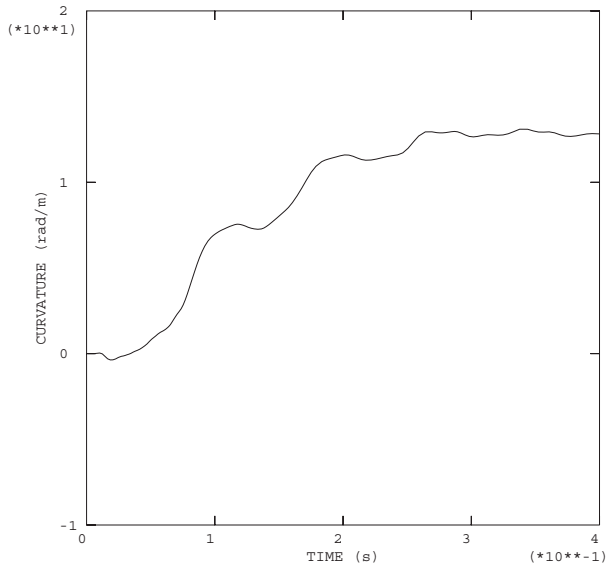


Figure 8: Curvature-time history for the element at the support.

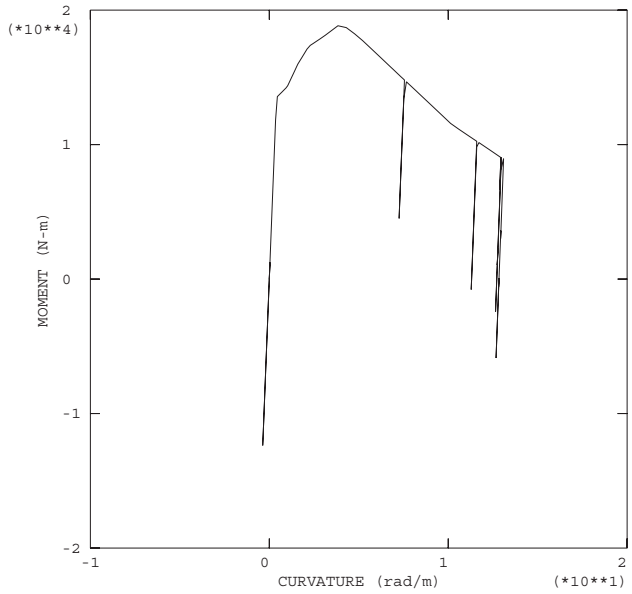


Figure 9: Moment versus curvature in the element at the support.

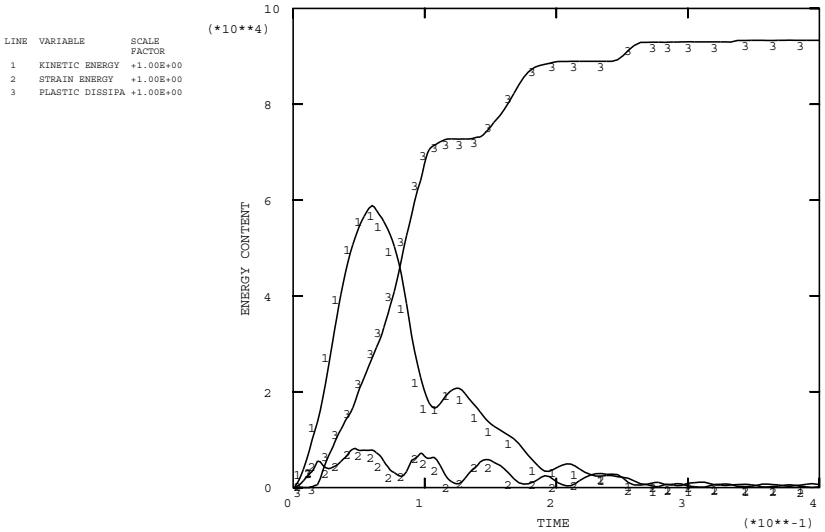


Figure 10: Energy history for the beam.

Detroit Edison pipe whip experiment

Products: Abaqus/Standard

Overview

This example models a simple, small-displacement pipe whip experiment conducted by the Detroit Edison Company. The problem conducted by the Detroit Edison Company involves rather small displacements but provides an interesting case because some (limited) experimental results are available. The results were reported by Esswein et al. (1978). It is a typical pipe whip restraint design case. It is a rather straightforward analysis because the restraint limits the motion and the geometry is so simple.

Geometry and model

The geometry and loading are shown in *Figure 1*. The pipe has a straight run of length 2.286 m (90 in), a very stiff elbow, and a cantilever "stick" 482.6 mm (19 in) long. A bursting diaphragm is installed at the end of the "stick" to initiate the blowdown. The restraint is a set of three U-bolts coupled together. The blowdown force history measured in the experiment is also shown in *Figure 1*. All dimensions, material properties, and this force history are taken from Esswein et al. (1978).

The horizontal pipe run is modeled with eight elements of type B23 (cubic interpolation beam with planar motion), and the stick is modeled with two elements of the same type. The elbow is treated as a fully rigid junction, so the node at the elbow is shared between the two branches. The bursting diaphragm structure is modeled as a lumped mass of 106.8 kg (0.61 lb s^2/in). The restraint is modeled as a single truss element. For the pipe the Young's modulus is 207 GPa (30×10^6 lb/in^2), the initial yield stress is 214 MPa (31020 lb/in^2), and the work hardening modulus is $846\varepsilon^{0.2}$ MPa ($122700\varepsilon^{0.2}$ lb/in^2) after yield. The restraint has an elastic stiffness of 131.35 MN/m (750000 lb/in), a yield force of 16681 N (3750 lb), and—when yielding—a force-displacement response $F = 2.2716\,\delta^{0.235}$ MN/m ($12971\delta^{0.235}$ lb/in). These values are taken from Esswein et al. (1978), where it is stated that they are based on measurements of static values with the stresses and forces increased by 50% in the plastic range to account for strain-rate effects. When it is known that strain-rate effects are important to the response it is preferable to model them directly, using a rate-dependent viscoplastic model. This has not been done in this case because the actual material is not specified.

Isotropic hardening is assumed for both the pipe and the restraint since the plastic flows are presumed to be in the large flow regime and not just incipient plasticity (where the Bauschinger effect can be important). The cross-section of the pipe is integrated with a seven-point Simpson rule: this should be of sufficient accuracy for this problem. Generally, in beam-like problems without repeated large magnitude excitation, a higher-order integration scheme would show only significantly different results at late times in the response, and then the differences are not too important in models of this rather unrefined level.

Esswein et al. (1978) provide the blowdown force-time history shown in *Figure 1*. This is applied as a point load at the end of the stick. In reality the fluid force during blowdown occurs at the piping elbows; but, since the displacements remain small, this detail is not important.

Solution control

Automatic time stepping is used, with an initial time increment of 100 μsec and the value of the half-increment residual tolerance, set to 4448 N (1000 lb). This value is based on actual force values expected (in this case, the blowdown force): HAFTOL is chosen to be about 10% of peak real forces. This should give good accuracy in the dynamic integration.

Results and discussion

The displacement of the node that hits the restraint is shown in *Figure 2*, and the force between the pipe and restraint is shown in *Figure 3*. Some experimental results from Esswein et al. (1978) are shown in *Figure 3*.

The analysis appears to predict the closure time and the peak force between the pipe and restraint quite well. However, the numerical solution (like the numerical solution given by Esswein et al., 1978) shows a slower force rise time than the experiment. A possible explanation may be the material model, where viscoplastic (strain-rate-dependent yield) effects have been modeled as enhanced yield values, as discussed above: this means that, at the high strain rate that occurs just after impact, the actual material can carry higher stresses than the model, and so will respond more stiffly. The oscillation in the gap force in *Figure 3* after the initial loading of the restraint is presumably caused by the difference in the basic natural frequencies of the restraint and the pipe: this oscillation is sufficiently severe to cause two slight separations.

Input files

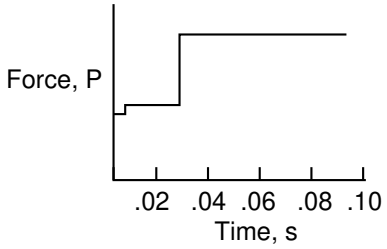
detroitedison.inp	Input data for this analysis.
detroitedison_postoutput.inp	**POST OUTPUT* analysis.

References

- Esswein, G., S. Levy, M. Triplet, G. Chan, and N. Varadavajan, Pipe Whip Dynamics, ASME Special Publication, 1978.

Figures

P		time
N	lb	s
32027	7200	0- 0.00045
32348	7272	0.00045 -0.0291
35230	7920	0.0291-



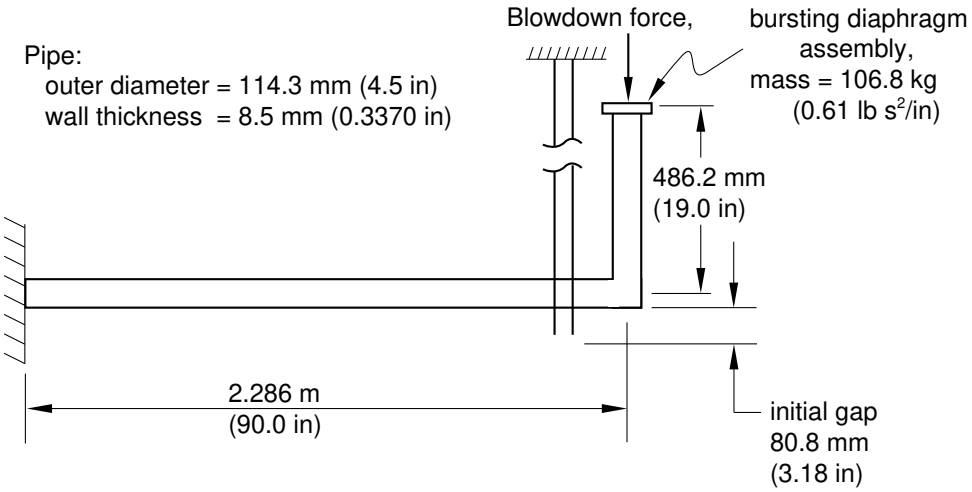


Figure 1: Detroit Edison experiment.

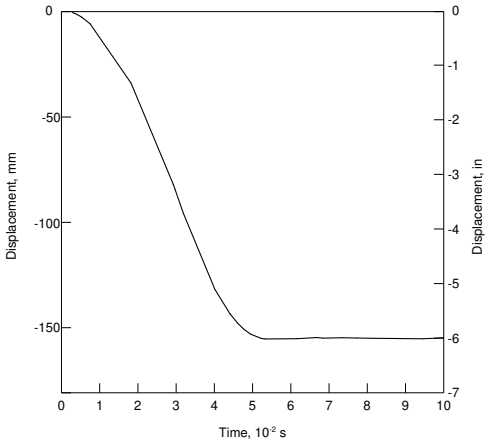


Figure 2: Displacement history at constrained end.

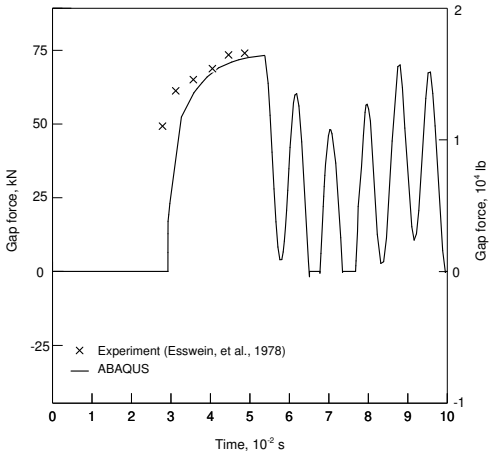


Figure 3: Gap force history.

Rigid projectile impacting eroding plate

Products: Abaqus/Explicit

Overview

This example illustrates impact, progressive failure, and the use of infinite elements.
The example simulates the oblique impact of a rigid spherical projectile onto a flat armor plate at a velocity of 1000 m/sec. A failure model is used for the plate, thus allowing the projectile to perforate the plate.

Problem description

The armor plate has a thickness of 10 mm and is assumed to be semi-infinite in size compared to the projectile. This is accomplished by using CIN3D8 infinite elements around the perimeter of the plate. The plate is modeled using 4480 C3D8R elements. The armor plate material has Young's modulus of 206.8 GPa, Poisson's ratio of 0.3, density of 7800 kg/m^3, yield stress of 1220 MPa, and a constant hardening slope of 1220 MPa. The material definition also includes a progressive failure model, which causes Abaqus/Explicit to remove elements from the mesh as they fail. Failure is assumed to occur at an equivalent plastic strain of 100%, at which point the element is removed from the model instantaneously. (The value of the failure strain is chosen somewhat arbitrarily; it is not intended to model any particular material.)

The sphere has a diameter of 20 mm and is assumed to be rigid, with a mass corresponding to a uniform material with a density of 37240 kg/m^3. The rotary inertia of the sphere is not needed in the model because we assume there is no friction between the sphere and the plate. Boundary conditions are applied to constrain the motion of the sphere in the y-direction. Two approaches for modeling the surface of the sphere are tested: using an analytical rigid surface and using R3D4 rigid elements. Analytical rigid surfaces are the preferred means for representing simple rigid geometries such as this in terms of both accuracy and computational performance. However, more complex three-dimensional surface geometries that occur in practice must be modeled with surfaces formed by element faces. Results for the faceted representations are presented here. The element formulation for the C3D8R elements is modified with the section controls. The advocated formulation for this problem uses the centroid kinematic formulation and a linear combination of stiffness and viscous hourglass control. Additional combinations of kinematic formulation and hourglass control are included for comparison.

Only half of the plate is modeled, using appropriate symmetry boundary conditions in the x–z plane. The model is shown in *Figure 1*. The complete sphere is modeled for visualization purposes. There are 17094 degrees of freedom in the model.

Since elements in the plate will fail and be removed from the model, nodes in the interior of the plate will be exposed to contact with the surface of the rigid sphere. Thus, contact must be modeled between the surface of the sphere, defined as an element-based surface, and a node-based surface that contains all of the nodes in the plate within a radius of 20 mm of the point of impact. (See *Eroding projectile impacting eroding plate* for an example in which element-based surfaces are used to model erosion.) In the primary input files contact pairs are used to define contact between the surface of the sphere and any of the nodes contained in the node set. Input files that use the general contact algorithm are also provided.

Results and discussion

The spherical projectile impacts the plate at 1000 m/sec at an angle of 30° to the normal to the plate. Deformed shapes at different stages of the analysis are shown in *Figure 2* through *Figure 4* for the centroid kinematic and the combined (viscous-stiffness form) hourglass section controls (analysis case pl3d_erode_ccs). Early in the analysis, shown in *Figure 2*, a relatively small amount of material has been eroded from the surface of the plate and the plate is still deforming under the sphere. In *Figure 3* the plate has been perforated and the projectile is

still in contact with the edge of the hole. In *Figure 4* the projectile has exited the plate and is moving away with a constant velocity. *Figure 5* and *Figure 6* show the history of the projectile's velocity (*Table 1* shows the analysis options used to obtain these results). The results show close agreement.

In *Figure 2* through *Figure 4* the failed elements have been eliminated by creating a display group in Abaqus/CAE that contains only the active elements.

Input files

pl3d_erode_ccs.inp	Model using the CENTROID kinematic and COMBINED hourglass section control options.
pl3d_erode_ccs_gcont.inp	Model using the CENTROID kinematic and COMBINED hourglass section control options and the general contact capability.
pl3d_erode_ces.inp	Model using the CENTROID kinematic and ENHANCED hourglass section control options.
pl3d_erode_ces_gcont.inp	Model using the CENTROID kinematic and ENHANCED hourglass section control options and the general contact capability.
sphere_n.inp	External file referenced in this input.
sphere_e.inp	External file referenced in this input.
pl3d_erode.inp	Model using the default section controls.
pl3d_erode_gcont.inp	Model using the default section controls and the general contact capability.
pl3d_erode_ale.inp	Model using the default section controls and the **ADAPTIVE MESH* option.
pl3d_erode_ocs.inp	Model using the ORTHOGONAL kinematic and COMBINED hourglass section control options.
pl3d_erode_ocs_gcont.inp	Model using the ORTHOGONAL kinematic and COMBINED hourglass section control options and the general contact capability.
pl3d_erode_oes.inp	Model using the ORTHOGONAL kinematic and ENHANCED hourglass section control options.
pl3d_erode_oes_gcont.inp	Model using the ORTHOGONAL kinematic and ENHANCED hourglass section control options and the general contact capability.
pl3d_erode_anl.inp	Model using an analytical rigid surface and the default section controls.

Tables

Table 1: Analysis section controls tested.

Analysis File	Relative CPU Time	Section Controls	
		Kinematic Formulation	Hourglass Control
pl3d_erode	1.0	average strain	integral viscoelastic
pl3d_erode_ocs	0.86	orthogonal	combined
pl3d_erode_oes	0.88	orthogonal	enhanced
pl3d_erode_ccs	0.73	centroid	combined
pl3d_erode_ces	0.75	centroid	enhanced

Figures

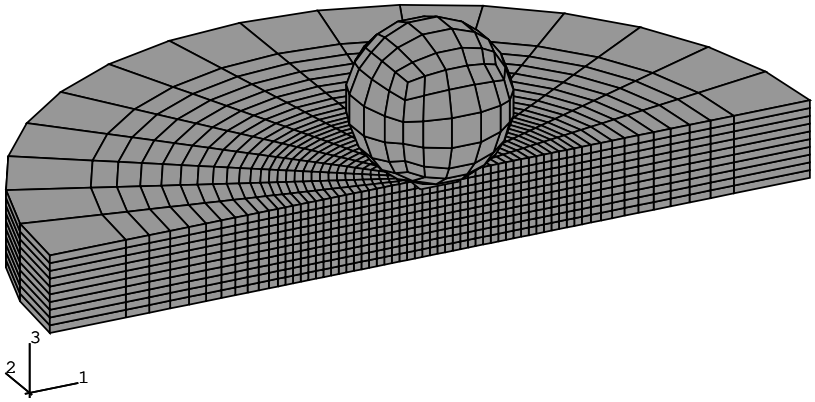


Figure 1: Undeformed mesh.

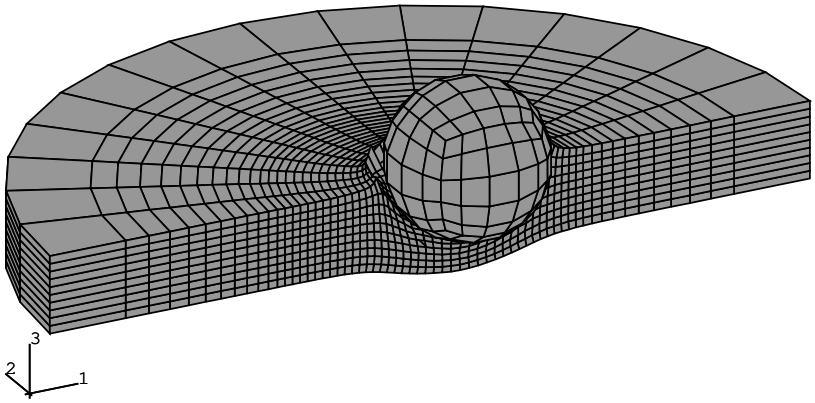


Figure 2: Deformed shape at 10 microseconds (analysis using the centroid kinematic and combined hourglass section controls).

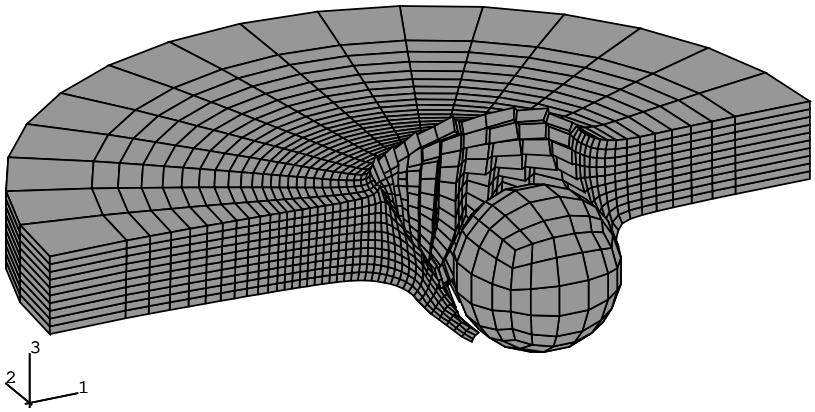


Figure 3: Deformed shape at 30 microseconds (analysis using the centroid kinematic and combined hourglass section controls).

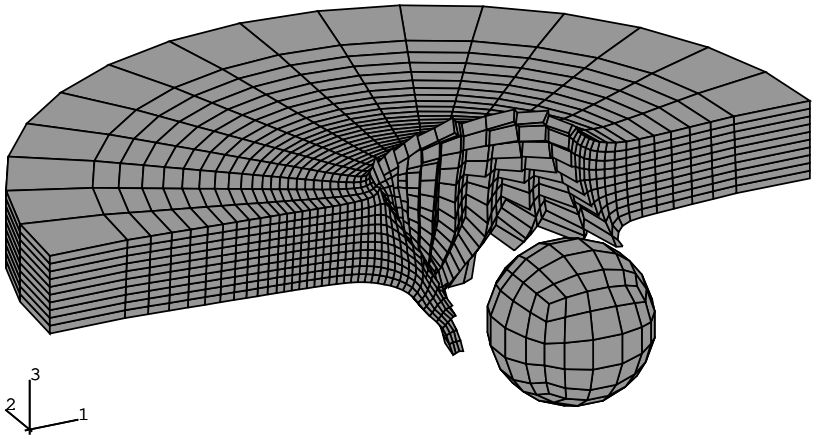


Figure 4: Deformed shape at 40 microseconds (analysis using the centroid kinematic and combined hourglass section controls).

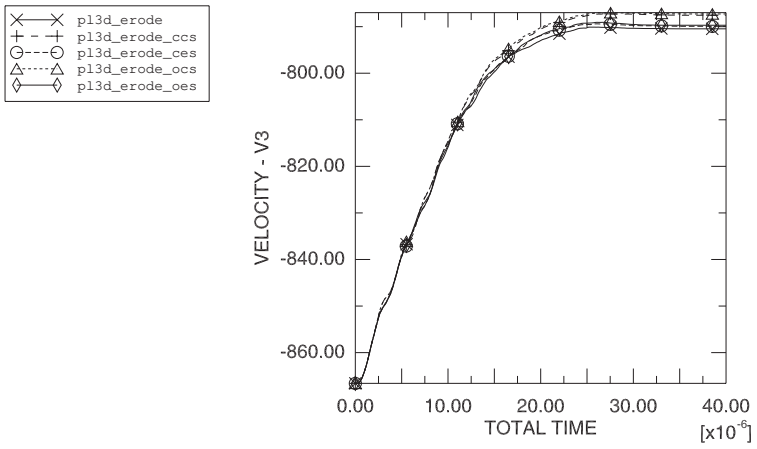


Figure 5: Vertical component of the projectile velocity.

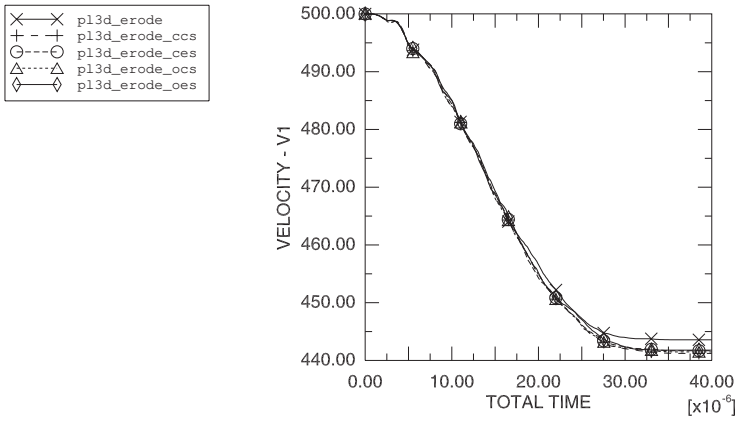


Figure 6: Horizontal component of the projectile velocity.

Eroding projectile impacting eroding plate

Products: Abaqus/Explicit

Overview

This example demonstrates the ability of the general contact algorithm to model surface erosion on multiple contacting bodies during high-speed impact.
The example simulates the oblique impact of a cylindrical projectile onto a flat armor plate at a velocity of 2000 m/sec. The same material model, which includes a failure model with progressive damage, is used for both the projectile and plate.

Problem description

The undeformed mesh is shown in *Figure 1*. The armor plate has a thickness of 3 mm. A relatively small rectangular region of the plate is modeled for simplicity, with fully fixed boundary conditions specified on three cutting planes and *y*-axis symmetry specified on one cutting plane. The projectile, which is 10 mm in length and has a radius of 1 mm, has an initial speed of 2000 m/sec. The cylindrical axis of the projectile is 20° from perpendicular to the plate, and the initial velocity of the projectile is aligned with its cylindrical axis. Half of the projectile is modeled, with *y*-axis symmetry specified on the cutting plane. The plate and projectile material properties are identical, with Young's modulus of 210 GPa, Poisson's ratio of 0.3, and density of 7800 kg/m^3. The yield stress of the material is specified as a function of the equivalent plastic strain at different equivalent plastic strain rates. The material definition also includes failure models with progressive damage, which causes Abaqus/Explicit to remove elements from the mesh as they fail. Both the ductile and shear initiation criteria are used: the ductile criterion is specified in terms of the plastic strain at the onset of damage as a tabular function of the stress triaxiality; the shear criterion is specified in terms of the plastic strain at the onset of damage as a tabular function of the shear stress ratio. The damage evolution energy is assumed to be 500 N/m.

During the analysis elements from both bodies are deleted due to material failure, which calls for the use of element-based surfaces that can adapt to the newly exposed surfaces of the remaining active elements. The general contact algorithm supports element-based surfaces that evolve in this manner (whereas the contact pair algorithm does not) with two distinct approaches. In the first approach all surface faces that may become exposed during the analysis, including faces that are originally in the interior of bodies, must be included at the beginning of the analysis. Only the interior faces that are expected to participate in contact are included in the contact domain in this analysis to minimize the memory use (including interior faces for all elements in the model would more than double the memory use). In the second approach the surface faces are generated and activated dynamically only when interior elements are exposed. This results in a possible memory savings compared to the first approach.

By default, the general contact algorithm does not include nodal erosion, so contact nodes will still take part in the contact calculations even after all of the surrounding elements have failed. These nodes act as free-floating point masses that can experience contact with the active contact faces. For comparison purposes, the analysis is also conducted including nodal erosion, which causes the nodes to be removed from the contact calculations once all surrounding elements have failed (and can result in computational savings). The momentum transfer associated with free-flying nodes is expected to be significant in this example, so nodal erosion is not recommended.

A plane strain version of the model is also included to demonstrate contact with eroding bodies for a two-dimensional simulation. Consistent boundary conditions, initial conditions, in-plane dimensions, and material properties are used for this model and the three-dimensional model. Other geometric characteristics are inconsistent between the two-dimensional and three-dimensional models, so results are expected to differ.

Results and discussion

Deformed shapes at different stages of the analysis are shown in *Figure 2* through *Figure 5* (only active elements are shown in these figures). As shown in *Figure 5*, the projectile eventually perforates the plate, with approximately the leading half of the projectile elements failing during the analysis. Some broken-off fragments with active elements can be observed in *Figure 5*. The nodes and exposed faces of such fragments can take part in contact. Nodes no longer attached to any active elements can take part in contact only in the analysis without nodal erosion (which corresponds to the primary input file). *Figure 6* compares total kinetic energy histories for the analyses conducted with and without nodal erosion, respectively. Approximately 32% of the initial kinetic energy is absorbed by the impact for the model without nodal erosion; whereas approximately 26% of the initial kinetic energy is absorbed for the model with nodal erosion.

Deformed shapes at different stages of the two-dimensional analysis are shown in *Figure 7*.

Input files

erode_material.inp	External file referenced in this input (material definition).
erode_proj_and_plate.inp	Model of impact of eroding projectile into eroding plate without nodal erosion.
erode_proj_and_plate2.inp	Model of impact of eroding projectile into eroding plate with nodal erosion.
erode_proj_and_plate2-dynamic.inp	Model of impact of eroding projectile into eroding plate with nodal erosion and with the contact surface generated dynamically.
erode_proj_and_plate2_2d.inp	Model of impact of two-dimensional eroding projectile into eroding plate with nodal erosion.

Figures

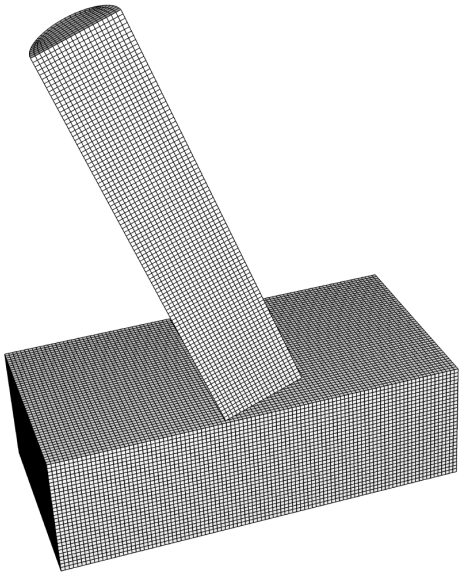


Figure 1: Undeformed mesh.

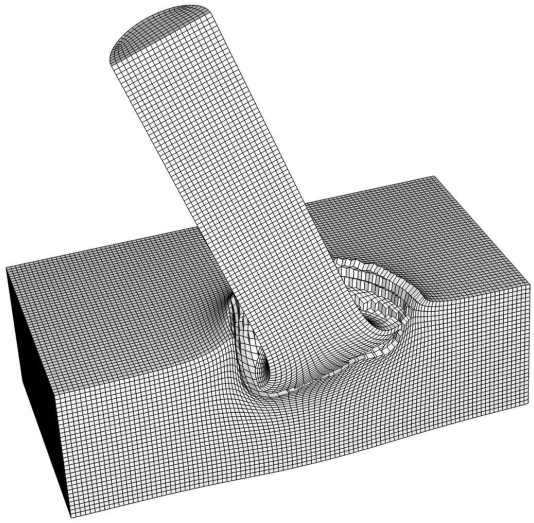


Figure 2: Deformed shape at 1.5 microseconds.

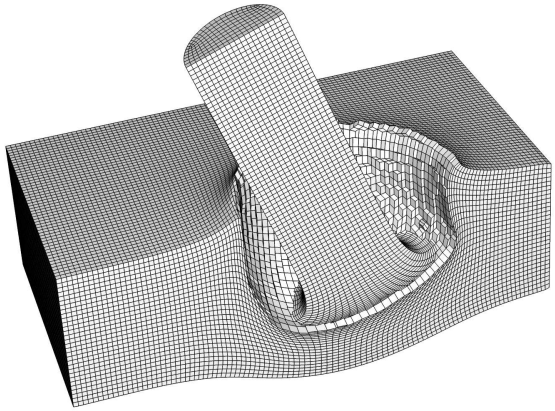


Figure 3: Deformed shape at 3 microseconds.

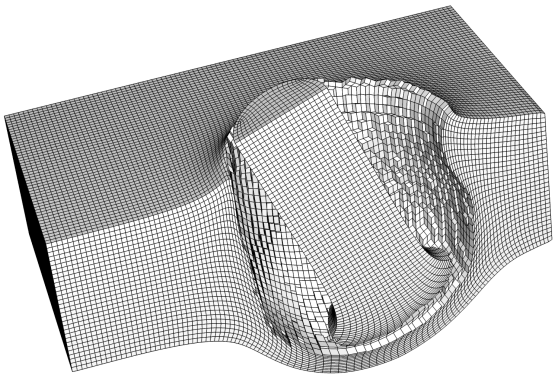


Figure 4: Deformed shape at 4.5 microseconds.

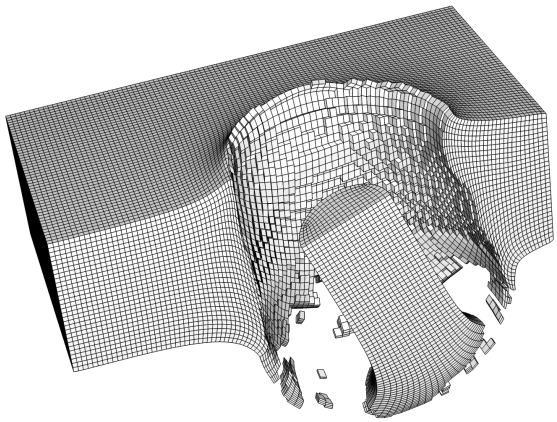


Figure 5: Deformed shape at 6 microseconds.

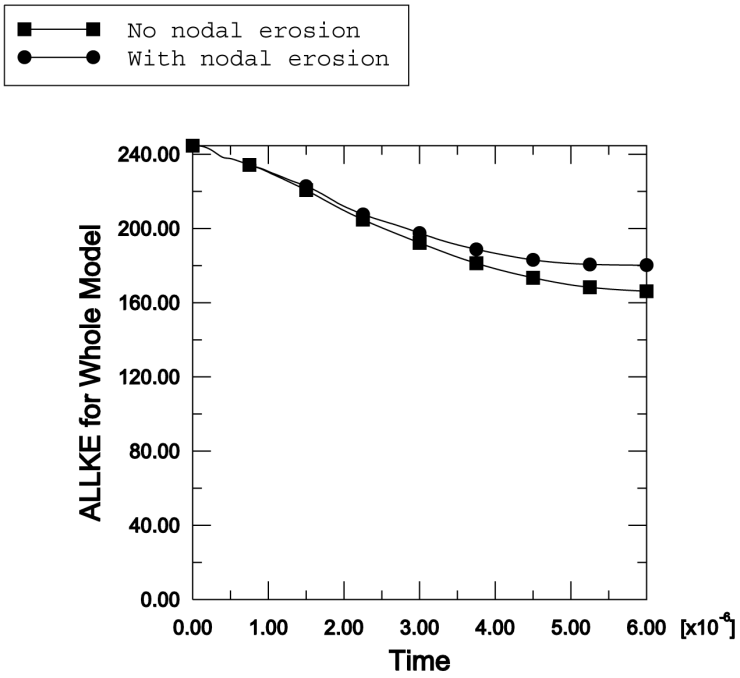


Figure 6: Kinetic energy history.

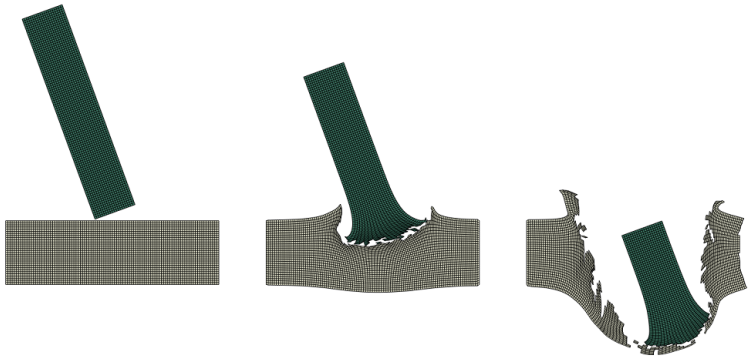


Figure 7: Two-dimensional eroding projectile impacting eroding plate.

Tennis racket and ball

Products: Abaqus/Explicit

Overview

This example illustrates contact between a deforming surface and a node set, the definition of initial stresses, and modeling of a fluid cavity filled with a compressible gas using the surface-based fluid cavity capability.
The example simulates the oblique impact of a tennis ball onto a racket at 6.706 m/sec (264 in/sec).

Problem description

The strings on the tennis racket are modeled using T3D2 truss elements. They are assumed to be linear elastic, with Young's modulus of 6.895 GPa (1.0×10^6 psi), Poisson's ratio of 0.3, and density of 1143 kg/m^3 (1.07×10^{-4} lb sec^{2}in^{-4}). The strings are under an initial tension of 44.48 N (10 lb), which is specified using initial stress conditions.

The frame is assumed to be rigid and is modeled using R3D4 elements. The nodes of the strings (truss elements) around the perimeter are the same nodes as those used for the R3D4 elements. The reference node for the rigid frame has boundary conditions applied to constrain all six degrees of freedom on the rigid body so that the frame does not move.

The tennis ball is modeled as a sphere, using 150 S4R shell elements. It is assumed to be made of rubber, modeled with general hyperelastic material properties as a Mooney-Rivlin material with the constants $C_{10} =$ 0.690 MPa (100 lb/in^2) and $C_{01} =$ 0.173 MPa (25 lb/in^2). Abaqus/Explicit requires some compressibility for hyperelastic materials. In the results shown here, $D_1 =$ 0.0145 MPa^{-1} (10^{-4} psi^{-1}). This gives an initial bulk modulus ($K_0 = 2/D_1$) that is 80 times the initial shear modulus $2\left(C_{10} + C_{01}\right)$. This ratio is lower than the ratio for typical rubbers, but the results are not particularly sensitive to this value in this case because the rubber is unconfined. A more accurate representation of the material's compressibility would be needed if the rubber were confined by stiffer adjacent components or reinforcement. Decreasing D_1 by an order of magnitude (thus increasing the initial bulk modulus by a factor of 10) has little effect on the overall results but causes a reduction in the stable time increment by a factor of $\sqrt{10}$ due to the increase in the bulk modulus. The density of the tennis ball is 1068 kg/m^3 (1.07×10^{-4} lb sec^{2}in^{-4}).

The tennis ball is under an initial internal pressure of 41 kPa (6 psi) in addition to the ambient atmospheric pressure of 100 kPa (14.7 psi). An element-based surface is defined on the inside of the tennis ball. This surface is used to define a fluid cavity filled with gas. The properties of the gas inside the tennis ball, molecular weight and molar heat capacity, are defined as part of the fluid behavior of a fluid cavity. The molecular weight and molar heat capacity of the gas are arbitrarily chosen as 0.062 kg (0.1367 lb) and 28.110 J/kg °K (112.847 lb in/lbm °K). Since the ball is impermeable to gas, the pressure of the gas will rise when the volume of the ball decreases, and vice versa. Static equilibrium gives the value of the initial biaxial membrane stresses in the shell elements of the sphere as $pr/2t =$ 155 kPa (22.5 psi) to balance the internal pressure (here p is the internal gas pressure, r is the radius of the sphere, and t is the tennis ball thickness). This initial state of stress in the ball is defined using initial stress conditions.

A coefficient of friction of 0.1 is specified between the ball and the strings. The ball impacts on the strings at 6.706 m/sec (264 in/sec) at an angle of 15°.

No attempt has been made to generate an accurate model of the ball and strings: the model parameters are chosen simply to provide a “soft” ball relative to the strings to illustrate contact effects.

The complete model is shown in *Figure 1*. There are 2241 degrees of freedom in the model.

An element-based surface is defined on the tennis ball. Since the truss elements are line elements, they do not form a planar surface. A node-based surface is defined that contains all the nodes of the strings. Contact pairs are then used to define contact between the element-based surface of the ball and any of the nodes defined in the node-based surface. An input file that uses the general contact algorithm is also provided.

Results and discussion

Figure 2 shows the position of the ball with respect to the strings in the undeformed configuration. The deformed shapes at different stages of the analysis are shown in *Figure 3* through *Figure 7*. *Figure 8* shows a time history of the energies for the model. These include the total internal energy (ALLIE), the kinetic energy (ALLKE), the viscous dissipation (ALLVD), the energy dissipated by friction (ALLFD), the external work (ALLWK), and the total energy balance for the model (ETOTAL). The total energy is seen to remain almost constant during the analysis, as it should. *Figure 9* and *Figure 10* give the history of pressure inside the ball and the history of the actual volume of the ball. It can be seen that both the gas pressure inside the ball and the ball volume stabilize after 10 msec.

Input files

tennis_surfcav.inp	Analysis using the contact pair approach.
tennis_gcont_surfcav.inp	Analysis using the general contact capability.
tennis_ef1.inp	External file referenced in all analyses.
tennis_ef2.inp	External file referenced in all analyses.

Figures

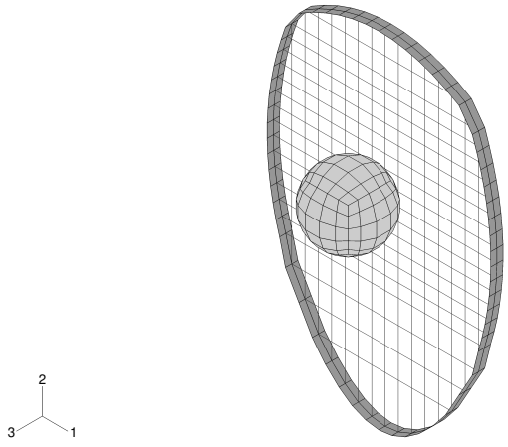


Figure 1: Undeformed mesh.

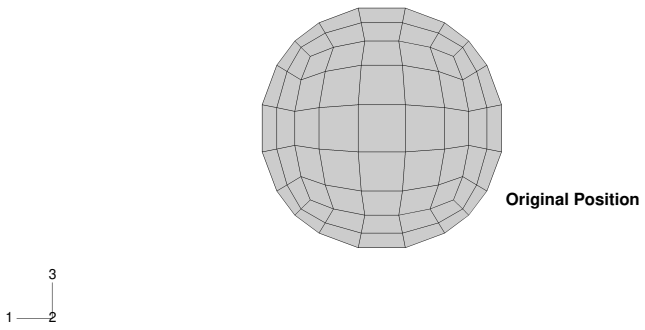


Figure 2: Original position of ball and strings.

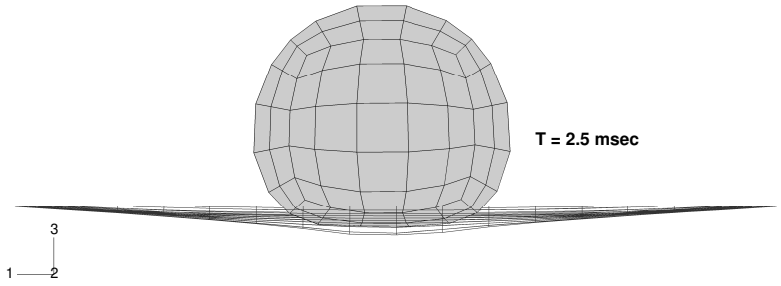


Figure 3: Deformed shape at 2.5 milliseconds.

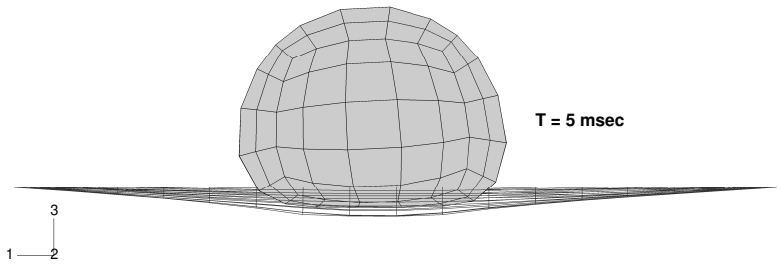


Figure 4: Deformed shape at 5 milliseconds.

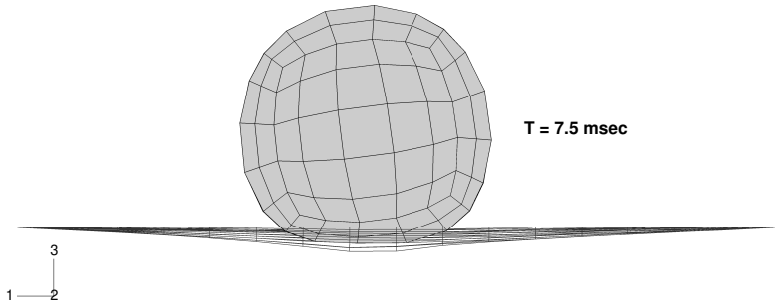


Figure 5: Deformed shape at 7.5 milliseconds.

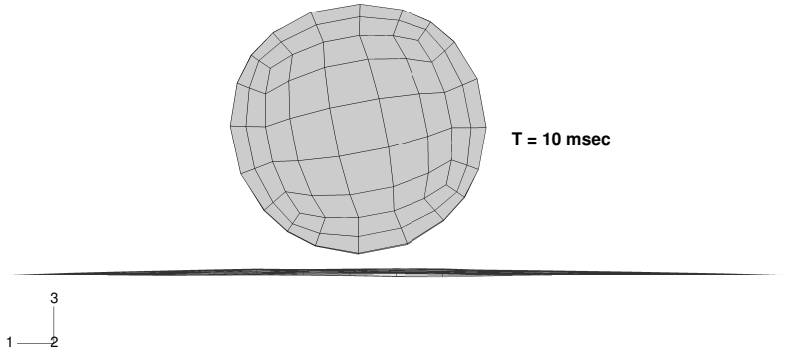


Figure 6: Deformed shape at 10 milliseconds.

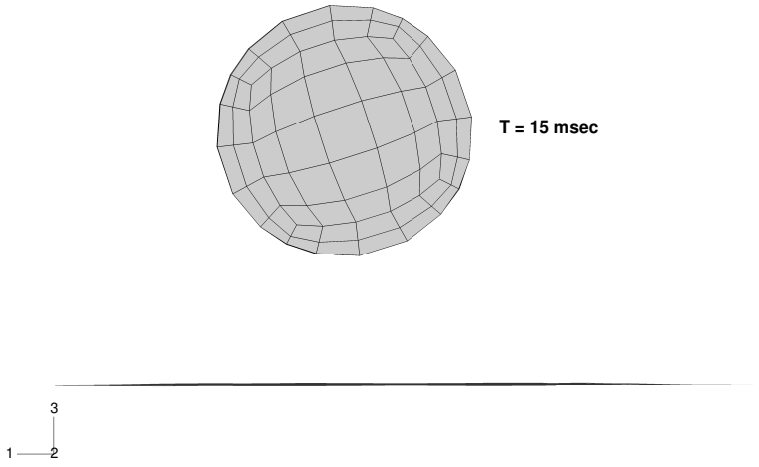


Figure 7: Deformed shape at 15 milliseconds.

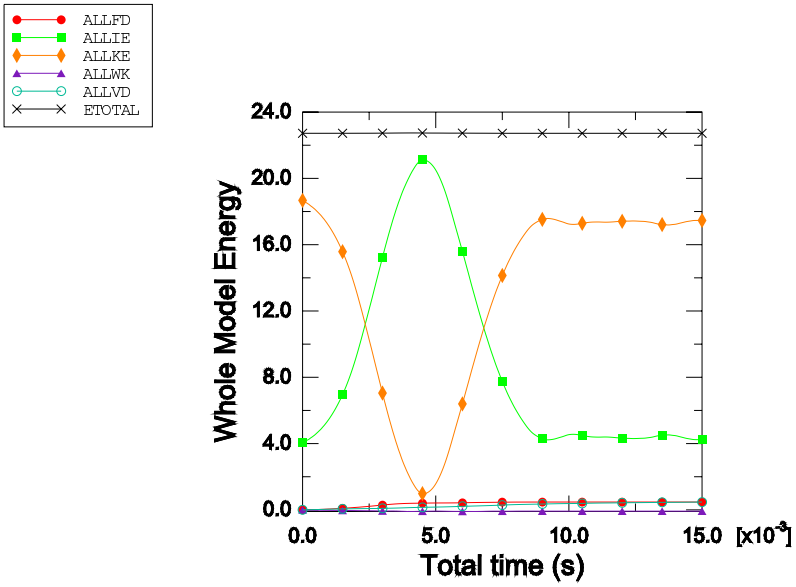


Figure 8: Energy histories.

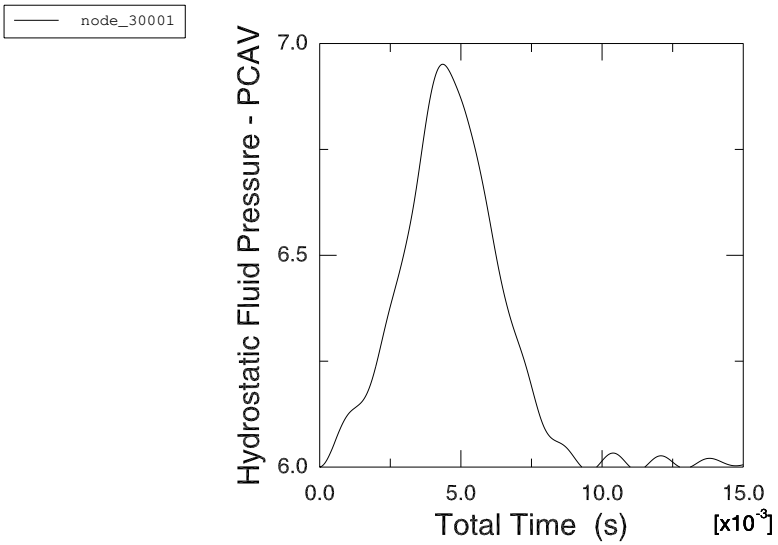


Figure 9: History of the gas pressure inside the tennis ball.

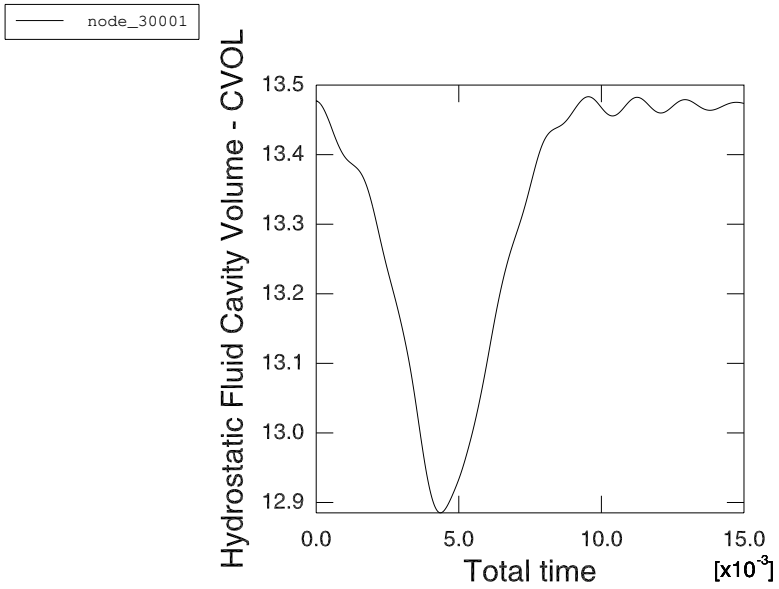


Figure 10: History of the ball volume.

Pressurized fuel tank with variable shell thickness

Products: Abaqus/Standard

Overview

This problem demonstrates the variable shell thickness capability in Abaqus.
The example is based on an analysis conducted by SOLVAY RESEARCH & TECHNOLOGY of a blow-molded, plastic fuel tank with dimensions similar to those considered here.

Geometry and model

The mesh shown in *Figure 1* is used in this example to model a fuel tank and its support straps. The mesh uses 2812 3-node shell elements (S3R), with the support straps modeled with 32 2-node beam elements (B31). Depending on the desired accuracy and detail of the solution, the analyst may identify some regions of the mesh in which additional refinement, or second-order elements, would be appropriate. The fuel tank would fit within a box of dimensions 450 mm × 200 mm × 680 mm. An internal pressure of 7×10^{-3} MPa is applied statically to the tank.

Analyses are conducted for a uniform shell thickness of 5 mm and for a spatially varying shell thickness in the range 1.38 mm to 9.35 mm (see *Figure 2*), which is a more accurate representation of the tank. The uniform thickness analysis provides a comparison to judge the effects of variable thickness. The overall volume of plastic modeled in the variable thickness analysis is about 93% of that in the uniform thickness analysis. For the variable thickness analysis the shell cross-section indicates that the shell thickness is to be interpolated from nodal values specified with nodal thickness. For elements with more than one integration point, this approach results in a thickness that can vary over the element.

The materials are modeled as isotropic elastic. The plastic fuel tank has a Young's modulus of 0.6 GPa and a Poisson's ratio of 0.3. The steel support straps have a Young's modulus of 206.8 GPa and a Poisson's ratio of 0.29. Geometrically nonlinear effects are significant in this example, so geometric nonlinearity is included in the step.

Results and discussion

Contour plots of the von Mises stress at the inner surface of the fuel tank (section point 1 in the shell elements) for the variable shell thickness and uniform shell thickness analyses are shown in *Figure 3* and *Figure 4*. The ratio of the maximum von Mises stress found in the variable thickness analysis to that found in the uniform thickness analysis is 1.5. For the variable thickness analysis, the maximum von Mises stress occurs at a location where the fuel tank skin is relatively thin (see *Figure 2* and *Figure 3*).

Examination of the y-component of displacement shows that the overall expansion of the tank in the y-direction is about 1.5% greater in the variable thickness analysis.

Input files

pressfueltank_variablethick.inp	Example using variable shell thickness.
pressfueltank_uniformthick.inp	Example using uniform shell thickness.
pressfueltank_node.inp	Nodal coordinate data for both models.
pressfueltank_shellelement.inp	Shell element connectivity data for both models.
pressfueltank_beamelement.inp	Beam element connectivity data for both models.

pressfueltank_shellthickness.inp Shell thickness data for the variable shell thickness model.

References

- SOLVAY RESEARCH & TECHNOLOGY, Plastic Processing Department, Rue de Ransbeek, 310, B-1120 Brussels, Belgium.

Figures

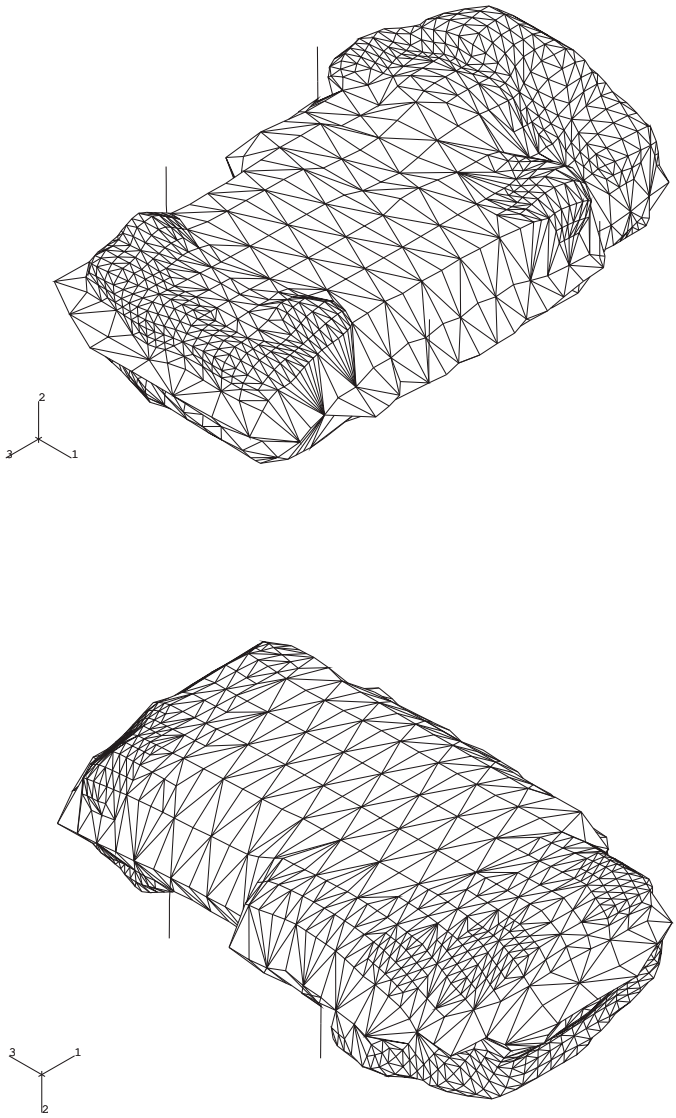


Figure 1: Fuel tank mesh with S3R and B31 elements.

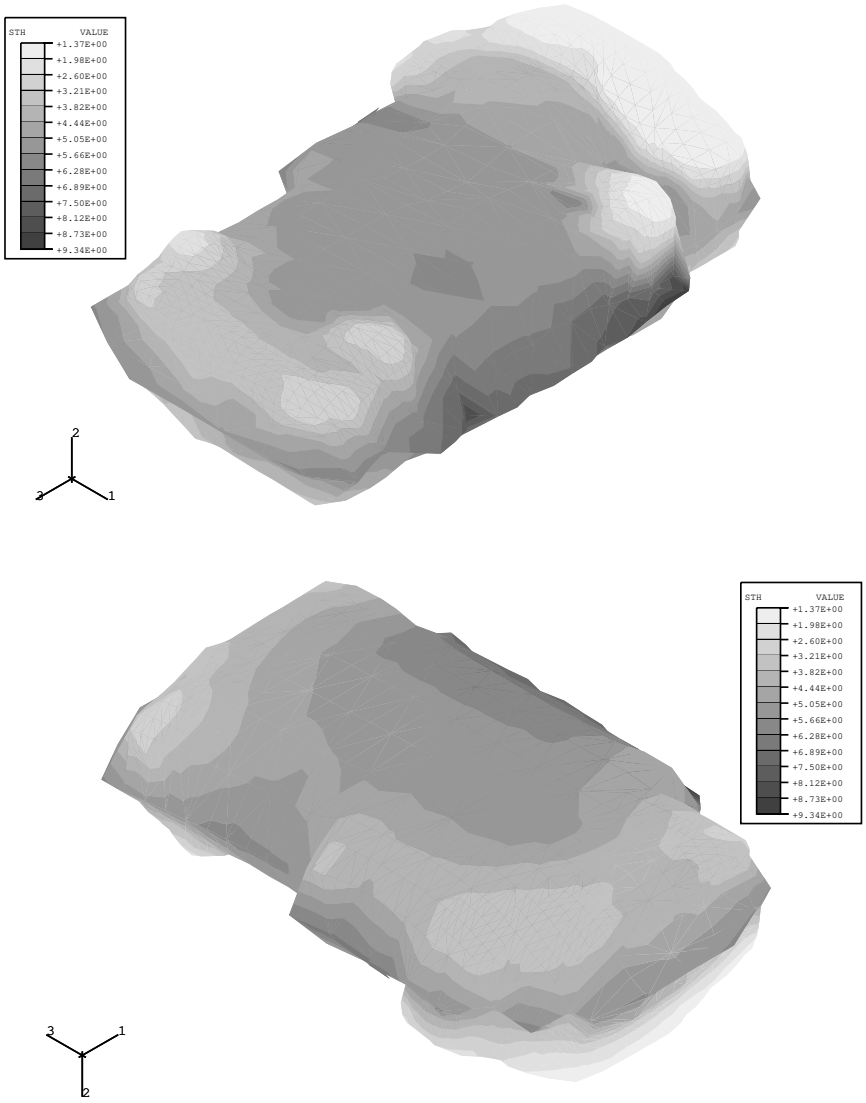


Figure 2: Shell thickness for variable thickness analysis.

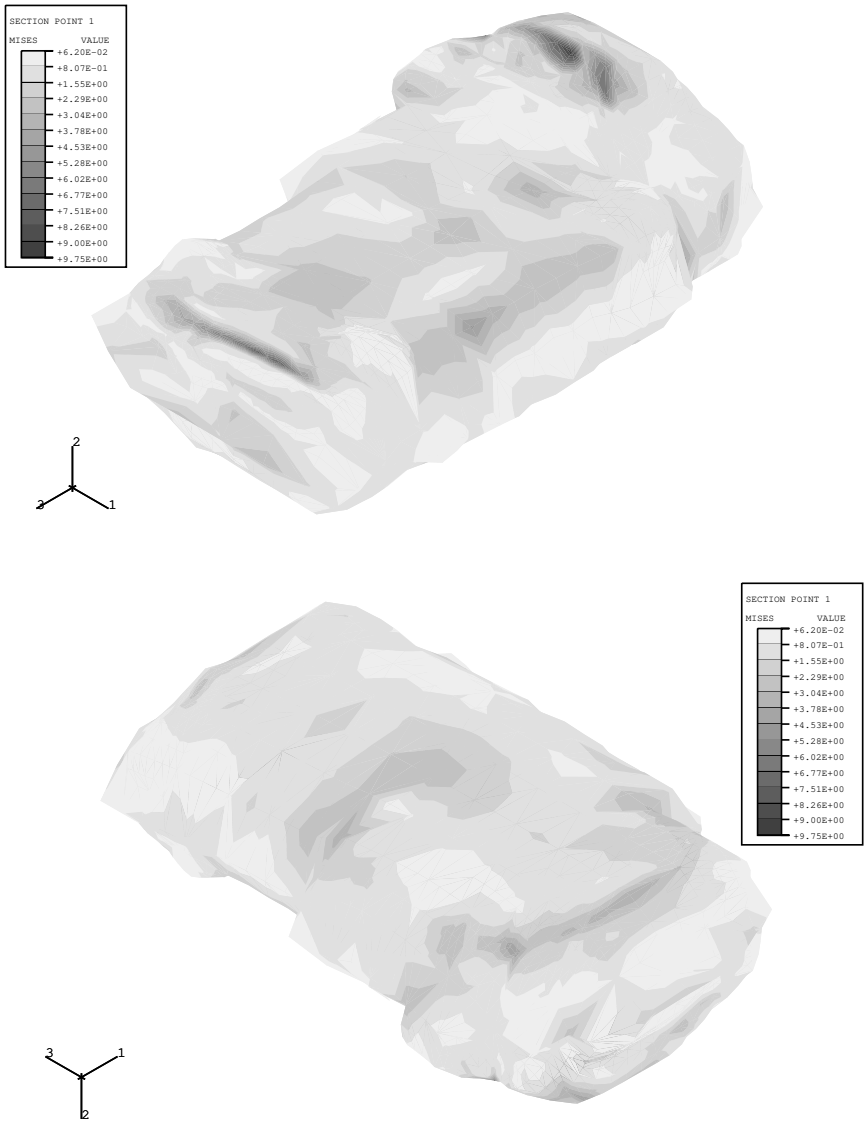


Figure 3: Von Mises stress solution for variable shell thickness analysis.

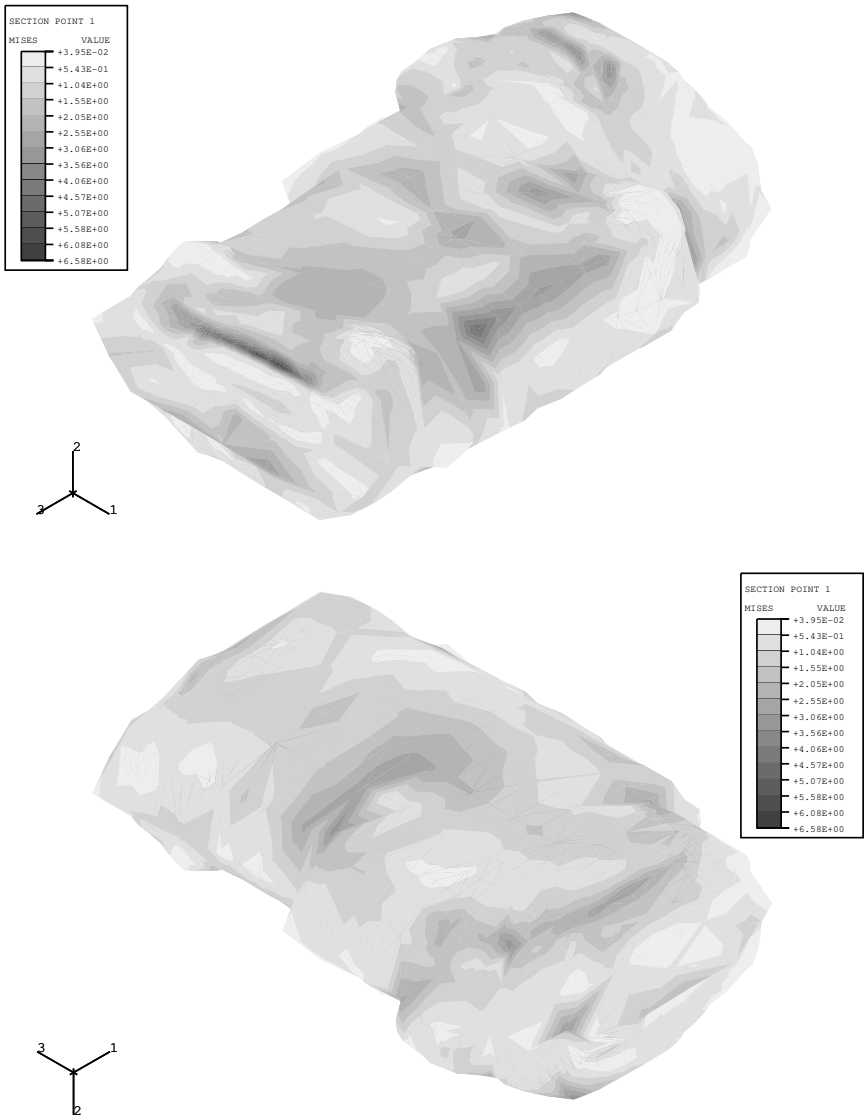


Figure 4: Von Mises stress solution for uniform shell thickness analysis.

Modeling of an automobile suspension

Products: Abaqus/Standard

Overview

This example verifies the accuracy of JOINTC elements and connector elements in a structure undergoing rigid rotation motions.
A secondary objective of this example is to demonstrate the use of equivalent rigid body motion output variables in Abaqus/Standard.

JOINTC elements (*Flexible Joint Element*) can be used to model the interaction between two nodes that are almost coincident geometrically and that represent a joint that has internal stiffness and/or damping. The behavior of the joint is defined in a local coordinate system (*Orientations*). This system rotates with the motion of the first node of the element and may consist of linear or nonlinear springs and dashpots arranged in parallel, coupling the corresponding components of relative displacement and of relative rotation in the joint. This feature can be used to model, for example, a rubber bearing in a car suspension.

In the connector element model the JOINTC elements are replaced by connector elements (see *Connector Elements*) with connection types CARTESIAN to define the translational behavior and ROTATION to define the rotational behavior. These connection types allow linear or nonlinear spring and dashpot behavior to be defined in a local coordinate system that rotates with the first node on the element. Several different connection types can be used to model the finite rotational response. See *Connection Types* for connection types using different finite rotation parametrizations. In this model the rotation magnitudes are assumed small. Hence, a rotation vector parametrization of the joint using ROTATION is appropriate.

Geometry and model

The structure analyzed is an automobile's left front suspension subassembly (see *Figure 1*). The physical components included in the assemblage are the tire, the wheel, the axle (hub), the A-arm (wishbone), the coil spring, and the frame. The tire is modeled with a JOINTC or connector element; a curved bar element has been attached for visualization. (The JOINTC or connector element is used because it is a convenient way of defining the tire's nonlinear stiffness in a local coordinate system.) The vertical stiffness of the wheel is represented by a beam element. The axle and A-arm are both modeled with beam elements. The axle is connected to the A-arm by a pin-type MPC in the JOINTC model or with connection type JOIN in the connector model. The coil spring is modeled by a SPRINGA element in the JOINTC model or with connection type AXIAL in the connector model. The automobile frame is represented by a MASS element. The top of the coil spring is connected directly to the frame, while the A-arm is connected to the frame by two JOINTC elements or two connector elements with connection types CARTESIAN and ROTATION (representing the A-arm bushings). The initial position represents a fully weighted vehicle, and the tire and coil spring have a corresponding initial preload.

The first step in the analysis allows the suspension system to reach equilibrium. The second step models the tire moving over a bump in the road. The bump is idealized as a triangular shape 100 mm high by 400 mm long, and the vehicle is assumed to travel at 5 km/hr. A second input file is used to show the effects of large rotation on the suspension response. This file includes an initial rotation step, which rigidly rotates the model by 90° about the vertical axis but is otherwise identical to the first input file. We expect the response from the two analyses to be the same.

In this example we are primarily interested in the equivalent rigid body motion of the A-arm and, in particular, in the average displacement and rotation.

Results and discussion

The results from the connector element models are qualitatively and quantitatively equivalent to the JOINTC models. Hence, only the JOINTC results are discussed further.

The vertical displacement histories in Step 2 are shown in *Figure 2* for the contact point of the tire with the ground, the wheel center, and the frame. *Figure 3* shows a series of overlaid displaced plots as the tire rolls up the bump.

To verify the behavior of the JOINTC elements with large rotations, the second model rigidly rotates the entire structure by 90° before applying the bump excitation. *Figure 4* shows the displacement time histories from the two models overlaid on the same plot. They are nearly identical.

In the analysis of deformable bodies undergoing large motions it is convenient to obtain information about the equivalent rigid body motions: average displacement and rotation, as well as linear and angular momentum about the center of mass. For this purpose Abaqus provides a set of equivalent rigid body output variables. As indicated above, a secondary objective of this example is to demonstrate the use of these output variables, which represent the average motion of the specified element set. Element output requests to the results file and to the data file are included. If no element set is specified, the average motion of the entire model is given. This type of output can only be requested in a direct-integration implicit dynamic analysis, and only elements that have a mass will contribute to the equivalent rigid body motion. For a precise definition of the equivalent rigid body motion of a deformable body, see *Equivalent rigid body dynamic motion*.

Figure 5 and *Figure 6* have been generated using the equivalent rigid body output variables. *Figure 5* shows the vertical motion of node 5001 (bearing point “A” of the A-arm), node 5080 (point of A-arm nearest the tire), and the average vertical motion of the A-arm (output variable UC3). As expected, the displacement of the center of mass of the component lies between the displacements of its two ends. *Figure 6* shows the average rigid body rotation of the A-arm component about its center of mass. Rigid body rotations are available about the three global axes. Here we are interested in the rotation about the global *X*-axis (output variable URC1). The standard output for rotations is radians, but the results have been scaled to plot the rotation in degrees.

Input files

jointcautosuspension.inp	Suspension analysis with JOINTC elements.
jointcautosuspension_rotated.inp	Rotated suspension analysis with JOINTC elements. This input file includes one extra (rotation) step but is otherwise identical to jointcautosuspension.inp.
jointcautosuspension_depend.inp	Identical to jointcautosuspension.inp, except that field-variable-dependent linear and nonlinear spring properties are used in the JOINTC elements.
connautosuspension.inp	Suspension analysis with connector elements.
connautosuspension_rotated.inp	Rotated suspension analysis with connector elements. This input file includes one extra (rotation) step but is otherwise identical to connautosuspension.inp.
connautosuspension_depend.inp	Identical to connautosuspension.inp, except that field-variable-dependent linear and nonlinear spring properties are used in the connector elements with connection types CARTESIAN, ROTATION, and AXIAL.

Figures

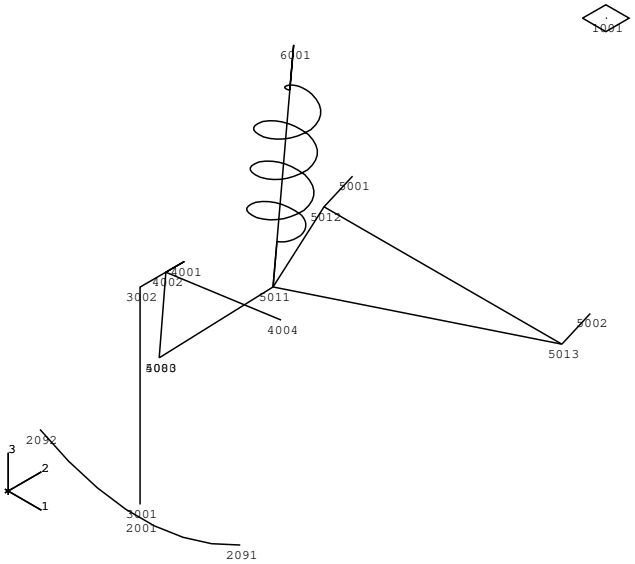


Figure 1: Left front automobile suspension.

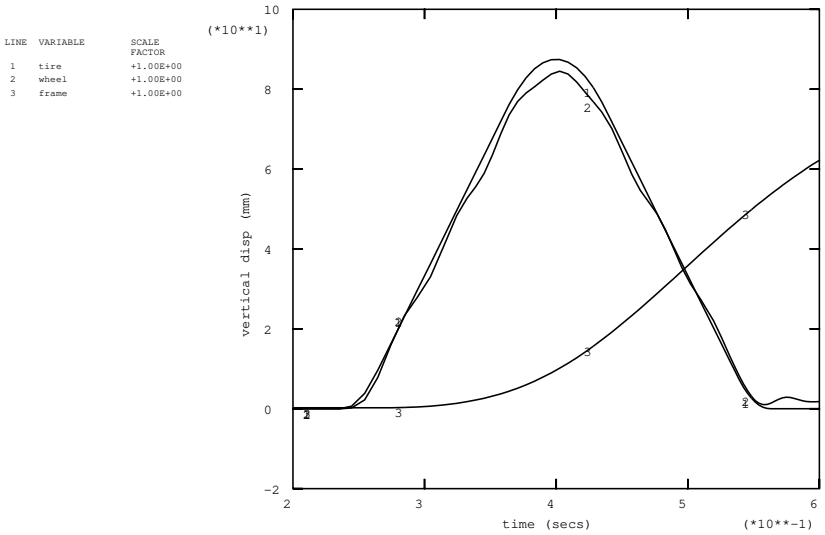


Figure 2: Displacement histories of tire, wheel center, and frame.

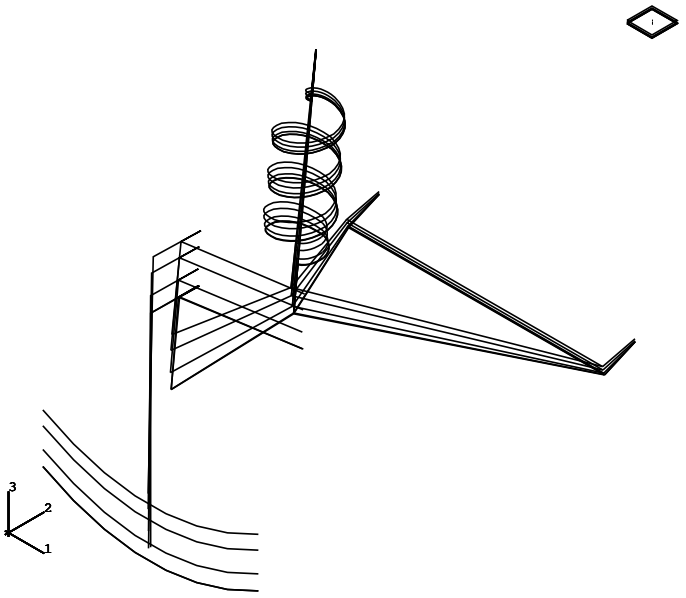


Figure 3: Displaced shapes during positive vertical tire motion.

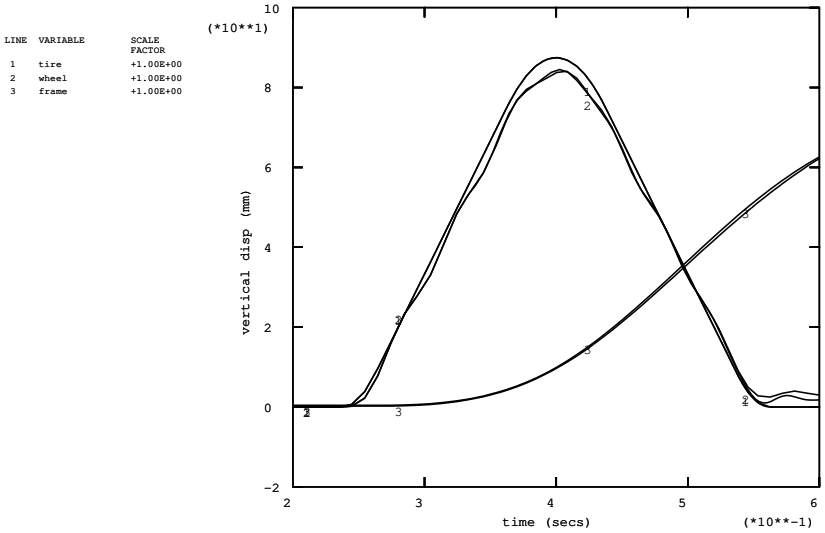


Figure 4: Overlay of unrotated and rotated suspension analyses.

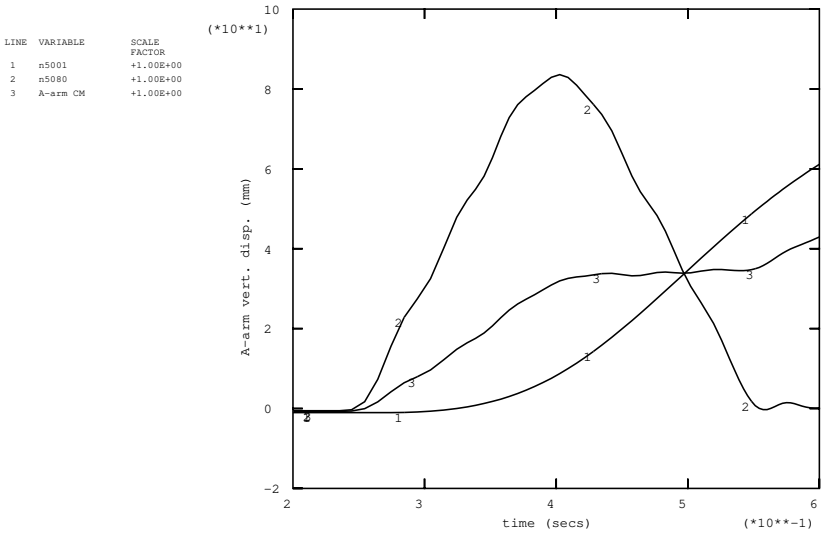


Figure 5: Vertical motion of the A-arm: average and nodal motions.

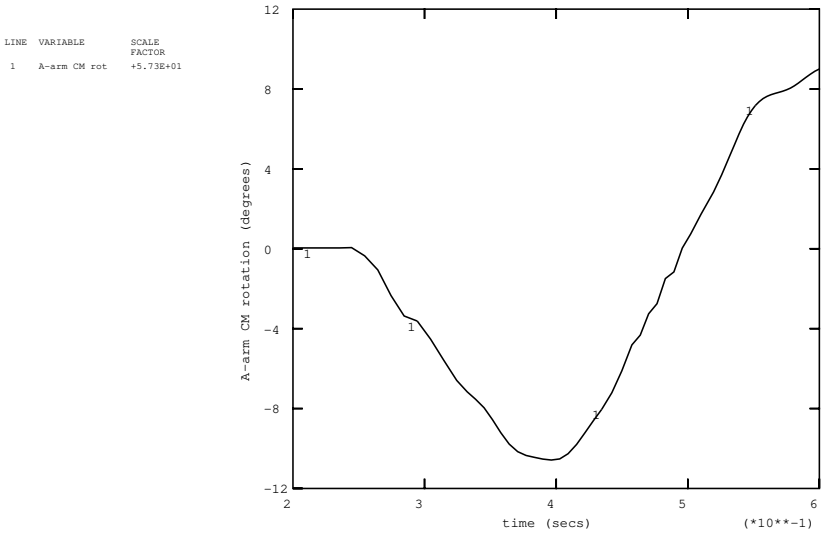


Figure 6: Average rotation of the A-arm about its center of mass.

Explosive pipe closure

Products: Abaqus/Explicit

Overview

This problem shows the following concepts: large deformation kinematics, equations of state, elastic-plastic material, transformations, detonation points.

Problem description

The units used in this analysis are referred to as c.g.μsec. Using these units, length is given in centimeters (cm), mass in grams (gm), and time is measured in microseconds (μsec). The stresses have units of mega bars (M bar). These units are commonly used in shock wave physics applications because the pressures tend to have values on the order of unity.

In this example problem two concentric pipes have the annulus between them filled with high explosive (HE). The inside radius of the inner pipe is 10 mm. The inside radius of the outer pipe is 20 mm. Both pipes are steel with a wall thickness of 2 mm. Each pipe is modeled with 6 elements in the radial direction, while the HE is modeled with 24 elements in the radial direction.

The steel pipe is an elastic, perfectly plastic material with Young's modulus of 221.1 GPa (2.211 M bar), Poisson's ratio of 0.279, yield strength of 430 MPa (.0043 M bar), and density of 7846 kg/m^3 (7.846 gm/cm^3).

The explosive material is modeled using the JWL equation of state with detonation wave speed = 7596 m/sec (.7596 cm/microsecond), A = 520.6 GPa (5.206 M bar), B = 5.3 GPa (0.053 M bar), R1 = 4.1, R2 = 1.2, ω = .35, density = 1900 kg/m^3 (1.9 gm/cm^3), and initial specific energy = 3.63 M joule/kg (0.0363 T erg/gm). The tension cutoff pressure is assumed to be zero and is specified using a tensile failure model. Refer to *Equation of State* for a description of this material model.

The explosive material is detonated at four points around the circumference of the cylinder. Because of the symmetry in this problem, only one-eighth of the pipe is modeled. *Figure 1* shows the original geometry and the location of the detonation point for the model. A transformed coordinate system is used to define the symmetry conditions along the sloping boundary. The interface between the explosive material and the steel is modeled using the no separation relationship for contact that allows for relative slip without separation between the two materials.

This analysis is run in two steps to reduce the amount of output written to the output database file. In the early part of the analysis, the deformations are not of much interest. Hence, the first step has a duration of 6 μsec. After 6 μsec the deformations are becoming significant. The second step has a duration of 1.5 μsec.

This analysis is run as both a two-dimensional case using CPE4R elements and as a three-dimensional case using C3D8R elements. In the three-dimensional case the displacements are constrained to be zero in the out-of-plane direction.

Results and discussion

Figure 2 through *Figure 5* show a sequence of the deformed shapes computed by Abaqus/Explicit for the two-dimensional case. The undeformed configuration is shown superimposed on the deformed shapes. Although not shown here, the results of the three-dimensional analysis are indistinguishable from those of the two-dimensional analysis.

This problem tests the features listed, but it does not provide independent verification of them.

Input files

eoscyl2d.inp	Two-dimensional case.
eoscyl3d.inp	Three-dimensional case.

Figures

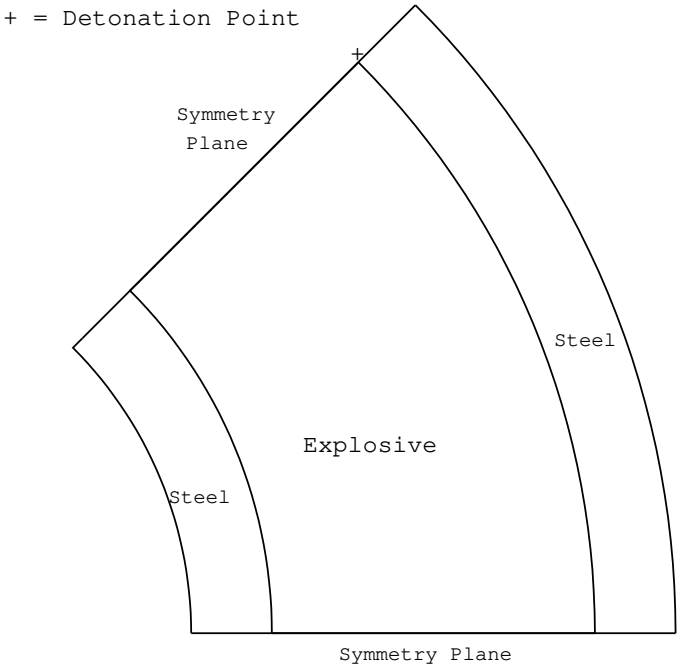


Figure 1: Original geometry.

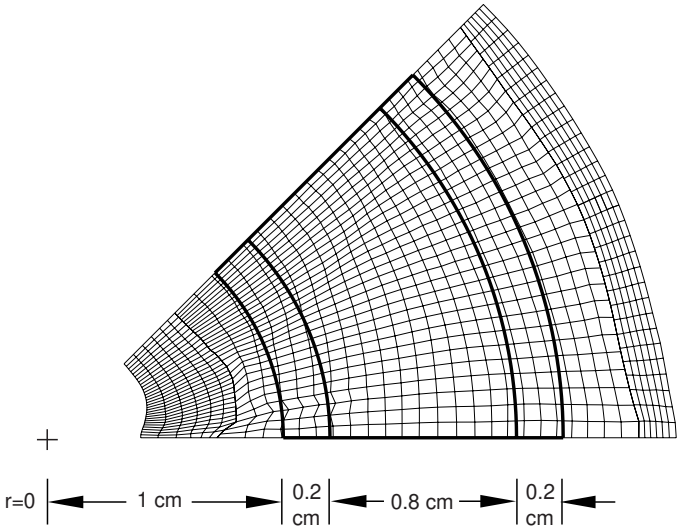


Figure 2: Deformed configuration after 6.0 μsec with undeformed configuration superimposed.

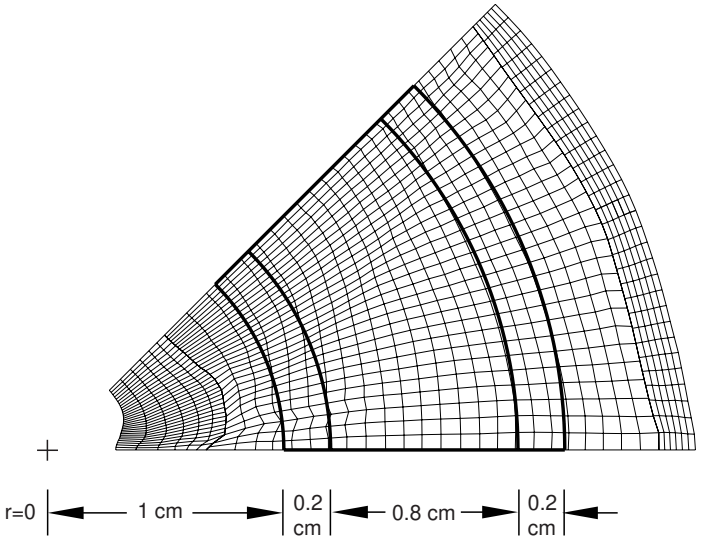


Figure 3: Deformed configuration after 6.5 μsec with undeformed configuration superimposed.

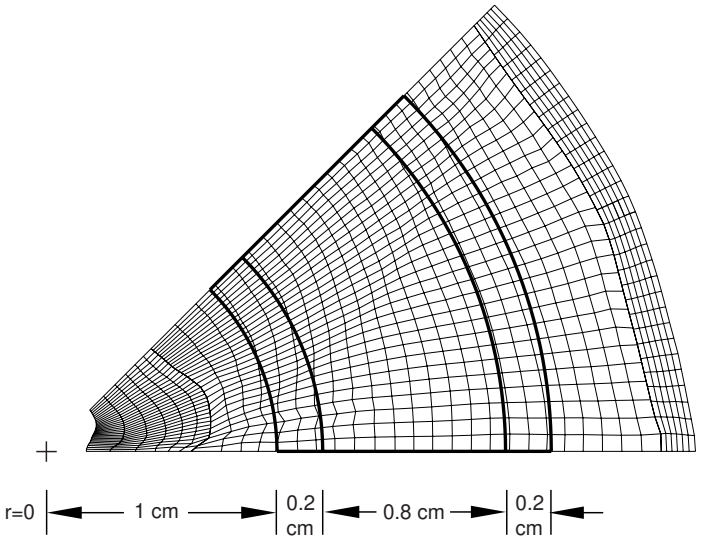


Figure 4: Deformed configuration after 7.0 μsec with undeformed configuration superimposed.

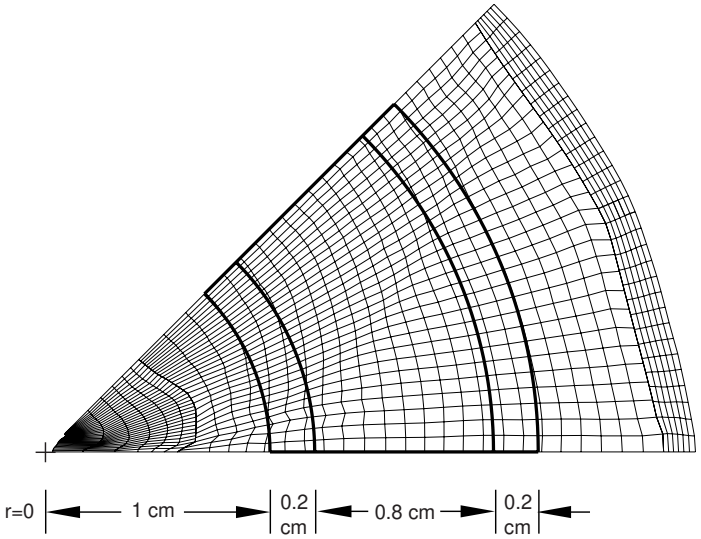


Figure 5: Deformed configuration after 7.5 μsec with undeformed configuration superimposed.

Knee bolster impact with general contact

Products: Abaqus/Explicit

Overview

This example illustrates the use of the general contact capability in a simulation involving large relative motion between potentially contacting surfaces.

Problem description

The model represents an automobile knee bolster assembly—the portion of the instrument panel that the occupant's legs impact in the event of a crash. The assembly consists of a hard plastic cover (the knee bolster) supported by a stiff steel substructure. Proper design of this assembly ensures that the occupant's energy is dissipated with a minimum of injury-causing forces. In this simulation the legs approach the knee bolster at 6 m/s, representing unrestrained motion following a 15 mph to dead stop crash event.

The components of the instrument panel are modeled using S3R and S4R shell elements. The bolster is made up of 2690 shell elements, with the material modeled as a von Mises, elastic strain hardening plastic material with a Young's modulus of 2.346 GPa, a Poisson's ratio of 0.4, a density of 1140 kg/m^3, and a yield stress of 11.7 MPa. The steel substructure is made up of 1648 elements, with the material modeled as a strain hardening steel with a Young's modulus of 207 GPa, a Poisson's ratio of 0.3, a density of 7700 kg/m^3, and a yield stress of 207 MPa. *Figure 1* shows the model geometry from the rear of the knee bolster prior to impact, and *Figure 2* shows the knee bolster and knee/leg assembly from a position outboard and behind the driver prior to impact.

The legs are represented as structural members with a surrounding rigid surface. The structural members, representing the bones, are modeled with B31 beam elements and T3D2 truss elements, with the material modeled as elastic with a Young's modulus of 207 GPa, a Poisson's ratio of 0.3, and a density of 7.7 kg/m^3. The rigid surfaces, representing the knee and shin, are modeled with R3D4 rigid elements. The body mass is modeled by distributing mass elements at various locations among the nodes of the structural elements.

In the input file *knee_bolster_nsm.inp* the upper body mass is distributed over the hip bones (B31 beam elements) as a nonstructural mass instead of point masses. The nonstructural mass contribution to an element increases both the element mass and the element rotary inertia, thus resulting in an increased stable time increment. This analysis completes in about 40% fewer increments when the nonstructural mass feature is used since the time increment for the problem is controlled by the beam elements. In the input file *knee_bolster_massadjust.inp*, the upper body mass is distributed over the hip bones using mass adjustment instead of point masses. The mass distribution and analysis results are similar to those using nonstructural mass. Input files for similar analyses using pipe elements (PIPE31) instead of beam elements are included.

Potential contact among the instrument panel assembly components and between the instrument panel and the legs is modeled using the general contact capability. The general contact inclusions option to automatically define an all-inclusive surface is used and is the simplest way to define contact in the model. In addition, a model that uses the alternative contact pair algorithm is provided; the contact definition is more tedious with the contact pair algorithm.

Initial velocities are defined on the leg components to approximate a 15 mph (6 m/s) crash condition. The hips are constrained to translate in the plane of the seat. The ankles are constrained consistent with fixed planting of the feet on the floor of the car. The dashboard substructure is fixed at locations where it would be welded to the automobile frame; deformations due to this impact are assumed to be confined to the explicitly modeled structure.

Results and discussion

Figure 3 shows the deformed shape of the bolster assembly after 100.0 ms.

Figure 4 shows the energy time history of the whole model: internal energy, kinetic energy, recoverable strain energy, and plastic dissipation. This figure shows that almost all of the body's initial kinetic energy has been transferred by the end of this simulation. Of this transferred amount a small amount has been transferred to elastic deformations in the instrument panel structure and bones, and the balance is lost to plastic dissipation.

Figure 5 shows the total knee and shin contact forces (filtered with the SAE 600 filter) measured against the displacement into the bolster. Consistent with the observations of the energy quantities, it is clear that the crash event is complete. The solution obtained using pipe elements is consistent with that using beam elements.

Acknowledgment

Abaqus would like to thank GE Plastics for supplying the model used in this example.

Input files

knee_bolster.inp	Input data for this analysis using the general contact capability.
knee_bolster_pipe.inp	Input file for similar analysis involving pipe elements and general contact.
knee_bolster_cpair.inp	Input data for this analysis using contact pairs.
knee_bolster_cpair_pipe.inp	Input file for similar analysis involving pipe elements and contact pairs.
knee_bolster_nsm.inp	Input data for this analysis using the general contact capability and the nonstructural mass capability to distribute the upper body mass over the hip bones.
knee_bolster_nsm_pipe.inp	Input file for similar analysis involving pipe elements, general contact, and nonstructural mass.
knee_bolster_massadjust.inp	Input data for this analysis using the general contact capability and mass adjustment to distribute the upper body mass over the hip bones.
knee_bolster_massadjust_pipe.inp	Input file for similar analysis involving pipe elements, general contact, and mass adjustment.
knee_bolster_ef1.inp	External file referenced by this analysis.
knee_bolster_ef2.inp	External file referenced by this analysis.
knee_bolster_ef3.inp	External file referenced by this analysis.

Figures

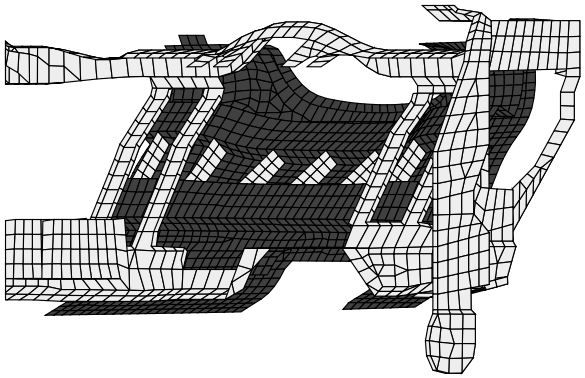


Figure 1: Initial configuration of the knee bolster model (view from behind the bolster).

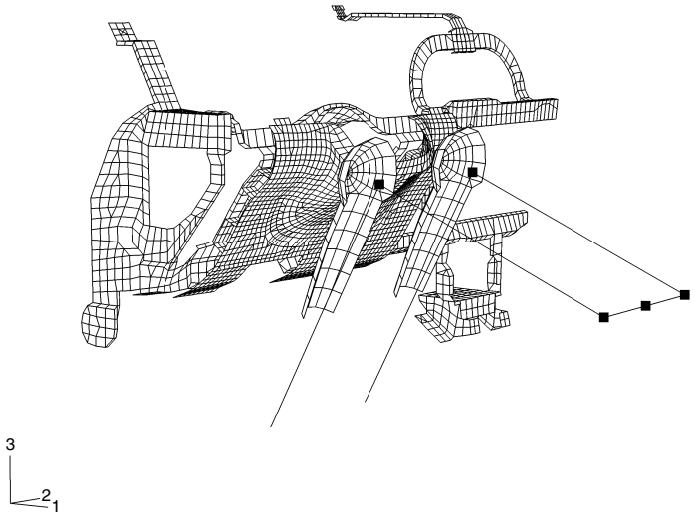


Figure 2: Initial configuration of the knee bolster model (view from outboard and behind the driver).

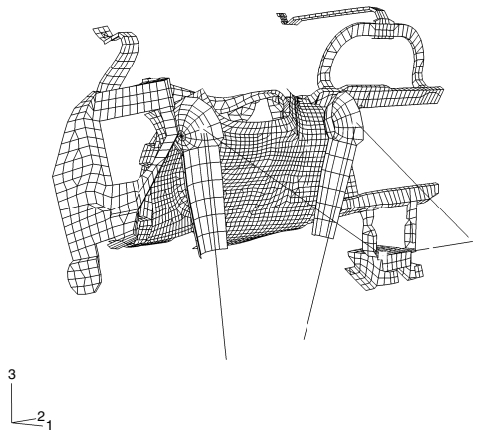


Figure 3: Deformed shape after 100 ms.

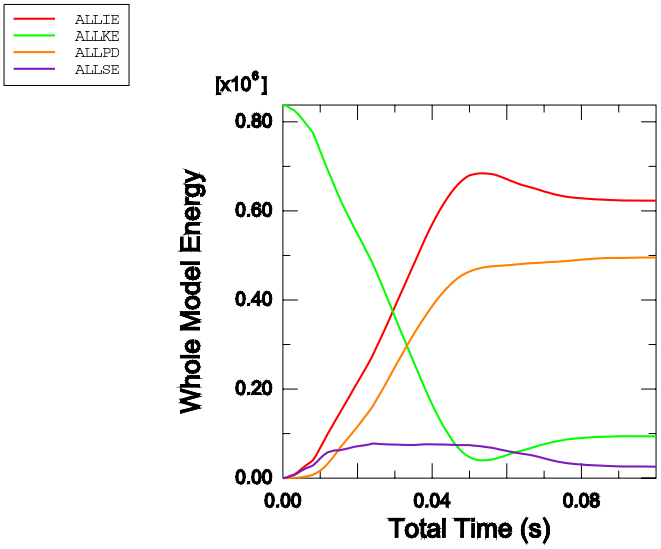


Figure 4: Time histories of the whole model: internal energy, kinetic energy, recoverable strain energy, and plastic dissipation.

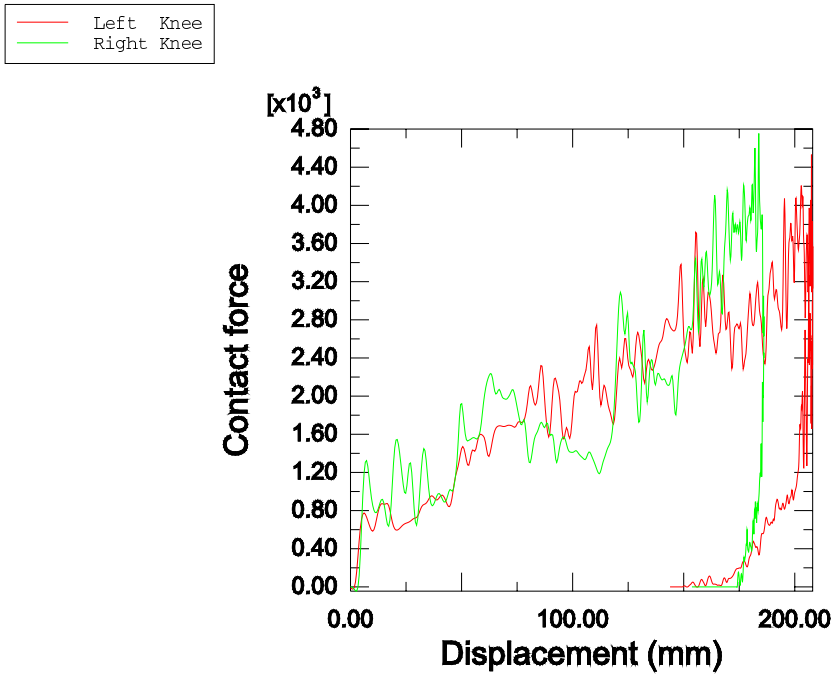


Figure 5: Front leg reaction forces measured against impact displacement.

Crimp forming with general contact

Products: Abaqus/Explicit

References:

- *About General Contact in Abaqus/Explicit*

Overview

This example illustrates the use of the general contact capability in a simulation involving a large number of contacting surfaces.
The general contact algorithm allows very simple definitions of contact with very few restrictions on the types of surfaces involved.

Geometry and model

This model simulates crimp forming. Modern automobiles contain several thousand crimp joints. In a crimp joint a multi-strand wire bundle is mechanically joined to an end terminal to provide electrical continuity across the joint. The portion of the terminal that is folded over and into the wire bundle during crimping is called the grip. Proper design of a crimp joint depends on a number of competing factors including the diameter and number of the wire strands; the thickness, length, and material of the grip; and the geometry and surface finish of the crimp tooling. Out-of-plane extrusion of the wire bundle and grip during crimping is a significant factor in crimp formation.

In this example the grip is 0.25 mm thick and has a 50% coin at the tips. Coining is done to help the grip arms curl over the wire bundle as they are pushed against the roof of the punch during crimping. The grip arm tips are 3.28 mm apart initially (wing tip width). A nineteen-strand wire bundle is used, with each strand having a 0.28 mm diameter. *Figure 1* shows the model geometry prior to crimp forming. *Figure 2* shows a close-up view of the wire-grip assembly.

The deformable wires and the grip are modeled with C3D8R elements. The punch and the anvil are modeled as rigid parts using R3D4 elements. The grip is made of a half hard copper alloy that is modeled as a von Mises elastic strain hardening plastic material with a Young's modulus of 112 GPa, a Poisson's ratio of 0.34, and a yield stress of 391 MPa. The wires are made of copper that is modeled as a strain hardening plastic material with a Young's modulus of 117 GPa, a Poisson's ratio of 0.35, and a yield stress of 241.5 MPa.

Analysis definition

An explicit dynamic simulation is used because the following aspects would present difficulties for a static analysis with Abaqus/Standard:

- The model has no static stability due to the free rigid body motion of the grip and wires.
- During crimping the grip arms buckle as they are turned by the punch downward into the bundle.
- There is complex multi-body contact in the analysis: between the grip arms and the nineteen wires, between each combination of two wires, and between the two grip arms.

The rigid punch has to undergo a downward stroke of 6.88 mm to complete the crimp forming. The punch is moved downward at varying speeds to conduct the analysis efficiently without having inertia effects significantly influence the solution. Initially the punch is moved at an average speed of 50 mm/sec to establish contact between the grip arms and the rigid punch. Then the punch is moved at 300 mm/sec until the tip of the grip arms reach

the roof of the punch. In the final phase the punch is slowed to about 20 mm/sec while the grip arms buckle and fold over into the wire bundle. The overall analysis time is about 0.12 seconds.

The general contact algorithm in Abaqus/Explicit is used for this analysis. The general contact inclusions option to automatically define an all-inclusive surface is used and is the simplest way to define contact in the model. Since this surface spans all the bodies in the model, self-contact for this surface includes interactions between all of the bodies. The contact pair algorithm cannot use surfaces that span multiple bodies; thus, using the contact pair approach for this model would be very tedious. Since there are 22 contacting parts in the model, 231 contact pairs would need to be defined to account for all the possible two-surface combinations, in addition to one contact pair needed to model self-contact for the grip.

Geometric feature edges of a model can also be considered for edge-to-edge contact by the general contact algorithm if a cutoff feature angle is specified. The feature angle is the angle formed between the normals of the two facets connected to an edge. Most of the interactions in this analysis can be detected by node-to-facet contact and, thus, do not rely on edge-to-edge contact; however, when the grip arms extrude out of the punch and contact the edge of the rigid punch, edge-to-edge contact is necessary to enforce contact accurately. The feature angle criterion of the surface property is used to specify a cutoff feature angle of 20° for this analysis; thus, all edges with feature angles greater than 20° are included in the general contact domain.

Coulomb friction is assumed between the individual wires, between the grip and anvil, between the punch and the grip, and between the two grip arms. The general contact property assignment is used to assign the appropriate friction coefficients to the various types of pairings.

The anvil is held motionless during the analysis. One end of the wire bundle is fully constrained, and the other end has no boundary conditions.

Results and discussion

Figure 3 shows the deformed shape of the crimp assembly after the grip arms have reached the roof of the punch (39 milliseconds). *Figure 4* shows the deformed shape of the crimp assembly after the grip arms have curved around the roof of the punch and partially folded over into the wire bundle (76 milliseconds). The grip arms buckle between *Figure 3* and *Figure 4*. *Figure 5* shows a cross-sectional view of the grip and wire bundle taken at the midlength of the grip. This figure shows the distortion of the wires after 107 milliseconds, when the punch has made a downward stroke of 6.605 mm. *Figure 6* shows the final deformed shape of the model (the rigid punch has been removed from the view for clarity). The grip arms have fully folded over into the wire bundle, and the punch has made its complete downward stroke. This figure also shows the out-of-plane extrusion of the wire bundle after distortion.

Figure 7 shows the final shape of the wire bundle without the surrounding grip. This figure shows that the originally round wires have been distorted during crimp formation. This distortion is essential for the correct formation of the crimp joint. The bare copper wires are actually covered by a thin layer of brittle copper oxide that forms on exposure of the copper to air. The goal of crimp forming is to break this oxide layer and expose the copper to the surface of the grip by inducing significant surface strains in each wire.

Input files

crimp_gcont.inp	Input data for this analysis. Surface names are used to identify regions for contact property assignment.
crimp_assembly.inp	External file referenced by this analysis.
crimp_gcont_matsurf.inp	Input data for this analysis. Material names are used to identify regions for contact property assignment.
crimp_assembly_matsurf.inp	External file referenced by this analysis.

References

- Villeneuve, G., D. Kulkarni, P. Bastnagel, and D. Berry, "Dynamic Finite Element Analysis Simulation of the Terminal Crimping Process," *42nd IEEE Holm Conference, Chicago, IL*, October 1996.
- Villeneuve, G., P. Bastnagel, D. Berry, and C. S. Nagaraj, "Determining the Factors Affecting Crimp Formation Using Dynamic Finite Element Analysis," *30th IICIT Connector and Interconnection Symposium, Anaheim, CA*, September 1997.
- Berry, D. T., "Development of a Crimp Forming Simulator," *Abaqus User's Conference Proceedings*, pp. 125–137, 1998.

Figures

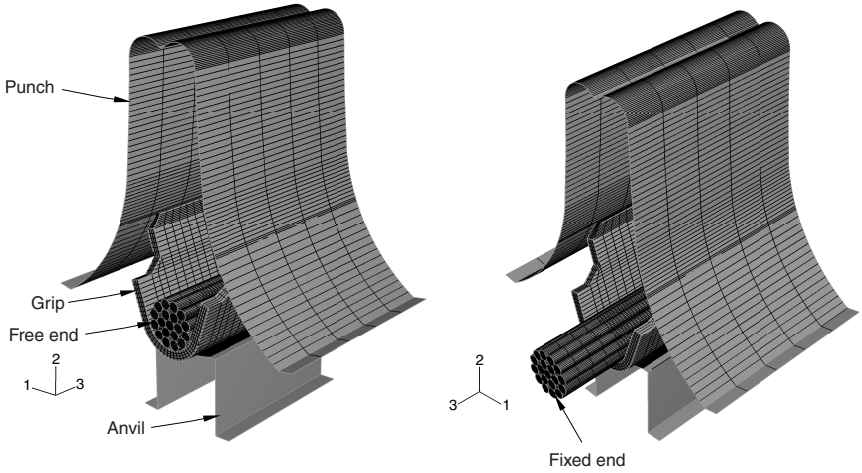


Figure 1: Initial configuration of the crimp forming model (opposite isometric views).

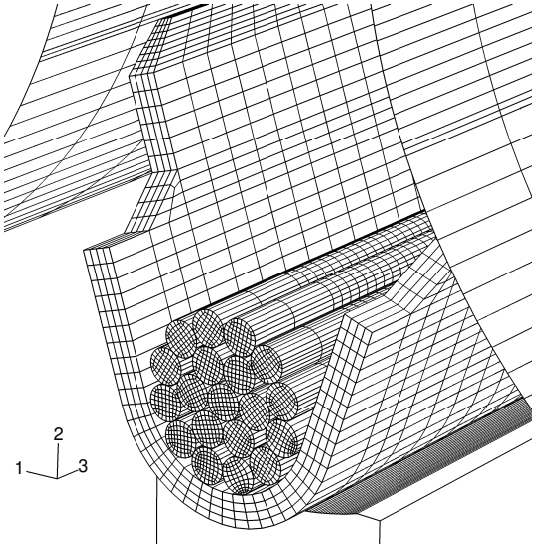


Figure 2: Close-up view of the wire-grip assembly.

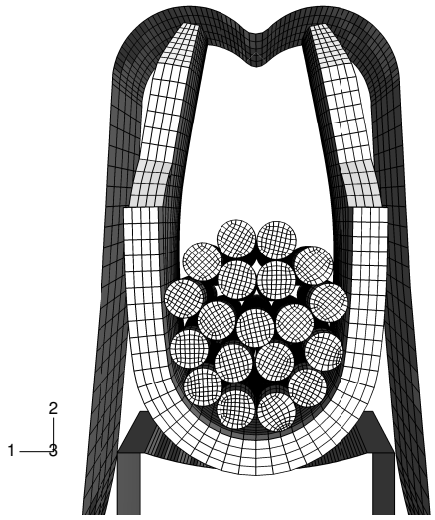


Figure 3: Deformed shape after 39 ms (front view).

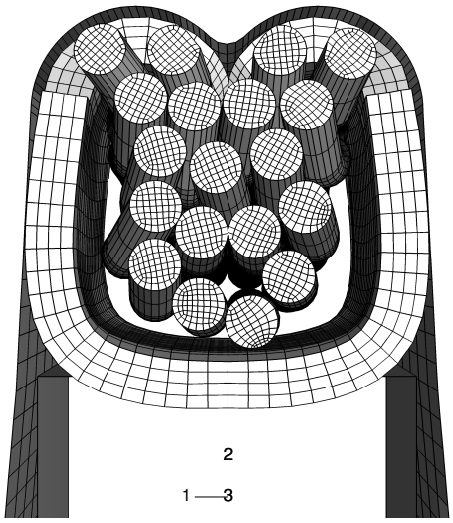


Figure 4: Deformed shape after 76 ms (front view).

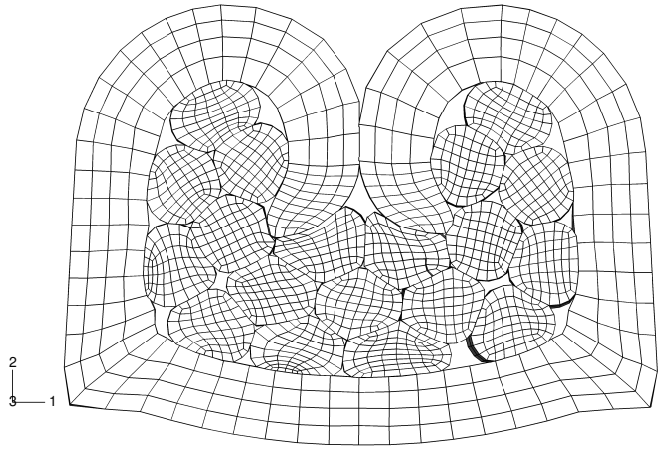


Figure 5: Cross-sectional view of the grip and wire bundle after 107 ms (the punch and anvil are not shown).

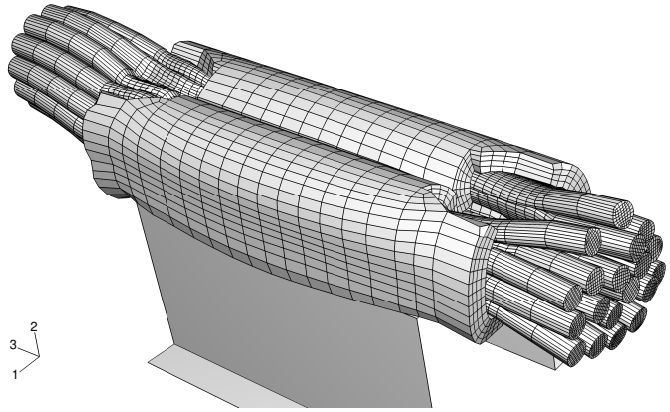


Figure 6: Isometric view of the final deformed configuration for the wire-grip assembly (the rigid punch is not shown).

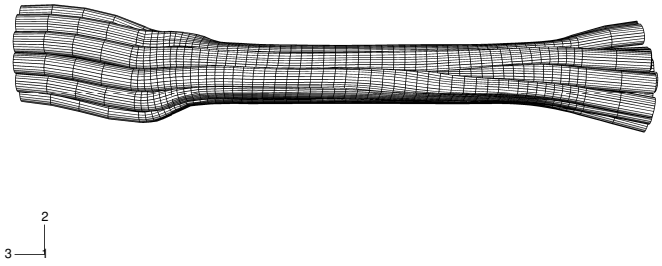


Figure 7: Final deformed shape of the wire bundle (the grip is not shown).

Collapse of a stack of blocks with general contact

Products: Abaqus/Explicit

References:

- *About General Contact in Abaqus/Explicit*

Overview

This example illustrates the use of the general contact capability in a simulation involving a large number of contacting bodies.
The general contact algorithm allows very simple definitions of contact with very few restrictions on the types of surfaces involved.

Problem description

The model simulates the collapse of a stack of blocks. The undeformed configuration of the model is shown in *Figure 1*. There are 35 blocks, and each block is $12.7 \times 12.7 \times 76.2$ mm ($0.5 \times 0.5 \times 3$ inches) in size. The blocks are stacked on a rigid floor. The stack is subjected to gravity loading. It is assumed that a key block near the bottom of the stack has been removed just before the start of the analysis, initiating the collapse.

Each block is modeled with a single C3D8R element. The use of a coarse mesh highlights the edge-to-edge contact capability of the general contact algorithm, because the majority of the block-to-block interactions do not result in penetrations of nodes into faces.

Two different cases are analyzed. In the first analysis the blocks are rigid. In the second analysis the blocks are deformable. In the latter case, the material of the block is assumed to be linear elastic with a Young's modulus of 12.135 GPa (1.76×10^6 Psi), a Poisson's ratio of 0.3, and a density of 577.098 kg/m^3 (5.4×10^{-5} lb s^2/in^4). Only the density is relevant for the analysis assuming rigid blocks. In addition, ENHANCED hourglass control is used for the deformable analysis. The rigid floor is modeled as a discrete rigid surface using a single R3D4 element.

This model involves a large number of contacting bodies. The general contact capability greatly simplifies the contact definition, since each of the 595 possible block-to-block pairings does not need to be specified individually. The general contact inclusions option to automatically define an all-inclusive surface is used and is the simplest way to define contact in the model. Coulomb friction with a friction coefficient of 0.15 is assumed between the individual blocks and between the blocks and floor. The general contact property assignment is used to assign this nondefault contact property.

By default, the general contact algorithm in Abaqus/Explicit accounts for edge-to-edge contact of perimeter edges on structural elements. Geometric feature edges of a model can also be considered for edge-to-edge contact by the general contact algorithm; including the geometric feature edges is crucial in this analysis. A cutoff feature angle of 20° is specified for the feature angle criterion of the surface property to indicate that all edges with feature angles greater than 20° should be considered for edge-to-edge contact. The feature angle is the angle formed between the normals of the two facets connected to an edge.

The magnitude of the gravity loading is increased by a factor of 10 to facilitate demonstration of the edge-to-edge contact capability with a short analysis time. The analysis is performed for a period of 0.15 seconds. For the analysis with rigid blocks there is no deformable element available in the model to control the stable time increment. A fixed time increment of 1×10^{-6} seconds is specified for this purpose, which is similar to the time increment used by the analysis with deformable blocks. The time increment chosen for the analysis with rigid

blocks will affect the penalty stiffness used by the contact algorithm since the penalty stiffness is inversely proportional to the time increment squared.

Results and discussion

Results are shown for the rigid body case. Results for the deformable case are very similar to the rigid model results.

Figure 2 shows the displaced shape of the block assembly after 0.0375 seconds. The stack of blocks has started to collapse under gravity loading. *Figure 3* shows a close-up view of the collapsing blocks after 0.1125 seconds. This figure clearly shows that the geometric feature edges of individual blocks contact each other during collapse. *Figure 4* shows the final configuration of the blocks. The stack has collapsed completely on the rigid surface.

Input files

blocks_rigid_gcont.inp	Input file for the rigid body analysis.
blocks_rigid_assembly.inp	External file referenced by the rigid body analysis.
blocks_deform_gcont.inp	Input file for the deformable analysis.
blocks_deform_assembly.inp	External file referenced by the deformable analysis.

Figures

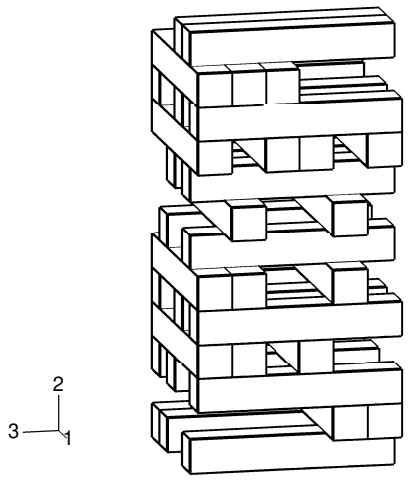


Figure 1: Initial configuration of the stack of blocks.

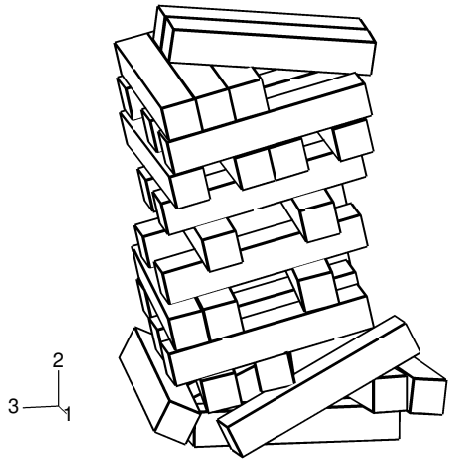


Figure 2: Displaced shape after 0.0375 s.

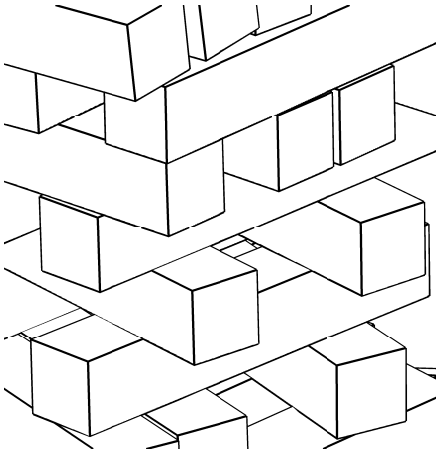


Figure 3: Close-up view of the collapsing blocks after 0.1125 s.

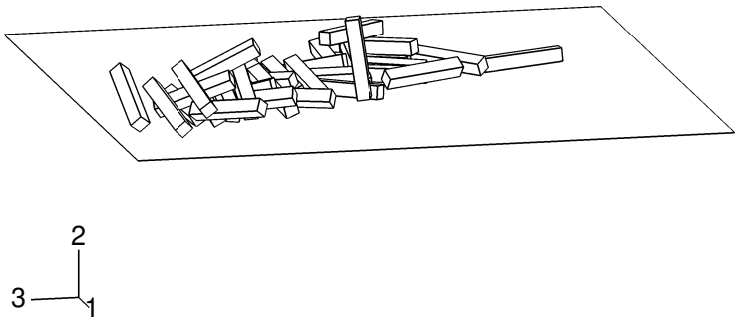


Figure 4: Final configuration of the model.

Cask drop with foam impact limiter

Products: Abaqus/Explicit

Overview

This example illustrates the use of an initial velocity condition and the analysis of a structure containing liquid and incorporating crushable foam to absorb impact energy.

A containment cask is partially filled with fluid and a foam impact limiter. The complete package is dropped a distance of 9.09 m (30 ft) onto a rigid surface, which results in an impact speed of 13.35 m/sec (525.3 in/sec). Experimental and numerical results for this problem have been reported by Sauvé et al. (1993). The numerical results given in the reference were obtained using a relatively coarse finite element mesh. In this example results are presented for the same coarse mesh as used in the reference and also for a more refined mesh. Both continuum meshes and particle methods are illustrated.

Model description

The containment cask shown in *Figure 2* consists of two compartments. The upper compartment surrounds the fluid and is made of stainless steel (304L). It has a height of 580 mm (22.8 in), a diameter of 300 mm (11.8 in), and a wall thickness of 4.76 mm (0.187 in). The top mild steel cover has a thickness of 9.52 mm (0.375 in). The water is filled to a depth of 522 mm (20.55 in), which is 90% of the container's capacity. *Figure 3* shows the original, coarse mesh of C3D8R elements used to model the fluid. Contact conditions are defined between the fluid and the inside of the upper compartment.

An impact limiter made of polyurethane foam is contained within the bottom mild steel compartment of the cask. The height of the foam impact limiter is 127.3 mm (5.01 in). *Figure 4* shows the coarse mesh used to model the foam. Contact conditions are defined between the foam and the inside of the bottom compartment of the cask. The foam impact limiter and the fluid/stainless steel liner are separated by a mild steel bulkhead with a thickness of 12.7 mm (0.5 in). A 12.7 mm (0.5 in) air gap exists between the top of the foam surface and this bulkhead.

In the experiment a pressure transducer is located in the polyurethane foam on the centerline of the cask at the top of the impact limiter. This result is compared with vertical stress-time histories taken from the element at the top of the foam model on the centerline.

Both axisymmetric and three-dimensional models are analyzed. *Figure 5* shows the three-dimensional model formed by assembling the parts shown in *Figure 2* through *Figure 4*. The equivalent axisymmetric model is shown in *Figure 6*. The corresponding smoothed particle hydrodynamic model incorporates PC3D elements for both the water and the foam parts of the model.

Contact pairs are defined between the solids and the shells. Element-based surfaces are defined on the shells, and node-based surfaces are defined containing the nodes on the outer surfaces of the solid or particle elements. Input files that use the alternative general contact algorithm are also provided. The shell thickness was not taken into account when the original meshes were designed, and the outer surface of the solids usually coincides with the midsurface of the enclosing shell. This would lead to an initial overclosure of one-half the shell thickness, but contact at the midsurface of the shell is enforced, as if the shell had zero thickness. The use of a node-based surface implies a pure main-secondary relationship for the contact pair. This is important in this problem because the default in Abaqus/Explicit when contact is defined between shells and solids is to define a pure main-secondary relationship with the solids as the main and the shells as the secondary. In this case the shell structures are much stiffer than the fluid and foam structures, so the main-secondary roles must be reversed.

For the axisymmetric model two cases using different section controls for the foam and fluid elements are analyzed. The first case uses a linear combination of stiffness and viscous hourglass control; the second case uses the default section controls (the integral viscoelastic form for hourglass control). The three-dimensional model also has two cases with different section controls for the foam and fluid elements. The first case uses the orthogonal kinematic formulation and combined (viscous-stiffness form) hourglass control; the second three-dimensional case uses the default section controls (the average strain kinematic formulation and the integral viscoelastic form for hourglass control). The section controls used are summarized in *Table 1*. Coarse and refined meshes are used for all analysis cases.

Material description

The general material properties are listed in *Table 2*. The material models for the water and foam are further described below.

Water:

The water is treated as a simple hydrodynamic material model. This provides zero shear strength and a bulk response given by

$$p = K\varepsilon_{vol},$$

where K is the bulk modulus with a value of 2068 MPa (300000 psi). This model is defined using the linear $U_s - U_p$ equation of state model provided in Abaqus/Explicit. The linear $U_s - U_p$ Hugoniot form, $p = f(\rho) + g(\rho) E_m$, is

$$p = \frac{\rho_0 c_0^2 \eta}{(1-s\eta)^2}\left(1 - \frac{\Gamma_0 \eta}{2}\right) + \Gamma_0 \rho_0 E_m,$$

where $\eta = 1 - \rho_0/\rho$ is the same as the nominal volumetric strain measure, ε_{vol}. Since $K = \rho_0 c_0^2$, setting the parameters $s = 0.0$ and $\Gamma_0 = 0.0$ gives the simple hydrostatic bulk response defined earlier. In this analysis $c_0 =$ 1450.6 m/sec (57100 in/sec) and $\rho_0 =$ 983.2 kg/m^3 (0.92×10^{-4} lb sec^{2}in^{-4}). The tension cutoff pressure is assumed to be zero and is specified using a tensile failure model. Refer to *Equation of State* for a description of this material model.

Foam:

The crushable foam model is used for the polyurethane foam. In this model the flow potential, h, is chosen as

$$h = \sqrt{\frac{9}{2}p^2 + q^2},$$

where q is the Mises equivalent stress and p is the hydrostatic pressure. The yield surface is defined as

$$\left[\left(\frac{p_t - p_c}{2} + p\right)^2 + \left(\frac{q}{M}\right)^2\right]^{\frac{1}{2}} = \frac{p_c + p_t}{2}.$$

Sauvé et al. use the “soils and crushable foams” model, which was originally defined in an unpublished report by Krieg (1978) and is based upon a Mises plasticity model in which the yield stress depends upon the mean volumetric pressure. The volumetric deformation allows for plastic behavior, defined by tabular data defining pressure versus volume strain. This model is easy to implement in an explicit dynamics algorithm and useful because the deviatoric and volumetric terms are only loosely coupled. However, it requires an experienced analyst to ensure that meaningful results are obtained, mainly because the model does not match physical behavior well under deviatoric straining.

To define the initial shape of the yield surface, the Abaqus/Explicit crushable foam model with volumetric hardening requires the initial yield stress in uniaxial compression, σ_0; the magnitude of the strength in hydrostatic tension, p_t; and the initial yield stress in hydrostatic compression, p_{c0}. Sauvé et al. define the pressure-dependent yield surface for the foam model as

$$\sigma_y^2 = 3.18 + 2.06p,$$

where the units of stress are MPa and pressure is positive in compression. To calibrate the Abaqus/Explicit crushable foam model to this pressure-dependent data, we observe that $p = \frac{1}{3}\sigma_y$ for the uniaxial compression case. Substituting this value for p in the above equation and solving for σ_y gives σ_0 = 2.16 MPa (313.3 psi). The value of p_t is obtained by solving the above equation for $\sigma_y = 0$, giving p_t = 1.54 MPa (223.8 psi). The value of p_{c0} is given in the reference as p_{c0} = 5.52 MPa (800.0 psi).

The pressure-volumetric strain data in the reference are given in *Table 3*. *Table 4* shows the uniaxial stress-plastic strain data converted to the form required for the Abaqus/Explicit volumetric hardening model. Each form of the data is plotted in *Figure 1*.

Results and discussion

The deformed geometries for the three-dimensional and the axisymmetric models at 5 msec are shown in *Figure 7* and *Figure 8*. The axisymmetric model is analyzed using combined hourglass control. The three-dimensional model uses orthogonal kinematic and combined hourglass control. *Figure 9* shows plots of the vertical stress versus time for the element located at the pressure transducer in the foam; results from the models with the previous section control options, as well as results from analyses using the default section control options, are reported for comparison (see *Table 1*). Axisymmetric and three-dimensional results are compared to the experimental pressure trace. The time origin of the experimental curve is not defined in the reference; therefore, the experimental curve is shifted so that the time when pressure in the transducer changes to a positive value is assumed to be the time at which impact occurs. The numerical pressure results show significant oscillations about the experimental results during the first 2 msec of the response. This is partly because the meshes are quite coarse and partly because pressure transducers in experiments exhibit inertia in their response and will not report sharp gradients in time. During the next 3 msec the numerical results correspond more closely with the experimental results. The analyses run with different section control options compare very well.

A more refined three-dimensional mesh is shown in *Figure 10*. The refined axisymmetric model is the same model used in the r–z plane. The deformed geometries for these models are shown in *Figure 11* (using orthogonal kinematic and combined hourglass control) and *Figure 12* (using combined hourglass control). The vertical stress histories for the refined models are shown in *Figure 13* for the same section controls used for the coarse meshes (see *Table 1*). The numerical results show less oscillation about the experimental results than those obtained with the coarse mesh. They compare well with the experimental results during the following 3 msec of the response. In addition, *Figure 11* shows that the refined mesh eliminates much of the fluid's hourglass-like response due to its zero shear strength. *Figure 14* shows a comparison of the vertical stress results for the three-dimensional refined model and the smoothed particle hydrodynamic model.

Input files

cask_drop_axi_cs.inp	Coarse axisymmetric model using COMBINED hourglass control.
cask_drop_3d_ocs.inp	Coarse three-dimensional model using ORTHOGONAL kinematic and COMBINED hourglass control.
cask_drop_3d_ocs_gcont.inp	Coarse three-dimensional model using ORTHOGONAL kinematic and COMBINED hourglass control and the general contact capability.
cask_drop_axi.inp	Coarse axisymmetric mesh using the default section controls.
cask_drop_3d.inp	Coarse three-dimensional mesh using the default section controls.

cask_drop_3d_gcont.inp	Coarse three-dimensional mesh using the default section controls and the general contact capability.
cask_drop_axi_r_cs.inp	Refined axisymmetric model using COMBINED hourglass control.
cask_drop_3d_r_ocs.inp	Refined three-dimensional model using ORTHOGONAL kinematic and COMBINED hourglass control.
cask_drop_3d_r_ocs_gcont.inp	Refined three-dimensional model using ORTHOGONAL kinematic and COMBINED hourglass control and the general contact capability.
cask_drop_axi_r.inp	Refined axisymmetric mesh using the default section controls.
cask_drop_3d_r.inp	Refined three-dimensional mesh using the default section controls.
cask_drop_3d_r_gcont.inp	Refined three-dimensional mesh using the default section controls and the general contact capability.
cask_drop_3d_sph.inp	Three-dimensional model using the smoothed particle hydrodynamic method.

References

- Krieg, R. D., "A Simple Constitutive Description for Soils and Crushable Foams," *SC-DR-72-0883, Sandia National Laboratories, Albuquerque, NM*, 1978.
- Sauvé, R. G., G. D. Morandin, and E. Nadeau, "Impact Simulation of Liquid-Filled Containers Including Fluid-Structure Interaction," *Journal of Pressure Vessel Technology*, vol. 115, pp. 68–79, 1993.

Tables

Table 1: Analysis section controls tested.

Analysis Label	Section Controls	
	Kinematic Formulation	Hourglass Control
AXI	n/a	integral viscoelastic
AXI CS	n/a	combined
3D	average strain	integral viscoelastic
3D OCS	orthogonal	combined

Table 2: Material properties.

Properties	A36	304L	Liquid	Foam
Density, ρ (kg/m^3)	8032	8032	983	305
Young's modulus, E (GPa)	193.1	193.1		.129
Poisson's ratio, ν	0.28	0.28		0
Yield stress, σ_{y0} (MPa)	206.8	305.4		
Bulk modulus, K (GPa)			2.07	
Hardening modulus, E_t (GPa)	0	1.52		

Table 3: Pressure-volumetric strain data.

ϵ_v	0	0.01	0.02	0.03	0.04	0.05	0.06	0.385	0.48	0.53	0.55
p (MPa)	0	2.76	4.14	5.17	5.52	5.86	6.21	10.34	19.31	39.30	82.74

Table 4: Uniaxial stress-plastic strain data.

ϵ^p	0.00	0.01	0.02	0.345	0.44	0.49	0.51	2.00
σ_c (MPa)	2.16	2.24	2.33	3.23	4.91	8.20	14.67	758.89

Figures

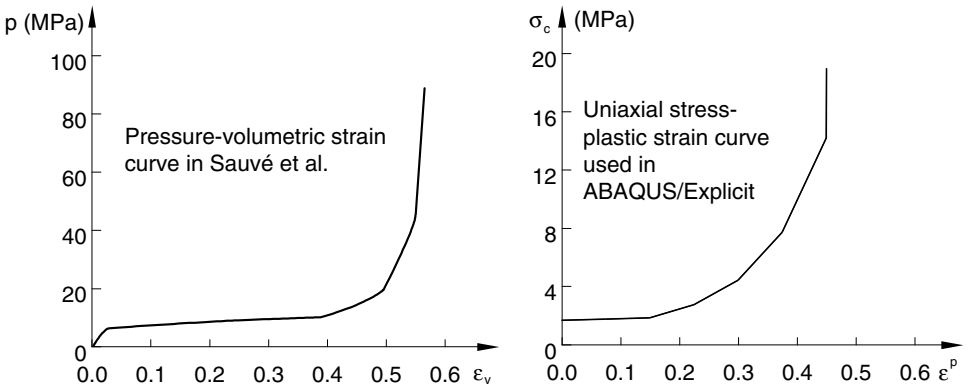


Figure 1: Foam hardening curves.

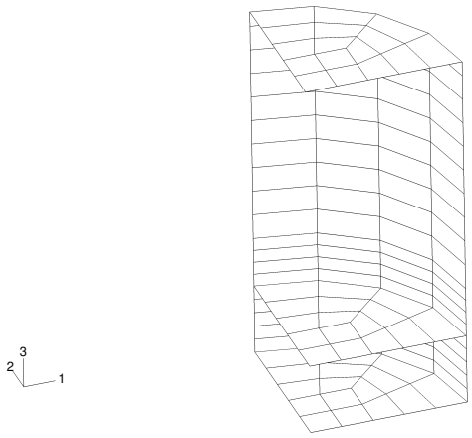


Figure 2: Containment structure mesh in the three-dimensional model (coarse mesh).

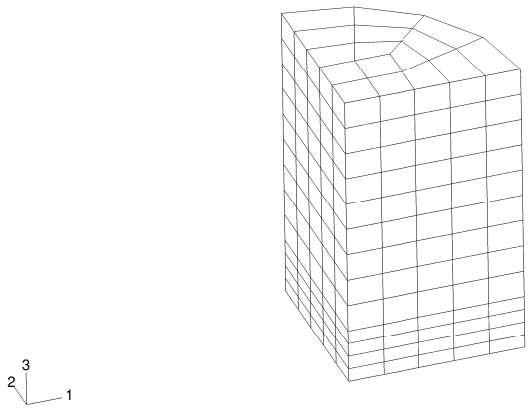


Figure 3: Fluid mesh in the three-dimensional model (coarse mesh).

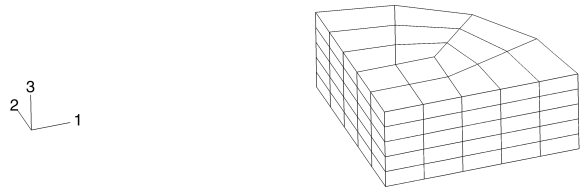


Figure 4: Foam mesh in the three-dimensional model (coarse mesh).

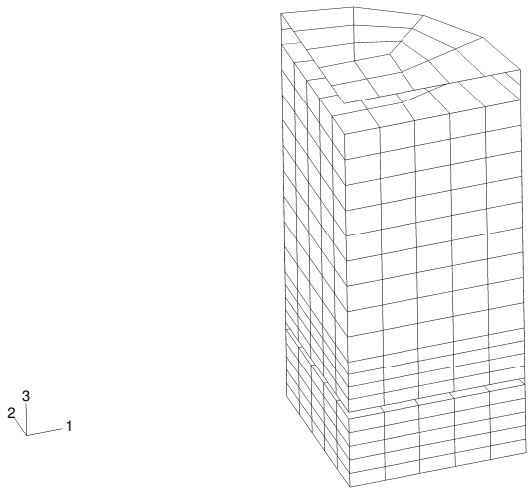


Figure 5: The complete three-dimensional model (coarse mesh).

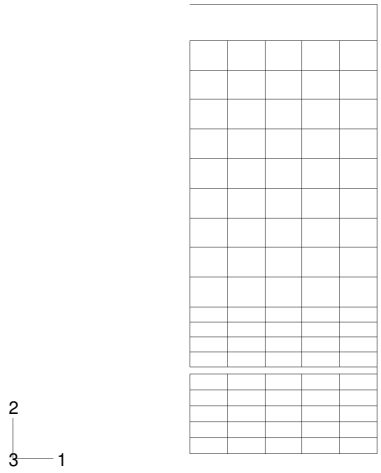


Figure 6: Axisymmetric model (coarse mesh).

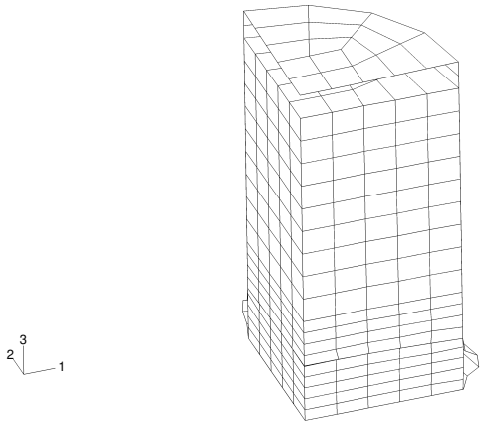


Figure 7: Three-dimensional deformed geometry using orthogonal element kinematics and combined hourglass control (coarse mesh).

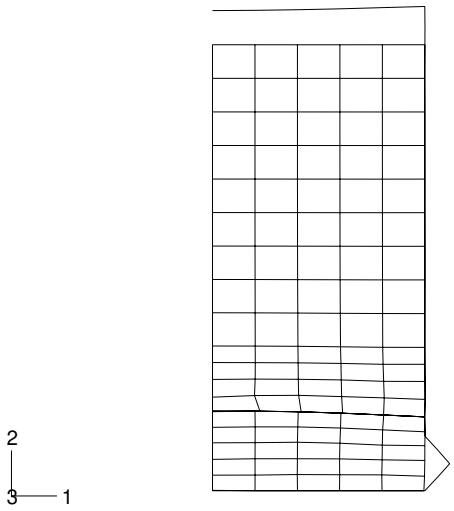


Figure 8: Axisymmetric deformed geometry using combined hourglass control (coarse mesh).

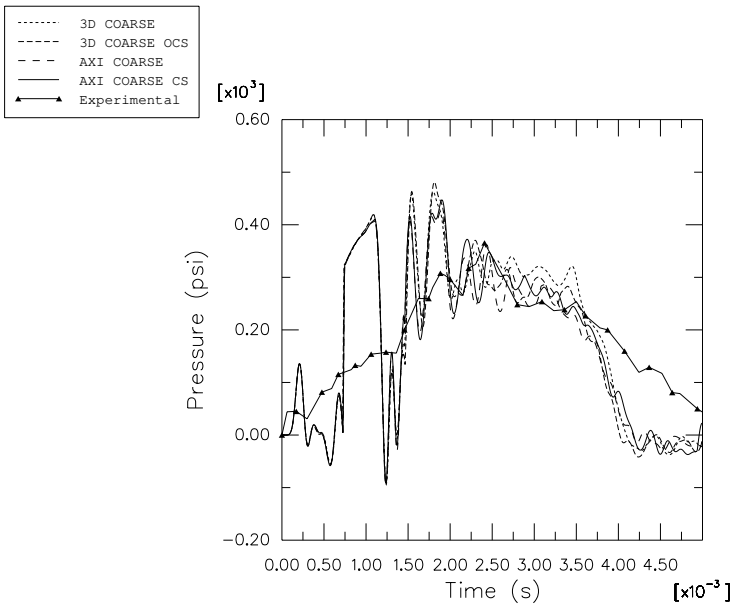


Figure 9: Vertical stress history in the foam (coarse mesh).

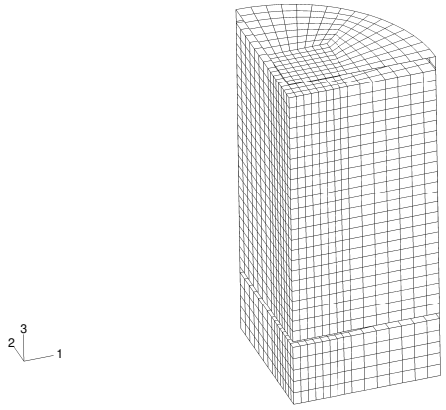


Figure 10: Refined mesh for the three-dimensional model.

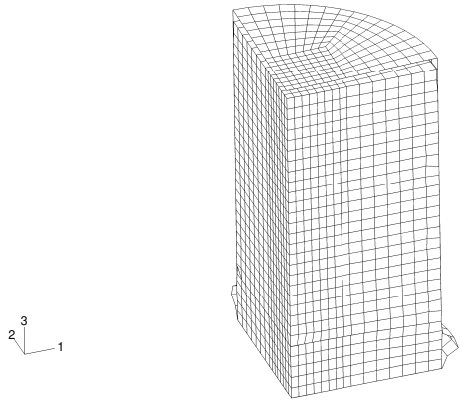


Figure 11: Three-dimensional deformed geometry using orthogonal element kinematics and combined hourglass control (refined mesh).

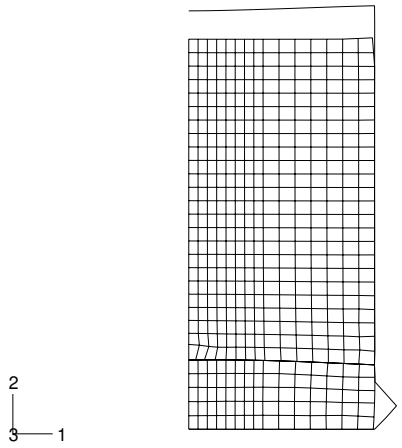


Figure 12: Axisymmetric deformed geometry using combined hourglass control (refined mesh).

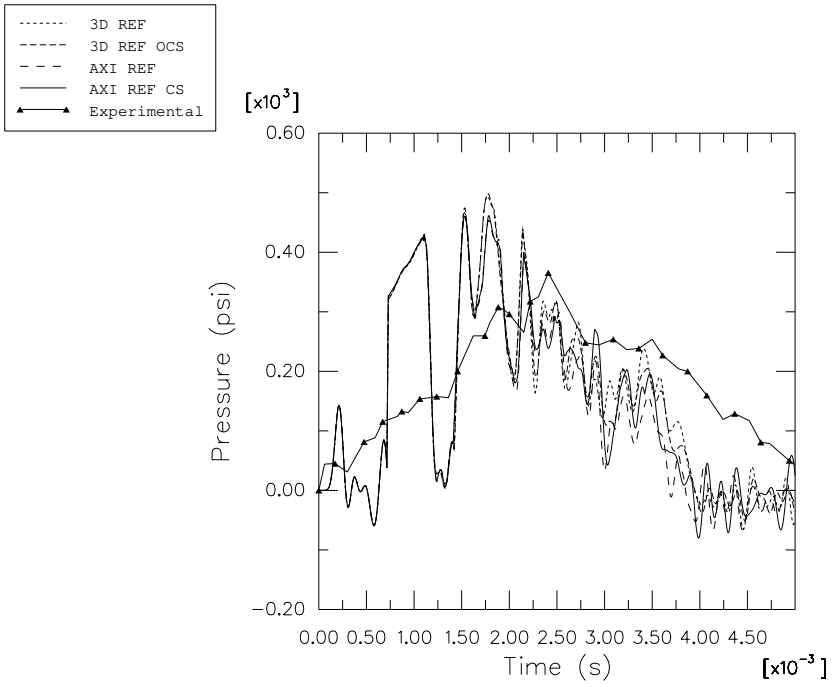


Figure 13: Vertical stress history in the foam (refined mesh).

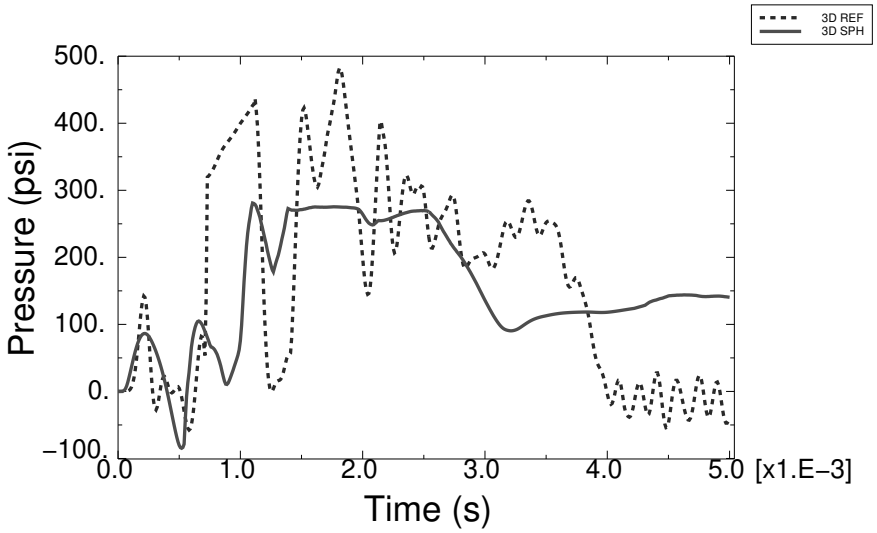


Figure 14: Vertical stress history comparison for the three-dimensional refined model and the smoothed particle hydrodynamic model.

Oblique impact of a copper rod

Products: Abaqus/Explicit

Overview

This example simulates a high velocity, oblique impact of a copper rod into a rigid wall.
Extremely high plastic strains develop at the crushed end of the rod, resulting in severe local mesh distortion. Adaptive meshing is used to reduce element distortion and to obtain an accurate and economical solution to the problem.

Problem description

The model geometry is depicted in *Figure 1*. A cylindrical rod, measuring 32.4 × 3.2 mm, impacts a rigid wall with an initial velocity of v_y=340 m/sec. The wall is perpendicular to the *x*–*z* plane and makes an angle of 30° with the *x*–*y* plane. The half-symmetric finite element model is shown in *Figure 2*. Symmetry boundary conditions are applied at the *y*=0 plane. The rod is meshed with CAX4R elements, and the wall is modeled as an analytical rigid surface using a three-dimensional cylindrical surface in conjunction with a rigid body constraint. Coulomb friction is assumed between the rod and the wall, with a friction coefficient of 0.2. The analysis is performed for a period of 120 microseconds.

The rod is modeled as a Johnson-Cook, elastic-plastic material with a Young's modulus of 124 GPa, a Poisson's ratio of 0.34, and a density of 8960 kg/m^3. The Johnson-Cook model is appropriate for modeling high-rate impacts involving metals. The Johnson-Cook material parameters are taken from Johnson and Cook (1985) in which the following constants are used: $A =$ 90 MPa, $n =$ 0.31, $m =$ 1.09, $C =$ 0.025, and $\dot{\varepsilon}_o =$ 1 s^{-1}. Furthermore, the melting temperature is 1058°C, and the transition temperature is 25°C. Adiabatic conditions are assumed with a heat fraction of 50%. The specific heat of the material is 383 J/Kg°C, and the thermal expansion coefficient is 0.00005°C^{-1}.

Adaptive meshing

A single adaptive mesh domain that incorporates the entire rod is defined. Symmetry boundary conditions are defined as Lagrangian surfaces (the default), and contact surfaces are defined as sliding contact surfaces (the default). Because the impact phenomenon modeled in this example is an extremely dynamic event with large changes in geometry occurring over a relatively small number of increments, it is necessary to increase the frequency and intensity of adaptive meshing. The frequency value is reduced to 1 increment from a default value of 10, and the number of mesh sweeps used to smooth the mesh is increased to 3 from the default value of 1. The default values are used for all other adaptive mesh controls.

Results and discussion

Deformed shape plots at 40, 80, and 120 microseconds are shown in *Figure 3*, *Figure 4* and *Figure 5*, respectively. The rod rebounds from the wall near the end of the analysis. High-speed collisions such as these result in significant amounts of material flow in the impact zone. A pure Lagrangian analysis of this finite element model fails as a result of excessive distortions. Continuous adaptive meshing allows the analysis to run to completion while retaining a high-quality mesh. The kinetic and internal energy histories are plotted in *Figure 6*. Most of the initial kinetic energy is converted to internal energy as the rod is plastically deformed. Both energy curves plateau as the rod rebounds from the wall.

Input files

ale_rodimpac_inclined.inp	Analysis using adaptive meshing.
ale_rodimpac_inclined_nodelem.inp	External file referenced by this analysis.

References

- Johnson, G. R., and W. H. Cook, “Fracture Characteristics of Three Metals Subjected to Various Strains, Strain Rates, Temperatures and Pressures,” Engineering Fracture Mechanics, 21, pp. 31–48, 1985.

Figures

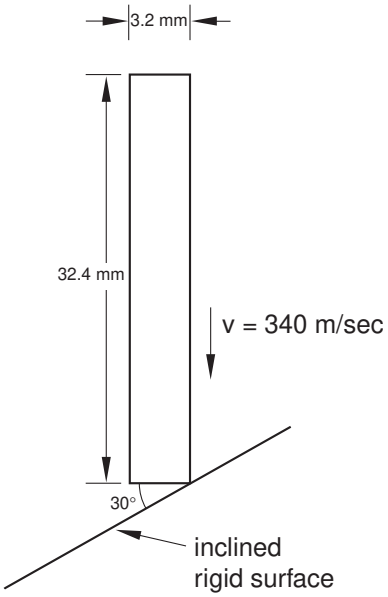


Figure 1: Model geometry.

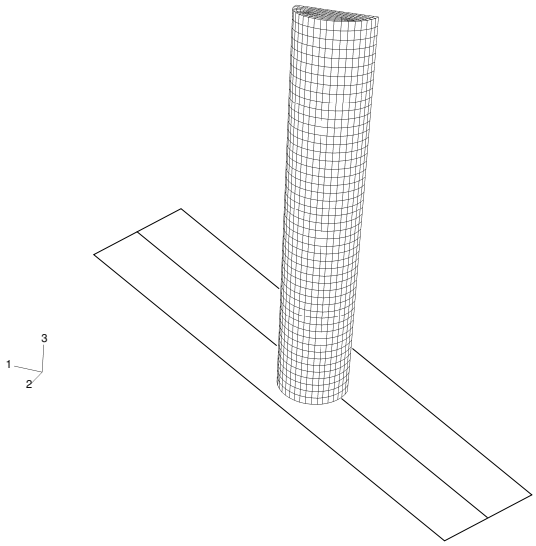


Figure 2: Initial configuration.

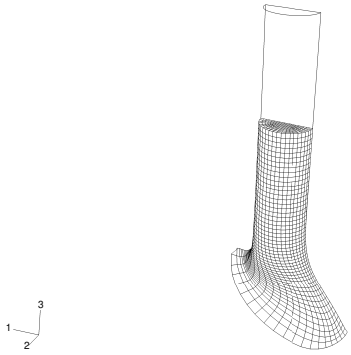


Figure 3: Deformed configuration at 40 microseconds.

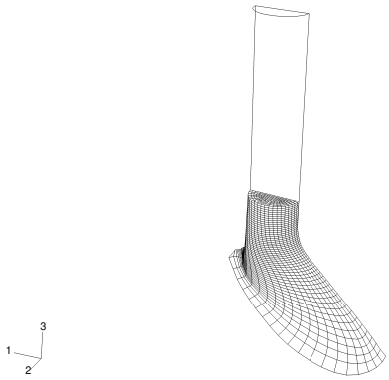


Figure 4: Deformed configuration at 80 microseconds.

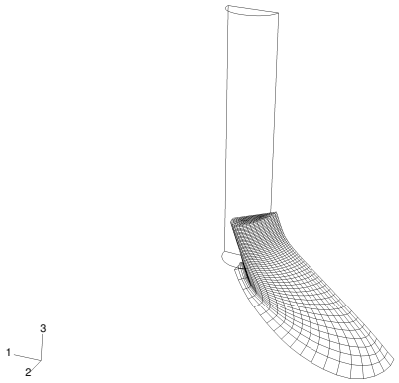


Figure 5: Deformed configuration at 120 microseconds.

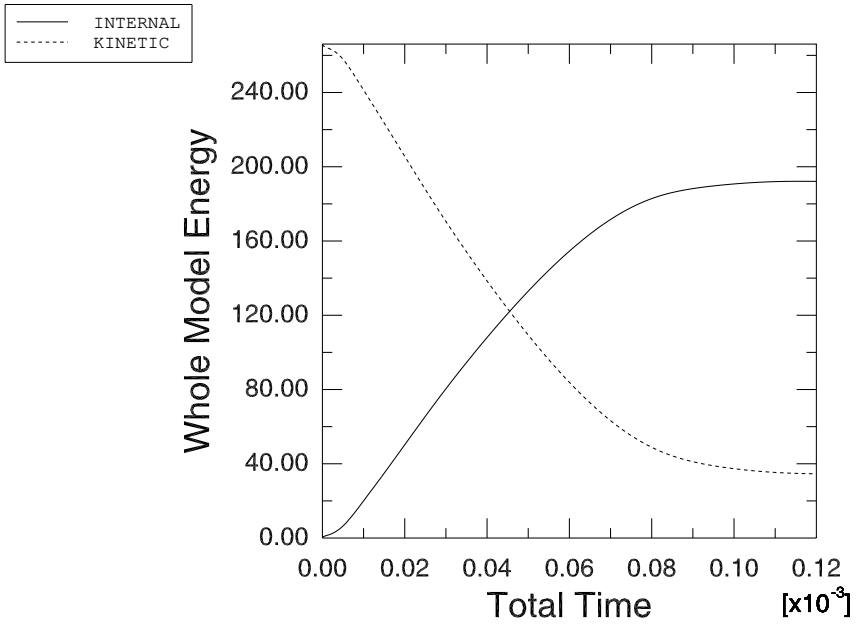


Figure 6: Time history of kinetic and internal energies of the rod.

Water sloshing in a baffled tank

Products: Abaqus/Explicit

Overview

This example illustrates the use of adaptive meshing to model an inviscid fluid sloshing inside a baffled tank. The overall structural response resulting from the coupling between the water and tank, rather than a detailed solution in the fluid, is sought. Adaptive meshing permits the investigation of this response over longer time periods than a pure Lagrangian approach would because the mesh entanglement that occurs in the latter case is prevented.

Problem description

The geometry for the problem is shown in *Figure 1*. The model consists of a baffled tank filled with water. The baffle, which is attached to the sides and top of the tank, does not penetrate the entire depth of the water. The tank measures 508 × 152.4 × 152.4 mm (20 × 6 × 6 inches), and the baffle measures 3.048 × 152.4 × 121.92 mm (0.12 × 6 × 4.8 inches). The tank is filled with 101.6 mm (4 inches) of water.

A cutaway view of the finite element model that displays the baffle and the water is shown in *Figure 2*. The top of the tank is not modeled because the water is not expected to come into contact with it. The tank is modeled as a rigid body and is meshed with R3D4 elements. The baffle is modeled as a deformable body and is meshed with S4R elements. A graded mesh of C3D8R elements is used for the water, with more refinement adjacent to the baffle where significant deformations are expected.

In sloshing problems water can be considered an incompressible and inviscid material. An effective method for modeling water in Abaqus/Explicit is to use a simple Newtonian viscous shear model and a linear $U_s - U_p$ equation of state for the bulk response. The bulk modulus functions as a penalty parameter for the incompressible constraint. Since sloshing problems are unconfined, the bulk modulus chosen can be two or three orders of magnitude less than the actual bulk modulus and the water will still behave as an incompressible medium. The shear viscosity also acts as a penalty parameter to suppress shear modes that could tangle the mesh. The shear viscosity chosen should be small because water is inviscid; a high shear viscosity will result in an overly stiff response. An appropriate value for the shear viscosity can be calculated based on the bulk modulus. To avoid an overly stiff response, the internal forces arising due to the deviatoric response of the material should be kept several orders of magnitude below the forces arising due to the volumetric response. This can be done by choosing an elastic shear modulus that is several orders of magnitude lower than the bulk modulus. If the Newtonian viscous deviatoric model is used, the shear viscosity specified should be on the order of an equivalent shear modulus, calculated as mentioned earlier, scaled by the stable time increment. The expected stable time increment can be obtained from a datacheck analysis of the model. This method is a convenient way to approximate a shear strength that will not introduce excessive viscosity in the material.

In addition, if a shear model is defined, the hourglass control forces are calculated based on the shear stiffness of the material. Thus, in materials with extremely low or zero shear strengths such as inviscid fluids, the hourglass forces calculated based on the default parameters are insufficient to prevent spurious hourglass modes. Therefore, a sufficiently high hourglass scaling factor is used to increase the resistance to such modes. This analysis methodology is verified in *Water sloshing in a pitching tank*.

For this example the linear $U_s - U_p$ equation of state is used with a wave speed of 45.85 m/sec (1805 in/sec) and a density of 983.204 kg/m^3 (0.92×10^{-4} lb sec^2/in^4). The wave speed corresponds to a bulk modulus of 2.07 MPa (300 psi), three orders of magnitude less than the actual bulk modulus of water, 2.07 GPa (3.0×10^5 psi). The shear viscosity is chosen as 1.5×10^{-8} psi sec. The baffle is modeled as a Mooney-Rivlin elastomeric material

with hyperelastic constants C_{10} = 689480 Pa (100 psi) and C_{01} = 172370 Pa (25 psi) and a density of 10900.74 kg/m^3 (1.02×10^{-3} lb sec^2/in^4).

The general contact capability is used for the entire domain. The bottom edge of the baffle has nodes in common with the underlying water surface. This prevents relative slip between the bottom edge of the baffle and the water immediately below it. The motion of the other edges of the baffle coincides with that of the tank.

The water is subjected to gravity loading. Consequently, an initial geostatic stress field is defined to equilibrate the stresses caused by the self-weight of the water. A velocity pulse in the form of a sine wave with an amplitude of 63.5 mm (2.5 inches) and a period of 2 seconds is prescribed for the tank in both the *x*- and *y*-directions simultaneously. All remaining degrees of freedom for the tank are fully constrained. The sloshing analysis is performed for two seconds.

Adaptive meshing

A single adaptive mesh domain that incorporates the water is defined. Sliding boundary regions are used for all contact surface definitions on the water (the default). Because the sloshing phenomenon modeled in this example results in large mesh motions, it is necessary to increase the frequency and intensity of adaptive meshing. The frequency value is reduced to 5 increments from a default value of 10, and the number of mesh sweeps used to smooth the mesh is increased to 3 from a default value of 1. The adaptive meshing algorithm is configured in this example so that the initial mesh gradation of the water is preserved while continuous adaptive meshing is performed. The default values are used for all other parameters and controls.

Results and discussion

Figure 3, *Figure 4*, and *Figure 5* show the deformed mesh configuration at $t =$ 1.2 s, $t =$ 1.6 s, and $t =$ 2.0 s, respectively. Four time histories of the vertical displacement of the water level are shown in *Figure 6*; these correspond to the water level at the baffle in the front and back of the left and right bays. The locations at which the time histories are measured are denoted A, B, C, and D in *Figure 1*. An analysis such as this could be used to design a baffle that attenuates sloshing at certain frequencies. Using adaptivity in Abaqus/Explicit is appropriate for sloshing problems in which the structural response is of primary interest. It is generally not possible to model such flow behaviors as splashing or complex free surface interactions. Furthermore, surface tension is not modeled.

Input files

ale_water_sloshing.inp	Input data for this analysis.
ale_water_sloshingel.inp	Element data.

Figures

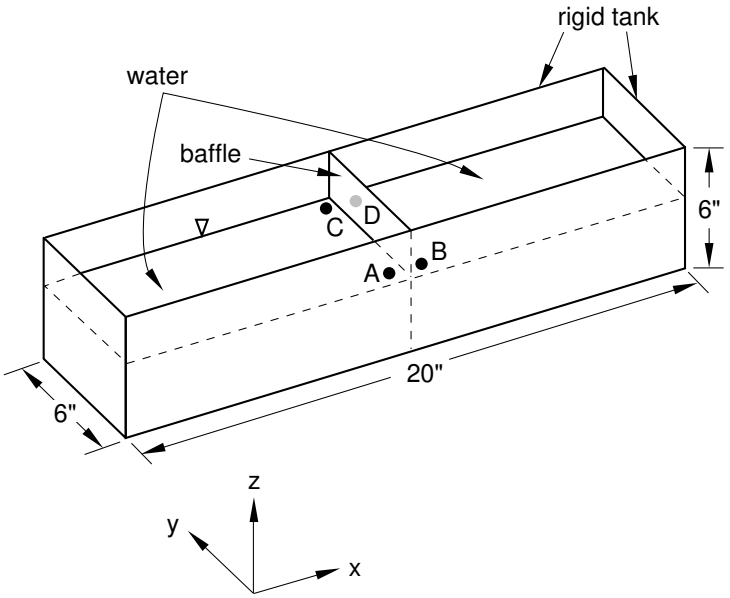


Figure 1: Model geometry.

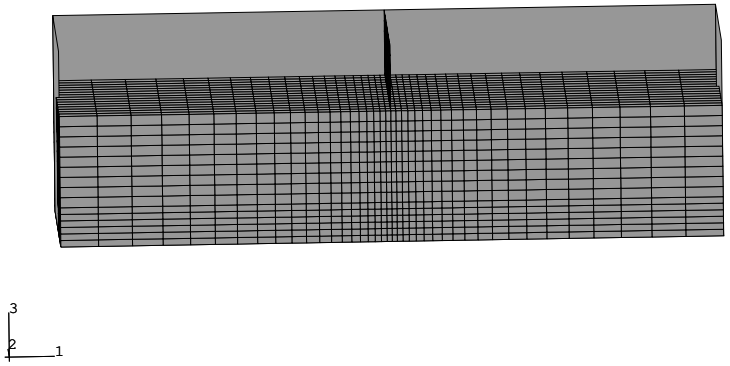


Figure 2: Initial configuration (front of rigid tank is not shown).

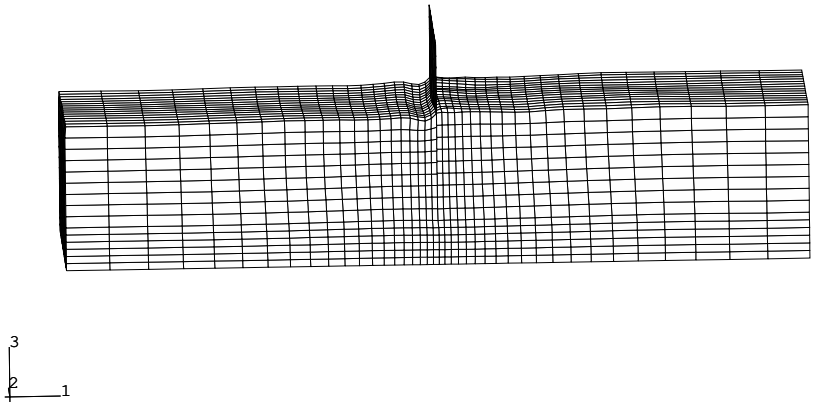


Figure 3: Deformed configuration of the water and the baffle at 1.2 seconds.

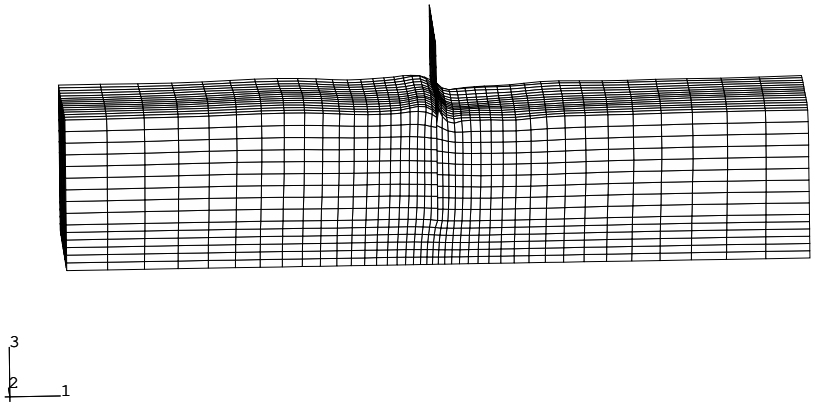


Figure 4: Deformed configuration of the water and the baffle at 1.6 seconds.

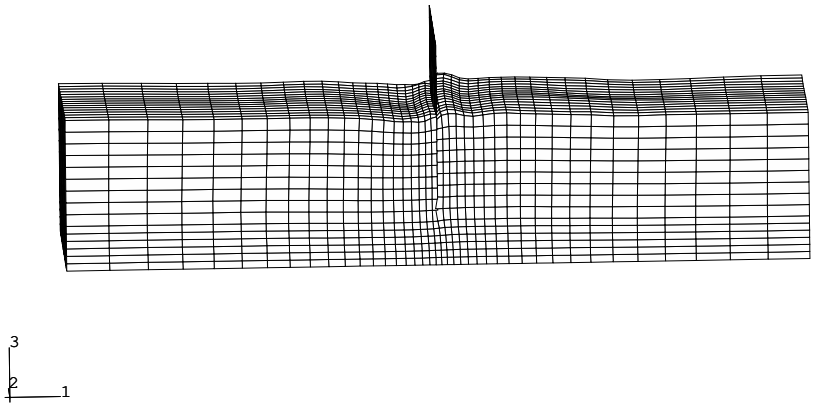


Figure 5: Deformed configuration of the water and the baffle at 2.0 seconds.

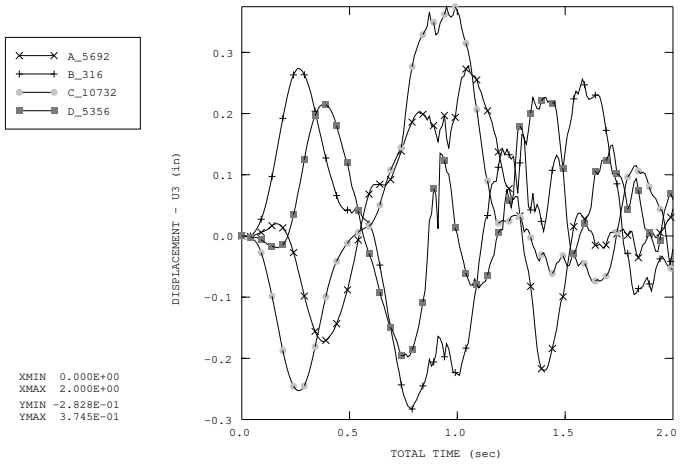


Figure 6: Time histories of the vertical displacement of the water at the baffle at both the front and back of the left and right bays.

Seismic analysis of a concrete gravity dam

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example illustrates a typical application of the concrete damaged plasticity material model for the assessment of the structural stability and damage of concrete structures subjected to arbitrary loading.

We consider an analysis of the Koyna dam, which was subjected to an earthquake of magnitude 6.5 on the Richter scale on December 11, 1967. This problem is chosen because it has been extensively analyzed by a number of investigators, including Chopra and Chakrabarti (1973), Bhattacharjee and Léger (1993), Ghrib and Tinawi (1995), Cervera et al. (1996), and Lee and Fenves (1998).

Problem description

The geometry of a typical non-overflow monolith of the Koyna dam is illustrated in *Figure 1*. The monolith is 103 m high and 71 m wide at its base. The upstream wall of the monolith is assumed to be straight and vertical, which is slightly different from the real configuration. The depth of the reservoir at the time of the earthquake is h_w = 91.75 m. Following the work of other investigators, we consider a two-dimensional analysis of the non-overflow monolith assuming plane stress conditions. The finite element mesh used for the analysis is shown in *Figure 2*. It consists of 760 first-order, reduced-integration, plane stress elements (CPS4R). Nodal definitions are referred to a global rectangular coordinate system centered at the lower left corner of the dam, with the vertical y-axis pointing in the upward direction and the horizontal x-axis pointing in the downstream direction. The transverse and vertical components of the ground accelerations recorded during the Koyna earthquake are shown in *Figure 3* (units of g = 9.81 m sec^{-2}). Prior to the earthquake excitation, the dam is subjected to gravity loading due to its self-weight and to the hydrostatic pressure of the reservoir on the upstream wall.

For the purpose of this example we neglect the dam–foundation interactions by assuming that the foundation is rigid. The dam–reservoir dynamic interactions resulting from the transverse component of ground motion can be modeled in a simple form using the Westergaard added mass technique. According to Westergaard (1933), the hydrodynamic pressures that the water exerts on the dam during an earthquake are the same as if a certain body of water moves back and forth with the dam while the remainder of the reservoir is left inactive. The added mass per unit area of the upstream wall is given in approximate form by the expression $\frac{7}{8}\rho_w\sqrt{h_w\,(h_w - y)}$, with $y \le h_w$, where ρ_w = 1000 kg/m^3 is the density of water. In the Abaqus/Standard analysis the added mass approach is implemented using a simple 2-node user element that has been coded in user subroutine *UEL*. In the Abaqus/Explicit analysis the dynamic interactions between the dam and the reservoir are ignored.

The hydrodynamic pressures resulting from the vertical component of ground motion are assumed to be small and are neglected in all the simulations.

Material properties

The mechanical behavior of the concrete material is modeled using the concrete damaged plasticity constitutive model described in *Concrete Damaged Plasticity* and *Damaged plasticity model for concrete and other quasi-brittle materials*. The material properties used for the simulations are given in *Table 1* and *Figure 4*. These properties are assumed to be representative of the concrete material in the Koyna dam and are based on the properties used by previous investigators. In obtaining some of these material properties, a number of assumptions are made. Of particular interest is the calibration of the concrete tensile behavior. The tensile strength is estimated to be 10% of the ultimate compressive strength (σ_{cu} = 24.1 MPa), multiplied by a dynamic amplification factor

of 1.2 to account for rate effects; thus, σ_{t0} = 2.9 MPa. To avoid unreasonable mesh-sensitive results due to the lack of reinforcement in the structure, the tensile postfailure behavior is given in terms of a fracture energy cracking criterion by specifying a stress/displacement curve instead of a stress-strain curve, as shown in *Figure 4*(a). This is accomplished with the postcracking stress/displacement curve. Similarly, tensile damage, d_t, is specified in tabular form as a function of cracking displacement by using the postcracking damage displacement curve. This curve is shown in *Figure 4*(b). The stiffness degradation damage caused by compressive failure (crushing) of the concrete, d_c, is assumed to be zero.

Damping

It is generally accepted that dams have damping ratios of about 2–5%. In this example we tune the material damping properties to provide approximately 3% fraction of critical damping for the first mode of vibration of the dam. Assuming Rayleigh stiffness proportional damping, the factor β required to provide a fraction ξ_1 of critical damping for the first mode is given as $\beta = 2\xi_1/\omega_1$. From a natural frequency extraction analysis of the dam the first eigenfrequency is found to be ω_1 = 18.61 rad sec^{-1} (see *Table 2*). Based on this, β is chosen to be 3.23×10^{-3} sec.

Loading and solution control

Loading conditions and solution controls are discussed for each analysis.

Abaqus/Standard analysis

Prior to the dynamic simulation of the earthquake, the dam is subjected to gravity loading and hydrostatic pressure. In the Abaqus/Standard analysis these loads are specified in two consecutive static steps, using a distributed load with the load type labels GRAV (for the gravity load) in the first step and HP (for the hydrostatic pressure) in the second step. For the dynamic analysis in the third step the transverse and vertical components of the ground accelerations shown in *Figure 3* are applied to all nodes at the base of the dam.

Since considerable nonlinearity is expected in the response, including the possibility of unstable regimes as the concrete cracks, the overall convergence of the solution in the Abaqus/Standard analysis is expected to be non-monotonic. In such cases automatically setting the time incrementation parameters is generally recommended to prevent premature termination of the equilibrium iteration process because the solution may appear to be diverging. The unsymmetric matrix storage and solution scheme is activated by specifying an unsymmetric equation solver for the step. This is essential for obtaining an acceptable rate of convergence with the concrete damaged plasticity model since plastic flow is nonassociated. Automatic time incrementation is used for the dynamic analysis of the earthquake, with the half-increment residual tolerance set to 10^7 and a maximum time increment of 0.02 sec.

Abaqus/Explicit analysis

While it is possible to perform the analysis of the pre-seismic state in Abaqus/Explicit, Abaqus/Standard is much more efficient at solving quasi-static analyses. Therefore, we apply the gravity and hydrostatic loads in an Abaqus/Standard analysis. These results are then imported into Abaqus/Explicit to continue with the seismic analysis of the dam subjected to the earthquake accelerogram. We still need to continue to apply the gravity and hydrostatic pressure loads during the explicit dynamic step. In Abaqus/Explicit gravity loading is specified in exactly the same way as in Abaqus/Standard. The specification of the hydrostatic pressure, however, requires some extra consideration because this load type is not currently supported by Abaqus/Explicit. Here we apply the hydrostatic pressure using user subroutine *VDLOAD*.

The Abaqus/Explicit simulation requires a very large number of increments since the stable time increment (6×10^{-6} sec) is much smaller than the total duration of the earthquake (10 sec). The analysis is run in double

precision to prevent the accumulation of round-off errors. The stability limit could be increased by using mass scaling; however, this may affect the dynamic response of the structure.

For this particular problem Abaqus/Standard is computationally more effective than Abaqus/Explicit because the earthquake is a relatively long event that requires a very large number of increments in Abaqus/Explicit. In addition, the size of the finite element model is small, and the cost of each solution of the global equilibrium equations in Abaqus/Standard is quite inexpensive.

Results and discussion

The results for each analysis are discussed in the following sections.

Abaqus/Standard results

The results from a frequency extraction analysis of the dam without the reservoir are summarized in *Table 2*. The first four natural frequencies of the finite element model are in good agreement with the values reported by Chopra and Chakrabarti (1973). As discussed above, the frequency extraction analysis is useful for the calibration of the material damping to be used during the dynamic simulation of the earthquake.

Figure 5 shows the horizontal displacement at the left corner of the crest of the dam relative to the ground motion. In this figure positive values represent displacement in the downstream direction. The crest displacement remains less than 30 mm during the first 4 seconds of the earthquake. After 4 seconds, the amplitude of the oscillations of the crest increases substantially. As discussed below, severe damage to the structure develops during these oscillations.

The concrete material remains elastic with no damage at the end of the second step, after the dam has been subjected to the gravity and hydrostatic pressure loads. Damage to the dam initiates during the seismic analysis in the third step. The evolution of damage in the concrete dam at six different times during the earthquake is illustrated in *Figure 6*, *Figure 7*, and *Figure 8*. Times t_1 = 3.96 sec, t_3 = 4.315 sec, and t_5 = 4.687 sec correspond to the first three large excursions of the crest in the upstream direction, as shown in *Figure 5*. Times t_2 = 4.163 sec and t_4 = 4.526 sec correspond to the first two large excursions of the crest in the downstream direction. Time t_6 = 10 sec corresponds to the end of the earthquake. The figures show the contour plots of the tensile damage variable, DAMAGET (or d_t), on the left, and the stiffness degradation variable, SDEG (or d), on the right. The tensile damage variable is a nondecreasing quantity associated with tensile failure of the material. On the other hand, the stiffness degradation variable can increase or decrease, reflecting the stiffness recovery effects associated with the opening/closing of cracks. Thus, assuming that there is no compressive damage ($d_c = 0$), the combination $d_t > 0$ and $d > 0$ at a given material point represents an open crack, whereas $d_t > 0$ and $d = 0$ represents a closed crack.

At time t_1, damage has initiated at two locations: at the base of the dam on the upstream face and in the region near the stress concentration where the slope on the downstream face changes.

When the dam displaces toward the downstream direction at time t_2, the damage at the base leads to the formation of a localized crack-like band of damaged elements. This crack propagates into the dam along the dam–foundation boundary. The nucleation of this crack is induced by the stress concentration in this area due to the infinitely rigid foundation. At this time, some partial tensile damage is also observed on several elements along the upstream face.

During the next large excursion in the upstream direction, at time t_3, a localized band of damaged elements forms near the downstream change of slope. As this downstream crack propagates toward the upstream direction, it curves down due to the rocking motion of the top block of the dam. The crack at the base of the dam is closed at time t_3 by the compressive stresses in this region. This is easily verified by looking at the contour plot of

SDEG at time t_3, which clearly shows that the stiffness is recovered on this region, indicating that the crack is closed.

When the load is reversed, corresponding to the next excursion in the downstream direction at time t_4, the downstream crack closes and the stiffness is recovered on that region. At this time tensile damage localizes on several elements along the upstream face, leading to the formation of a horizontal crack that propagates toward the downstream crack.

As the upper block of the dam oscillates back and forth during the remainder of the earthquake, the upstream and downstream cracks close and open in an alternate fashion. The dam retains its overall structural stability since both cracks are never under tensile stress during the earthquake. The distribution of tensile damage at the end of the earthquake is shown in *Figure 8* at time t_6. The contour plot of the stiffness degradation variable indicates that, except at the vicinity of the crack tips, all cracks are closed under compressive stresses and most of the stiffness is recovered. No compressive failure is observed during the simulation. The damage patterns predicted by Abaqus are consistent with those reported by other investigators.

Abaqus/Explicit results

Figure 9 shows the distribution of tensile damage at the end of the Abaqus/Explicit simulation. Two major cracks develop during the earthquake, one at the base of the dam and the other at the downstream change of slope. If we compare these results with those from the analysis in Abaqus/Standard (see *Figure 8* at time t_6), we find that Abaqus/Standard predicted additional damage localization zones on the upstream face of the dam. The differences between the results are due to the effect of the dam–reservoir hydrodynamic interactions, which are included in the Abaqus/Standard simulation via an added-mass user element and are ignored in Abaqus/Explicit. This is easily verified by running an Abaqus/Standard analysis without the added-mass user element. The results from this analysis, shown in *Figure 10*, are consistent with the Abaqus/Explicit results in *Figure 9* and confirm that additional damage to the upstream wall occurs when the hydrodynamic interactions are taken into account.

Input files

Abaqus/Standard input files

koyna_freq.inp	Frequency analysis of the Koyna dam.
koyna_std.inp	Seismic analysis of the Koyna dam, including hydrodynamic interactions.
koyna2_std.inp	Seismic analysis of the Koyna dam, not including hydrodynamic interactions.
koyna_haccel.inp	Transverse ground acceleration record.
koyna_vaccel.inp	Vertical ground acceleration record.
addedmass_uel.f	User subroutine *UEL* used by koyna_std.inp to model hydrodynamic interactions via the added mass technique.
koyna_std_to_xpl.inp	Analysis of the pre-seismic state of the Koyna dam. These results are imported by koyna_xpl.inp.

Abaqus/Explicit input files

koyna_xpl.inp	Seismic analysis of the Koyna dam, not including hydrodynamic interactions; requires import of the results from koyna_std_to_xpl.inp.
koyna_hp_vdload.f	User subroutine *VDLOAD* used by koyna_xpl.inp to specify hydrostatic pressure.
koyna2_xpl_std.inp	Analysis of the post-seismic state of the Koyna Dam; requires import of the results from koyna_xpl.inp.

References

- Bhattacharjee, S. S., and P. Léger, "Seismic Cracking and Energy Dissipation in Concrete Gravity Dams," *Earthquake Engineering and Structural Dynamics*, vol. 22, pp. 991–1007, 1993.
- Cervera, M., J. Oliver, and O. Manzoli, "A Rate-Dependent Isotropic Damage Model for the Seismic Analysis of Concrete Dams," *Earthquake Engineering and Structural Dynamics*, vol. 25, pp. 987–1010, 1996.
- Chopra, A. K., and P. Chakrabarti, "The Koyna Earthquake and the Damage to Koyna Dam," *Bulletin of the Seismological Society of America*, vol. 63, no. 2, pp. 381–397, 1973.
- Ghrib, F., and R. Tinawi, "An Application of Damage Mechanics for Seismic Analysis of Concrete Gravity Dams," *Earthquake Engineering and Structural Dynamics*, vol. 24, pp. 157–173, 1995.
- Lee, J., and G. L. Fenves, "A Plastic-Damage Concrete Model for Earthquake Analysis of Dams," *Earthquake Engineering and Structural Dynamics*, vol. 27, pp. 937–956, 1998.
- Westergaard, H. M., "Water Pressures on Dams during Earthquakes," *Transactions of the American Society of Civil Engineers*, vol. 98, pp. 418–433, 1933.

Tables

Table 1: Material properties for the Koyna dam concrete.

Young's modulus:	E = 31027 MPa
Poisson's ratio:	ν = 0.15
Density:	ρ = 2643 kg/m^3
Dilation angle:	ψ = 36.31^o
Compressive initial yield stress:	σ_{c0} = 13.0 MPa
Compressive ultimate stress:	σ_{cu} = 24.1 MPa
Tensile failure stress:	σ_{t0} = 2.9 MPa

Table 2: Natural frequencies of the Koyna dam.

Mode	Natural Frequency (rad sec^{-1})	
	Abaqus	Chopra and Chakrabarti (1973)
1	18.86	19.27
2	49.97	51.50
3	68.16	67.56
4	98.27	99.73

Figures

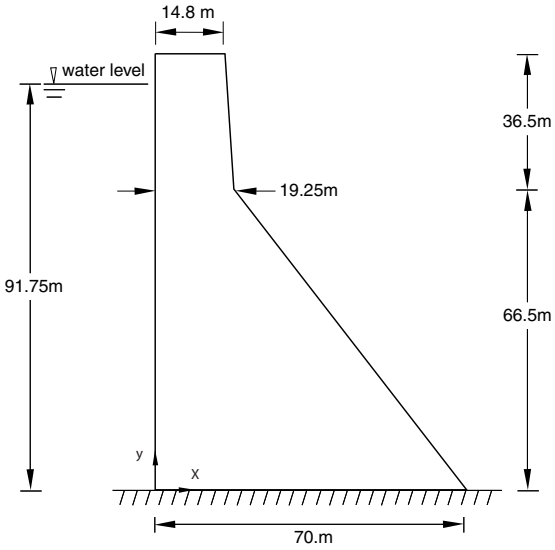


Figure 1: Geometry of the Koyna dam.

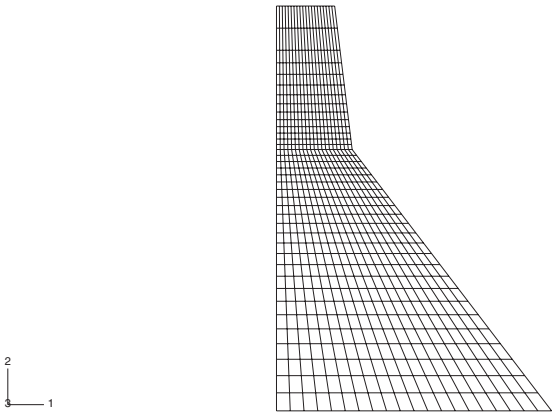


Figure 2: Finite element mesh.

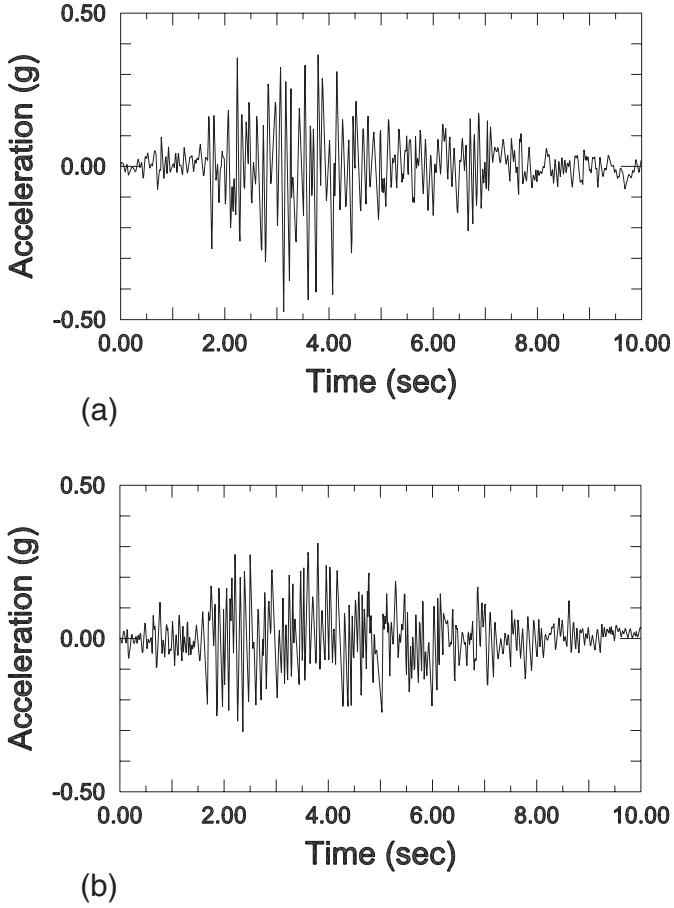


Figure 3: Koyna earthquake: (a) transverse and (b) vertical ground accelerations.

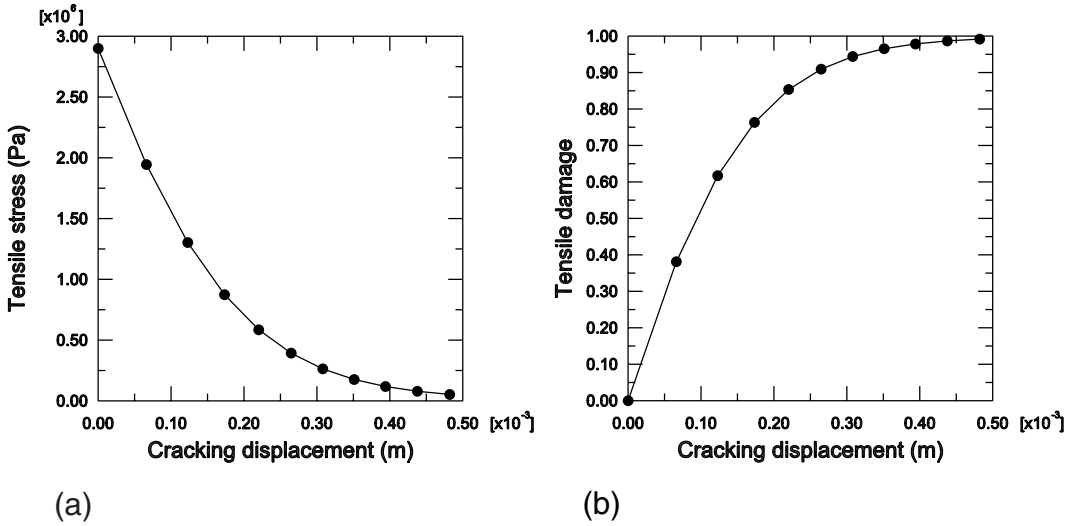


Figure 4: Concrete tensile properties: (a) tension stiffening and (b) tension damage.

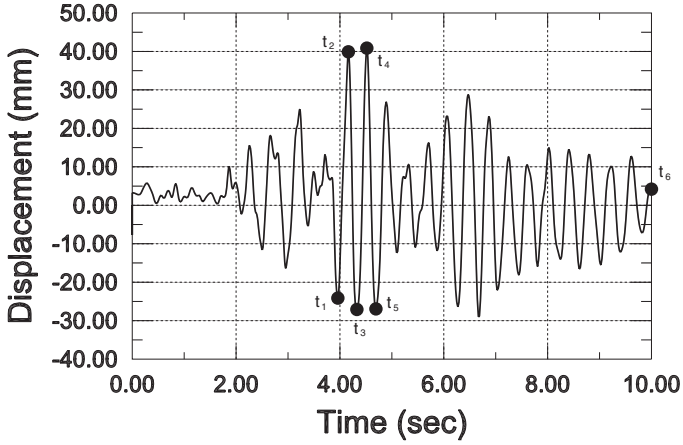


Figure 5: Horizontal crest displacement (relative to ground displacement).

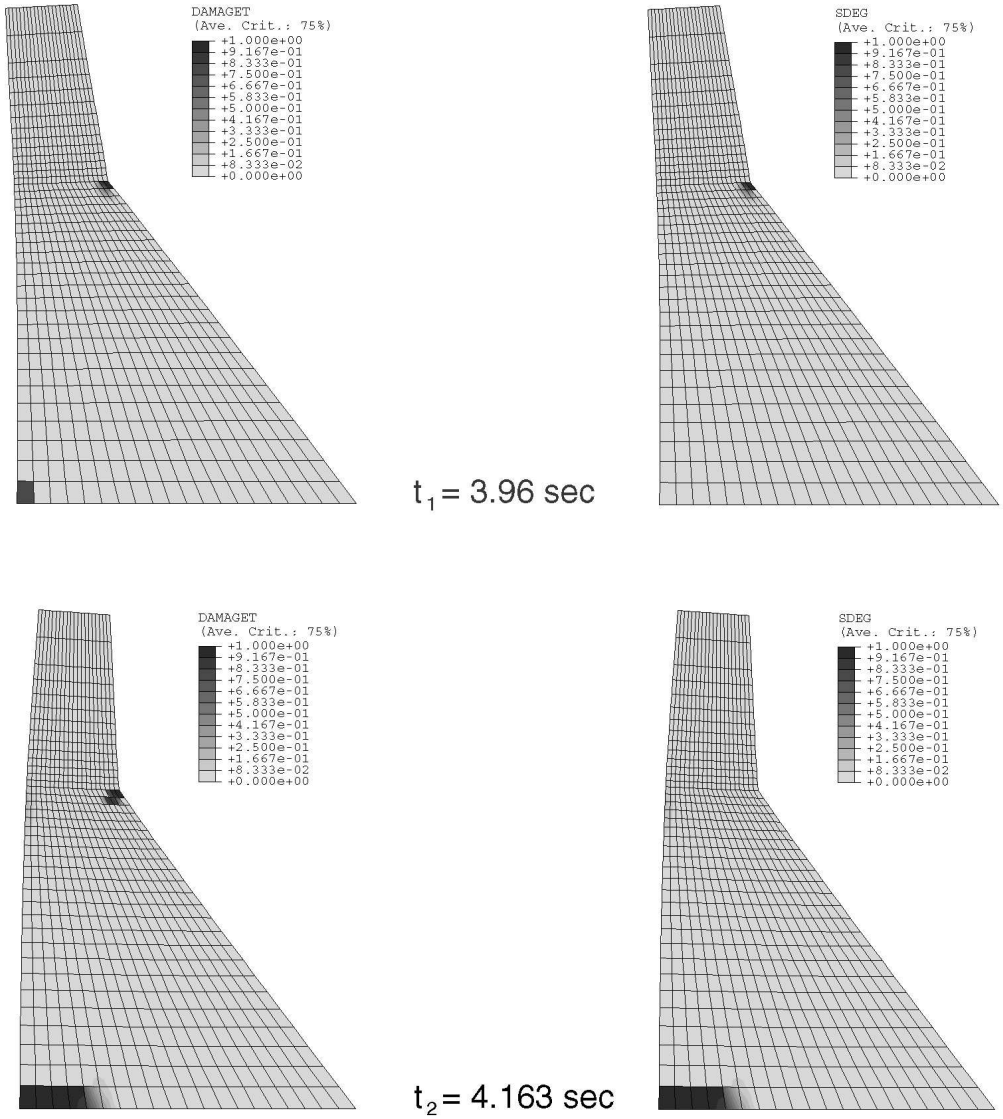


Figure 6: Evolution of tensile damage (Abaqus/Standard); deformation scale factor = 100.

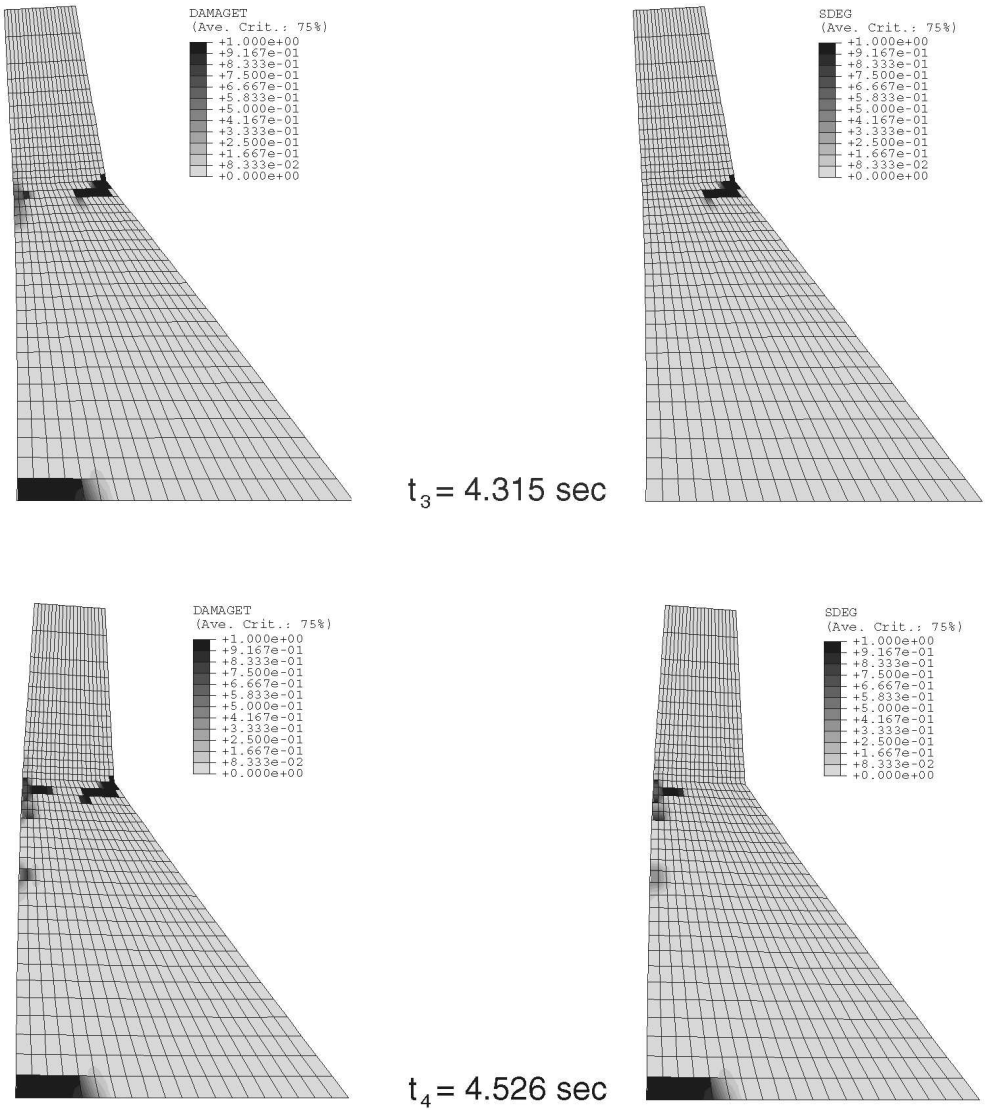


Figure 7: Evolution of tensile damage (Abaqus/Standard); deformation scale factor = 100.

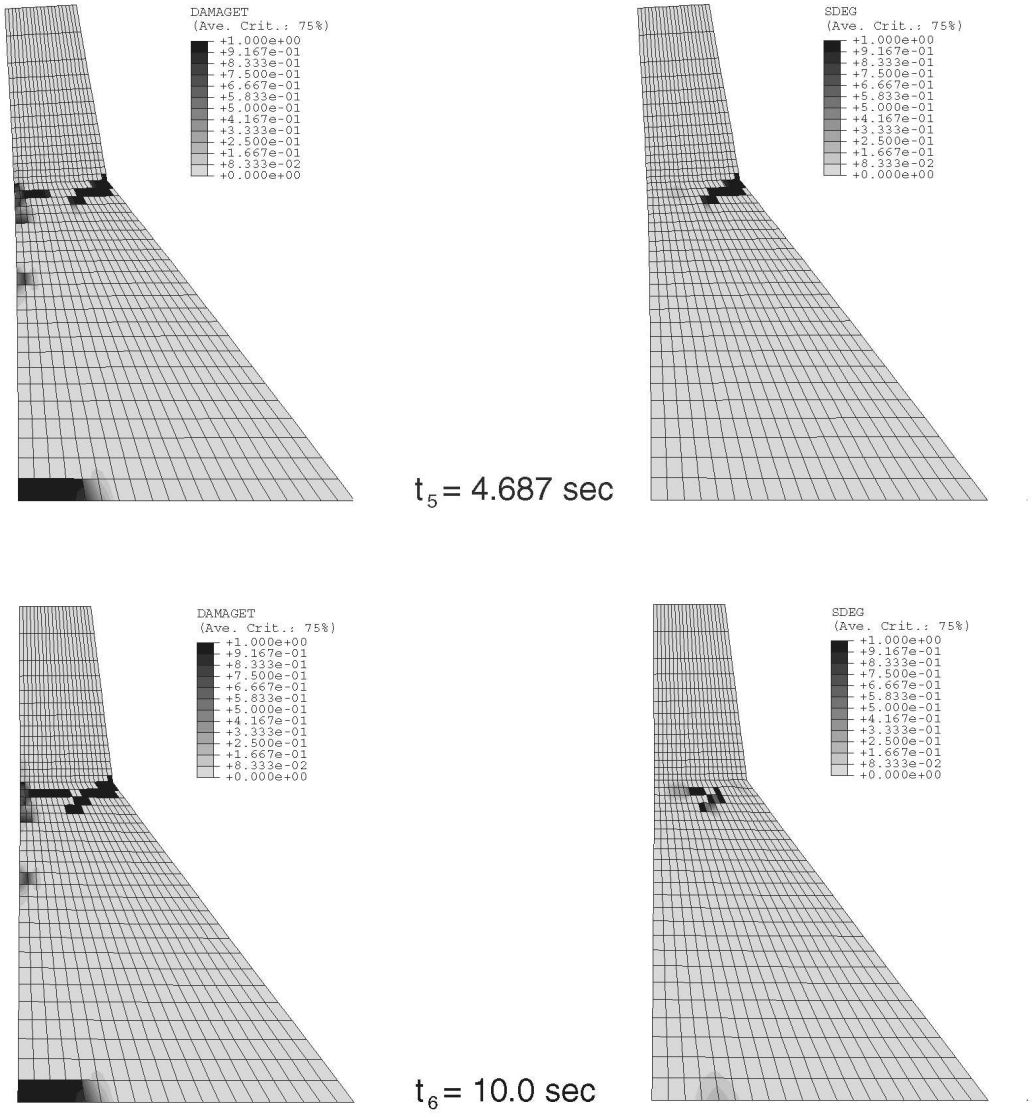


Figure 8: Evolution of tensile damage (Abaqus/Standard); deformation scale factor = 100.

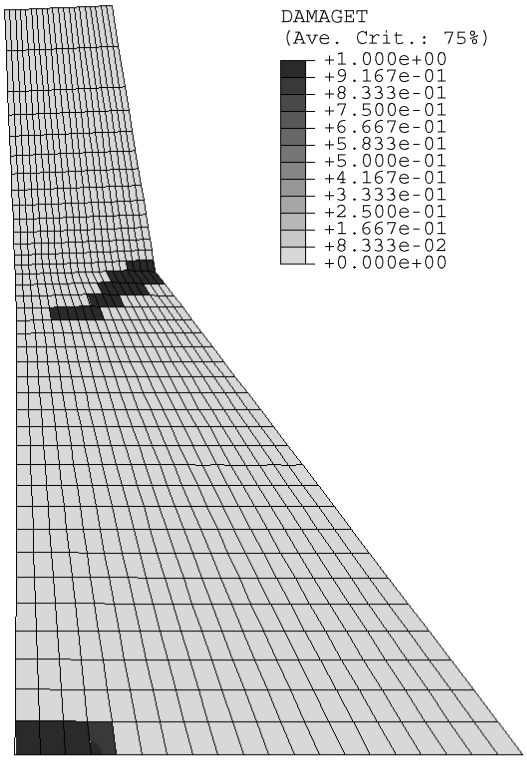


Figure 9: Tensile damage at the end of the Abaqus/Explicit simulation without dam–reservoir hydrodynamic interactions; deformation scale factor = 100.

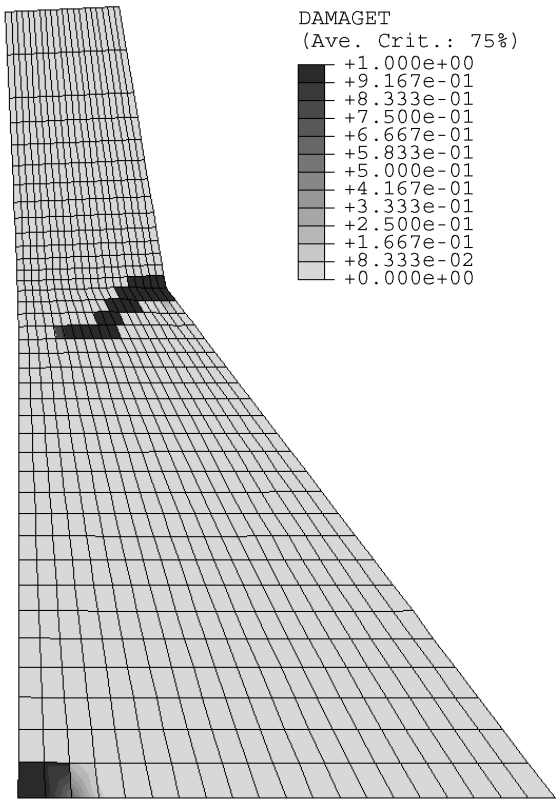


Figure 10: Tensile damage at the end of the Abaqus/Standard simulation without dam–reservoir hydrodynamic interactions; deformation scale factor = 100.

Progressive failure analysis of thin-wall aluminum extrusion under quasi-static and dynamic loads

Products: Abaqus/Explicit

References:

- *About Progressive Damage and Failure*
- **DAMAGE INITIATION*
- **DAMAGE EVOLUTION*
- *Progressive damage and failure of ductile metals*

Overview

In this example problem we consider the overall deformation and failure behavior of a thin-wall, double-chambered aluminum extrusion under quasi-static three-point bending and dynamic axial loading conditions.

The following Abaqus features are demonstrated:

- using ductile, shear, and Müschenborn-Sonne forming limit diagram (MSFLD) damage initiation criteria to study the initiation of failure due to three different mechanisms: ductile fracture, shear band formation, and necking instability, respectively; and
- modeling progressive failure of components using damage evolution and element removal.

The overall load-displacement response and the fracture patterns are compared with the experimental results given by Hooputra et al. (2004).

Application description

New materials such as aluminum and magnesium alloys and high-strength steels are being introduced increasingly in automotive components to reduce weight and, hence, to increase overall vehicle performance. These materials typically have low ductility at fracture compared to traditional steels and may suffer damage and failure under crash loading conditions. A typical component made of sheet metal may undergo damage due to a number of mechanisms including void nucleation and coalescence, shear band formation, and necking instability. Thus, to obtain reliable predictions from crashworthiness simulations, it is essential to model damage initiation and progressive failure due to various failure mechanisms as well as modeling accurate plastic deformation behavior.

Geometry

The three-point bending and the axial crushing configurations are shown in *Figure 1* and *Figure 6*, respectively. The overall dimensions of the aluminum extrusion are L=500 mm, W=95 mm, and H=68 mm for the three-point bending case and L 396.5 mm, W=95 mm, and H=68 mm for the axial crushing case. The thickness of the sheet is 2.5 mm for both cases.

Materials

The material used in this study is an extruded aluminum alloy EN AW-7108 T6. This material behaves in an elastic-plastic manner and can undergo damage due to either one or a combination of the following damage mechanisms: nucleation and coalescence of voids, shear band formation, and necking instability.

Boundary conditions and loading

The three-point bending configuration consists of the aluminum extrusion supported on two rigid cylinders and loaded in the transverse direction by another rigid cylinder (*Figure 1*). In the axial crushing simulation, one end of the aluminum extrusion is supported by a fixed rigid base and the other end is subjected to an instantaneous velocity by a planar rigid impactor (*Figure 6*).

Abaqus modeling approaches and simulation techniques

Two loading cases are considered. The first case consists of a quasi-static three-point bending configuration where the part is loaded transversely to the extrusion direction. In the second case the part is subjected to a dynamic loading in the axial (extrusion) direction.

Summary of analysis cases

Case 1	Quasi-static three-point bending simulation.
Case 2	Dynamic axial crushing simulation.

The sections that follow discuss the analysis considerations that are applicable to both cases.

Mesh design

In both cases the mesh is similar to that used by Hooputra et al. (2004). The aluminum extrusion is meshed with a uniform mesh consisting primarily of 4-node shell elements (S4R). In the axial crushing case some 3-node shell elements (S3R) are also used. The planar dimensions of the elements are an order of magnitude larger than the shell thickness. The simulations with this mesh yield results in agreement with the experimental observations. No mesh refinement studies were conducted.

Materials

The details of the Abaqus models used for constitutive behavior and progressive damage analysis are discussed below. Guidelines for obtaining the material parameters from experimental data are also provided.

Elastoplasticity

Hooputra et al. (2004) have shown that the extruded aluminum alloy EN AW-7108 T6 displays plastic orthotropy due to the nature of the extrusion processing and have used the Barlat symmetric yield locus (Barlat et al., 1991) to fit the experimental data. In this example we neglect the orthotropy and assume both the elastic and the plastic behavior to be isotropic with the yield surface described by the Mises yield function (see *Inelastic Behavior*). The assumption of isotropic plasticity may appear to be too restrictive for the accurate prediction of failure in extruded alloys. However, in crashworthiness simulations the assumption of isotropy usually yields acceptable results when compared with experimental observations, as shown in the results obtained in this example. Nevertheless, you should compare your simulation results with experimental data to check the validity of the isotropic plasticity assumption.

Damage initiation

Metal sheets and thin-walled extrusions made of aluminum alloys may fail due to one or a combination of the following failure mechanisms (Hooputra et al. 2004): ductile failure due to nucleation, growth, and coalescence of voids; shear failure due to fracture within shear bands; and failure due to necking instabilities. If the model consists of shell elements, a criterion for the last failure mechanism is necessary because the size of the localized neck is of the order of the sheet thickness and, hence, cannot be resolved with shell elements of dimensions one order of magnitude larger than the thickness.

Abaqus/Explicit offers a number of damage initiation criteria to model the onset of necking instabilities in sheet metals. These include the Forming Limit Diagram (FLD), Forming Limit Stress Diagram (FLSD), Müschenborn-Sonne Forming Limit Diagram (MSFLD), and Marciniak-Kuczynski (M-K) criteria. The first three criteria utilize the experimentally measured forming limit curves in the appropriate strain or stress spaces. The last criterion introduces virtual thickness imperfections in the sheet metal and analyzes the deformation in the imperfection zone to determine the onset of the instability (see *Damage Initiation for Ductile Metals*).

The strain-based FLD criterion is limited to applications where the strain path is linear. On the other hand, the stress-based FLSD criterion is relatively insensitive to changes in the strain path. However, this apparent independence of the stress-based limit curve due to the strain path may simply reflect the small sensitivity of the yield stress to changes in the plastic deformation. The M-K criterion can capture the effects of nonlinear strain paths accurately; however, it is computationally expensive, especially if large numbers of imperfection orientations are introduced. It has been verified that the results obtained using the MSFLD criterion are similar to those obtained using the M-K criterion but with a much reduced computational expense (see *Progressive damage and failure of ductile metals*). Therefore, in this example we choose the MSFLD damage initiation criterion for necking instability.

For specifying the MSFLD damage initiation criterion, the forming limit curve of the material is required. In Abaqus this criterion can be specified by converting the forming limit curve from the space of major versus minor strains to the space of equivalent plastic strain versus ratio of principal strain rates. Abaqus also allows direct specification of the forming limit curve for the MSFLD criterion (see *Müschenborn-Sonne Forming Limit Diagram (MSFLD) Criterion*). We use the forming limit curve based on the experimental work of Hooputra (2005). This curve is assumed to be valid at both the quasi-static and the dynamic strain rates. The parameter OMEGA used in conjunction with the MSFLD criterion to provide filtering of numerical noise in the evaluation of the ratio of principal strain rates is set to 0.001 in both cases (see *Damage Initiation for Ductile Metals*); this value is recommended for crashworthiness simulations.

Damage due to initiation, growth, and coalescence of voids leads to ductile failure in metals; the formation of cracks within shear bands leads to shear failure. Abaqus offers phenomenological damage initiation criteria for both of these mechanisms. The ductile criterion is specified by providing the equivalent plastic strain at the onset of ductile damage as a function of stress triaxiality and strain rate. Similarly, the shear criterion is specified by providing the equivalent plastic strain at the onset of shear damage as a function of shear stress ratio and strain rate (see *Damage Initiation for Ductile Metals*). The data required for both of these criteria may be difficult to obtain through direct experimentation since it would require experiments spanning a range of stress triaxiality and shear stress ratio that may be difficult to achieve. Hooputra et al. (2004) have given simplified analytical expressions for the ductile and the shear failure criteria that require only a limited number of experiments. In this example we adopt those expressions; however, we ignore the orthotropy of the ductile fracture to be consistent with the assumption of isotropic plasticity made earlier.

For the ductile damage initiation criterion the equivalent plastic strain is given by the following function of the stress triaxiality, η (Hooputra et al, 2004):

$$\bar{\varepsilon}_D^{pl}\left(\eta, \dot{\bar{\varepsilon}}^{pl}\right) = \frac{\varepsilon_T^+ \sinh\left[k_0\left(\eta^- - \eta\right)\right] + \varepsilon_T^- \sinh\left[k_0\left(\eta - \eta^+\right)\right]}{\sinh\left[k_0\left(\eta^- - \eta^+\right)\right]},$$

where ε_T^+ and ε_T^- correspond to the equivalent plastic strain at ductile damage initiation for equibiaxial tensile and equibiaxial compressive deformation, respectively. For isotropic materials the stress triaxiality in equibiaxial tensile deformation state, η^+, is $2/3$, and in equibiaxial compressive deformation state, η^-, is $-2/3$. The definition of η in Abaqus, as a ratio of the equivalent mean stress to the Mises equivalent stress, differs from that used by Hooputra et al. (2004) by a factor of $1/3$. Consequently, the value of k_0 used in the above expression is three times the value used in Hooputra et al. (2004). The above expression has three parameters that must be obtained experimentally: ε_T^+, ε_T^-, and k_0. These parameters depend on the material, strain rate, and possibly the temperature. For each strain rate of interest, three experiments are needed at different values of stress triaxiality to obtain the

three material parameters. ε_T^+ can be obtained directly from the Erichsen test ($\eta = \eta^+$). Three-point bending of sheet coupons (with width/thickness > 4) under plane strain tension ($\eta = 1/\sqrt{3}$) and fracture at the notch root of waisted tensile coupons in uniaxial tension ($\eta = 1/3$) may provide two additional experiments to determine ε_T^- and k_0. In the Erichsen and three-point bending experiments the local fracture strain can be derived by placing a grid on the specimen's surface; in the waisted tensile experiment the fracture strain can be obtained from the sheet thickness in the fracture plane (Hooputra et al., 2004). For the aluminum alloy used in this example the experimentally obtained ductile failure parameters at quasi-static and dynamic strain rates (250 s^{-1}) are listed in *Table 1*.

For the shear damage initiation criterion the equivalent plastic strain at the onset of damage is given by the following function of the shear stress ratio, θ_s (Hooputra et al, 2004):

$$\bar{\varepsilon}_S^{pl}\left(\theta_s, \dot{\bar{\varepsilon}}^{pl}\right) = \frac{\varepsilon_S^+ \sinh[f(\theta_s - \theta_s^-)] + \varepsilon_S^- \sinh[f(\theta_s^+ - \theta_s)]}{\sinh[f(\theta_s^+ - \theta_s^-)]},$$

where $\theta_s = (1 - k_s\eta)/\phi$ with $\phi = \tau_{max}/\sigma_{eq}$, and ε_S^+ and ε_S^- correspond to the equivalent plastic strain at shear damage initiation for equibiaxial tensile and equibiaxial compressive deformation, respectively. The parameters θ_s^+ and θ_s^- correspond to the values of θ_s at $\eta = \eta^+$ and $\eta = \eta^-$, respectively. This expression has four parameters that must be determined experimentally: k_s, ε_S^+, ε_S^-, and f. These parameters depend on the material and strain rate. Hooputra et al. (2004) have used tensile specimens with a groove (rectangular cross-section and groove depth=half the sheet thickness) at $45°$ to the loading direction ($\theta_s = 1.469$), specially designed tensile specimens with a groove parallel to the loading direction (pure shear, $\theta_s = 1.732$), and Erichsen tests ($\theta_s = 1.6$) in conjunction with the above expression to determine ε_S^+, ε_S^-, and f. The value of the material parameter k_s is taken as 0.3. For the aluminum alloy used in this example the experimentally obtained shear failure parameters at quasi-static and dynamic strain rates (250 s^{-1}) are listed in *Table 2*.

Using the aforementioned expressions and the material parameters listed in *Table 1* and *Table 2*, tabular data for ductile and shear damage initiation criteria can be generated as a function of stress triaxiality and shear stress ratio, respectively. This tabular data is provided in the Abaqus input files. The above expressions may give very high values of the equivalent plastic strain at damage initiation when the stress triaxiality or the shear stress ratio is very small. A cutoff value of the equivalent plastic strain can be provided in such cases.

Damage evolution and element removal

Damage evolution occurs once the damage initiation criteria are satisfied. Plastic displacement-based linear damage evolution law is used for each of the three damage initiation criterion. The value of the plastic displacement at which the damage variable reaches 1 is taken as 0.1. The default maximum degradation rule is used, and the elements are removed from the mesh once the maximum degradation has occurred (see *Maximum Degradation and Choice of Element Removal*).

Initial conditions

For the axial crushing simulation a velocity initial condition is specified at the reference node of the planar rigid impactor in the global 1-direction.

Boundary conditions

For the three-point bending simulation all the degrees of freedom at the reference node of the rigid supports are constrained. A velocity boundary condition in the global 2-direction is specified at the reference node of the rigid punch with all the remaining degrees of freedom constrained.

For the axial crushing simulation all the degrees of freedom at the reference node associated with the rigid support are constrained. Furthermore, all of the degrees of freedom except that associated with the global 1-direction are constrained at the reference node of the planar rigid impactor.

Loads

The velocity boundary condition at the rigid punch applies the load in the three-point bending simulation. In the case of the axial crushing simulation the initial velocity of the planar rigid impactor loads the component.

Constraints

Rigid body constraints are specified in both cases to form element-based rigid bodies. These rigid bodies form the support and apply loads to the aluminum extrusion.

Interactions

For the three-point bending simulation a contact pair interaction is defined between the rigid punch and the node-based surface of the aluminum extrusion component. A general contact interaction is defined between the rigid cylinders forming the support and the element-based surface of the aluminum extrusion component. In addition, self-contact is defined between the element-based surfaces of the extruded component. A friction coefficient of 0.05 is used for the contact between the rigid cylinders and the extruded component, and a value of 0.15 is used for the self-contact.

For the axial crushing simulation a contact pair interaction is defined between the extruded component and the rigid support as well as between the component and the rigid impactor. A general contact interaction is used for self-contact between the surfaces of the extruded component. A friction coefficient of 0.15 is used for all of the contact interactions in this case.

For both the three-point bending and axial crushing cases, a penalty-type mechanical constraint is used for all of the contact pair definitions.

Analysis steps

Both the three-point bending and the axial crushing analyses consist of one explicit dynamic step. The total simulation times in the three-point bending and the axial crushing cases are 0.072 s and 0.0501 s, respectively. Both the analyses consider geometric nonlinearity and use automatic time incrementation using element-by-element stable time increment estimates.

Output requests

For both cases the field output request includes the following quantities: displacement, stress, strain, element status, and damage initiation criteria variables. The history output request consists of displacement, velocity, acceleration, and reaction force at the reference point of the top rigid cylinder (for the three-point bending simulation) and at the reference point of the rigid impactor and the supporting base (for the axial crushing simulation). Energy output variables are requested for the entire model.

Discussion of results and comparison of cases

The overall deformed shape of the aluminum extrusion obtained from the three-point bending simulation is shown in *Figure 2*, and the experimentally observed deformed shape (Hooputra et al. 2004) is shown in *Figure 3*. The elements that have failed at the end of the simulation are shown in *Figure 4*, mapped into the undeformed configuration. Good qualitative agreement is seen between the simulation results and experimental observations. The load-displacement history of the punch obtained from the simulation is compared with three different experimental results in *Figure 5*. Again, a very good match is observed, indicating the reliability of the simulation results. In *Figure 5* the simulation results are plotted after applying the Butterworth filter with a cutoff frequency of 1000 (see *Applying Butterworth filtering to an X–Y data object*).

The overall deformed shape including the failure patterns obtained from the axial crushing simulation is shown in *Figure 7*. The deformed shape and the failure patterns are qualitatively similar to those observed experimentally (*Figure 8*). The overall force-displacement response from the simulation (filtered using the Butterworth filter with a cutoff frequency of 1500) is compared with the results from three different experiments (Hooputra, 2005) in *Figure 9*. Again, a good qualitative match is seen, and the numerical results are within the experimentally observed scatter.

In conclusion, the results from both the quasi-static three-point bending and the dynamic axial crushing simulations match the experimental data very well. It is also concluded that the use of progressive damage and failure is essential to capture the overall deformation and failure behavior of thin-wall aluminum extrusion.

Input files

Case 1: Three-point bending

threepointbending_alextrusion.inp Input file to create and analyze the model.

Case 2: Axial crushing

axialcrushing_alextrusion.inp Input file to create and analyze the model.

References

- Barlat, F., D. J. Lege, and J. C. Brem, "A Six-Component Yield Function for Anisotropic Materials," International Journal of Plasticity, vol. 7, pp. 693–712, 1991.
- Hooputra, H., H. Gese, H. Dell, and H. Werner, "A Comprehensive Failure Model for Crashworthiness Simulation of Aluminium Extrusions," International Journal of Crashworthiness, vol. 9, pp. 449–463, 2004.
- Hooputra, H., Private Communication, 2005.

Tables

Table 1: Experimentally determined ductile failure parameters (Hooputra et al., 2004).

Parameter	Quasi-static	Dynamic ($\dot{\varepsilon} = 250\,s^{-1}$)
ε_T^+	0.26	0.44
ε_T^-	193.0	1494.0
k_0	5.277	8.6304

Table 2: Experimentally determined shear failure parameters (Hooputra et al., 2004).

Parameter	Quasi-static	Dynamic ($\dot{\varepsilon} = 250\,s^{-1}$)
ε_S^+	0.26	0.35
ε_S^-	4.16	1.2
f	4.04	2.05

Parameter	Quasi-static	Dynamic ($\dot{\epsilon} = 250\, s^{-1}$)
k_s	0.3	0.3

Figures

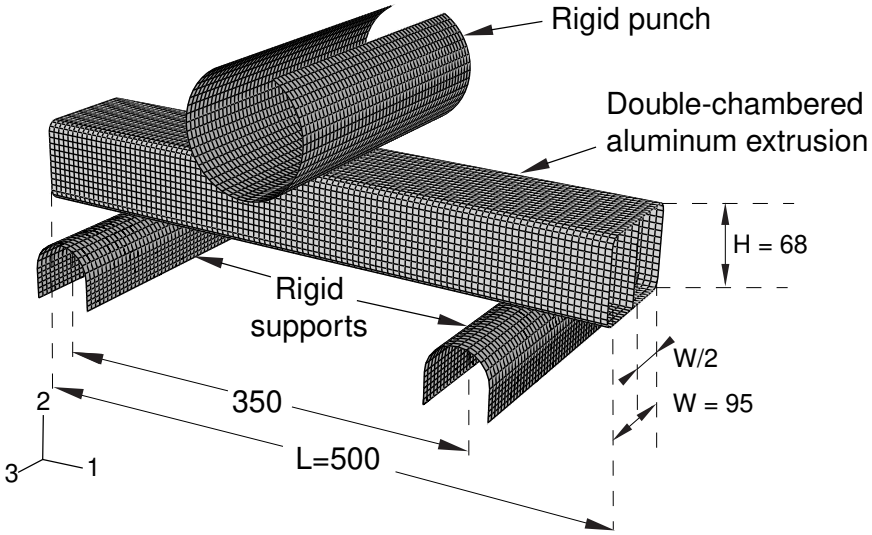


Figure 1: Three-point bending configuration: geometry and finite element mesh.

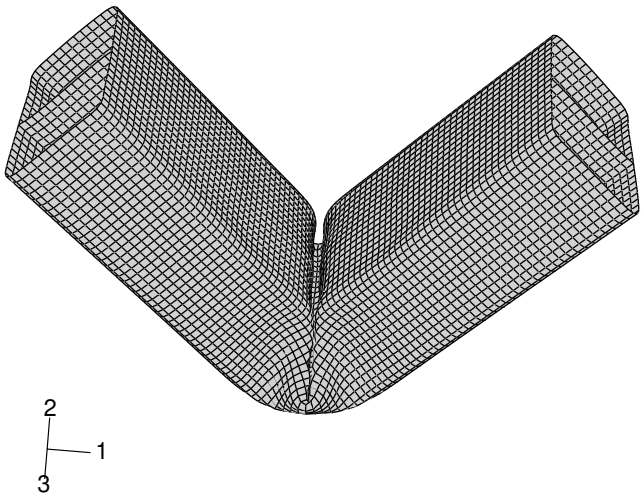


Figure 2: Final deformed shape of the aluminum extrusion in the quasi-static three-point bending simulation.

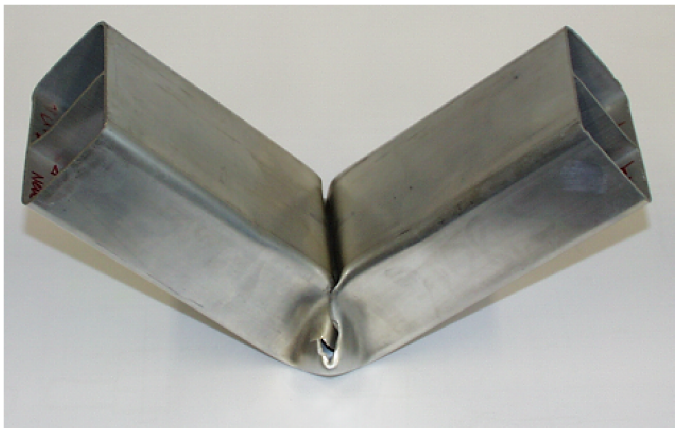


Figure 3: Deformed shape of the aluminum extrusion in the quasi-static three-point bending experiment (Hooputra et al., 2004).

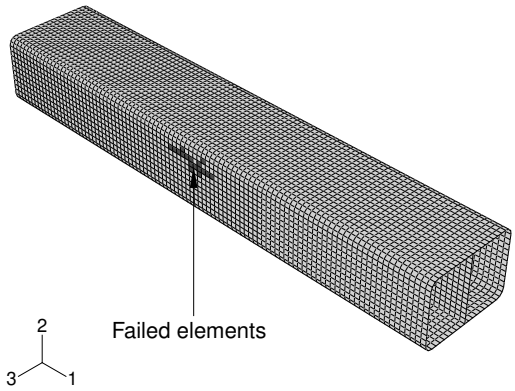


Figure 4: Completely failed elements at the end of the three-point bending simulation mapped into the undeformed configuration.

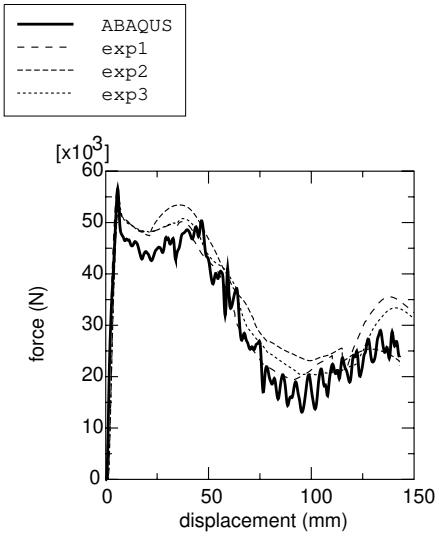


Figure 5: Comparison of the force-displacement response obtained from the three-point bending simulation with the experimental results of Hooputra et al. (2004).

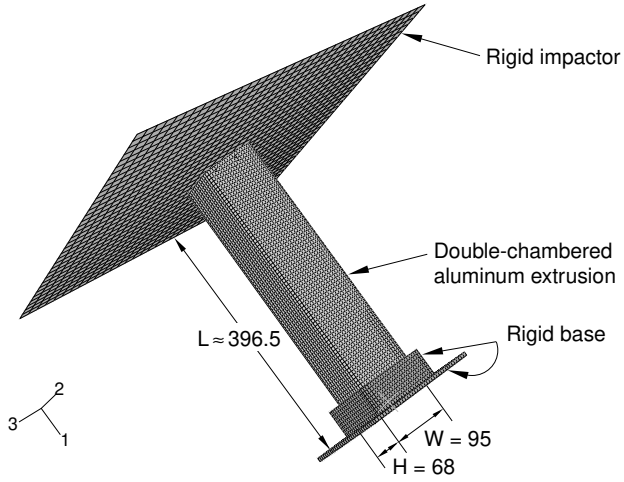


Figure 6: Axial crushing configuration: geometry and finite element mesh.

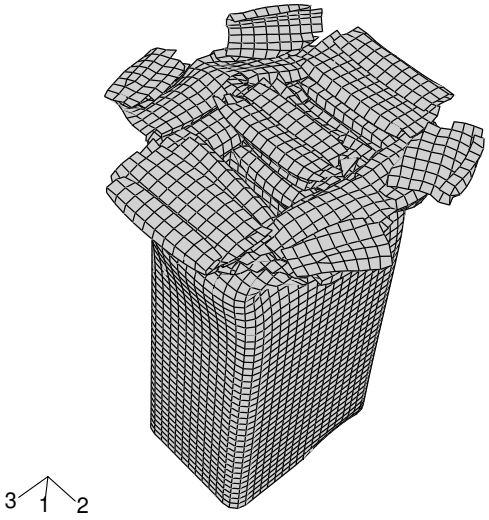


Figure 7: Final deformed shape of the aluminum extrusion in the dynamic axial crushing simulation.



Figure 8: Deformed shape of the aluminum extrusion in the dynamic axial crushing experiment (Hooputra et al., 2004).

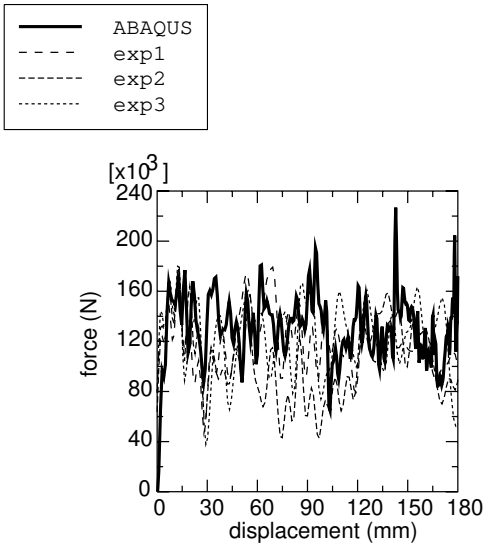


Figure 9: Comparison of the force-displacement response obtained from the axial crushing simulation with the experimental results of Hooputra et al. (2004).

Impact analysis of a pawl-ratchet device

Products: Abaqus/Standard

References:

- *Implicit Dynamic Analysis Using Direct Integration*
- *Defining General Contact in Abaqus/Standard*
- **CONTACT*
- **DYNAMIC*

Overview

This example illustrates the use of the direct-integration implicit dynamic procedure with general contact for an example involving impact and significant sliding between contacting surfaces.

Application description

Pawl-ratchet mechanisms exist in a variety of applications such as ratchet lever hoists (also known as come-along winches; see *Figure 1*) and parking brakes. Some care should be taken when using these devices to minimize dynamic impact between the pawl and ratchet teeth—for example, the lever of a ratchet lever hoist should be controlled such as to gradually engage the antireverse mechanism, and a car should be stopped before putting its transmission into "park." This analysis provides a preliminary assessment of the integrity of a pawl-ratchet mechanism under misuse conditions in which dynamic effects are significant when the pawl engages a ratchet tooth.

Geometry

The model is shown in *Figure 2*, which somewhat resembles the ratchet mechanism within the tool shown in *Figure 1*; however, this model is not representative of design details of any specific product. The ratchet mechanism is intended to restrict clockwise motion of the ratchet wheel. The pawl and ratchet wheel are each attached to a stiff support using cylindrical bolts. The ratchet-bolt and pawl-bolt interfaces do not restrict axial rotation of the ratchet and pawl. A torsional spring, which is not shown in the figure, tends to keep the pawl in contact with the ratchet wheel. The ratchet wheel is attached to a hollow shaft. A cable and a lever would also be attached to the shaft in a ratchet lever hoist; however, these aspects are not modeled explicitly.

The model uses a millimeter-tonne-second unit system. The hollow shaft has a length of 40 mm and a radius of 20 mm. The radius of the shaft hole is 12 mm. The width of both the pawl and the ratchet wheel is 8 mm. The radius of the wheel hole is 8 mm, and the radius of the pawl hole is 4 mm. The wheel outer diameter is 100 mm.

Materials

The materials considered in this example are assumed to have linear elastic, perfectly plastic behavior except for the support, which is considered rigid with fixed position. The deformable bodies have an elastic modulus of 21×10^4 MPa, Poisson's ratio of 0.3, and density of 7.9×10^{-9} tonne/mm^3. The yield strength of the pawl is 600 MPa, and the yield strength of the other deformable bodies is 800 MPa.

Interactions

The model employs the general contact capabilities available in Abaqus/Standard to simulate contact interactions between the various parts.

Abaqus modeling approaches and simulation techniques

The analysis is performed using implicit dynamic steps. The first step establishes contact between the pawl and the ratchet wheel, thereby establishing appropriate initial conditions for the second step. Inertia effects are considered during the first step only to avoid issues associated with initially unconstrained rigid body modes that could be troublesome for a purely static procedure; therefore, the quasi-static implicit dynamic application option is used in the first step.

The remainder of the analysis considers the effect of a suddenly applied, sustained net torque acting on the ratchet shaft that leads to impact between the pawl and a ratchet tooth. This applied net torque is meant to approximate the effect of suddenly removing an applied torque that had been counteracting a working load. In a ratchet lever hoist (see *Figure 1*) this applied net torque approximates the effect of suddenly removing the load on the lever before the pawl is properly engaged with the ratchet (while the hoist is carrying a load). The response due to the applied torque is modeled with a single (second) step in a preliminary analysis and alternatively with three separate steps for the pre-impact, impact, and post-impact phases to facilitate use of smaller time increments for a more accurate analysis of the highly nonlinear impact phase. The moderate dissipation implicit dynamic application option is used for steps modeling the response to the applied torque, which is appropriate for most dynamic contact simulations.

This analysis is meant to approximate a situation in which the cable loading is due to gravity acting on a mass attached to the cable. Since the cable and this mass are not modeled directly, the inertial effect of the mass is approximated with additional rotary inertia on the shaft. In general, if the cable were included in the model, one could expect that the solution may be affected by the cable dynamic response and the details of cable-shaft interactions and self-contact of the cable where it is wrapped onto the shaft. However, in this preliminary study, the model simplifications are assumed to provide reasonably representative results with respect to the severity of the impact loading.

Mesh design

The shaft mesh uses fully integrated first-order continuum (C3D8) elements. All other parts are meshed with reduced integration first-order continuum (C3D8R) elements.

Materials

The deformable materials use the Mises plasticity model without hardening. The material parameters were discussed previously.

Initial conditions

All of the bodies are initially at rest. The positioning of the pawl and ratchet wheel at the beginning of the analysis is shown in *Figure 2*. A small initial gap exists between the pawl and the ratchet. Initially, the pawl is not in static equilibrium due to initial loading by a torsional spring, as discussed in *Constraints* below.

Boundary conditions

The reference node of the support (which is considered rigid) has all components of translation and rotation fixed. The annular face of the shaft away from the ratchet wheel is constrained to a reference node that is positioned on the shaft axis, at the center of the annular face. Zero displacement along the global X- and global Y-axes is prescribed for this reference node.

Loads

After the initialization step, force and torque act on the ratchet shaft due to cable loading associated with a 0.2 tonne mass under gravitational loading (as discussed previously, this mass and the cable are not modeled explicitly). A force of 1960 N and torque of 42,140 Nmm act on the ratchet shaft via a distributing coupling

reference node. These concentrated loads are applied instantaneously at the beginning of the second step and remain constant in time.

The applied torque leads to impact loading (modeled with general contact), which is of primary interest in this analysis.

Constraints

All translations and rotations for the rigid support are fixed. The circumferential surface of the bolts is partitioned in two halves (with respect to a plane perpendicular to their axis) with one half being tied to the corresponding hole in the support part, using a tie constraint. A distributing coupling is employed to apply concentrated force and torque to the outer cylindrical surface of the hollow shaft. One of the annular faces of the shaft is tied to the ratchet wheel. A kinematic coupling is employed to constrain the other annular face of the shaft (i.e., the face away from the ratchet wheel) to a reference node at the center of this face.

A torsional spring—modeled using a connector element—is used for maintaining contact between the pawl and the ratchet wheel. One end of the connector is coupled using a distributing coupling to the outer cylindrical surface of the pawl.

Interactions

Contact is defined using the default general contact inclusions option. This is the easiest way to ensure that Abaqus/Standard will enforce any mechanical interactions between various parts of the model. A friction coefficient of 0.15 is specified for all interactions. Circumferential smoothing is applied to all cylindrical surfaces (associated with bolts and bolt holes), which was facilitated by using Abaqus/CAE to create the model (Abaqus/CAE assigns circumferential smoothing by default to cylindrical surfaces).

A connector element (CONN3D2) using a revolute connection type with a prescribed reference angle is employed for representing the torsional spring, which maintains contact between the pawl and ratchet. The rotational stiffness of the connector element is 2000 Nmm/rad. A reference angle of 45° is defined for the connector behavior. When in static equilibrium, the pawl presses the ratchet with a contact force of about 31.5 N.

Results and discussion

The evolution of the angular velocity of the ratchet wheel is shown in *Figure 3* for the preliminary analysis that uses two load steps. This angular velocity remains zero during the first step because no net torque acts on the ratchet wheel during that step. Once the torque loading is applied after the first step, an approximately linear increase in the absolute value of this angular velocity occurs while the pawl slides along the ratchet; during this time the net torque acting on the ratchet wheel (affected by the applied torque and frictional forces) is approximately constant, so the angular acceleration is also nearly constant. The sign of the angular acceleration changes upon impact. Large contact forces transmitted between the pawl and ratchet tooth eventually lead to a rebound. This simulation completes before any secondary rebounds occur (the analysis duration could be extended if secondary rebounds were of interest).

In a subsequent 4-step analysis, the initialization step remains the same but the response to torque loading is modeled with three steps instead of one step to facilitate specification of smaller time increments during impact. These three steps correspond to the pre-impact, impact, and post-impact phases that are apparent in *Figure 3*. Usage of smaller time increments during impact enables more accurate modeling of the highly nonlinear response during this phase. *Figure 4* shows contour plots of equivalent plastic strain at the end of the 2-step and 4-step analyses in the vicinity of the pawl-to-ratchet impact for the imagined mechanism. The maximum equivalent plastic strain, which occurs at the tip of the pawl, is approximately 8% in the 2-step analysis and approximately 10% in the 4-step analysis. The 4-step analysis uses a time increment during impact about one-fifth of that in the 2-step analysis. The maximum equivalent plastic strain is quite insensitive to further reductions in the time increment during impact. The difference between the equivelent plastic strain plots shown in *Figure 4* demonstrates the importance of using an adequately small time increment to model portions of an analysis with sudden changes

in acceleration. Less severe plastic deformation is predicted in the ratchet tooth and, as shown on the right side of *Figure 5*, in the vicinity of the pawl-bolt interface.

The evolution of kinetic energy (ALLKE) and energy dissipation due to yielding (ALLPD) is plotted for the 2-step and 4-step analyses in *Figure 6*. Plastic dissipation occurs while the ratchet wheel decelerates after impact; differences in plastic dissipation for the two analyses are again apparent in *Figure 6*. The various phases of the simulation can be identified in the kinetic energy plot, including the pre-impact, impact, rebound, and post-rebound phases.

Input files

pawlratchet_impact.inp	Implicit dynamic 2-step analysis.
pawlratchet_impact_4step.inp	Implicit dynamic 4-step analysis.

Figures

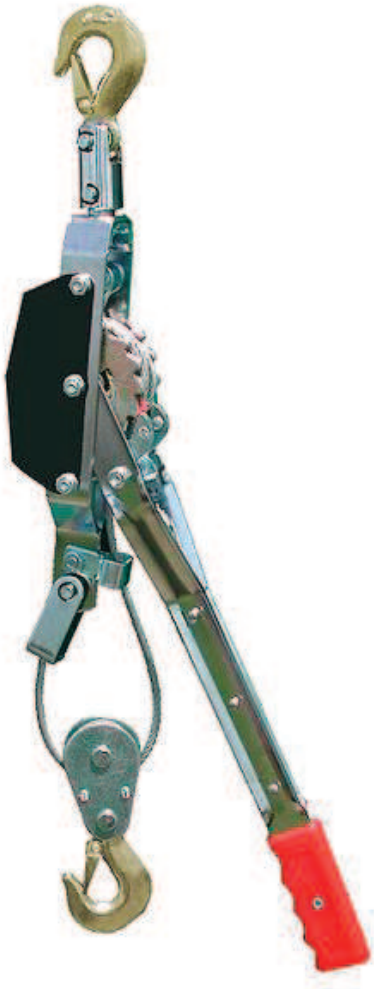


Figure 1: Ratchet lever hoist (come-along winch).

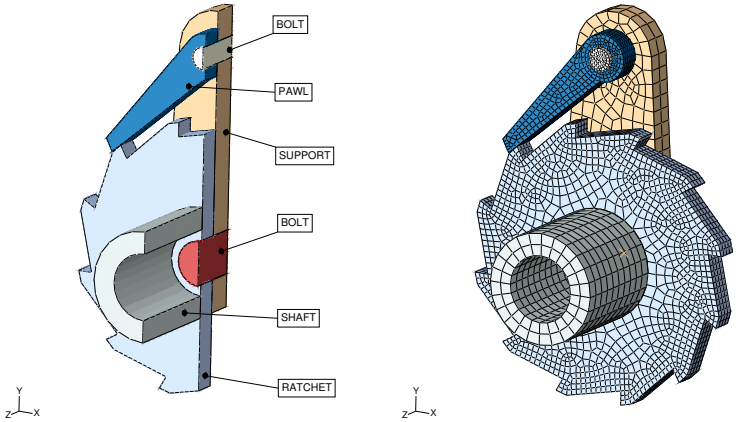


Figure 2: Geometry and mesh of the pawl-ratchet model.

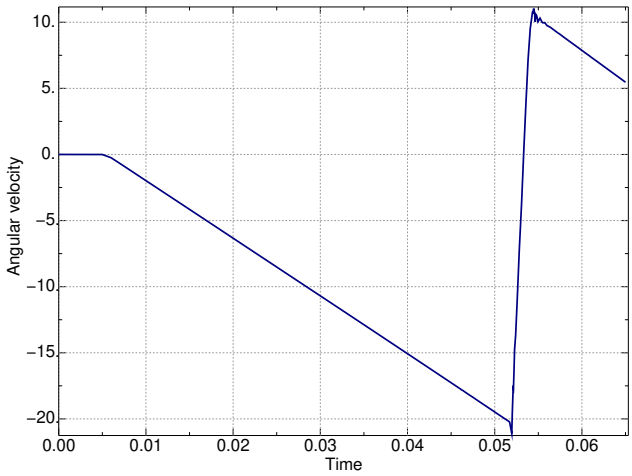


Figure 3: Evolution of angular velocity of the ratchet wheel.

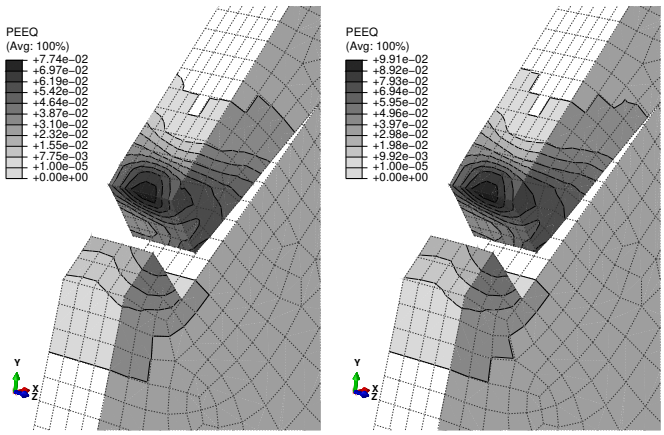


Figure 4: Distribution of equivalent plastic strain (PEEQ) near pawl tip for 2-step analysis (left) and 4-step analysis (right).

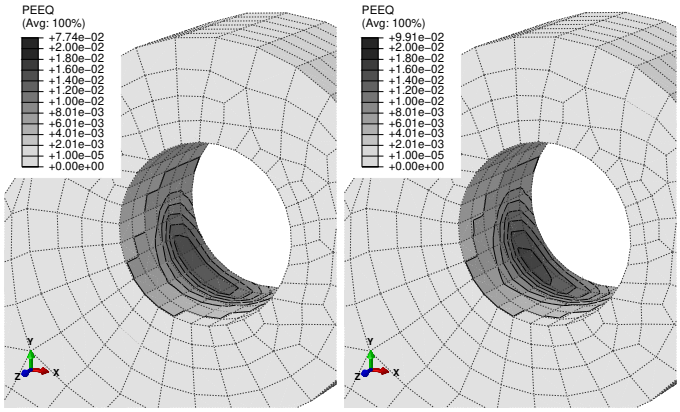


Figure 5: Distribution of equivalent plastic strain (PEEQ) near pawl hole for 2-step analysis (left) and 4-step analysis (right).

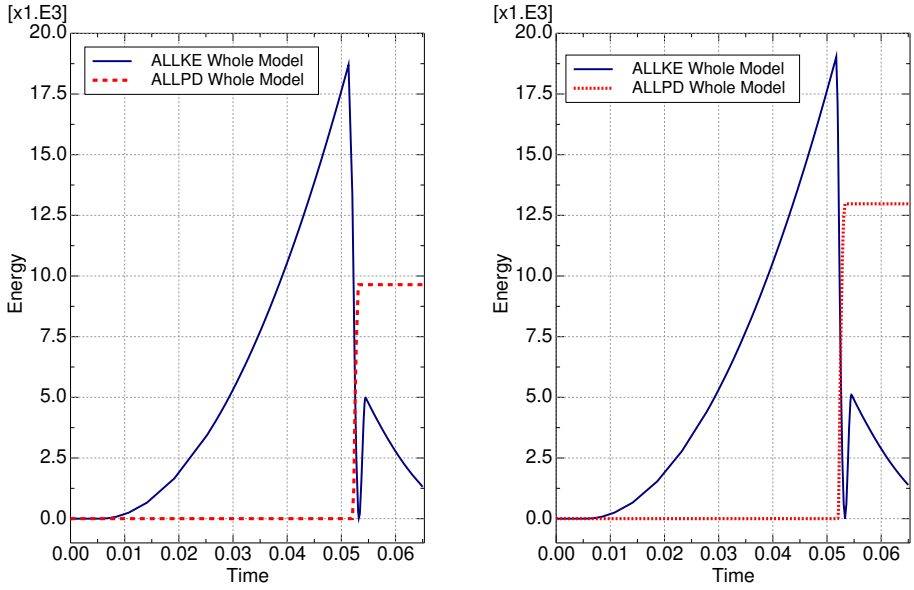


Figure 6: Evolution of the kinetic and plastically dissipated energy for 2-step analysis (left) and 4-step analysis (right).

High-velocity impact of a ceramic target

Products: Abaqus/Explicit

References:

- *About Progressive Damage and Failure*
- *Extended Drucker-Prager Models*
- **DAMAGE INITIATION*
- **DAMAGE EVOLUTION*
- **DRUCKER PRAGER*

Overview

In this example the Johnson-Holmquist-Beissel and the Johnson-Holmquist material models are explored to investigate the penetration velocity of a gold projectile impacting on a silicon carbide target. The computed results are compared with published results given by Holmquist and Johnson (2005).

The following Abaqus features are demonstrated:

- using the Johnson-Holmquist-Beissel (JHB) and the Johnson-Holmquist (JH-2) ceramic material models, which are available in Abaqus/Explicit as built-in user materials, to study the high-velocity impact of a silicon carbide target;
- achieving a similar material response for ceramics with proper calibration of the Drucker-Prager plasticity and the equation of state functionality in Abaqus/Explicit; and
- comparing numerical results with published results.

Application description

Ceramic materials are commonly used in armor protection applications. In recent years Johnson, Holmquist, and their coworkers have developed a series of constitutive relations to simulate the response of ceramic materials under large strain, high-strain rate, and high-pressure impacting conditions.

Geometry

The initial configuration is shown in *Figure 1*. Both target and projectile are of cylindrical shape. The silicon carbide target has a radius of 7.5 mm and a length of 40 mm. The gold projectile has a radius of 0.375 mm and a length of 30 mm.

Materials

The target material is silicon carbide. This material is very hard and mainly used under compressive load conditions and can only sustain very little tension. Typical applications include bulletproof vests and car brakes due to its high endurance. The strength has a dependence on pressure. In high-speed impact applications, damage to the material plays an important role in the evolution of the strength. The totally failed silicon carbide will not sustain any load. The projectile is gold, which is soft compared to the target material.

Initial conditions

An initial velocity of 4000 m/s is prescribed for the projectile.

Interactions

The projectile will penetrate into the target due to the high-speed impact.

Abaqus modeling approaches and simulation techniques

Three cases are investigated, each using a different approach to model the silicon carbide material: the first case uses the JHB material model, the second uses the JH-2 model, and the third case uses a combination of several Abaqus options to obtain a similar constitutive model within a more general framework. The Lagrangian description is used for both projectile and target. General contact with surface erosion is used for all three cases. Element deletion and node erosion are considered. A tonne-millimeter-second unit system was chosen for all simulations.

Summary of analysis cases

Case 1	JHB (built-in user material)
Case 2	JH-2 (built-in user material)
Case 3	Combination of Drucker-Prager plasticity, equation of state, and Johnson-Cook rate dependence

The sections that follow discuss the analysis considerations that are applicable to all cases.

Analysis types

An Abaqus/Explicit dynamic analysis is used for all the simulations. The total duration for the penetration process is 7 μs.

Mesh design

An 11.5° slice of the cylinders is modeled. There are five elements along the radial direction of the projectile. The element size along the radial direction for the target is nearly the same as for the projectile. Due to the large radius ratio between the projectile and the target, there are 343,980 elements for the target and 2000 elements for the projectile. *Figure 2* shows part of the meshes used for the analysis.

Materials

The different material models used for the silicon carbide target are discussed in detail in subsequent sections. The JHB and JH-2 models are available as built-in user materials for Abaqus (i.e., via *VUMAT* subroutines that are built-in). These built-in materials are invoked by using material names starting with `ABQ_JHB` and `ABQ_JH2`, respectively. For descriptions of the ceramic material models, see "Analyzing ceramics with the Johnson-Holmquist and Johnson-Holmquist-Beissel material models" in the Dassault Systèmes Knowledge Base at *http://support.3ds.com/knowledge-base/*.

The material for the projectile is gold. The density is 19,240 kg/m^3. The shear modulus is 27.2 GPa. The hydrodynamic behavior is described by the Mie-Grüneisen equation of state. The linear $U_s - U_p$ Hugoniot form is used. The parameters are c_0 = 2946.16 m/s, s = 3.08623, and Γ_0 = 2.8. The strength is 130 MPa described as a perfect plasticity. A ductile damage initiation criterion with the equivalent plastic strain of 0.2 at the onset of damage is used. The fracture energy is chosen as 0 for the damage evolution.

Initial conditions

Initial velocity conditions of 4000 m/s are specified for all the nodes of the projectile in the axial direction toward the target.

Boundary conditions

All the nodes on the symmetry axis, which is set up as the global x-direction, can move only along this axis, so zero velocity boundary conditions are prescribed for both the y- and z-directions. To satisfy the axial symmetry boundary conditions, a cylindrical coordinate system is established. The circumferential degrees of freedom for all the nodes on the two side surfaces except the nodes on the symmetry axis of both the target and the projectile are prescribed with zero velocity boundary conditions. The nodes on the non-impacting end of the target are fixed along the axial direction.

Interactions

General contact is used to model the interactions between the projectile and the target. The interior surface of both the target and the projectile is included to enable element removal.

Analysis steps

There is only one explicit dynamic analysis step, during which the penetration takes place.

Output requests

In addition to the standard output identifiers available in Abaqus, the solution-dependent state variables described in *Table 1* and *Table 2* are also available for output for the JHB and JH-2 models, respectively.

Constitutive models for ceramic materials under high-velocity impact

This section provides a detailed description of the different constitutive models for ceramic materials that are used to model the silicon carbide target for each of the cases considered.

Case 1: Johnson-Holmquist-Beissel model

This case uses the JHB model for the silicon carbide target. The JHB material parameters for silicon carbide given in Holmquist and Johnson (2005) are used in this study. They are listed in *Table 3*.

The JHB model consists of three main components: a representation of the deviatoric strength of the intact and fractured material in the form of a pressure-dependent yield surface, a damage model that transitions the material from the intact state to a fractured state, and an equation of state (EOS) for the pressure-density relation that can include dilation (or bulking) effects as well as a phase change (not considered in this study).

Strength

The strength of the material is expressed in terms of the von Mises equivalent stress, σ, and is a function of the pressure, P, the dimensionless equivalent strain rate, $\dot{\epsilon}^* = \dot{\bar{\epsilon}}^{pl}/\dot{\epsilon}_0$ (where $\dot{\bar{\epsilon}}^{pl}$ is the equivalent plastic strain rate and $\dot{\epsilon}_0$ is the reference strain rate), and the damage variable, D ($0 \le D \le 1$). For the intact (undamaged) material, $D = 0$, whereas $D = 1$ for a fully damaged material.

For a dimensionless strain rate of $\dot{\epsilon}^* = 1.0$, the strength of the intact material ($D = 0$) takes the form

$$\sigma = \sigma_i \left(P + T\right) / \left(P_i + T\right) \quad \text{if} \quad \mathrm{P} \le \mathrm{P_i},$$

$$\sigma = \sigma_i + \left(\sigma_i^{\max} - \sigma_i\right) \left\{1 - \exp[-\alpha_i \left(P - P_i\right)]\right\} \quad \text{if} \quad \mathrm{P} \ge \mathrm{P_i},$$

where $\alpha_i = \sigma_i / \left[\left(\sigma_i^{\max} - \sigma_i\right) \left(P_i + T\right)\right]$ and P_i, T, σ_i, and $\sigma_i^{\max}$ are material parameters. The strength of the fractured material ($D = 1$) is given by

$$\sigma = \sigma_f P/P_f \quad \text{if} \quad \mathrm{P} \le \mathrm{P_f},$$
$$\sigma = \sigma_f + \left(\sigma_f^{\max} - \sigma_f\right)\{1 - \exp[-\alpha_f (P - P_f)]\} \quad \text{if} \quad \mathrm{P} \ge \mathrm{P_f},$$

where $\alpha_f = \sigma_f / \left[\left(\sigma_f^{\max} - \sigma_f\right) P_f\right]$ and P_f, σ_f, and $\sigma_f^{\max}$ are material parameters.

The intact and fractured strengths above are for a dimensionless strain rate of $\dot{\epsilon}^* = 1.0$. The effect from strain rates is incorporated by the Johnson-Cook strain rate dependence law as $\sigma = \sigma_0 (1 + C \ln \dot{\epsilon}^*)$, where σ_0 is the strength corresponding to $\dot{\epsilon}^* = 1.0$. Plastic flow is volume preserving and is governed by a Mises flow potential.

Damage

The damage initiation parameter, ω, accumulates with plastic strain according to

$$\omega = \sum \frac{\Delta \bar{\epsilon}^{pl}}{\bar{\epsilon}_f^{pl}(P)},$$

where $\Delta \bar{\epsilon}^{pl}$ is the increment in equivalent plastic strain and $\bar{\epsilon}_f^{pl}(P)$ is the equivalent plastic strain to fracture under constant pressure, defined as

$$\bar{\epsilon}_f^{pl} = D_1 (P^* + T^*)^{D_2} \; ; \quad \bar{\epsilon}_{f,\min}^{pl} \le \bar{\epsilon}_f^{pl} \le \bar{\epsilon}_{f,\max}^{pl},$$

where D_1 and D_2 are material constants and $P^* = P/\sigma_i^{\max}$ and $T^* = T/\sigma_i^{\max}$. The optional parameters $\bar{\epsilon}_{f,\min}^{pl}$ and $\bar{\epsilon}_{f,\max}^{pl}$ are provided for additional flexibility to limit the minimum and maximum values of the fracture strain.

The JHB model assumes that the material fails immediately, $D = 1$ when $\omega = 1$. For other values of $\omega < 1$, there is no damage ($D = 0$) and the material preserves its intact strength.

Pressure

The equations for the pressure-density relationship without phase change are used in this study and are listed here.

$$P = K_1 \mu + K_2 \mu^2 + K_3 \mu^3 \quad \text{if} \quad \mu \ge 0,$$
$$P = K_1 \mu \quad \text{if} \quad \mu \le 0,$$

where $\mu = \rho/\rho_0 - 1$. In the above, K_1, K_2, and K_3 are constants (K_1 is the initial bulk modulus); ρ is the current density; and ρ_0 is the reference density. The model includes the effects of dilation or bulking that occur when brittle materials fail by including an additional pressure increment, ΔP, such that

$$P = K_1 \mu + K_2 \mu^2 + K_3 \mu^3 + \Delta P.$$

The pressure increment is determined from energy considerations as

$$\Delta P = -K_1 \mu_f + \sqrt{(K_1 \mu_f)^2 + 2\beta K_1 \Delta U},$$

where μ_f is the current value of μ at the time of failure and β is the fraction of the elastic energy loss converted to potential hydrostatic energy ($0 \le \beta \le 1$). The bulking pressure is computed only for failure under compression ($\mu_f > 0$).

Case 2: Johnson-Holmquist model

The second case uses the JH-2 model. Unlike the JHB model, the JH-2 model assumes that the damage variable increases progressively with plastic deformation. The material parameters used for the JH-2 model are listed in *Table 4*. The JH-2 model similarly consists of three components.

Strength

The strength of the material is expressed in terms of the normalized von Mises equivalent stress as

$$\sigma^* = \sigma_i^* - D\left(\sigma_i^* - \sigma_f^*\right),$$

where σ_i^* is the normalized intact equivalent stress, σ_f^* is the normalized fractured equivalent stress, and D is the damage variable. The normalized equivalent stresses (σ^*, σ_i^* and σ_f^*) have the general form $\sigma^* = \sigma/\sigma_{\mathrm{HEL}}$, where σ is the actual von Mises equivalent stress and σ_{HEL} is the equivalent stress at the Hugoniot elastic limit (HEL). The model assumes that the normalized intact and fractured stresses can be expressed as functions of the pressure and strain rate as

$$\sigma_i^* = A(P^* + T^*)^N \left(1 + C\ln\dot{\epsilon}^*\right) \le \sigma_i^{\max},$$
$$\sigma_f^* = B(P^*)^M \left(1 + C\ln\dot{\epsilon}^*\right) \le \sigma_f^{\max}.$$

The material parameters are $A, B, C, M, N,$ and the optional limits for the strengths $\sigma_i^{\max}$ and $\sigma_f^{\max}$. The normalized pressure is defined as $P^* = P/P_{\mathrm{HEL}}$, where P is the actual pressure and P_{HEL} is the pressure at the HEL. The normalized maximum tensile hydrostatic pressure is $T^* = T/P_{\mathrm{HEL}}$, where T is the maximum tensile pressure that the material can withstand.

Damage

The damage initiation parameter, ω, accumulates with plastic strain according to

$$\omega = \sum \frac{\Delta\bar{\epsilon}^{pl}}{\bar{\epsilon}_f^{pl}(P)},$$

where $\Delta\bar{\epsilon}^{pl}$ is the increment in equivalent plastic strain and $\bar{\epsilon}_f^{pl}(P)$ is the equivalent plastic strain to fracture under constant pressure, defined as

$$\bar{\epsilon}_f^{pl} = D_1(P^* + T^*)^{D_2} \quad \bar{\epsilon}_{f,\min}^{pl} \le \bar{\epsilon}_f^{pl} \le \bar{\epsilon}_{f,\max}^{pl}.$$

The JH-2 model assumes that the damage variable increases gradually with plastic deformation by setting $D = \omega$.

Pressure

The equations for the pressure-density relationship are similar to the JHB model.

$$P = K_1\mu + K_2\mu^2 + K_3\mu^3 \quad \text{if} \quad \mu \ge 0,$$
$$P = K_1\mu \quad \text{if} \quad \mu \le 0,$$

where $\mu = \rho/\rho_0 - 1$. The model includes the effects of dilation or bulking that occur when brittle materials fail by including an additional pressure increment, ΔP, such that

$$P = K_1\mu + K_2\mu^2 + K_3\mu^3 + \Delta P.$$

The pressure increment is determined from energy considerations as

$$\Delta P_{t+\Delta t} = -K_1 \mu_{t+\Delta t} + \sqrt{(K_1 \mu_{t+\Delta t} + \Delta P_t)^2 + 2\beta K_1 \Delta U},$$

where β is the fraction of the elastic energy loss converted to potential hydrostatic energy ($0 \le \beta \le 1$).

Case 3: Drucker-Prager model

We use a calibrated Drucker-Prager plasticity model and an equation of state to obtain a material behavior that is similar to that of the JH-2 model. In this way we are not confined to follow the specific expressions and, subsequently, the material parameters of the JH models. The calibration of the Drucker-Prager plasticity model and the equation of state is described below.

Strength

We use the general exponent form of the extended Drucker-Prager model (*Extended Drucker-Prager Models*) which, after some manipulations, can be written as follows:

$$\sigma = \frac{1}{a^{1/b}} (P + P_t)^{1/b}.$$

We have replaced q with σ and p with P to be consistent with the symbols used in the JH-2 models. This expression is very similar to that of the intact strength of the JH-2 model

$$\sigma_i = \frac{\sigma_{\text{HEL}} A}{P_{\text{HEL}}^N} (P + T)^N.$$

Comparing these two expressions, the equations to calibrate the material parameters in the Drucker-Prager model can be obtained as

$$a = \frac{P_{\text{HEL}}}{(A\sigma_{\text{HEL}})^{1/N}},$$
$$b = \frac{1}{N},$$
$$P_t = T.$$

After substituting the values for silicon carbide in *Table 4*, we find $a = 3.920173 \times 10^{-3}$ MPa $^{1-\frac{1}{N}}$ and $b = 1.53846$. After obtaining the material parameters a and b, the uniaxial compression yield stress, σ_c, can be calibrated by solving the following equation.

$$P_t = a\sigma_c^b - \frac{\sigma_c}{3} = T.$$

For this silicon carbide, σ_c = 6605.66 MPa. The Johnson-Cook type rate dependence can also be used together with the Drucker-Prager plasticity model.

Damage

The ductile damage initiation criterion in Abaqus (*Damage Initiation for Ductile Metals*) can be calibrated to reproduce the damage criterion used in the JH-2 damage model. The ductile criterion requires the specification of the equivalent plastic strain at the onset of damage as a function of the stress triaxiality. Along the intact strength curve of the JH-2 model, the stress triaxiality is given as

$$-\frac{P}{\sigma_i} = -P \Big/ \left[\sigma_{\text{HEL}} A \left(\frac{P}{P_{\text{HEL}}} + \frac{T}{P_{\text{HEL}}}\right)^N\right].$$

Given $\bar{\epsilon}_f^{pl}$, the damage evolution relation for the JH-2 model gives the following expression for the pressure P:

$$P = P_{\mathrm{HEL}} \left(\frac{\bar{\epsilon}_f^{pl}}{D_1} \right)^{1/D_2} - T.$$

Substituting this expression into the stress triaxiality expression, we finally obtain the functional relationship between $\bar{\epsilon}_f^{pl}$ and the stress triaxiality needed for the ductile damage initiation criterion. An additional consideration when specifying the damage initiation criterion for the Drucker-Prager plasticity model is that the definition of the equivalent plastic strain is different from that used in the JH-2 model. The two are related by the following plastic work statement:

$$\sigma \dot{\bar{\epsilon}}^{pl} = \sigma_c \dot{\bar{\epsilon}}_c^{pl}.$$

In summary, we specify some sampling points for $\bar{\epsilon}^{pl}$ and calculate the corresponding pressure P and, in turn, the intact strength σ_i for the JH-2 model. We calculate the stress triaxiality using these P and σ_i values. We convert the $\bar{\epsilon}^{pl}$ values into $\bar{\epsilon}_c^{pl}$ through the above plastic work statement. In this manner, we finally obtain a table of data pairs for the equivalent plastic strain at damage initiation and the stress triaxiality for the Drucker-Prager model.

Pressure

The Mie-Grüneisen equation of state (*Mie-Grüneisen Equations of State*) is used to describe the hydrodynamic behavior of the silicon carbide material. The linear $U_s - U_p$ Hugoniot form is used. Without the energy contribution (Γ_0 = 0.0), the pressure is expressed as

$$P = \frac{\rho_0 c_0^2 \eta}{(1 - s\eta)^2},$$

where $\eta = 1 - \rho_0/\rho = \mu/(1+\mu)$. Using a Taylor expansion with respect to μ, the linear and quadratic coefficients of the polynomial can be identified as K_1 and K_2 in the pressure density relation for the JH-2 model, which gives

$$K_1 = \rho_0 c_0^2,$$
$$K_2 = \rho_0 c_0^2 (2s - 1).$$

We solve for the parameters c_0 and s. The values are c_0 = 8272.2 m/s and s = 1.32.

Discussion of results and comparison of cases

The penetration depths from the three models and the published results in the reference by Holmquist and Johnson (2005) at 3 µs, 5 µs, and 7 µs are listed in *Table 5*. All results from the three models match the published results well. Especially, the results obtained with the Drucker-Prager model are in satisfactory agreement with all other results obtained with the JH models. The final configurations for the JHB, JH-2, and Drucker-Prager models are shown in *Figure 3*, *Figure 4*, and *Figure 5*, respectively. The wave propagation results can be improved by increasing the angle of the wedge.

Input files

Case 1: JHB material model

exa_impactsiliconcarbide_jhb.inp	Input file to create and analyze the model.

Case 2: JH-2 material model

exa_impactsiliconcarbide_jh2.inp Input file to create and analyze the model.

Case 3: Drucker-Prager material model

exa_impactsiliconcarbide_dp.inp Input file to create and analyze the model.

References

- Holmquist, T. J., Johnson, G. R., “Characterization and Evaluation of Silicon Carbide for High-Velocity Impact,” Journal of Applied Physics, vol. 97, 093502, 2005.
- Johnson, G. R., Holmquist, T. J., “An Improved Computational Constitutive Model for Brittle Materials,” High Pressure Science and Technology–1993, New York, AIP Press, 1993.

Tables

Table 1: Solution-dependent state variables defined in JHB model.

Output variables	Symbol	Description
SDV1	$\bar{\epsilon}^{pl}$	Equivalent plastic strain PEEQ
SDV2	$\dot{\bar{\epsilon}}^{pl}$	Equivalent plastic strain rate
SDV3	ω	Damage initiation criterion
SDV4	D	Damage variable
SDV5	δP	Pressure increment due to bulking
SDV6	σ	Yield strength
SDV7	$\mu_{\max}$	Maximum value of volumetric strain μ
SDV8	μ	Volumetric strain $\mu = \rho/\rho_0 - 1$
SDV9	MpStatus	Material point status: 1 if active, 0 if failed

Table 2: Solution-dependent state variables defined in JH-2 model.

Output variables	Symbol	Description
SDV1	$\bar{\epsilon}^{pl}$	Equivalent plastic strain PEEQ
SDV2	$\dot{\bar{\epsilon}}^{pl}$	Equivalent plastic strain rate
SDV3	ω	Damage initiation criterion
SDV4	D	Damage variable
SDV5	δP	Pressure increment due to bulking
SDV6	σ	Yield strength
SDV7	μ	Volumetric strain $\mu = \rho/\rho_0 - 1$

Output variables	Symbol	Description
SDV8	MpStatus	Material point status: 1 if active, 0 if failed

Table 3: Material parameters for JHB model.

Line 1	ρ_0	G	σ_i	P_i	σ_f	P_f	C	$\dot{\epsilon}_0$
	3215 kg/m^3	193 GPa	4.92 GPa	1.5 GPa	0.1 GPa	0.25 GPa	0.009	1.0
Line 2	T	$\sigma_i^{\max}$	$\sigma_f^{\max}$	β				
	0.75 GPa	12.2 GPa	0.2 GPa	1.0				
Line 3	D_1	D_2	$\bar{\epsilon}_{f,min}^{pl}$	FS				
	0.16	1.0	999	0.2				
Line 4	K_1	K_2	K_3					
	220 GPa	361 GPa	0 GPa					
Line 5	$\overline{K}_1$	$\overline{K}_2$	$\overline{K}_3$	μ_1	μ_2	μ_0	h	
	0	0	0	0	0	0	0	

Table 4: Material parameters for JH-2 model.

Line 1	ρ_0	G	A	N	B	M	C	$\dot{\epsilon}_0$
	3215 kg/m^3	193 GPa	0.96	0.65	0.35	1.0	0.009	1.0
Line 2	T	$\sigma_i^{\max}$	$\sigma_f^{\max}$	HEL	p_{HEL}	β		
	0.75 GPa	1.24	0.132	11.7 GPa	5.13 GPa	1.0		
Line 3	D_1	D_2	$\bar{\epsilon}_{f,\max}^{pl}$	$\bar{\epsilon}_{f,\min}^{pl}$	FS	lDamage		
	0.48	0.48	1.2	0.0	0.2	0		
Line 4	K_1	K_2	K_3					
	220 GPa	361 GPa	0 GPa					

Table 5: Results comparison with the reference results.

Results	Penetration depth		
	3 μs	5 μs	7 μs
Published results	7.23 mm	12.05 mm	16.87 mm
JHB model	7.19 mm	12.10 mm	16.88 mm
JH-2 model	7.18 mm	12.03 mm	16.89 mm
Drucker-Prager model	7.18 mm	11.80 mm	16.52 mm

Figures

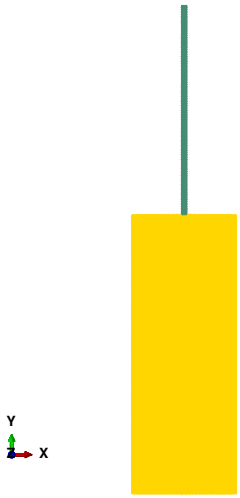


Figure 1: Silicon carbide target and gold projectile: geometry.

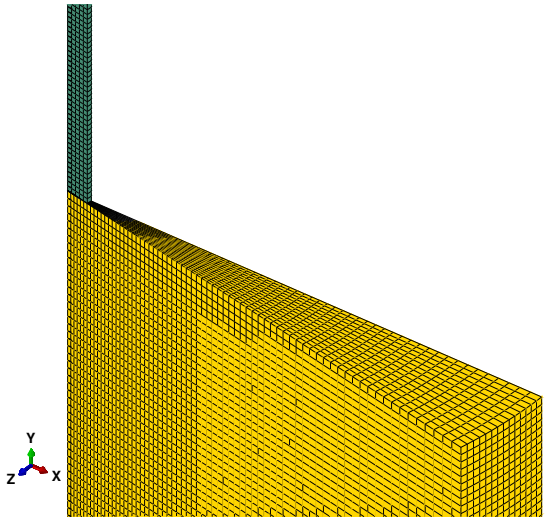


Figure 2: Silicon carbide target and gold projectile: mesh.

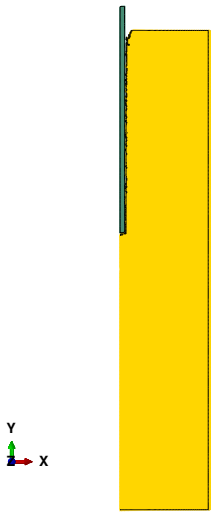


Figure 3: Final configuration at 7 µs using the JHB model.

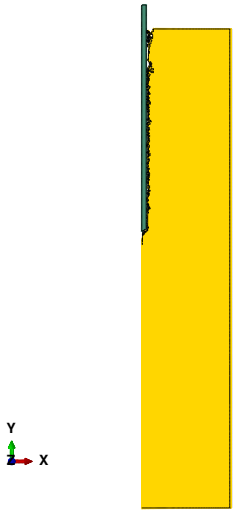


Figure 4: Final configuration at 7 µs using the JH-2 model.

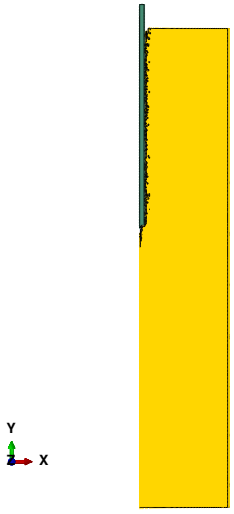


Figure 5: Final configuration at 7 µs using the Drucker-Prager model and the Mie-Grüneisen equation of state.

Confined compression of multiple two-dimensional cylinders with different material properties

Products: Abaqus/Explicit

Overview

This example illustrates the use of general contact in Abaqus/Explicit for a two-dimensional analysis.

Application description

In this example a simple U-shaped container is simulated using rigid elements (R2D2), and it is initially filled with two-dimensional cylinders modeled as plane stress elements (CPS3 and CPS4R). A plane stress state approximates relatively thin bodies, so this model may be representative of short cylinders. The model could be converted to approximating the response for long cylinders by switching to the corresponding plane strain element types (CPE3 and CPE4R).

In the second part of the analysis, the deformable cylinders are highly compressed by a horizontal rigid lid.

Abaqus modeling approaches and simulation techniques

Materials

Two different materials are used in this example. The cylinders colored green in *Figure 2* are modeled using a hyperelastic material (polynomial model with N=1; Mooney-Rivlin) with the constants C_{10}=4.48632 MPa, C_{01}=4.48632 MPa, and D_1=0.02229 MPa^{-1}, such that the initial shear modulus $\mu_0 = 2(C_{10} + C_{01})$=17.9452 MPa, and the initial bulk modulus $K_0 = \frac{2}{D_1}$=89.72633 MPa. The density is set to 1.71E–8 tonne/mm^3. The cylinders colored white in *Figure 2* are linear elastic with Young's modulus of 70,000 MPa, Poisson's ratio of 0.3, and density of 7.8E–8 tonne/mm^3. The shear and bulk modulus of the white cylinders are $\mu = \frac{E}{2(1+\upsilon)}$=26.923 MPa and $K = \frac{E}{3(1-2\upsilon)}$=58.333 MPa, respectively; therefore, the white cylinders are three orders of magnitude stiffer than the green cylinders.

Boundary conditions

The rigid container is fully constrained. The rigid lid has a specified displacement in the negative *Y*-direction equal to –54.0 mm applied using a linear amplitude. The cylinders are free to move in the plane except for contact.

Loads

The entire system is subjected to gravity loading in the *Y*-direction.

Interactions

Contact is defined using general contact with the option to automatically include the whole exterior surface (see *Figure 1*). This type of automated contact definition is much simpler than defining a priori all possible contact pairs that may interact during the simulation. The lid and container surfaces are modeled as overlapping to ensure

that the contents remain inside the container, and contact exclusions are specified as part of the contact definition to avoid considering contact between the lid and container surfaces during the simulation.

A friction coefficient of 0.1 is used as the default for all contact interactions.

Results and discussion

The cylinders are subjected only to gravity loading in the initial phase of the analysis, such that the cylinders settle on the bottom of the container (see middle image of *Figure 2*). Next, the rigid lid is moved downward to compress the cylinders (see right side of *Figure 2*). As expected, due to the large stiffness discrepancy between the two materials, most deformation occurs in the green cylinders, which are made of the softer, hyperelastic material.

Figure 3 shows that, as expected, the reaction force on the reference node of the rigid lid is very small during the first phase of the analysis and increases exponentially as the lid compresses the cylinders.

Due to the high number of increments (> 1 million), the analysis is run in double precision to prevent the accumulation of round-off errors.

Input files

xpl_2dgencont_compression.inp	Input data for this analysis.

Figures

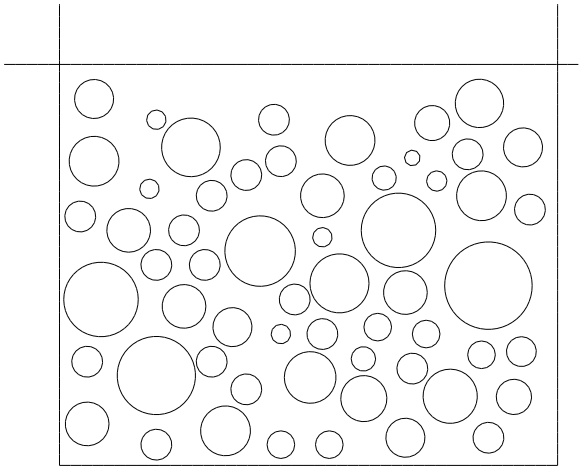


Figure 1: General contact surface definition at the initial configuration.

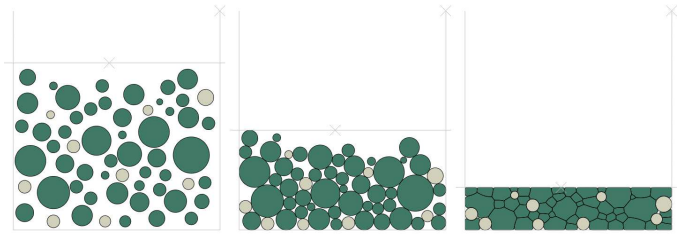


Figure 2: Compression of the two-dimensional cylinders at different stages of the analysis.

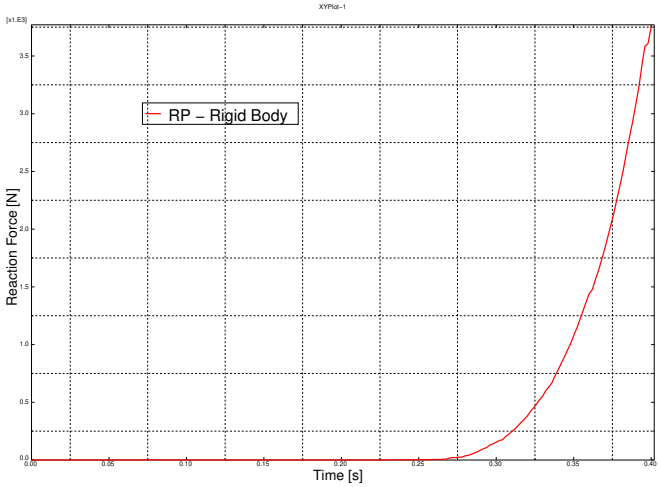


Figure 3: Reaction force in Y-direction of the reference node of the lid.

Mode-based dynamic analyses

In this section:

- *Analysis of a rotating fan using substructures and cyclic symmetry*
- *Linear analysis of the Indian Point reactor feedwater line*
- *Response spectra of a three-dimensional frame building*
- *Brake squeal analysis*
- *Dynamic analysis of antenna structure utilizing residual modes*
- *Steady-state dynamic analysis of a vehicle body-in-white model*

Analysis of a rotating fan using substructures and cyclic symmetry

Products: Abaqus/Standard

Overview

This example illustrates the single and multi-level substructure capability of Abaqus for problems where the part being modeled consists of repeated structures.
The example demonstrates the capability of Abaqus to analyze cyclic symmetric models using the cyclic symmetry analysis technique. Some of the limitations of modeling a structure using substructures or cyclic symmetry are also discussed.

Geometry and material

The structure is a fan consisting of a central hub and four blades, as shown in *Figure 1*. The blades and the hub are made up of S4R shell elements. The material is elastic, with a Young's modulus of 200 GPa and a Poisson's ratio of 0.29. The density of the material is 7850 kg/m^3. All nodes along the mounting hole in the hub are fixed.

Models

Four different models are considered, as follows:

1. The fan is modeled as a single structure (no substructures).
2. One quadrant of the fan, consisting of a quarter of the hub and a single blade, is reduced to a substructure. The fan is then modeled with four substructures (a single-level substructure). During substructure generation all degrees of freedom are retained for the nodes along the edges of the hub in each quadrant as well as one node at the blade tip (see *Figure 2*).
3. A single fan blade is reduced to a substructure, which is then combined with one-quarter of the hub to form a higher level substructure. Four of these substructures are then combined to form the fan (similar to the single-level substructure), thus forming a multi-level substructure. Nodes along the base of the fan blade and one node at the tip of the blade have all their degrees of freedom retained during generation of the fan blade substructure as shown in *Figure 2*. At the higher level substructure generation stage, nodes along the edge of the hub in each quadrant as well as the node at the blade tip have their degrees of freedom retained.
4. One quadrant of the fan, consisting of a quarter of the hub and a single blade, is modeled with and without substructures as a datum sector for the cyclic symmetry analysis technique. Two surfaces, which are at 90° to each other, are chosen to serve as the secondary and main surfaces for the cyclic symmetry surface-based tie constraint. The finite element mesh contains matching nodes on the symmetry surfaces; therefore, both surfaces are defined with lists of nodes or node set labels. The axis of cyclic symmetry is parallel to the global z-axis and passes through the point on the x–y plane with coordinates (3.0, 3.0). The cyclic symmetry model is shown in *Figure 3*. The entire model consists of four repetitive sectors.

Both a frequency analysis and a static analysis are performed on the first three models. Static analysis followed by a frequency extraction and a modal-based steady-state dynamic analysis are performed for the cyclic symmetry model. Stress- and load-stiffening effects due to the centrifugal loading on the fan are built into the substructure stiffness during generation using a preload step with large-displacement formulation. To get the proper stress stiffening in the hub of the multi-level substructure, the centrifugal load defined in the lowest-level substructure (the blade) needs to be captured with a substructure load case and must be applied as a substructure load in the next-level substructure.

To improve the representation of the substructure's dynamic behavior in the global analysis, *m* dynamic modes, which are extracted using an eigenfrequency extraction step, are included during the substructure generation using eigenmode selection. The reduced mass matrix obtained with the default value of $m = 0$ corresponds to the Guyan reduction technique, while $m > 0$ corresponds to the restrained mode addition technique. In the "Results and discussion" section below the solution obtained for the model without substructures (the "full model") is used as the reference solution.

For the cyclic symmetry model without substructures the eigenvalue extraction procedure is performed on the preloaded structure. The nonlinear static step has the centrifugal load applied to the blade. Eigenvalues are requested using the Lanczos eigenvalue solver, which is the only eigensolver that can be used for an eigenfrequency analysis with the cyclic symmetry analysis technique. The specification of cyclic symmetry modes in an eigenvalue analysis is demonstrated in one problem. This makes it possible to extract only the eigenmodes that have the requested cyclic symmetry. When this option is omitted, the eigenvalues are extracted for all possible (three) cyclic symmetry modes. In the discussion that follows the solution obtained for the cyclic symmetry model is compared to the solution for the entire 360° model (the reference solution). An eigenvalue analysis without the preload step is performed for the cyclic symmetry model with substructures. Twenty eigenvalues are extracted and compared to the reference solution obtained for the entire 360° model with substructures. The third step in the cyclic symmetry model problems is a frequency-domain, modal-based, steady-state procedure. It calculates the response to pressure loads projected on a specific cyclic symmetry mode.

Results and discussion

Results for the frequency analysis and the static analysis appear below.

Frequency analysis for models with substructures

Frequencies corresponding to the 15 lowest eigenvalues are extracted and tabulated in *Table 1* for each model. To study the effect of retaining dynamic modes during substructure generation, the substructure models are run after extracting 0, 5, and 20 dynamic modes during substructure generation.

While the Guyan reduction technique ($m = 0$) yields frequencies that are reasonable compared to those of the full model, the values obtained with 5 retained modes are much closer to full model predictions, especially for the higher eigenvalues. Increasing the number of retained modes to 20 does not yield a significant improvement in the results, consistent with the fact that in the Guyan reduction technique the choice of retained degrees of freedom affects accuracy, while for the restrained mode addition technique the modes corresponding to the lowest frequencies are by definition optimal.

When substructures are used in an eigenfrequency analysis, it is to be expected that the lowest eigenfrequency in the substructure model is higher than the lowest eigenfrequency in the corresponding model without substructures. This is indeed the case for the single-level substructure analysis, but for the multi-level substructure analysis the lowest eigenfrequency is *below* the one for the full model. This occurs because the stress and load stiffness for the lowest-level substructure (the blade) are generated with the root of the blade fixed, whereas in the full model the root of the blade will move radially due to the deformation of the hub under the applied centrifugal load. Hence, the substructure stiffness is somewhat inaccurate. Since the radial displacements at the blade root are small compared to the overall dimensions of the model (of order 10^{-3}), the resulting error should be small, as is observed from the results.

Table 2 shows what happens if the NLGEOM parameter is omitted during the preloading steps. It is clear that the results are significantly different from the ones that take the effect of the preload on the stiffness into account. In this case the lowest eigenfrequency in the substructure models is indeed above the lowest eigenfrequency in the model without substructures.

Static analysis for models with substructures

A static analysis of the fan is carried out about the preloaded base state by applying a pressure load of 10^5 Pa normal to the blades of the fan. The axial displacement of the outer edge of the fan blade due to the pressure load is monitored at nodes along path ***AB***, as shown in *Figure 1*. The results are shown in *Figure 4*; there is good agreement between the solutions for the substructure models and the full model.

While substructures can be generated from models that exhibit nonlinear response, it must be noted that, once created, a substructure always exhibits linear response at the usage level. Hence, a preloaded substructure will produce a response equivalent to that of the response to a linear perturbation load on a preloaded full model. Consequently, the full model is analyzed by applying the centrifugal preload in a general step and the pressure load in a linear perturbation step. Since an analysis using substructures is equivalent to a perturbation step, the results obtained do not incorporate the preload deformation. Thus, if the total displacement of the structure is desired, the results of this perturbation step need to be added to the base state solution of the structure.

Steady-state analysis with preload for the cyclic symmetry model

A modal-based, steady-state analysis of the fan is carried out about the preloaded base state, as shown in *fan_cyclicsymmodel_ss.inp*. In the general static step, which includes nonlinear geometry, the centrifugal load is applied to the datum sector. Only symmetric loads can be applied in general static steps with the cyclic symmetry analysis technique. A sequence of three eigenvalue extraction and steady-state dynamics steps follows the preload step. Each eigenvalue extraction requests only one cyclic symmetry mode that is used in the load projection in the steady-state dynamic analysis that follows. The analysis specifies that modes belonging to the cyclic symmetry modes 0, 1, and 2 should be extracted. The computed eigenvalues are identical to those obtained for the entire 360° model, as shown in *Table 1*. The additional information obtained during the eigenvalue extraction is the cyclic symmetry mode number associated with each eigenvalue. In the case of 4 repetitive sectors, all the eigenvalues corresponding to cyclic symmetry mode 1 appear in pairs; the eigenvalues corresponding to modes 0 and 2 are single. The lowest two eigenvalues correspond to cyclic symmetry mode 1, followed by the single eigenvalues corresponding to cyclic symmetry modes 2 and 0. For a comparison with the cyclic symmetry model option, the eigenvalue problem is also modeled with multi-point constraint type CYCLSYM (see *fansubstr_mpc.inp*). To verify the use of substructures with the cyclic symmetry model, it was determined that the eigenvalues obtained with *fansubstr_cyclic.inp* were identical to those obtained with *fansubstr_1level_freq.inp*. The last step is the modal-based, steady-state dynamic analysis. A pressure load is applied to the entire structure as projected onto three different cyclic symmetry modes.

Input files

fan_cyclicsymmodel_ss.inp	Cyclic symmetry model with static, eigenvalue, and steady-state dynamics steps with the load projected onto the cyclic modes 0, 1, and 2, respectively.
fansubstr_1level_freq.inp	Single-level substructure usage analysis with a frequency extraction step.
fansubstr_1level_static.inp	Single-level substructure usage analysis with a static step.
fansubstr_multi_freq.inp	Multi-level substructure usage analysis with a frequency extraction step.
fansubstr_multi_static.inp	Multi-level substructure usage analysis with a static step.
fansubstr_freq.inp	Frequency extraction without substructures.
fansubstr_static.inp	Static analysis without substructures.
fansubstr_mpc.inp	Single-level usage analysis demonstrating the use of cyclic symmetry MPCs.
fansubstr_gen1.inp	Substructure generation for a single blade used in the multi-level substructure generation file fansubstr_gen2.inp.

fansubstr_gen2.inp	Multi-level substructure generation used infansubstr_multi_freq.inp and fansubstr_multi_static.inp.
fansubstr_gen3.inp	Single-level substructure generation used infansubstr_1level_freq.inp, fansubstr_1level_static.inp, and fansubstr_mpc.inp.
fansubstr_cyclic.inp	Single-level substructure with the cyclic symmetry model used in a frequency analysis.

Tables

Table 1: Comparison of natural frequencies for single-level and multi-level substructures with the values for the model without substructures.

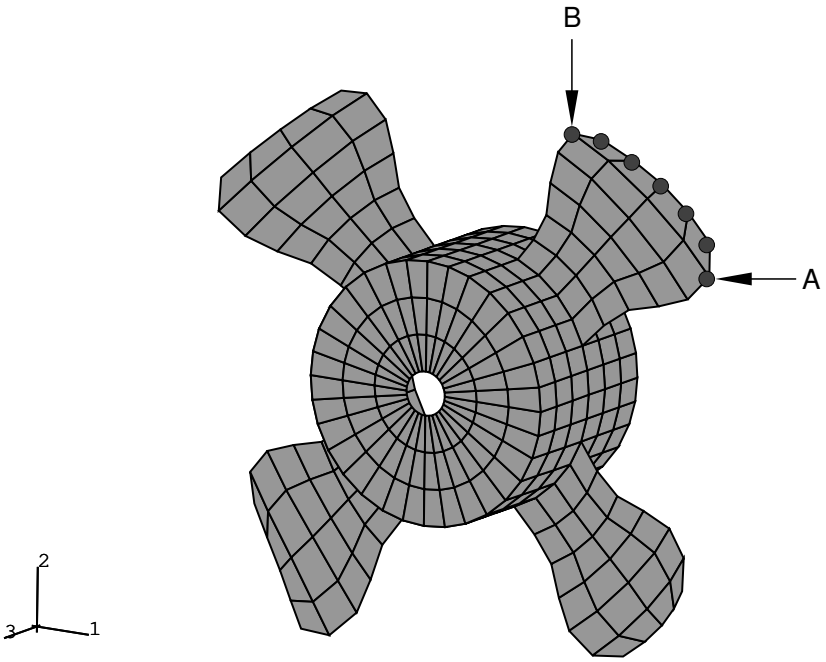
Eigenvalue no. cycles/sec	With substructuring: 1 level			With substructuring: 2 levels			Full model
	m=0	m=5	m=20	m=0	m=5	m=20	
1	6.9477	6.7901	6.7891	6.7655	6.6269	6.6258	6.7890
2	6.9477	6.7901	6.7891	6.7655	6.6269	6.6258	6.7890
3	8.0100	7.7207	7.7198	7.8162	7.5563	7.5552	7.7198
4	8.2009	7.8816	7.8810	8.1986	7.8813	7.8807	7.8810
5	11.341	11.020	11.010	11.123	10.802	10.792	11.009
6	11.341	11.020	11.010	11.123	10.802	10.792	11.009
7	12.529	11.930	11.912	11.539	11.142	11.124	11.910
8	14.751	14.397	14.346	13.450	13.256	13.211	14.348
9	17.787	14.432	14.432	17.208	14.455	14.455	14.431
10	18.922	14.779	14.775	18.797	14.751	14.747	14.774
11	21.250	14.779	14.775	19.860	14.751	14.747	14.774
12	21.250	16.034	15.995	19.860	15.645	15.623	15.991
13	28.250	17.699	17.624	28.066	17.129	17.057	17.624
14	28.691	19.034	19.019	28.628	18.914	18.901	19.008
15	28.691	21.333	21.178	28.628	20.014	19.885	21.176

Table 2: Comparison of natural frequencies for single-level and two-level substructures with the full model values without the use of the NLGEOM parameter.

Eigenvalue no. cycles/sec	With substructuring		Full model
	1 level	2 levels	
1	4.4795	4.4795	4.4795
2	4.4795	4.4795	4.4795
3	4.5511	4.5511	4.5511
4	4.8889	4.8889	4.8889
5	9.5431	9.5431	9.5426
6	9.5431	9.5431	9.5426
7	9.7921	9.7921	9.7918
8	12.632	12.633	12.632

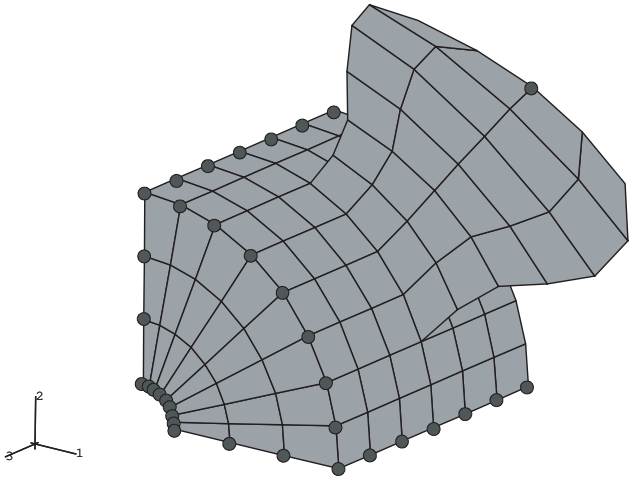
Eigenvalue no. cycles/sec	With substructuring		Full model
	1 level	2 levels	
9	14.005	14.005	14.005
10	14.336	14.336	14.336
11	14.336	14.336	14.336
12	15.489	15.489	15.489
13	16.861	16.861	16.860
14	18.241	18.241	18.232
15	19.036	19.036	19.034

Figures

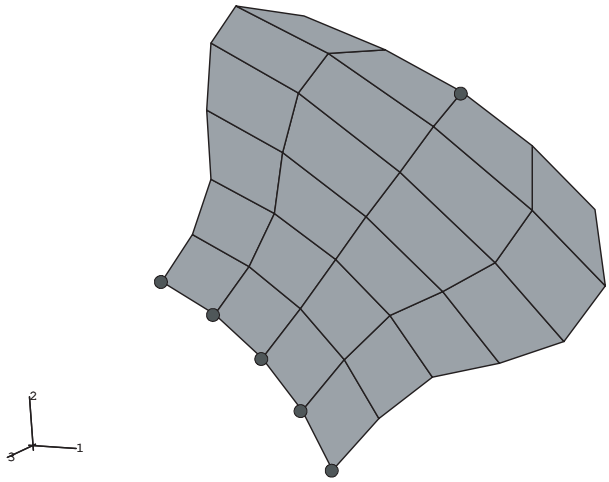


Full model indicating monitored nodes.

Figure 1: Mesh used for the complete fan model.



(a) One-level substructure showing retained nodes.



(b) First level of the two-level substructure showing retained nodes.

Figure 2: Substructures generated.

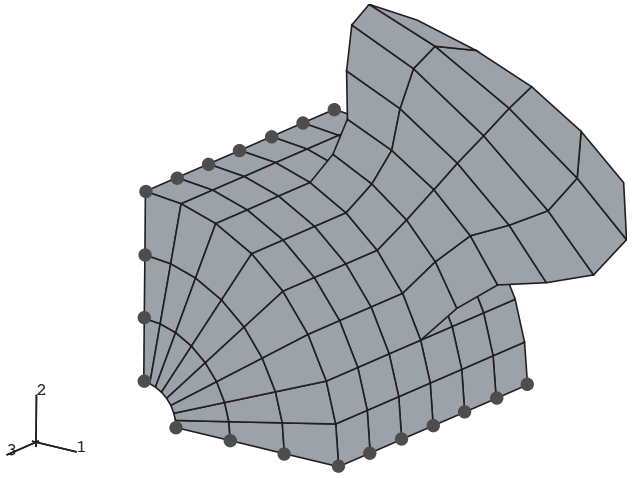


Figure 3: Datum sector for cyclic symmetry model.

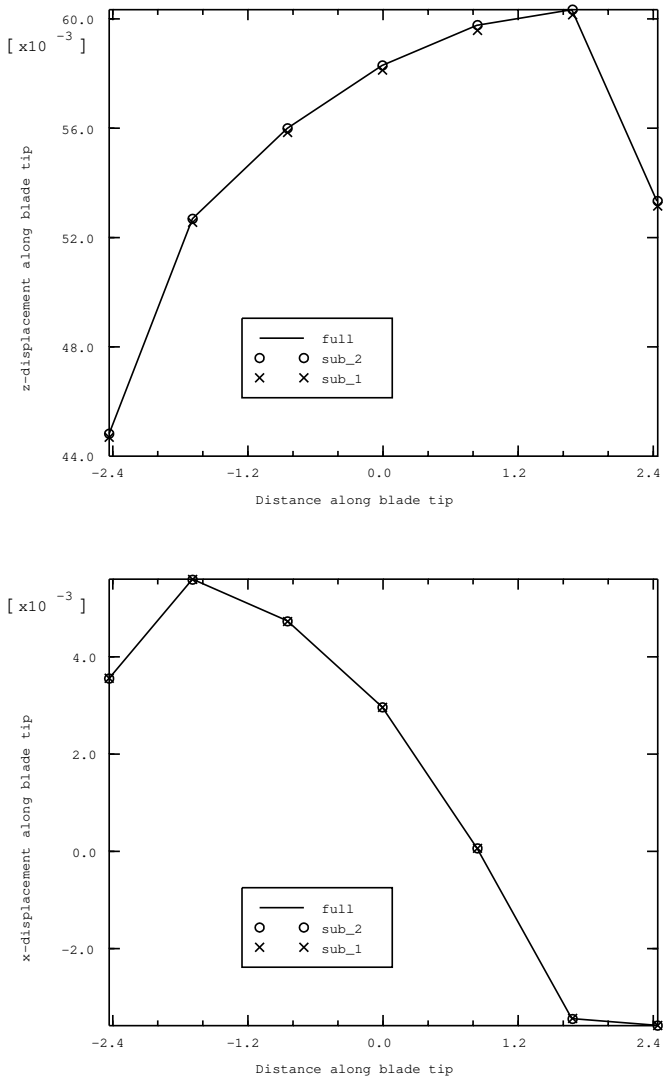


Figure 4: Displacements due to pressure loading along path AB.

Linear analysis of the Indian Point reactor feedwater line

Products: Abaqus/Standard

Overview

This example shows the linear analysis of an actual pipeline from a nuclear reactor and illustrates some of the issues that must be addressed in performing seismic piping analysis.

The pipeline is the Indian Point Boiler Feedwater Pipe fitted with modern supports, as shown in *Figure 1*. This pipeline was tested experimentally in EPRI's full-scale testing program. The model corresponds to Configuration 1 of the line in Phase III of the testing program. The experimental results are documented in EPRI Report NP–3108 Volume 1 (1983).

We first verify that the geometric/kinematic model is adequate to simulate the dynamic response accurately. For this purpose we compare predictions of the natural frequencies of the system using a coarse model and a finer model, as well as two substructure models created from the coarse mesh. These analyses are intended to verify that the models used in subsequent runs provide accurate predictions of the lower frequencies of the pipeline. We then perform linear dynamic response analysis in the time domain for one of the "snap-back" loadings applied in the physical test (EPRI NP–3108, 1983) and compare the results with the experimental measurements. The linear dynamic response analysis results are also compared with the results of direct integration analysis (integration of all variables in the entire model, as would be performed for a generally nonlinear problem). This is done primarily for cross-verification of the two analysis procedures. These snap-back response analyses correspond to a load of 31136 N (7000 lb) applied at node 25 in the z-direction, with the pipe filled with water. This load case is referred to as test S138R1SZ in EPRI NP–3108.

We also compute the pipeline's response in the frequency domain to steady excitation at node 27 in the z-direction. Experimental data are also available for comparison with these results.

Geometry and model

Geometrical and material properties are taken from EPRI NP–3108 (1983). The supports are assumed to be linear springs for the purpose of these linear analyses, although their actual response is probably nonlinear. The spring stiffness values are those recommended by Tang et al. (1985). The pipe is assumed to be completely restrained in the vertical direction at the wall penetration.

In the experimental snap-back test used for the comparison (test S138R1SZ), the pipe is full of water. The density of the general beam section is, therefore, adjusted to account for the additional mass of the water by computing a composite (steel plus water) mass per unit length of pipe.

The pipeline is modeled with element type B31. This is a shear flexible beam element that uses linear interpolation of displacement and rotation between two nodes, with transverse shear behavior modeled according to Timoshenko beam theory. The element uses a lumped mass matrix because this provides more accurate results in test cases.

The coarse finite element model uses at least two beam elements along each straight run, with a finer division around the curved segments of the pipe to describe the curvature of the pipe with reasonable accuracy. Separate nodes are assigned for all spring supports, external loading locations, and all the points where experimental data have been recorded. The model is shown in *Figure 2*. This mesh has 74 beam elements.

In typical piping systems the elbows play a dominant role in the response because of their flexibility. This could be incorporated in the model by using the ELBOW elements. However, ELBOW elements are intended for applications that involve nonlinear response within the elbows themselves and are an expensive option for linear response of the elbows, which is the case for this study. Therefore, instead of using elbow elements, we modify the geometrical properties of beam elements to model the elbows with correct flexibility. This is done by calculating the flexibility factor, k, for each elbow and modifying the moments of inertia of the beam cross-sections

in these regions. The flexibility factor for an elbow is a function of two parameters. One is a geometric parameter, λ, defined as

$$\lambda = \frac{tR}{r^2\sqrt{1-\nu^2}},$$

where t is the wall thickness of the curved pipe, R is the bend radius of the centerline of the curved pipe, r is the mean cross-sectional radius of the curved pipe, and ν is Poisson's ratio. The other parameter is an internal pressure loading parameter, ψ. For thick sections (like the ones used in this pipe), ψ has negligible effect unless the pressures are very high and the water in this case is not pressurized. Consequently, the flexibility factor is a function of λ only.

For the elbows in this pipeline $\lambda =$ 0.786 for the 203 mm (8 in) section and $\lambda =$ 0.912 for the 152 mm (6 in) section. The corresponding flexibility factors obtained from Dodge and Moore (1972) are 2.09 and 1.85. These are implemented in the model by modifying the moments of inertia of the beam cross-sections in the curved regions of the pipeline.

Abaqus provides two different options for introducing geometrical properties of a beam cross-section. For general beam sections all geometric properties (area, moments of inertia) can be given without specifying the shape of the cross-section. The material data, including the density, are given on the same option. Alternatively, the geometrical properties of the cross-section can be given by using a beam section. With this option the cross-section dimensions are given, and Abaqus calculates the corresponding cross-sectional behavior by numerical integration, thus allowing for nonlinear material response in the section. When this option is used, the material properties—including density and damping coefficients—are introduced in the material definition associated with the section. This approach is more expensive for systems in which the cross-sectional behavior is linear, since numerical integration over the section is required each time the stress must be computed. Thus, in this case we use a general beam section.

To verify that the mesh will provide results of adequate accuracy, the natural frequencies predicted with this model are compared with those obtained with another mesh that has twice as many elements in each pipe segment. *Table 1* shows that these two meshes provide results within 2% for the first six modes and generally quite similar frequencies up to about 30 Hz. Based on this comparison the smaller model, with 74 beam elements, is used for the remaining studies (although the larger model would add little to the cost of the linear analyses, which for either case would be based on the same number of eigenmodes: only in direct integration would the cost increase proportionally with the model size).

Substructure models

In Abaqus the dynamic response of a substructure is defined by a combination of Guyan reduction and the inclusion of some natural modes of the fully restrained substructure. Guyan reduction consists of choosing additional physical degrees of freedom to retain in the dynamic model that are not needed to connect the substructure to the rest of the mesh. In this example we use only Guyan reduction since the model is small and it is easy to identify suitable degrees of freedom to retain. A critical modeling issue with this method is the choice of retained degrees of freedom: enough degrees of freedom must be retained so that the dynamic response of the substructure is modeled with sufficient accuracy. The retained degrees of freedom should be such as to distribute the mass evenly in each substructure so that the lower frequency response of each substructure is modeled accurately. Only frequencies up to 33 Hz are generally considered important in the seismic response of piping systems such as the one studied in this example, so the retained degrees of freedom must be chosen to provide accurate modeling of the response up to that frequency.

In this case the pipeline naturally divides into three segments in terms of which kinematic directions participate in the dynamic response, because the response of a pipeline is generally dominated by transverse displacement. The lower part of the pipeline, between nodes 1 and 23, is, therefore, likely to respond predominantly in degrees of freedom 1 and 2; the middle part, between nodes 23 and 49, should respond in degrees of freedom 2 and 3; and the top part, above node 49, should respond in degrees of freedom 1 and 3. Comparative tests (not documented)

have been run to verify these conjectures, and two substructure models have been retained for further analysis: one in which the entire pipeline is treated as a single substructure, and one in which it is split into three substructures. In the latter case all degrees of freedom must be retained at the interface nodes to join the substructures correctly. At other nodes only some translational degrees of freedom are retained, based on the arguments presented above.

The choice of which degrees of freedom to retain can be investigated inexpensively in a case such as this by numerical experiments—extracting the modes of the reduced system for the particular set of retained degrees of freedom and comparing these modes with those of the complete model. The choices made in the substructure models used here are based somewhat on such tests, although insufficient tests have been run to ensure that they are close to the optimal choice for accuracy with a given number of retained variables. For linear analysis of a model as small as this one, achieving an optimal selection of retained degrees of freedom is not critical because computer run times are short: it becomes more critical when the reduced model is used in a nonlinear analysis or where the underlying model is so large that comparative eigenvalue tests cannot be performed easily. In such cases the inclusion of natural modes of the substructure is desirable. The substructure models are shown in *Figure 2*.

Damping

"Damping" plays an essential role in any practical dynamic analysis. In nonlinear analysis the "damping" is often modeled by introducing dissipation directly into the constitutive definition as viscosity or plasticity. In linear analysis equivalent linear damping is used to approximate dissipation mechanisms that are not modeled explicitly.

Experimental estimates of equivalent linear damping, based on three different methods, are found in EPRI NP–3108 (see Table 7–6, Table 7–7, and Figure 7–15 of that report). For the load case and pipe configuration analyzed here, those results suggest that linear damping corresponding to 2.8% of critical damping in the lowest mode of the system matches the measured behavior of the structure, with the experimental results also showing that the percentage of critical damping changes from mode to mode. Despite this all the numerical analyses reported here assume the same damping ratio for all modes included in the model, this choice being made for simplicity only.

For linear dynamic analysis based on the eigenmodes, Abaqus allows damping to be defined as a percentage of critical damping in each mode, as structural damping (proportional to nodal forces), or as Rayleigh damping (proportional to the mass and stiffness of the structure). Only the last option is possible when using direct integration, although other forms of damping can be added as discrete dashpots or in the constitutive models. In this case results are obtained for linear dynamic analysis with modal and Rayleigh damping and for direct integration with Rayleigh damping. For linear dynamic analysis based on the eigenmodes, modal damping can be specified using the mode numbers or using the frequency ranges.

Results and discussion

Results are shown for four geometric models: the "coarse" (74 element) model, which has a total of 435 degrees of freedom; a finer (148 element) model, which has a total of 870 degrees of freedom; a model in which the pipeline is modeled as a single substructure (made from the coarse model), with 59 retained degrees of freedom; and a model in which the pipeline is modeled with three substructures (made from the coarse model), with 65 retained degrees of freedom.

The first comparison of results is the natural frequencies of the system, as they are measured and as they are predicted by the various models. The first 24 modes are shown in *Table 1*. These modes span the frequency range from the lowest frequency (about 4.3 Hz) to about 43 Hz. In typical seismic analysis of systems such as this, the frequency range of practical importance is up to 33 Hz; on this basis these modes are more than sufficient.

Only the first six modes of the actual system have been measured, so any comparison at higher frequencies is between the numerical calculations reported here and other similar computations. The results obtained with the four models correlate quite well between themselves, suggesting that the mesh and the choices of retained degrees

of freedom in the substructure models are reasonable. It is particularly noteworthy that the results for the substructure models correspond extremely well with those provided by the original model, considering the large reduction in the number of degrees of freedom for the substructures. The results also correlate roughly with the analysis results obtained by EDS and reported in EPRI NP-3108: except for modes 3 and 4 the frequencies are within 10% of the EDS numbers. For the first three modes the Abaqus results are lower than those reported by EDS. This suggests the possibility that the Abaqus model may be too flexible. The SUPERPIPE values are significantly higher than any of the other data for most modes, and the Abaqus and the EDS results diverge from the test results after the first four modes.

The results of the time history analyses are summarized in *Table 2* to *Table 5*. These analyses are based on using all 24 modes of the coarse model. Typical predicted response plots are shown in *Figure 3* to *Figure 7*. In many cases of regular, beam-type, one-dimensional structures, the first few modes will generally establish the dynamic behavior. Although the pipeline has an irregular shape, it is worth checking how much the higher modes influence the results. This is done in this case by comparing the results using the first six modes only with the results obtained with 24 modes. The highest discrepancy (20%) is found in the predicted accelerations at certain degrees of freedom. All other results show at most 5–10% differences (see *Figure 3* and *Figure 4*). This conclusion is also supported by the steady-state results.

All the Abaqus results are reasonably self-consistent, in the sense that Rayleigh and modal damping and modal dynamics and direct integration all predict essentially the same values. The choice of 2.8% damping seems reasonable, in that oscillations caused by the snap-back are damped out almost completely in 10 seconds, which corresponds to the measurements.

Unfortunately there is poor correlation between predicted and measured support reactions and maximum recorded displacements. The test results and the corresponding computations are shown in *Table 2* and *Table 3*. All the models give essentially the same values. The initial reactions and displacements are computed for a snap-back load of 31136 N (7000 lb) applied at node 25 (node 417 in EPRI report NP–3108) in the z-direction. The maximum recorded displacements occur at node 27 (node 419 in EPRI report NP–3108) in the y- and z-directions. It is assumed that the supports are in the positions relative to the pipe exactly as shown in *Figure 1*. The scatter in the experimental measurements makes it difficult to assess the validity of the stiffness chosen for the spring supports. The maximum displacement predicted at node 27 in the z-direction is almost twice that measured. This again implies the possibility that, at least in the area near this node, the model is too flexible.

The generally satisfactory agreement between the natural frequency predictions and poor agreement between the maximum displacements and reactions suggests that improved modeling of the supports may be necessary. In this context it is worthwhile noting that the experimental program recorded significantly different support parameters in different tests on the pipeline system.

Table 4 shows the results for displacement and acceleration for node 27 (which has the largest displacement). All the computed results are higher than the experimental values. The largest discrepancies between the measurements and the analysis results are in the predictions of peak forces in the springs, summarized in *Table 5*. Results obtained with the various models differ by less than 10%: these differences are caused by the differences in the models, different types of damping, and—for the direct integration results—errors in the time integration (for the modal dynamic procedure the time integration is exact). The principal cause of the discrepancies between the measurements and the computed values is believed to be the assumption of linear response in the springs in the numerical models. In reality the spring supports are either rigid struts or mechanical snubbers (Configuration 2). Especially when snubbers are used, the supports perform as nonlinear elements and must be modeled as such to reflect the support behavior accurately. Interestingly, even with the assumption of linear support behavior, the character of the oscillation is well-predicted for many variables.

The last group of numerical results are frequency domain calculations obtained using the steady-state dynamic analysis. The response corresponds to steady harmonic excitation at node 27 in the z-direction by a force with a peak amplitude of 31136 N (7000 lb). Such frequency domain results play a valuable role in earthquake analysis because they define the frequency ranges in which the structure's response is most amplified by the excitation. Although it is expected that the first few natural frequencies will be where the most amplification occurs, the results show clearly that some variables are strongly amplified by the fifth and sixth modes. This is observed both in the simulations and in the experimental measurements. Measured experimental results are available for

the acceleration of node 33 (node 419 in EPRI NO–3108) in the z-direction and for the force in spring FW-R-21. The character of curves obtained with Abaqus agrees well with the experimental results (see *Figure 8* and *Figure 9*), but the values differ significantly, as in the time domain results. The peak acceleration recorded is 2.0 m/s^2 (78.47 in/s^2), at the first natural frequency, while the analysis predicts 4.0 m/s^2 (157.5 in/s^2). Likewise, the peak force value recorded is 2.0 kN (450 lb), compared to 5.9 kN (1326 lb) predicted. The discrepancies are again attributed to incorrect estimates of the support stiffness or to nonlinearities in the supports.

Input files

indianpoint_modaldyn_coarse.inp	**MODAL DYNAMIC* analysis with modal damping using the coarse model.
indianpoint_modaldyn_3sub.inp	**MODAL DYNAMIC* analysis using the three substructure model.
indianpoint_3sub_gen1.inp	First substructure generation referenced by the analysis indianpoint_modaldyn_3sub.inp.
indianpoint_3sub_gen2.inp	Second substructure generation referenced by the analysis indianpoint_modaldyn_3sub.inp.
indianpoint_3sub_gen3.inp	Third substructure generation referenced by the analysis indianpoint_modaldyn_3sub.inp.
indianpoint_sstate_sinedwell.inp	**STEADY STATE DYNAMICS* analysis corresponding to the sine dwell test performed experimentally using the coarse model.
indianpoint_direct_beam_coarse.inp	Direct integration analysis using the coarse model with the **BEAM SECTION* option.
indianpoint_sstate_modaldamp.inp	**STEADY STATE DYNAMICS* analysis with modal damping, covering a range of frequencies using the coarse model.
indianpoint_modaldyn_1sub.inp	**MODAL DYNAMIC* analysis with one substructure.
indianpoint_1sub_gen1.inp	Substructure generation referenced by the analysis indianpoint_modaldyn_1sub.inp.
indianpoint_direct_beamgensect.inp	Direct integration using the coarse model with **BEAM GENERAL SECTION* instead of **BEAM SECTION*, which, thus, runs faster on the computer since numerical integration of the cross-section is avoided.
indianpoint_modaldyn_elmatrix1.inp	**MODAL DYNAMIC* analysis that reads and uses the substructure matrix written to the results file in indianpoint_3sub_gen1.inp, indianpoint_3sub_gen2.inp, and indianpoint_3sub_gen3.inp.
indianpoint_modaldyn_elmatrix2.inp	Reads and uses the element matrix written to the results file in indianpoint_modaldyn_3sub.inp.
indianpoint_modaldyn_elmatrix3.inp	Reads and uses the substructure matrix written to the results file in indianpoint_1sub_gen1.inp.
indianpoint_modaldyn_elmatrix4.inp	Reads and uses the element matrix written to the results file in indianpoint_modaldyn_1sub.inp.
indianpoint_modaldamp_rayleigh.inp	**MODAL DAMPING* analysis with modal Rayleigh damping using the coarse mesh with the **BEAM SECTION* option.
indianpoint_dyn_rayleigh_3sub.inp	**DYNAMIC* analysis with Rayleigh damping using the three substructure model.
indianpoint_rayleigh_3sub_gen1.inp	First substructure generation referenced by the analysis indianpoint_dyn_rayleigh_3sub.inp.
indianpoint_rayleigh_3sub_gen2.inp	Second substructure generation referenced by the analysis indianpoint_dyn_rayleigh_3sub.inp.
indianpoint_rayleigh_3sub_gen3.inp	Third substructure generation referenced by the analysis indianpoint_dyn_rayleigh_3sub.inp.

indianpoint_modaldyn_unsorted.inp	One substructure **MODAL DYNAMIC* analysis with unsorted node sets and unsorted retained degrees of freedom.
indianpoint_unsorted_gen1.inp	Substructure generation with unsorted node sets and unsorted retained degrees of freedom referenced by the analysis indianpoint_modaldyn_unsorted.inp.
indianpoint_lanczos.inp	Same as indianpoint_modaldyn_coarse.inp, except that it uses the Lanczos solver and the eigenvectors are normalized with respect to the generalized mass.
indianpoint_restart_normdisp.inp	Restarts from indianpoint_lanczos.inp and continues the eigenvalue extraction with the eigenvectors normalized with respect to the maximum displacement.
indianpoint_restart_bc.inp	Restarts from indianpoint_lanczos.inp and continues the eigenvalue extraction with modified boundary conditions.
indianpoint_overlapfreq.inp	Contains two steps, which extract eigenvalues with overlapping frequency ranges.

References

- Consolidated Edison Company of New York, Inc., EDS Nuclear, Inc., and Anco Engineers, Inc., *Testing and Analysis of Feedwater Piping at Indian Point Unit 1, Volume 1: Damping and Frequency*, EPRI NP-3108, vol. 1, July 1983.
- Dodge, W. G., and S. E. Moore, "Stress Indices and Flexibility Factors for Moment Loadings in Elbows and Curved Pipe," *WRC Bulletin*, no. 179, December 1972.
- Tang, Y. K., M. Gonin, and H. T. Tang, "Correlation Analysis of In-situ Piping Support Reactions," *EPRI correspondence with Abaqus*, May 1985.

Tables

Table 1: Comparison of natural frequencies (Hz).

Mode	Anco (experiment)	EDS	SUPER PIPE	Abaqus			
				coarse mesh	finer mesh	single sub	three subs
1	4.20	4.30	5.30	4.25	4.26	4.25	4.25
2	6.80	6.80	8.10	6.27	6.25	6.27	6.27
3	8.30	8.80	12.00	7.29	7.29	7.30	7.30
4	12.60	10.60	13.30	12.80	12.66	12.87	12.86
5	15.40	13.00	14.40	13.18	13.14	13.19	13.20
6	16.70	14.50	15.90	13.90	13.75	13.91	13.92
7		16.20	18.30	15.11	15.98	14.34	14.39
8			19.40	16.30	16.07	16.24	16.31
9			20.20	16.89	16.81	16.43	16.43
10			22.20	17.43	17.82	17.17	17.20
11				18.02	19.07	18.10	18.10
12				19.58	20.10	20.05	20.01
13				23.43	21.45	23.98	24.00

Mode	Anco (experiment)	EDS	SUPER PIPE	Abaqus			
				coarse mesh	finer mesh	single sub	three subs
14				23.99	22.13	24.47	24.47
15				24.27	23.58	24.97	24.96
16				24.80	24.15	25.34	25.28
17				26.82	26.84	27.63	27.56
18				29.53	30.18	30.31	30.55
19				30.61	30.60	31.08	31.06
20				30.95	32.58	31.43	31.43
21				31.52	33.11	32.00	31.98
22				33.50	35.08	33.76	33.77
23				39.09	39.65	39.75	39.97
24				39.86	43.25	42.98	42.97

Table 2: Comparison of initial support reactions. Snap-back Test No. S138R1SZ; 31136 N (7000 lb) at node 25, z-direction.

NODE	SUPPORT	Anco TEST		Abaqus	
		N	(lb)	N	(lb)
15	FW-R-11	−8000	(−1798.6)	−11712	(−2633)
22	FW-R-13	30000	(6744.6)	29352	(6599)
23	FW-R-14	−252	(−56.7)	−3754	(−844)
35	FW-R-17	23625	(5311.4)	102	(−22.8)
35	FW-R-18	10025	(2553.8)	−18468	(−4152)
39	FW-R-20	24000	(5395.7)	4212	(947)
39	FW-R-21	−24500	(−5508.1)	25016	(5624)
49	FW-R-23	8000	(1798.6)	24348	(5474)
53	FW-R-24	−4324	(−972.1)	−4057	(−912)
53	FW-R-25	2000	(449.6)	−816	(−183)
56	FW-R-27	432	(97.1)	−1801	(−405)
56	FW-R-28	156	(35.1)	−799	(−180)

Table 3: Comparison of maximum displacements.

Abaqus NODE No.	Anco NODE No.	Measured		Abaqus	
		mm	(in)	mm	(in)
27	419-Y	−16.0	(−.630)	−26.85	(−1.057)
27	419-Z	37.81	(1.49)	65.72	(2.587)

Table 4: Peak displacement and acceleration values at node 27.

Variable	Measured (Anco)	Abaqus		
		Modal, 2.8% modal damping	Modal, Rayleigh damping	Direct integration
u_y (mm)	−0.024/0.024	−0.029/0.029	−0.031/0.031	−0.031/0.031
u_z (mm)	−0.038/0.038	−0.058/0.066	−0.062/0.059	−0.063/0.068
$\ddot{u}_z$ (m/s^2)	−47.6/40.9	−42.1/50.8	−49.6/49.9	−83.8/91.0
The high acceleration amplitude reported for the Abaqus direct integration analysis occurs only during the first few increments, after which it reduces to −31.6/48.6 m/s^2.				

Table 5: Peak reaction forces at supports (in kN).

Support number	Measured (Anco)	Abaqus		
		Modal, 2.8% modal damping	Modal, Rayleigh damping	Direct integration
FW-R-11	−16.44/19.22	−19.80/13.42	−19.90/14.26	−21.82/15.76
FW-R-13	−15.10/29.91	−18.94/24.45	−19.46/23.61	−28.50/21.98
FW-R-14	−7.22/12.00	−9.34/7.35	−10.23/10.00	−12.54/9.03
FW-R-17	34.40/26.20	−7.50/10.59	−.17/9.25	−7.91/10.97
FW-R-18	−14.30/14.40	−33.26/32.06	−33.58/31.61	−33.46/32.63
FW-R-20	−25.60/26.90	−7.54/8.79	−7.98/8.50	−8.07/10.60
FW-R-21	−24.50/23.80	−25.55/24.47	−26.38/25.30	−27.78/25.26
FW-R-23	−15.30/16.00	−25.39/24.63	−26.06/25.36	−25.40/24.35
FW-R-24	−9.61/7.30	−7.17/6.87	−7.69/7.20	−8.23/8.64
FW-R-25	−6.77/6.21	−3.48/4.36	−3.34/4.55	−7.13/4.71
FW-R-27	−3.76/3.04	−4.12/4.00	−3.78/3.80	−4.29/4.43
FW-R-28	−1.10/1.82	−1.53/1.08	−1.62/1.15	−1.79/1.44

Figures

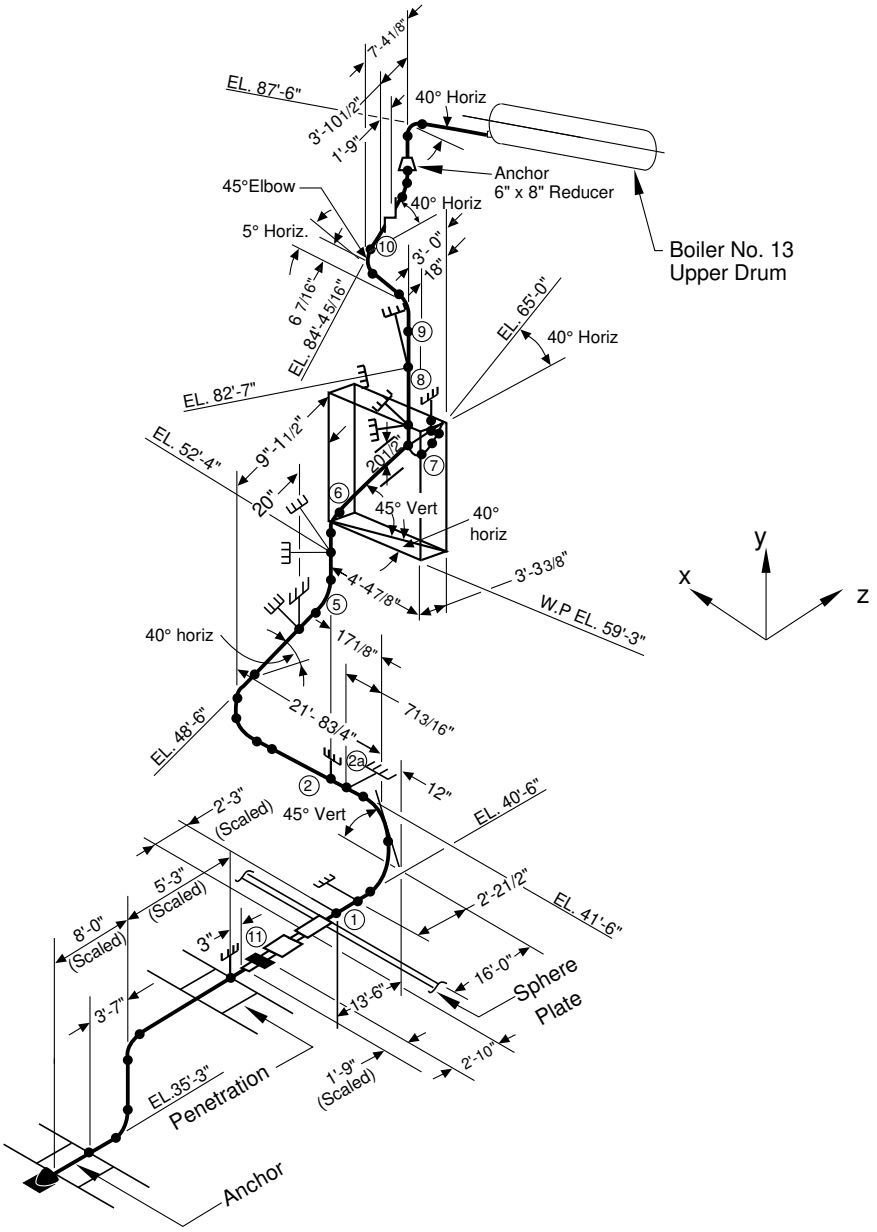


Figure 1: Indian Point boiler feedwater line: modern supports, Configuration 1.

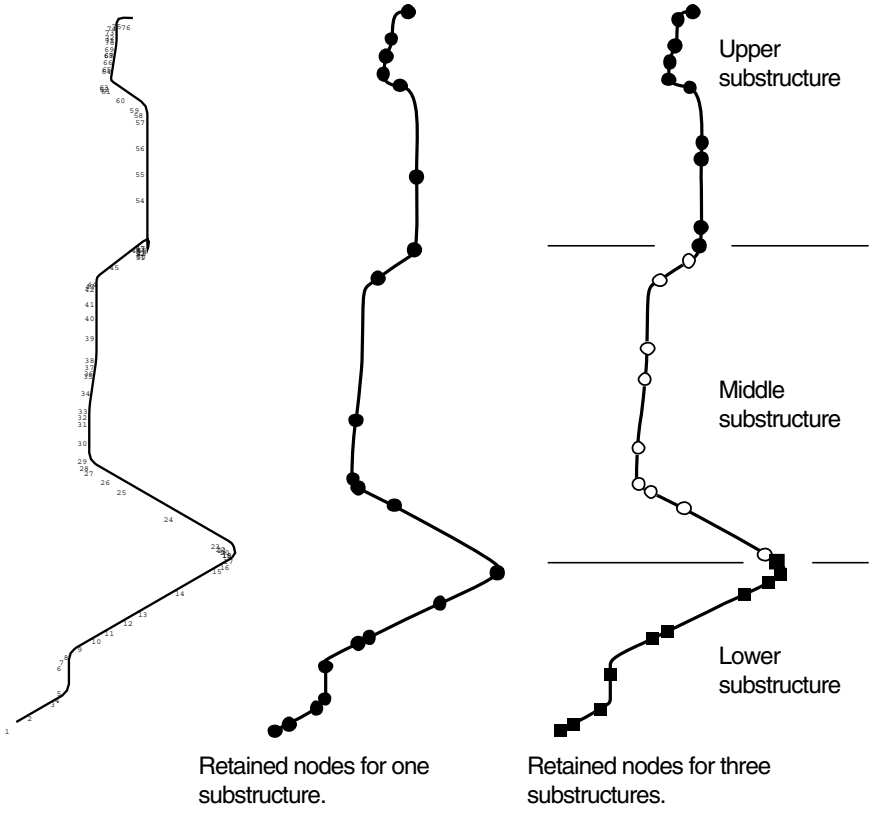


Figure 2: Basic mesh and substructure models.

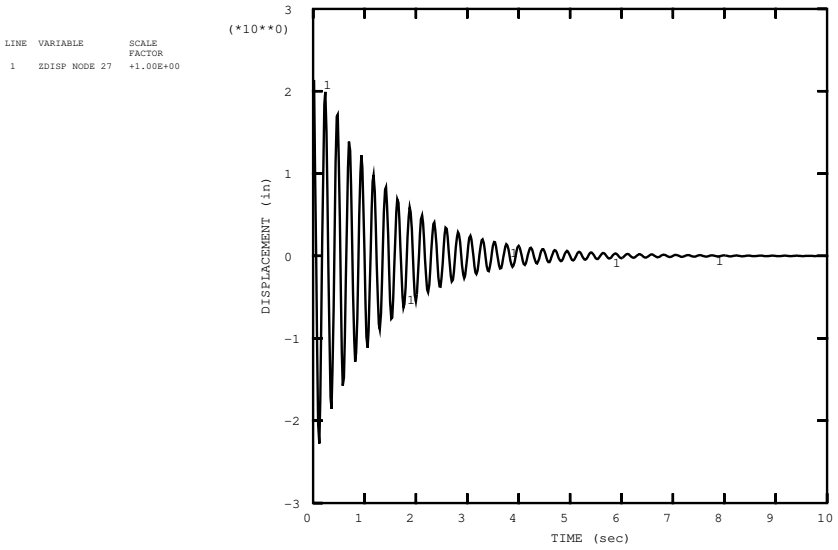


Figure 3: z-displacement at node 27, modal analysis with 24 modes.

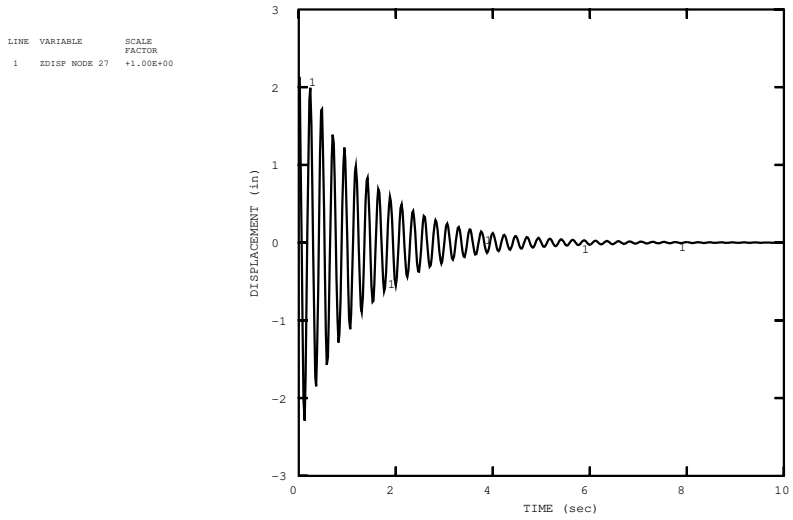


Figure 4: z-displacement at node 27, modal analysis with 6 modes.

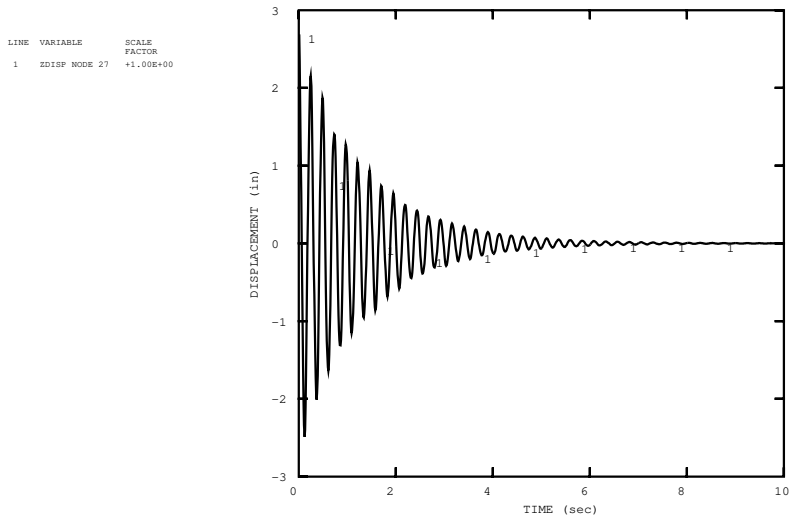


Figure 5: z-displacement at node 27, direct integration analysis.

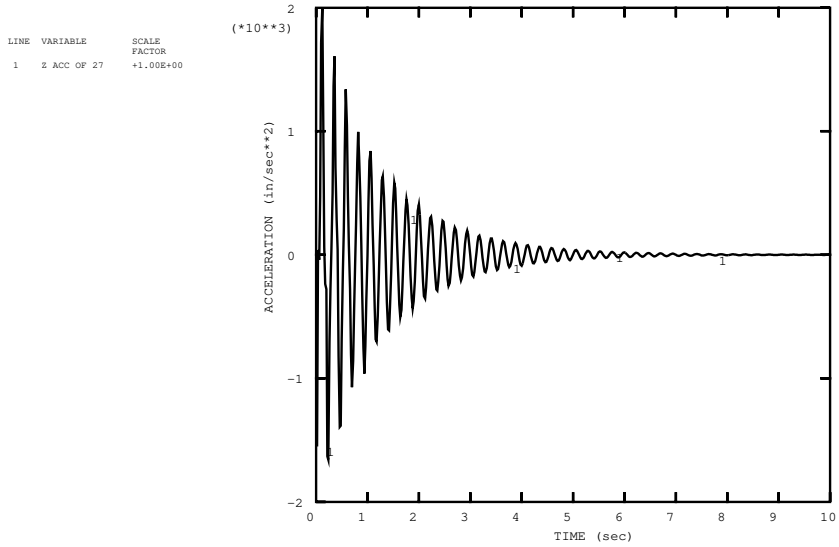


Figure 6: z-direction acceleration at node 27, modal analysis with 24 modes.

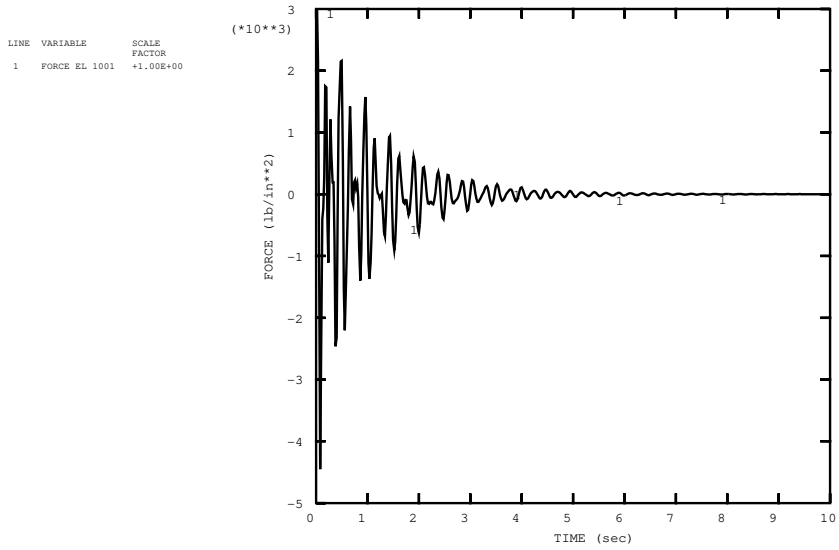


Figure 7: Force in spring support FW–R–11, modal analysis with 24 modes.

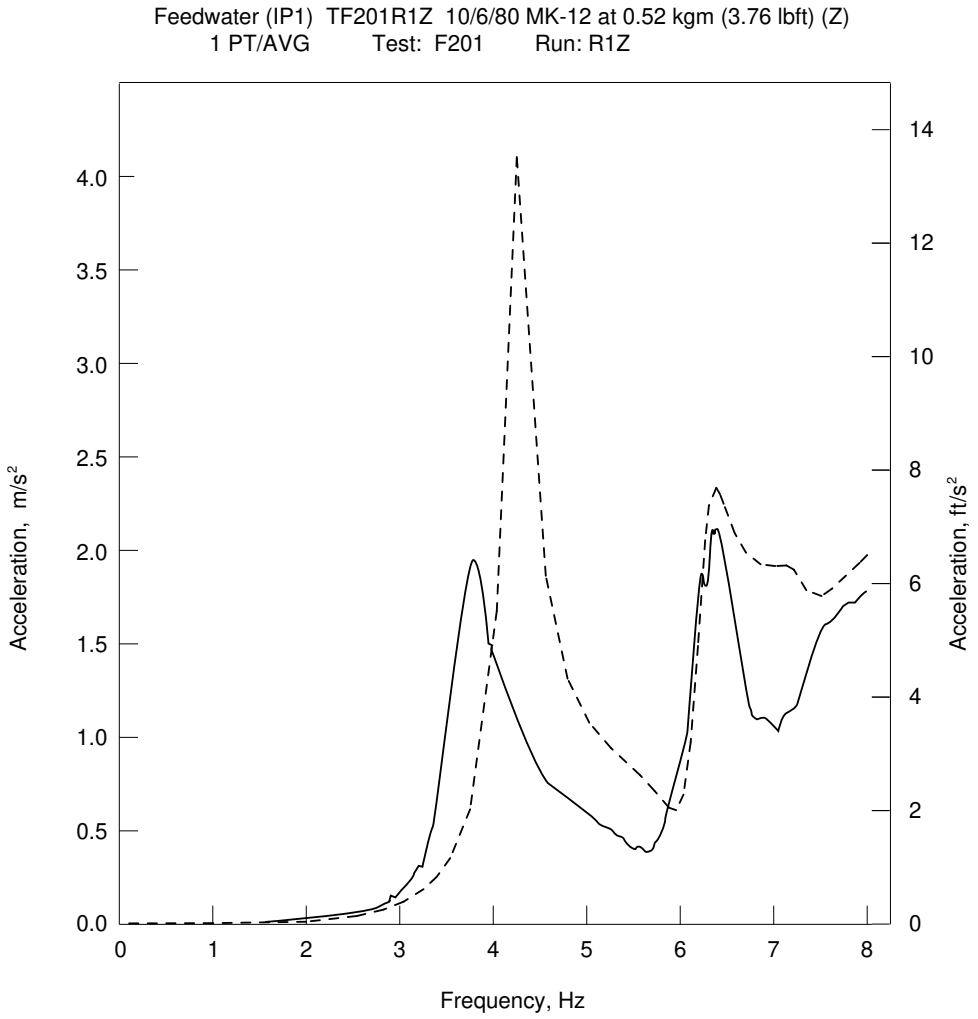


Figure 8: Comparison of z-direction acceleration at node 33 between experimental steady-state results (solid line) and Abaqus (dashed line).

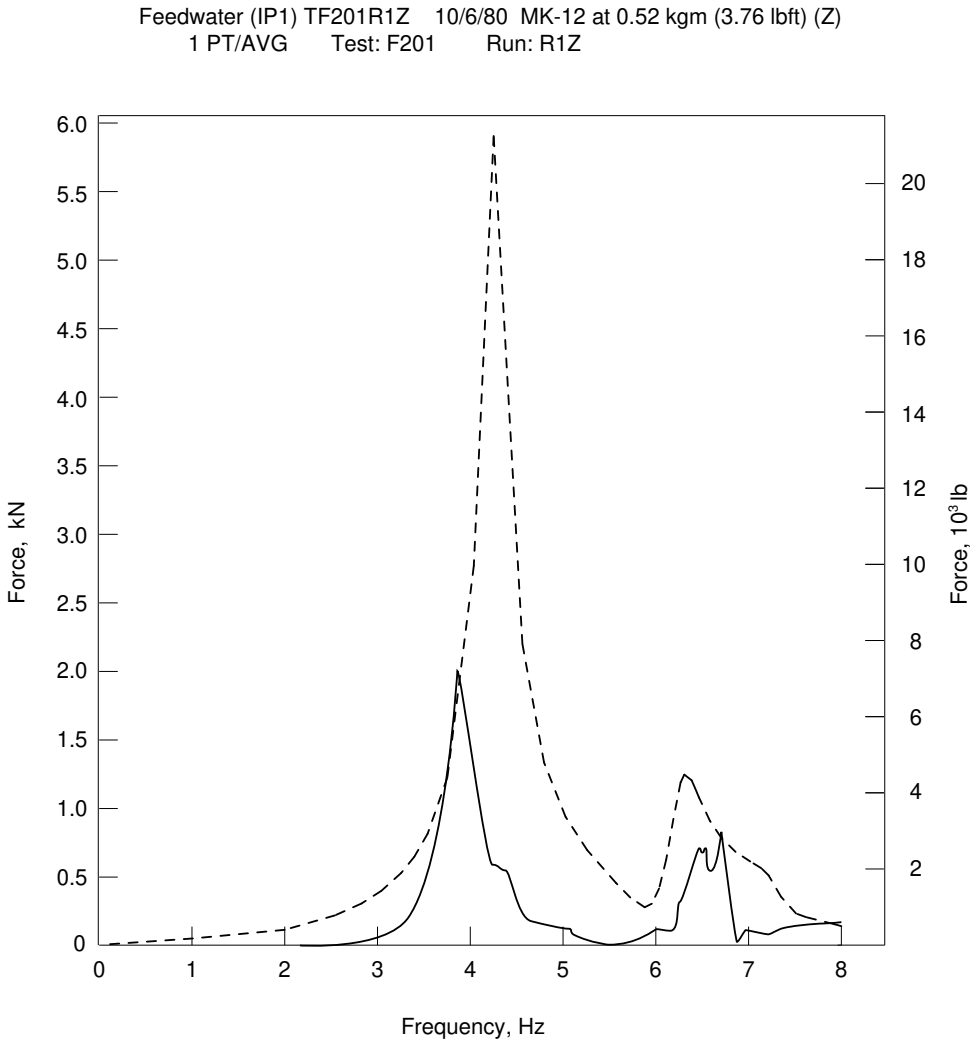


Figure 9: Comparison of force in spring support FW–R–21 between experimental steady-state results (solid line) and Abaqus (dashed line).

Response spectra of a three-dimensional frame building

Products: Abaqus/Standard

Overview

This example verifies the different summation methods for natural modes in the response spectrum procedure. To compare the five different methods that are available in Abaqus, a three-dimensional model with closely spaced modes is examined.

Geometry and model

A four-story steel-frame building based on the frame described in the referenced paper (*Wilson and Kiureghian, 1981*) is analyzed. By closely matching the geometric dimensions, we can reasonably compare results.

All columns in the building have the same geometric properties. However, as shown in *Figure 1*, the properties of the beams in Frames 1 and 2 are different, as compared to those in Frames 3 and 4, to move the center of mass of the structure away from its geometric center. Eigenvalue extraction performed on the model shows that many of the 30 modes that cover the frequency range up to 40 Hz are closely spaced. An acceleration spectrum based on the El Centro earthquake record is applied in the x–y plane. The Fortran program given in *frameresponsespect_acc.f* is used to generate the spectrum. The frequency range is chosen between 0.1 Hz and 40 Hz, and the number of points at which the spectrum is calculated is set at 501. Only one spectrum curve is requested for 2% damping.

Results and discussion

As described in *Linear analysis of a rod under dynamic loading*, for structures with well-separated modes the TENP and the CQC methods reduce to the SRSS method, while the NRL and the ABS methods give similar results. Hence, for such structures, two summation rules would suffice, with ABS providing the more conservative results. However, when structures with closely spaced modes are analyzed, all five summation rules can yield very different results. This is even more apparent in three-dimensional problems. In the present example, the plane of the earthquake motion lies along the x-axis, so we expect that the structural response will be dominated by Frames 1 and 3 and will result in a significant base shear in the x-direction. All five methods are compared against a modal time history response using the same El Centro acceleration record in *Table 1*, where the base shear forces are summed up in the plane of each frame S_i, where i is the frame number. This comparison shows that the best approximation is generated by the CQC method. The other methods overestimate the shear in the y-direction, and some of them underestimate the base shear in the x-direction. The CQC method is generally recommended for asymmetrical three-dimensional problems with closely spaced structural modes. This method takes into account the sign of the mode shapes through cross-modal correlation factors and can correctly predict the response in directions perpendicular to the direction of excitation.

Input files

frameresponsespect_freq.inp	**FREQUENCY* analysis.
frameresponsespect_rs.inp	**RESPONSE SPECTRUM* analysis.
frameresponsespect_modal.inp	**MODAL DYNAMIC* analysis. To run this file, the user must fetch the file cantilever_quakedata.inp and copy it to `QUAKE.AMP`.

frameresponsespect_acc.f	Fortran program that will produce the acceleration spectrum needed to run frameresponsespect_rs.inp. To run this program, the user must fetch the file cantilever_quakedata.inp and copy it to `QUAKE.AMP`.

References

- Wilson, E. L., and A. Der Kiureghian, "A Replacement for the SRSS Method in Seismic Analysis," *Earthquake Engineering and Structural Dynamics*, vol. 9, pp. 187–194, 1981.

Tables

Table 1: Comparison of base shear forces for different summation methods.

Method	S_1 (kip)	S_2 (kip)	S_3 (kip)	S_4 (kip)
Time history	–25.5	14.0	–37.0	–22.8
ABS/ALG	48.3	48.3	65.1	65.1
SRSS/ALG	18.4	18.4	24.8	24.8
TENP/ALG	28.9	28.9	35.9	35.9
NRL/ALG	25.9	25.9	34.6	34.6
CQC/ALG	23.3	13.0	29.0	20.7

Figures

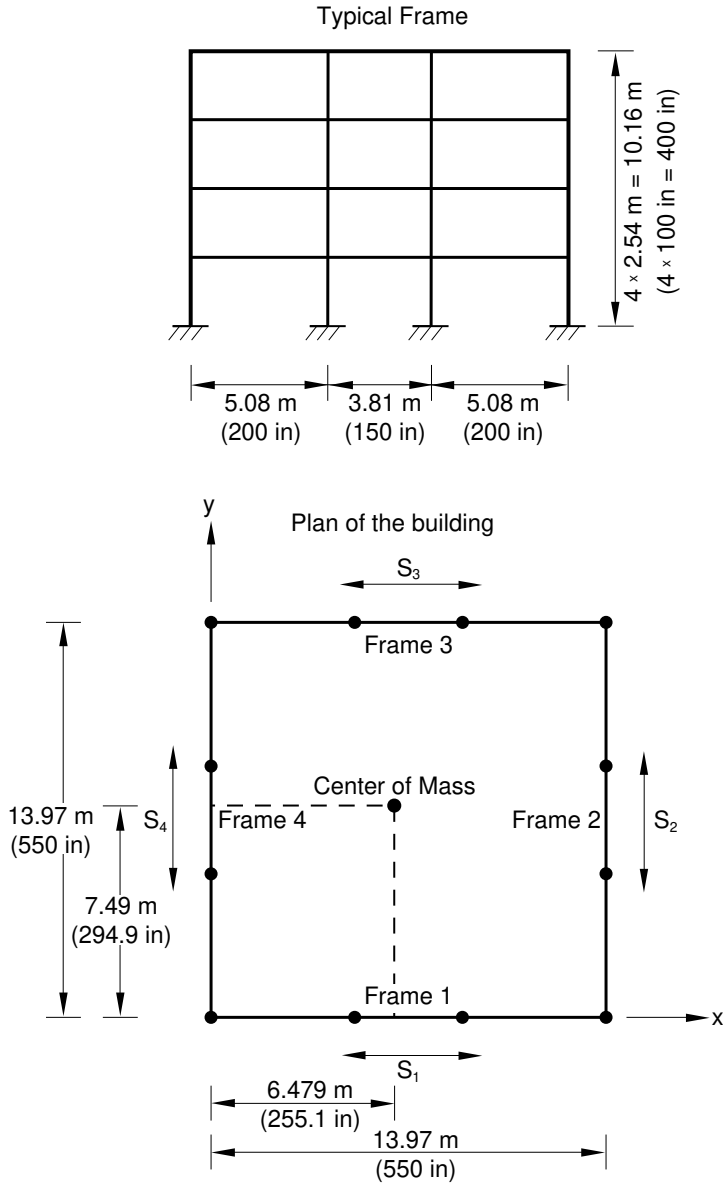


Figure 1: Three-dimensional frame system.

Brake squeal analysis

Products: Abaqus/Standard

References:

- *Complex Eigenvalue Extraction*

Overview

This example illustrates the use of the complex eigenvalue extraction procedure in a brake squeal analysis.

Disc brakes operate by pressing a set of brake pads against a rotating disc. The friction between the pads and the disc causes deceleration, but it may also induce a dynamic instability of the system, known as brake squeal. One possible explanation for the brake squeal phenomenon is the coupling of two neighboring modes. Two modes, which are close to each other in the frequency range and have similar characteristics, may merge as the friction contribution increases. When these modes merge at the same frequency (become coupled), one of them becomes unstable. The unstable mode can be identified during complex eigenvalue extraction because the real part of the eigenvalue corresponding to an unstable mode is positive. The brake system design can be stabilized by changing the geometry or material properties of the brake components to decouple the modes.

The purpose of this analysis is to identify the unstable modes (if they exist) in a particular disc brake system.

Problem description and model definition

The brake model used in this example is a simplified version of a disc brake system used in domestic passenger vehicles. The simplified model consists of a rotor and two pads positioned on both sides of the rotor. The pads are made of an organic friction material, which is modeled as an anisotropic elastic material. The rotor has a diameter of 288 mm and a thickness of 20 mm and is made of cast iron. The back plates and insulators are positioned behind the pads and are made of steel. In this problem material damping is ignored. The mesh (shown in *Figure 1*) is generated using C3D6 and C3D8I elements. Contact is defined between both sides of the rotor and the pads using the small-sliding contact formulation. Initially, the friction coefficient is set to zero.

Contact between the rotor and the pads is established initially in the first step by applying pressure to the external surfaces of the insulators. In the next step a rotational velocity of ω=5 rad/s is imposed on the rotor using the prescribed rotational motion. The imposed velocity corresponds to braking at low velocity. The friction coefficient, μ, is also increased to 0.3 using a change to friction properties. In general, the friction coefficient can depend on the slip rate, contact pressure, and temperature. If the friction coefficient depends on the slip rate, the velocity imposed by the motion predefined field is used to determine the corresponding value of μ. The friction coefficient is ramped from zero up to the desired value to avoid the discontinuities and convergence problems that may arise because of the change in friction coefficient that typically occurs when bodies in contact are moving with respect to each other. This issue is described in detail in *Static Stress Analysis*. The unsymmetric solver is used in this step. At the end of the step a steady-state braking condition is obtained.

In the next step the eigenvalue extraction procedure is performed in this steady-state condition. Because the complex eigensolver uses the subspace projection technique, the real eigenvectors are extracted first to define the projection subspace. In the eigenvalue extraction procedure the tangential degrees of freedom are not constrained at the contact nodes at which a velocity differential is defined. One hundred real eigenmodes are extracted and, by default, all of them are used to define the projection subspace. The subspace can be reduced by selecting the eigenmodes to be used in the complex eigenvalue extraction step. The complex eigenvalue analysis is performed up to 10 kHz (the first 55 modes).

Results and discussion

Figure 2 shows the damping ratio as a function of frequency. The damping ratio is defined as

$-2 * Re(\lambda) / |Im(\lambda)|$, where λ is a complex eigenvalue. A negative value of the damping ratio indicates an unstable mode. In the range of interest the complex eigensolver found an unstable mode at the frequency 2.0 kHz with a damping ratio of −0.0138. *Figure 3* presents the unstable mode as the combined magnitude of both the real and imaginary components of the complex eigenvector.

In the second analysis the value of the friction coefficient between the rotor and the pads is increased to 0.5. Because the real eigenvectors do not differ significantly between the problems with different friction coefficients, the eigenspace determined in the first analysis can be reused as the projection subspace for the second complex eigenvalue extraction procedure. The analysis is restarted after the frequency extraction step of the preceding run, and two new procedure steps are defined. In the first nonlinear static step the friction coefficient is increased by changing friction properties, followed by the complex eigenvalue extraction in the second step. The results are presented in the form of a damping ratio plot, shown in *Figure 4*. In this case four unstable modes at 2.0, 3.0, 8.0, and 9.0 kHz are found. As the friction coefficient is increased, more neighboring modes couple and become unstable.

Acknowledgments

SIMULIA would like to thank Dr. Li Jun Zeng of TRW Automotive for supplying the disc brake model used in this example.

Input files

brake_squeal.inp	The initial brake squeal analysis with a friction coefficient μ=0.3.
brake_squeal_node.inp	Nodal coordinates for the brake model.
brake_squeal_elem.inp	Element definitions for the brake model.
brake_squeal_res.inp	The restarted brake squeal analysis with a friction coefficient μ=0.5.

Figures

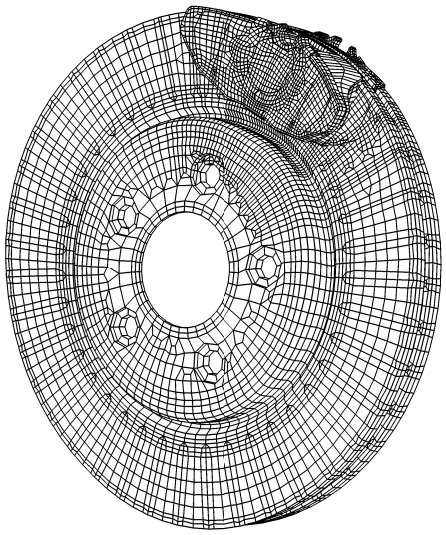


Figure 1: Geometry and mesh of the disc brake system.

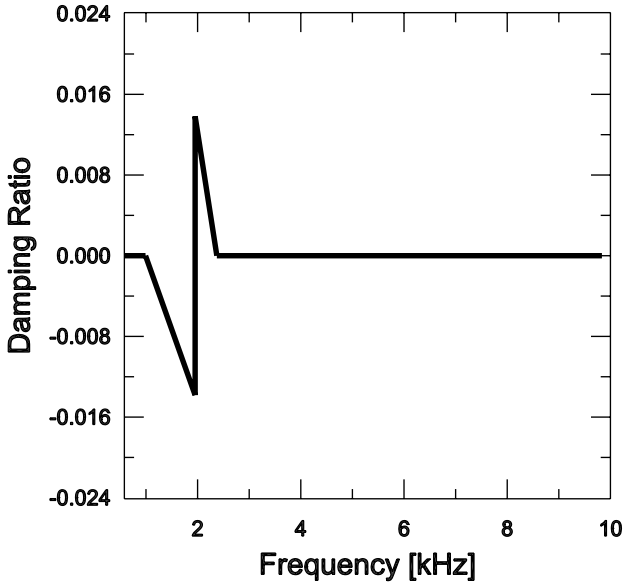


Figure 2: Damping ratios for the analysis with μ=0.3.

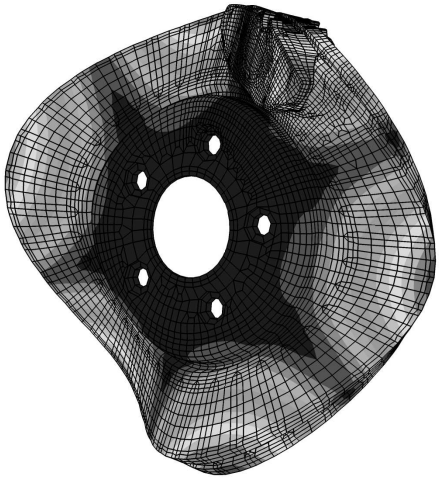


Figure 3: The unstable mode at 2.0 kHz in the analysis with μ=0.3.

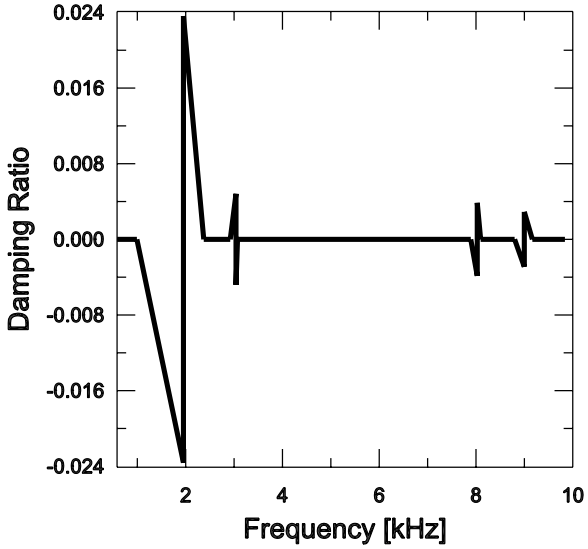


Figure 4: Damping ratios for the analysis with μ=0.5.

Dynamic analysis of antenna structure utilizing residual modes

Products: Abaqus/Standard

Overview

This example illustrates the use of residual modes in a modal dynamic analysis.

The residual modes capability is a cost-effective approach to correct inaccuracies due to modal truncation that are often prevalent in modal dynamic analyses. For performance reasons, usually only a relatively small subset of the total possible eigenmodes is extracted for the model. This set of eigenmodes typically is chosen to cover adequately the frequency content of the applied loads in the modal dynamic analysis. However, this criterion alone does not guarantee that the system will be represented adequately by this set of eigenmodes; consequently, inaccuracies can occur in the modal dynamic solution. Fortunately, adding a few residual modes to the set of eigenmodes can significantly improve the solution at a relatively low cost compared to extracting additional eigenmodes arbitrarily.

Geometry and problem description

The model used for this study is a relatively simple antenna structure that has shell elements for the dish with beam elements for the support, as shown in *Figure 1*. The objective is to calculate the total force in the main support beam due to a force dynamically applied to the antenna. The bottom of the antenna is completely fixed to the ground. The force is applied in the vertical direction as a step function of magnitude 1.78 lbf, as shown in *Figure 2*.

Models

Four models are created to demonstrate modal truncation inaccuracies and comparisons of the results using residual modes with a relatively small number of eigenmodes to the results using no residual modes but many eigenmodes.

The finite element discretization and excitation environment are the same in all models. The only difference between the models is the number of eigenmodes extracted and subsequently used in the modal dynamic analysis. More specifically, the first three models use 10, 20, and 50 eigenmodes, respectively; and the fourth model uses 10 eigenmodes in addition to a single residual mode.

Procedure

Following the usual procedure for modal dynamics, the eigenmodes are first extracted in a frequency step. This step is followed by a modal dynamics step in which the forcing excitation is applied.

To activate the residual mode capability, a static perturbation step must precede the frequency extraction step. The same loading pattern that is given in the modal dynamics step must be applied in the static perturbation step. In addition, residual modes must be extracted in the frequency step. The modal dynamics step remains the same as in the case without residual modes.

Results and discussion

Since the magnitude of the applied load is 1.78 lbf, the static solution for the total axial force in the main support beam is also 1.78 lbf. When the modal dynamic analysis is performed, modal truncation inaccuracies become evident. *Figure 3* shows the force results using 10, 20, and 50 eigenmodes. For the 10-eigenmode case, the peak

force in the support beam is only 0.4 lbf. If 20 eigenmodes are included, the peak force jumps to 1.7 lbf. If 50 eigenmodes are used, the peak force is 2.5 lbf and the correct static response of 1.78 lbf is recovered.

The answers vary because of modal truncation. The excitation history is a step function and, thus, includes the full span of frequency content. Similarly, since the excitation is applied at a single point, it can be accurately represented in the modal analysis only by using a large span of eigenmodes. Since a very large number of eigenmodes are potentially excited due to the temporal and spatial characteristics of the excitation, it is necessary to extract and use many eigenmodes. It is possible to determine how many eigenmodes are needed by looking at the total effective mass of the extracted modes or by performing a convergence study by systematically increasing the number of eigenmodes used in the modal dynamics analysis, similar to the study conducted above.

A much less time-consuming approach (in terms of human intervention and analysis cost) for achieving more accurate results is to use the residual modes capability. Due to the nature of the loading in this analysis, the residual mode capability is particularly useful in capturing the mode shape that dominates the static response. *Figure 4* shows that a single residual mode combined with only 10 eigenmodes yields the correct solution. The residual modes approach (with a total of only 11 modes) produces essentially the same accuracy as using 50 pure eigenmodes.

Input files

antenna_10.inp	Modal dynamic analysis of antenna structure using 10 system eigenmodes.
antenna_20.inp	Modal dynamic analysis of antenna structure using 20 system eigenmodes.
antenna_50.inp	Modal dynamic analysis of antenna structure using 50 system eigenmodes.
antenna_10_resvec.inp	Modal dynamic analysis of antenna structure using 10 eigenmodes plus residual modes.

Figures

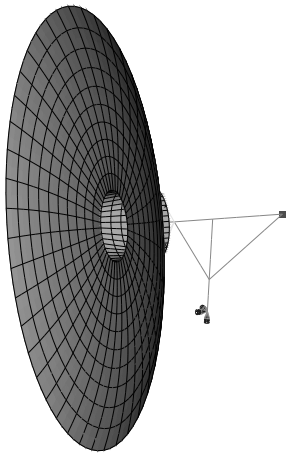


Figure 1: Antenna model.

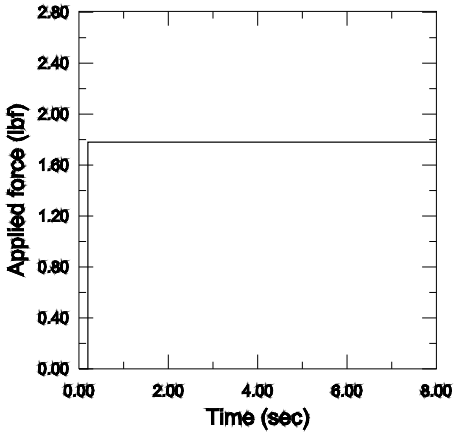


Figure 2: Excitation history.

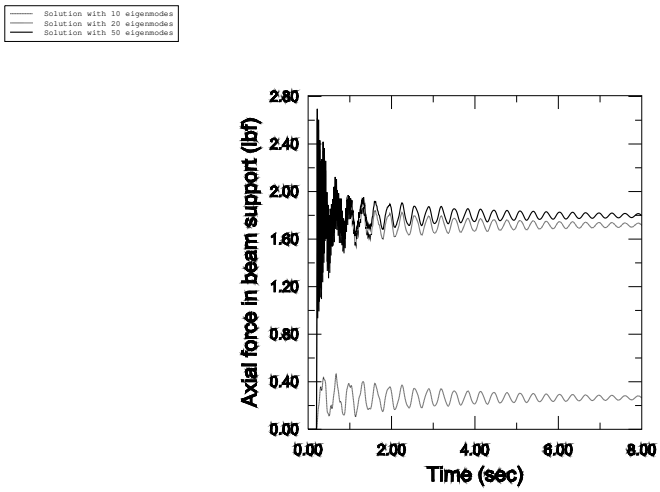


Figure 3: Axial force in main support beam due to dynamic excitation using 10, 20, and 50 eigenmodes.

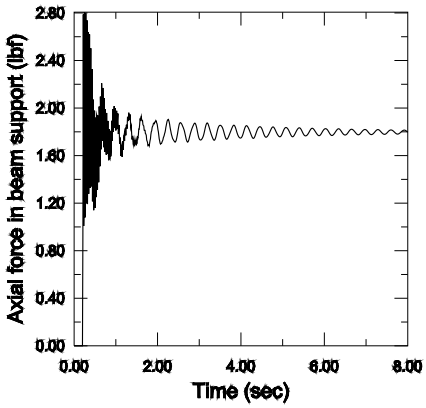


Figure 4: Axial force in main support beam due to dynamic excitation using 10 eigenmodes and residual modes.

Steady-state dynamic analysis of a vehicle body-in-white model

Products: Abaqus/Standard

References:

- *Using the SIM Architecture for Modal Superposition Dynamic Analyses*
- *Direct-Solution Steady-State Dynamic Analysis*
- *Natural Frequency Extraction*
- *Mode-Based Steady-State Dynamic Analysis*
- *Subspace-Based Steady-State Dynamic Analysis*
- **FREQUENCY*
- **STEADY STATE DYNAMICS*
- *Steady-state dynamics with nondiagonal damping using the AMS eigensolver*

Overview

This example examines the structural behavior of a vehicle body-in-white (BIW) model in terms of eigenmodes and frequency response functions.

The following Abaqus features for frequency extraction and steady-state dynamic analysis are demonstrated:

- using the automatic multilevel substructuring (AMS) eigensolver in the frequency extraction step along with residual modes;
- projecting a global material structural damping operator during the frequency extraction step using the AMS eigensolver;
- using the SIM-based steady-state dynamic analysis procedure with material structural damping; and
- demonstrating the performance benefit of the SIM-based steady-state dynamic analysis procedure using the AMS eigensolver compared to the subspace-based steady-state dynamic analysis procedure with the Lanczos eigensolver.

Application description

The model shown in *Figure 1* was obtained from the National Highway Traffic Safety Administration website (*https://www.nhtsa.gov/*). This model includes material structural damping.

The primary goal of this example is to demonstrate the significant performance improvement of the SIM-based steady-state dynamic analysis procedure using the AMS eigensolver compared to the subspace-based steady-state dynamic analysis procedure. Prior to performing the steady-state dynamic analysis, the undamped eigensolution is computed using the AMS eigensolver. The global cutoff frequency of this model is 300 Hz; therefore, the global eigenmodes below 300 Hz are extracted. During the reduction phase of the AMS eigensolver, all the substructure eigenmodes below 1500 Hz are extracted and used to calculate the global eigensolution with the

AMS_{cutoff_1} default value of 5. In addition, the material structural damping operator is projected onto the global eigenmode subspace.

A second model identical to the one shown in *Figure 1* demonstrates the use of frequency-dependent structural damping.

Geometry

The model consists of the bare metal shell of the frame body including fixed windshields. This model has 127,213 elements and 794,292 active degrees of freedom. A total of 1107 fasteners are defined to model spot welds in the frame body.

Materials

Linear elastic materials are used for all shell elements, and material structural damping with a value of 0.01 is applied to the materials for the ceiling, floor, hood, and side body of the vehicle. This type of damping results in a full system of modal equations that must be solved for each frequency point.

In a second model, the constant value of structural damping used in the first model is replaced with frequency-dependent structural damping (see *Table 1*), using the definition in the table. The frequency-dependence of the structural damping in the second model is such that $s = 0.01$ for frequencies lower than 50 Hz and $s = 0.1$ for frequencies higher than 100 Hz, with a linear transition between the two regions.

Table 1: Frequency-dependent structural damping coefficients.

s	f (Hz)
0.01	1
0.01	50
0.1	100
0.1	150

Boundary conditions and loading

This structure is not constrained; therefore, there are six rigid body modes in the model. Two concentrated loads are applied to the nodes at the two pivot points on the bottom of the vehicle floor to simulate the rolling motion of the vehicle in the steady-state dynamic analysis.

Abaqus modeling approaches and simulation techniques

The objective of this analysis is an understanding of the overall structural response of a body-in-white model due to rolling motion. The response is evaluated by studying the frequency responses at both a drive point node and a transfer point node.

Compared to the subspace-based steady-state dynamic analysis procedure with the Lanczos eigensolver, the SIM-based steady-state dynamic analysis procedure using the AMS eigensolver shows significant performance benefit with acceptable accuracy. The accuracy of the SIM-based steady-state dynamic solution is evaluated by comparing the frequency response functions for the SIM-based and subspace-based steady-state dynamic analyses.

Drive point and transfer point responses are calculated at nodes 840950 and 874962, respectively, to demonstrate the accuracy of the SIM-based steady-state dynamic analysis procedure with eigenmodes computed by the AMS eigensolver. This model has 1210 global eigenmodes below the global cutoff frequency, including 6 rigid body modes. The AMS eigensolver approximates 1188 global eigenmodes below the global cutoff frequency of 300 Hz, and the frequency response solutions below 150 Hz are calculated at 1 Hz increments. Two residual modes are also added to demonstrate a means of improving efficiency by using fewer global modes.

Summary of analysis cases

Case 1	SIM-based steady-state dynamic analysis using AMS.
Case 2	Subspace-based steady-state dynamic analysis using Lanczos.

Case 3	Direct-solution steady-state dynamic analysis.

Analysis types

The frequency extraction step is followed by a steady-state dynamic analysis step. The global cutoff frequency for the frequency extraction step is 300 Hz, and the frequency range considered is 1–150 Hz at 1 Hz increments. For the subspace-based steady-state dynamic analysis, 1000 eigenmodes below 300 Hz are used. For the SIM-based steady-state dynamic analysis, all the eigenmodes below 300 Hz (approximately 1200 eigenmodes) are used.

Discussion of results and comparison of cases

The accuracy of the natural frequencies computed by the AMS eigensolver is evaluated assuming that the natural frequencies computed by the Lanczos eigensolver are exact, and relative errors are shown in *Figure 2*. These errors are less than 1.6% at frequencies below the global cutoff frequency of 300 Hz. Below 150 Hz, the region of particular interest, the errors are less than 0.25% and the resonance peaks in the frequency response function are reasonably accurate.

The accuracy of the steady-state dynamics solutions is evaluated qualitatively by comparing the frequency response curves. The response curves are compared at two different output locations: a drive point node and a transfer point node.

Figure 3 shows the point inertance at node 840950:

Curve	Procedure
`SSD DIRECT`	Direct-solution steady-state dynamic analysis
`SSD SP`	Subspace-based steady-state dynamic analysis
`SSD AMS`	SIM-based steady-state dynamic analysis
`SSD AMS RES`	SIM-based steady-state dynamic analysis with residual modes

The curves are almost indistinguishable for frequencies below 125 Hz. To improve the accuracy of the response functions, residual modes can be added to compensate for high-frequency truncation errors. Calculation of residual modes is requested in the AMS eigenvalue extraction step by specifying the nodal points where forces are applied. For this solution (`SSD AMS RES`), the global cutoff frequency is set to 225 Hz instead of 300 Hz to demonstrate that the global cutoff frequency can be lowered without losing accuracy. By lowering the global cutoff frequency to 225 Hz and adding two residual modes in the global eigenmode subspace, a speed up of approximately 25% can be achieved.

Figure 4 shows the transfer inertance at the node located at the bottom of the driver's seat; the same curve labels discussed above are used in this figure. The frequency response computed by the SIM-based steady-state dynamic analysis procedure nearly matches those computed by the subspace-based steady-state dynamic analysis procedure and the direct-solution steady-state dynamic analysis procedure for the frequencies below 125 Hz. To improve accuracy for frequencies higher than 125 Hz, more eigenmodes should be extracted and used.

The overall performance comparison among the three steady-state dynamic analysis procedures is summarized in *Table 2* (analysis cost is independent of whether or not residual modes are used). Performance was evaluated on a Linux/x86-64 server with a single processor. The amount of available memory was set to 16 GB for each run. It is clear that the SIM-based steady-state dynamic analysis procedure using the AMS eigensolver shows excellent performance compared to the subspace-based steady-state dynamic analysis procedure using the Lanczos eigensolver or the direct-solution steady-state dynamic analysis procedure.

Figure 5 shows the transfer inertance at the node located at the bottom of the driver's seat for the model with frequency-dependent structural damping. As expected from the frequency-dependent structural damping definition in *Table 1*, the transfer inertance is identical to that for the constant $s = 0.01$ structural damping shown in *Figure*

4 for frequencies lower than 50 Hz and to that for the constant $s = 0.1$ structural damping for frequencies higher than 100 Hz.

Input files

biw_modeldata.inp	Input file for the BIW model data.
biw_freq_ams.inp	Frequency extraction analysis using the AMS eigensolver for the BIW model.
biw_ssd_ams.inp	SIM-based steady-state dynamic analysis using the AMS eigensolution for the BIW model.
biw_freq_ams_res.inp	Frequency extraction analysis (including residual modes) using the AMS eigensolver for the BIW model.
biw_ssd_ams_res.inp	SIM-based steady-state dynamic analysis using the AMS eigensolution including residual modes for the BIW model.
biw_freq_lnz.inp	Frequency extraction analysis using the Lanczos eigensolver for the BIW model.
biw_ssd_lnz.inp	Subspace-based steady-state dynamic analysis using the Lanczos eigensolution for the BIW model.
biw_ssd_dir.inp	Direct-solution steady-state dynamic analysis for the BIW model.
biw_modeldata_freq_dep_sdamp.inp	Input file for the BIW model data with frequency-dependent structural damping.
biw_ssd_dir_freq_dep_sdamp.inp	Direct-solution steady-state dynamic analysis for the BIW model with frequency-dependent structural damping.

Tables

Table 2: Overall performance comparison of the steady-state dynamic analysis procedures (hh:mm:ss).

Performance	SSD with AMS	SSD SP with Lanczos	SSD DIRECT
Total wall clock time	00:06:31	03:20:45	05:58:54
Wall clock time for eigensolver	00:05:35	01:36:24	N/A
Wall clock time for SSD	00:00:56	01:44:21	05:58:54

Figures

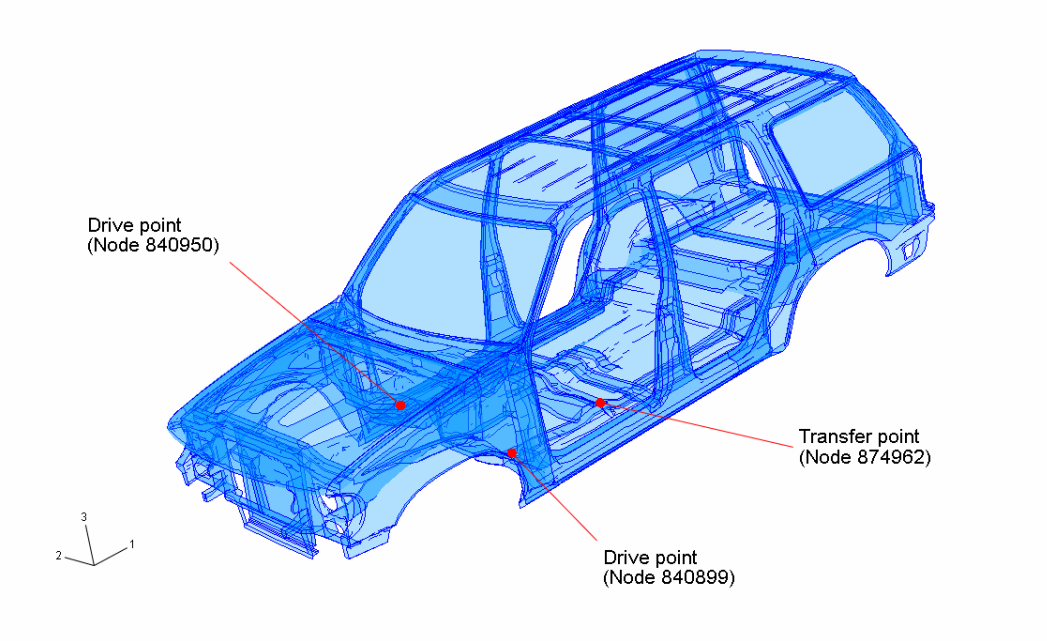


Figure 1: Body-in-white model of sport utility vehicle.

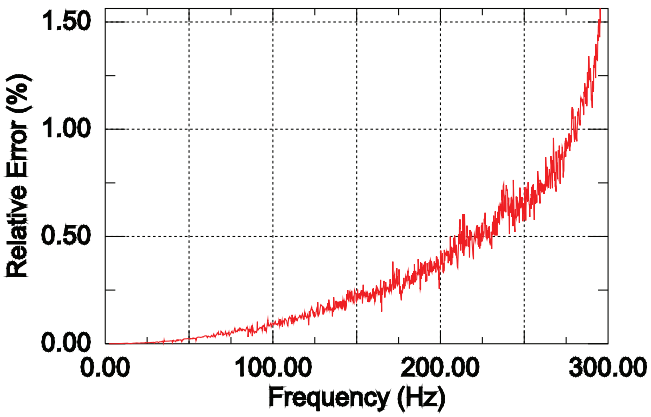


Figure 2: The relative errors of the natural frequencies computed by AMS with respect to the natural frequencies computed by Lanczos below the global cutoff frequency of 300 Hz.

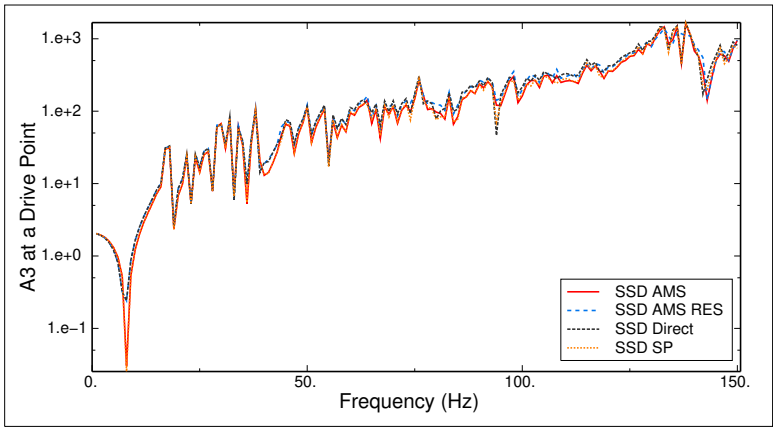


Figure 3: Point inertance at node 840950.

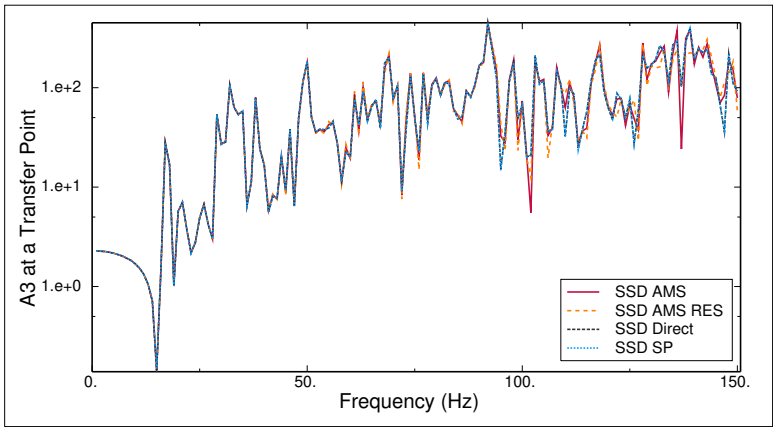


Figure 4: Transfer inertance at node 874962.

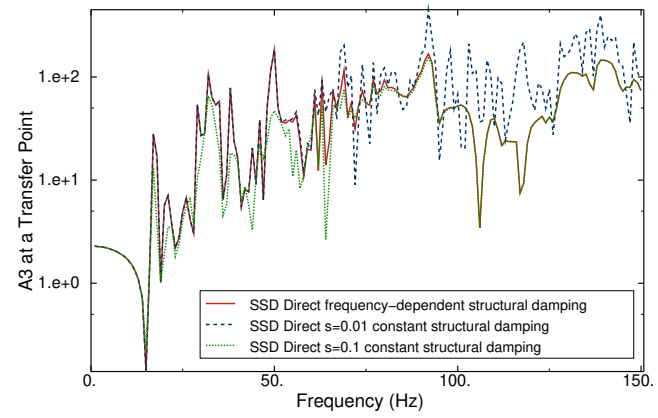


Figure 5: Transfer inertance at node 874962 for the model with frequency-dependent structural damping.

Eulerian and co-simulation analyses

In this section:

- *Rivet forming*
- *Impact of a water-filled bottle*
- *Dynamic impact of a scooter with a bump*

Rivet forming

Products: Abaqus/Explicit Abaqus/CAE

References:

- *Eulerian Analysis*
- *Eulerian Elements*
- *Eulerian analyses*
- **EULERIAN SECTION*
- **INITIAL CONDITIONS*

Overview

This example investigates the effectiveness of a particular rivet by simulating its compression (also referred to as the forming process).

The following Abaqus features are demonstrated:

- using the coupled Eulerian-Lagrangian (CEL) analysis formulation to analyze a solid mechanics model undergoing extreme deformation, and
- comparing results from the CEL-based analysis to those from an identical model using the traditional Lagrangian formulation.

Application description

Rivets are a type of fastener designed to create permanent attachments between two or more sheets of material. A rivet design typically consists of a cylindrical body with two diameters: the smaller diameter is inserted through a hole in the overlapping sheets, then both ends of the rivet are compressed. The compression effectively expands the diameters of the rivet body, pinching the sheets of material between the two ends of the rivet (see *Figure 1*). Different rivet designs and applications will undergo different deformations, but the basic principle remains the same in all cases.

Three questions are of particular importance in this study:

- Does the rivet deform appropriately during the forming process?
- After the forming process, does the rivet retain enough strength to maintain a hold on the fastened materials?
- Is the rivet installation tool capable of forming the rivet?

The displacements during the forming simulation indicate whether or not the rivet deforms appropriately. After the deformation, the strength of the rivet is based largely on its material properties; examining the equivalent plastic strain in the rivet gives some indication of potential damage or strength degradation in the material. To assess the effect of the rivet on the installation tool, reaction forces in the tool can be compared to known force capacities in standard installation tools.

Geometry

The rivet used in these analyses is a simple multidiameter cylinder, as described above. To aid the deformation of the smaller end of the rivet, a hemispherical portion is removed from the center of the cylinder. *Figure 2* shows the dimensions of the rivet model.

To simulate the forming, the rivet is placed in a hole at the center of a circular plate. Circular dies representing the installation tool are positioned at the top and bottom end of the rivet (see *Figure 3*).

Materials

The rivet in this model is composed of an elastic-plastic steel with a density of 7.85×10^{-9} t/mm^3, Young's modulus of 2.1×10^5 N/mm^2, Poisson's ratio of 0.266, and the onset of plastic yielding occurring at 3.0×10^5 N/mm^2.

The plate and dies are assumed to be significantly harder than the rivet, and no deformation of these parts is expected.

Boundary conditions and loading

The forming process is simulated through the enforcement of displacement boundary conditions. The plate is constrained to a fixed location. The top die is displaced downward a distance of 3 mm, while at the same time the bottom die is displaced upward a distance of 2 mm.

Interactions

Contact interactions must be enforced between the rivet and all installation tool components; the deformation of the rivet depends on contact loads delivered through the displacement of the tool. The tool components never come into contact with each other, so interactions between the plate and dies can be neglected.

Abaqus modeling approaches and simulation techniques

The rivet forming simulation is conducted in Abaqus/Explicit using two fundamentally different element formulations. The traditional Lagrangian formulation generally offers accuracy and computational efficiency, but pure Lagrangian models tend to exhibit mesh distortion and an associated loss of accuracy when undergoing extreme deformations. The Eulerian formulation trades some accuracy of geometry and results for robustness in analyses involving very large deformations; in situations where the Lagrangian formulation yields an unreliable solution or no solution at all, the Eulerian formulation can be used to obtain a reasonable solution.

Lagrangian and Eulerian elements can be combined in the same model using a technique known as coupled Eulerian-Lagrangian (CEL) analysis. In a CEL analysis bodies that undergo large deformations are meshed with Eulerian elements, while stiffer bodies in the model are meshed with more efficient Lagrangian elements.

The rivet forming analysis is performed using both a pure Lagrangian approach, in which the rivet, plate, and dies are all modeled with Lagrangian elements; and a coupled Eulerian-Lagrangian approach, in which the rivet is modeled with Eulerian elements while the plate and dies are modeled with Lagrangian elements.

Summary of analysis cases

Case 1	Pure Lagrangian rivet forming analysis.
Case 2	Coupled Eulerian-Lagrangian (CEL) rivet forming analysis.

The following sections detail some modeling techniques that are common to both analysis cases.

Analysis Types

Both analysis cases are conducted using quasi-static explicit dynamic procedures. The forming takes place over the course of a single step lasting 1 ms.

Materials

The material for the rivet uses an isotropic hardening Mises plasticity model. The stress-strain data points used to define the plastic behavior are shown in *Table 1*.

Boundary conditions

In both analysis cases the plate and dies are modeled as Lagrangian bodies imposed with rigid body constraints. A boundary condition preventing displacements and rotations is imposed on the reference point for the plate body. Boundary conditions are also applied to each die reference point to prevent them from rotating or displacing, except in the vertical 3-direction: the boundary condition on the top reference point displaces it 3 mm in the negative 3-direction, and the boundary condition on the bottom reference point displaces it 2 mm in the positive 3-direction. The application of the boundary conditions is governed by an amplitude that ramps the displacement linearly from zero to full displacement over the course of 0.8 ms; the dies are then fixed in place for the final 0.2 ms of the analysis.

Constraints

As mentioned above, rigid body constraints are applied to the plate and two dies. These components are assumed to be significantly harder than the rivet, and they do not deform during the forming process. The rigid body constraints improve computational efficiency and allow the use of simple boundary conditions to initiate the forming.

Output requests

Field output is specifically requested for the equivalent plastic strains in the model (PEEQ). In addition, history output for the reaction force in the 3-direction (RF3) is requested at the reference point for each of the dies.

Pure Lagrangian analysis case

The first analysis case uses four Lagrangian bodies meshed from discrete geometric part instances. In the pure Lagrangian case the geometry of the model corresponds directly to the shapes of the parts being modeled, making the assembly process fairly intuitive.

Mesh design

The rivet is meshed with C3D8R elements using a global mesh seed of 0.25 mm (see *Figure 4*).

The plate and dies are also meshed with C3D8R elements, but the rigid body constraints applied to these parts makes the element selection somewhat arbitrary. Unmeshed analytical rigid surfaces could have been used to model the plate and dies, but rigid body constraints are used to maintain consistency with the CEL model.

Interactions

A general contact definition enforces contact interactions between all bodies in the model. A frictionless, hard contact model governs all interactions.

Solution controls

Although large deformations are expected in the analysis, no special solution controls or analysis techniques (such as adaptive meshing) are applied to the model, allowing a straightforward comparison between the pure Lagrangian model and the CEL model.

CEL analysis case

In the second analysis case the rivet is modeled using Eulerian elements. The plate and dies are still rigid bodies. The modeling approach in the CEL analysis has some distinct differences from the pure Lagrangian case.

Mesh design

In the Eulerian formulation the mesh does not generally correspond to the geometry of the part being modeled; rather, the placement of the material within the Eulerian mesh defines the geometry of the part. The Eulerian mesh does not deform or displace; only the materials within the mesh can move. Typically, the Eulerian mesh is an arbitrary collection of regular hexahedral elements that fully encompasses the region in which material might exist during the analysis.

In this example the Eulerian part is a rectangular prism measuring 17 × 17 × 11.5 mm meshed with EC3D8R elements. A global mesh seed of 0.25 mm dictates the element size.

This mesh does not define the geometry of the rivet; rather, the mesh defines the domain in which the rivet material can exist. The rivet geometry is defined by assigning steel material to a portion of this mesh corresponding to the shape of the rivet, as discussed in the *Initial conditions* section below. One strength of the Eulerian technique is the ability to define a regular, high-quality mesh independent of the geometry of the part being modeled.

It is important that the Eulerian mesh is large enough to contain the rivet material completely as it deforms; if material reaches the edge of the mesh, it flows out of the model and is lost to the simulation.

Initial conditions

The rivet geometry is defined using a material assignment initial condition on the Eulerian mesh. The material assignment specifies which elements in the mesh initially contain steel. Each element is designated a percentage (known as the volume fraction), which represents the portion of that element filled with steel. For partially filled elements Abaqus positions the material in the element such that it forms a continuous surface with the material in adjacent elements. The end result is a distribution of material in the mesh corresponding to the rivet geometry, as seen in *Figure 5*. You can use the view cut manager in the Visualization module of Abaqus/CAE to visualize the extent of a material within an Eulerian mesh, as discussed in *Viewing output from Eulerian analyses*.

The material assignment is created with the aid of the volume fraction tool in Abaqus/CAE. The volume fraction tool calculates the overlap between an Eulerian mesh and some reference geometric part. To use the volume fraction tool for this analysis case, the entire Lagrangian assembly (including the Lagrangian rivet) is copied from the previous analysis case and positioned inside the Eulerian mesh (see *Figure 6*). The Lagrangian rivet serves as the reference part, and the volume fraction tool creates a discrete field that associates each element in the Eulerian mesh with a volume fraction based on the amount of space occupied by the rivet within that element. This discrete field can then be used as the basis for a material assignment predefined field in Abaqus/CAE.

Interactions

A general contact definition enforces contact interactions between all rigid bodies and Eulerian materials in the model. General contact does not enforce contact between the rigid bodies and the Eulerian elements; the rigid bodies can pass unimpeded through the Eulerian mesh until they encounter a material within the mesh. As with the pure Lagrangian case, a frictionless, hard contact model governs all interactions.

It is generally not advisable to model Lagrangian-Eulerian contact near the boundary of the Eulerian mesh. The inflow or outflow of materials at the mesh boundary can lead to improper contact constraint enforcement. Therefore, the Eulerian mesh extends a few elements past the contact interfaces between the dies and the rivet.

General contact does not enforce interactions between analytical rigid surfaces and Eulerian materials, which is why the tool components must be modeled as Lagrangian parts with rigid body constraints.

Output requests

In addition to the field and history output requests used in the pure Lagrangian analysis case, the Eulerian volume fraction output variable (EVF) is requested as field output to visualize geometric results.

Discussion of results and comparison of cases

Figure 7 shows the deformed meshes for the pure Lagrangian and the CEL analysis cases. (To view the results of the CEL analysis, use the view cut manager as described in *Viewing output from Eulerian analyses*.) The pure Lagrangian analysis runs to completion, but the mesh becomes extremely distorted along the bottom of the rivet—results from such an irregular mesh might be unreliable. The Eulerian analysis exhibits a similar deformed shape but retains a high-quality, regular mesh.

Computational efficiency

In general, an Eulerian analysis is more expensive than a comparable Lagrangian analysis in terms of run times and file sizes. This performance tradeoff should be weighed against the benefits of Eulerian robustness for large deformations when choosing an analysis formulation.

Difficulties with contact

Figure 8 shows the contact interface between the rivet and the plate for both the pure Lagrangian and the CEL cases. Both cases demonstrate some undesirable penetration of the rivet into the plate.

In the pure Lagrangian case the penetrations are a direct result of mesh distortion. As the rivet's facets spread out, there are fewer constraint points per given area, and portions of the facets are able to pass unconstrained into the plate surface. While the Lagrangian formulation is generally adept at simulating contact, a severely deformed mesh can cause noisy, inconsistent contact enforcement.

In the CEL case the penetrations result largely from approximations used to visualize Eulerian material. The boundary of an Eulerian material does not correspond to a discrete element face. As discussed previously, Abaqus determines the location of material in an Eulerian mesh based on a volume fraction within each element; the volume fractions are averaged and interpolated to calculate a smooth material surface during visualization. Therefore, the material boundary displayed in the Visualization module of Abaqus/CAE is an approximation based on numerical averaging, not geometric properties. The approximation causes the apparent penetrations at contact interfaces and explains why sharp corners in a Lagrangian model appear rounded in an Eulerian model. Despite the apparent penetrations, coupled Eulerian-Lagrangian contact does not suffer from the inconsistent constraint enforcement associated with pure Lagrangian mesh distortion, and contact between a solid Lagrangian body and an Eulerian material typically delivers reliable results.

In both analysis cases contact penetrations can be alleviated through the use of a finer mesh: smaller elements lead to reduced mesh distortion in the pure Lagrangian case, and additional elements provide additional sample points for more accurate volume fraction averaging in the Eulerian case.

Interpreting results

A contour plot of the equivalent plastic strains in a cross-section of the deformed rivet appears in *Figure 9* for both analysis cases. The results are similar, but the regions of maximum plastic strain occur in slightly different areas. In the Eulerian rivet the peak strain occurs near the corner where the rivet meets the bottom of the plate; this region undergoes extreme bending and stretching during the forming. In the Lagrangian rivet the peak strain occurs in the elements with the most severe distortion. For small to moderate deformation, the Eulerian approach provides results that are comparable to the traditional Lagrangian approach (though at a higher computational cost); for large deformation the Eulerian results appear more reliable than the Lagrangian results.

Figure 10 plots the reaction forces in both dies over the course of the forming process. The results for the two formulations are comparable during the first half of the analysis. However, the force plots for the pure Lagrangian

analysis case exhibit some noise during the latter half of the forming and subsequently diverge from the Eulerian force plots. The noise is likely the result of contact difficulties in the model (discussed above), which in turn lead to an uneven transfer of force between the tooling and the rivet. After 0.8 ms, when the dies have fully displaced, the forces in the Eulerian case exhibit some relaxation; the forces in the Lagrangian case remain steadily higher than the corresponding Eulerian forces due to the extreme deformation in the mesh.

Python scripts

rivet_forming.py	Script to generate the pure Lagrangian and the CEL models in Abaqus/CAE.

Input files

rivet_forming_lag.inp	Input file for the pure Lagrangian model.
rivet_forming_cel.inp	Input file for the CEL model.

Tables

Table 1: Plasticity data for rivet material.

Yield stress (N/mm^2)	Plastic strain
300	0
450	0.02
750	0.34
140	0.35

Figures

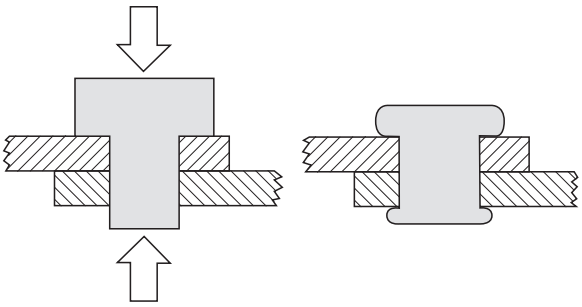


Figure 1: Using a rivet to fasten two sheets of material.

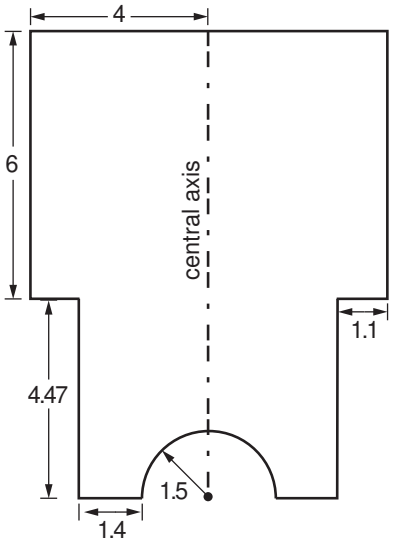


Figure 2: Rivet model geometry. All dimensions are in millimeters.

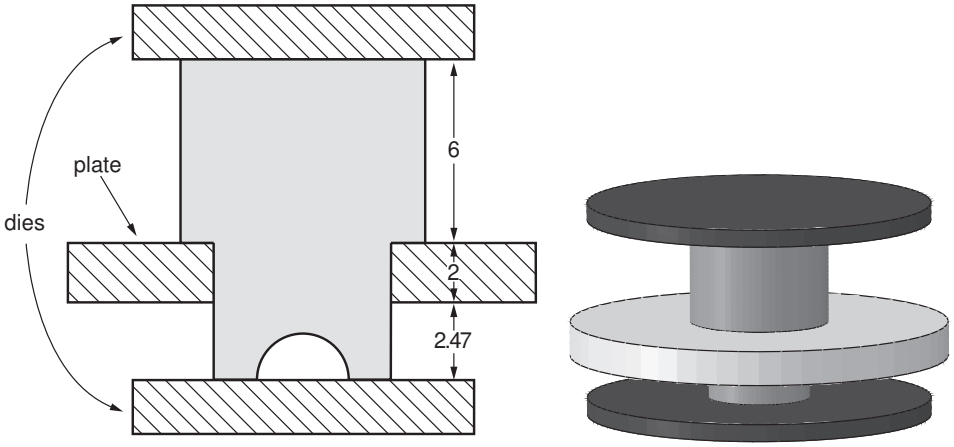


Figure 3: Rivet assembled with the forming tools. All dimensions are in millimeters.

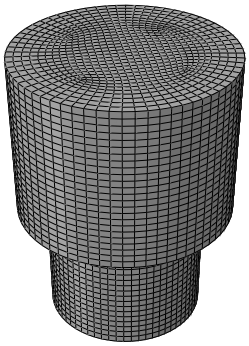


Figure 4: Rivet mesh in the pure Lagrangian case.

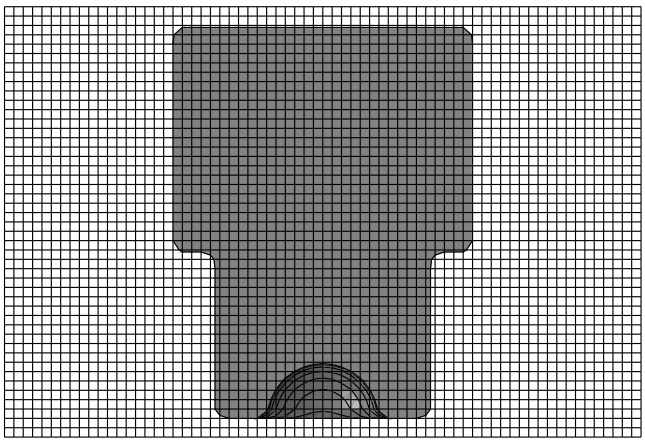


Figure 5: Cross-section of the Eulerian mesh with the rivet material assigned.

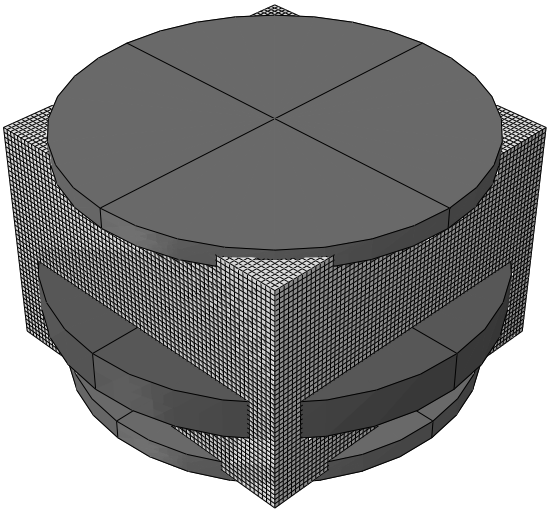


Figure 6: The Lagrangian parts are assembled inside of the Eulerian mesh.

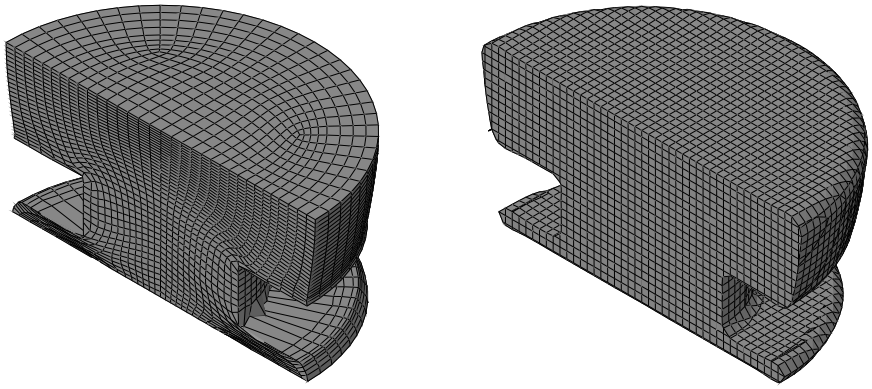


Figure 7: Deformed configuration for the pure Lagrangian case (left) and the CEL case (right).

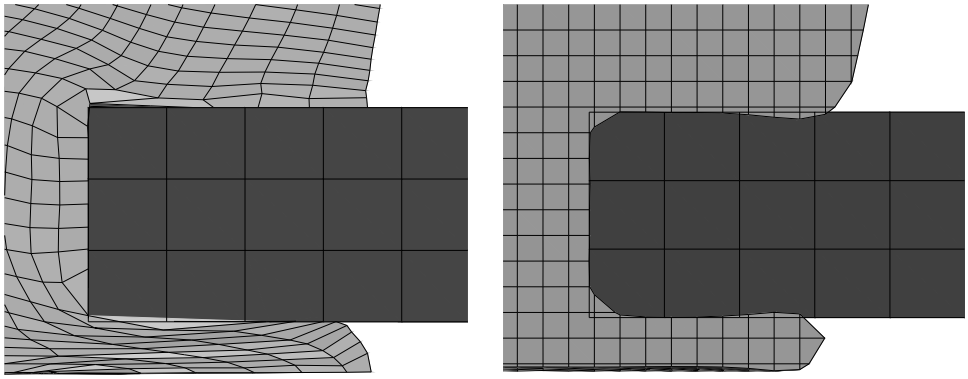


Figure 8: Contact penetrations between the rivet and plate for the pure Lagrangian case (left) and the CEL case (right).

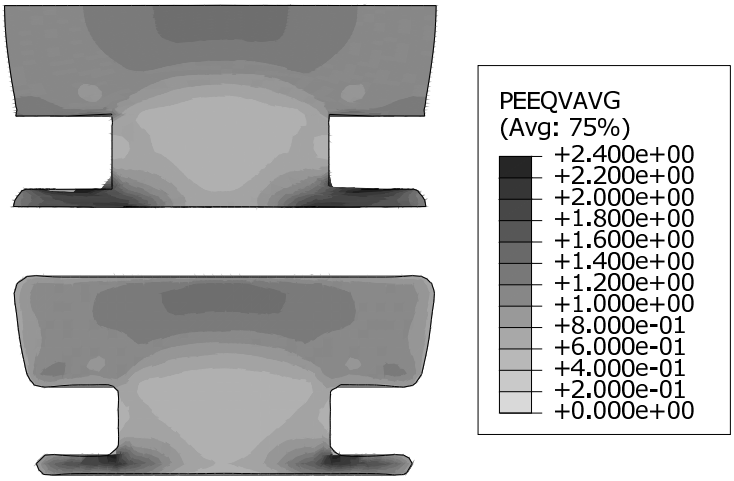


Figure 9: Equivalent plastic strain in the pure Lagrangian model (top) and the CEL model (bottom).

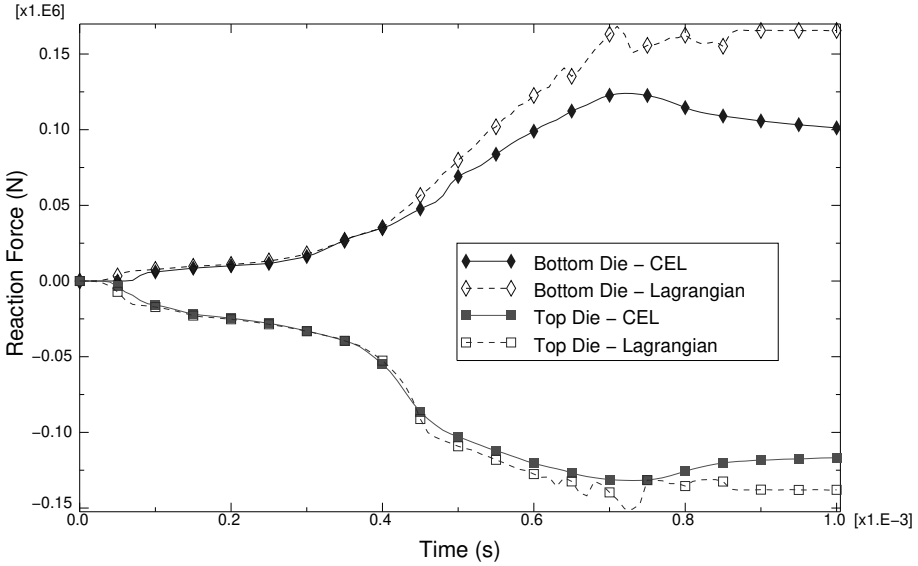


Figure 10: Reaction forces in the forming dies.

Impact of a water-filled bottle

Products: Abaqus/Explicit

References:

- *Eulerian Analysis*
- *Smoothed Particle Hydrodynamics*
- *Finite Element Conversion to SPH Particles*
- *Classical Metal Plasticity*
- *Equation of State*
- *Eulerian analyses*
- *Creating discrete fields for material volume fractions*

Overview

This example shows a drop test of a fluid-filled plastic bottle.

The following Abaqus features are demonstrated:

- using the volume fraction tool in Abaqus/CAE to model complex material distributions in an Eulerian element mesh,
- using the Eulerian-Lagrangian contact formulation to simulate a highly dynamic event involving a fluid material (modeled using Eulerian elements) interacting with structural boundaries (modeled using Lagrangian elements),
- using the smoothed particle hydrodynamic (SPH) method to analyze a highly dynamic event in a purely Lagrangian environment, and
- converting continuum finite elements to SPH particles to analyze the highly dynamic event.

Application description

Simulation is commonly used in the consumer packaging industry to reduce the time and cost associated with physical prototyping. Drop tests, which simulate an object falling and impacting a hard surface, are often used to investigate the object's response under harsh handling conditions.

This example involves a fluid-filled plastic bottle falling from a height of roughly 300 mm onto a flat, rigid floor. The bottle, as shown in *Figure 1*, is a rectangular jug made of high-density polyethylene. The bottle is filled almost completely (about 95%) with water. A realistic simulation for the bottle must account for both the exterior forces on the bottle from the floor impact, as well as the interior forces of the water pushing against the bottle. Resulting stresses and strains in the bottle can be used to determine its structural feasibility.

Geometry

Figure 1 shows the pertinent dimensions for the bottle and cap model. The bottle has a uniform thickness of 0.5 mm, with the exception of the rim around the bottle's mouth; the rim is 0.65 mm thick. The bottle's cap is modeled as a separate part instance and positioned on the mouth of the bottle; the cap is uniformly 1 mm thick.

The bottle strikes the floor at a skew angle, with one of the bottom corners experiencing the initial impact. *Figure 2* shows the assembled model. Initially the water in the bottle is distributed according to a gravitational response; that is, the boundary of the water is parallel to the horizontal floor, not to the bottom of the skewed bottle.

Materials

The bottle is constructed of high-density polyethylene that follows an isotropic plastic hardening model.

The water is treated as a nearly incompressible, nearly inviscid Newtonian fluid.

Boundary conditions and loading

The entire model is subject to a gravity load. The rigid floor is fixed in place.

Interactions

The bottle contacts three different model components over the course of the analysis: the floor, the bottle cap, and the water within the bottle. All of these contact interactions are assumed to be frictionless.

Abaqus modeling approaches and simulation techniques

The primary challenge to solving this problem is the highly transient fluid-structure interaction between the water and the bottle. In this example two methods are studied. The coupled Eulerian-Lagrangian (CEL) analysis technique in Abaqus/Explicit is well suited to handling problems of this nature. The SPH method can be used to model the violent sloshing associated with the impact.

Summary of analysis cases

Case 1	Coupled Eulerian-Lagrangian analysis.
Case 2	SPH analysis.
Case 3	Finite element conversion to SPH particle analysis.

Mesh design

The bottle is imported as an orphan mesh of S3R and S4R elements. The cap geometry is meshed with S4R elements. *Figure 3* shows the assembled mesh for the bottle and cap. The floor is meshed with surface elements of type SFM3D4R; a rigid body constraint is subsequently applied to these elements (see *Constraints* below).

Materials

An elastic-plastic material definition is used for the polyethylene in the bottle, with the isotropic hardening curve defined by the data points in *Table 1*; the onset of plastic yielding occurs at 8.618 N/mm^2, and failure occurs at a strain of 0.59. Damage is incorporated using a ductile damage definition. The high-density polyethylene has a density of 8.76×10^{-7} kg/mm^3, Young's modulus of 903.114 N/mm^2, and Poisson's ratio of 0.39.

The water is modeled using the linear $U_s - U_p$ Hugoniot form of the Mie-Grüneisen equation of state; the equation parameters appear in *Table 2*. The water has a density of 9.96×10^{-7} kg/mm^3 and bulk modulus of 2.094 GPa.

Boundary conditions

An encastre boundary condition is applied to the reference point of the rigid floor to fix it in place for the duration of the analysis.

Loads

A gravity load is applied to the bottle, cap, and water instances. An acceleration of –9800 mm/s^2 is applied in the z-direction.

Predefined fields

Instead of simulating the full dropping event from the initial position, the bottle, cap, and water instances are positioned close to the floor and prescribed an initial velocity predefined field. An initial velocity of −2444 mm/s in the z-direction corresponds to the speed that would be attained by an object falling about 300 mm from rest under typical gravitational acceleration.

Constraints

A rigid body constraint applied to the floor part instance makes the floor a simple, undeformable surface.

Interactions

General contact is defined for the model. General contact enforces interactions between the bottle and other Lagrangian components, such as the cap and floor. The default frictionless hard contact property governs all interactions.

In an actual bottle, the cap would be firmly attached to the neck of the bottle. This interaction between the bottle and the cap is considered insignificant for the purposes of this analysis. Instead of adding undue cost to the analysis by modeling a threaded connection or tie constraint between the two parts, the cap is allowed to freely detach from the bottle during the simulation.

Output requests

Displacements, velocities, and accelerations are requested for the bottle and cap. Logarithmic strain (LE) and equivalent plastic strain (PEEQ) are requested for the bottle to assess its structural response. Damage initiation criteria (DMICRT) and element status (STATUS) are also requested for the polyethylene bottle to track potential failure in these components. Finally, contact stresses (CSTRESS) and contact forces (CFORCE) are requested for the Lagrangian part instances.

Case 1 Coupled Eulerian-Lagrangian analysis

This case shows the use of the Eulerian-Lagrangian contact formulation to simulate the bottle drop with the fluid material (modeled using Eulerian elements) interacting with structural boundaries (modeled using Lagrangian elements). In addition, the volume fraction tool in Abaqus/CAE is used to model complex material distributions in an Eulerian element mesh.

Analysis types

The full simulation is conducted in a single explicit dynamic step lasting 0.05 s.

A few distorted Lagrangian elements in the model are controlling the stable time increment, thus dictating the time to complete the analysis. Using the semi-automatic mass scaling method, the time increment can be increased to 7.528×10^{-7} s, reducing the analysis cost by nearly half, while increasing the total mass by only 0.11 percent. Alternatively, using the mass adjustment method, the mass can be redistributed among the elements of the bottle to achieve the same time increment without affecting the total mass of the bottle. A large number of elements have mass in excess of what is required to achieve the above stable time increment. Only 0.12 percent of this excess mass is redistributed to the remaining elements to raise their time increment to the specified value. In both methods, the change in the mass distribution is very small and does not significantly influence the analysis results.

Analysis techniques

The Eulerian element formulation allows the analysis of bodies undergoing severe deformation without the difficulties traditionally associated with mesh distortion. In an Eulerian mesh material flows through fixed

elements, so a well-defined mesh at the start of an analysis remains well-defined throughout the analysis. Although Eulerian material boundaries are more approximate than traditional Lagrangian element boundaries, the Eulerian formulation allows you to capture extreme deformation phenomena such as fluid flow. The water is modeled using an Eulerian element domain. The bottle—which, although significantly stiffer than the water, is not rigid—is modeled using traditional Lagrangian shell elements. The general contact algorithm in Abaqus/Explicit tracks and enforces contact between the Eulerian material boundary and the Lagrangian elements, enabling effective simulation of the fluid-structure interaction.

Mesh design

The Eulerian mesh, which serves as the domain through which the water material can flow, is a 300 × 250 × 200 mm rectangular prism of EC3D8R elements. Each Eulerian element is a regular cube measuring 5 mm on an edge. The Eulerian mesh completely encompasses the bottle and cap, and it extends slightly below the floor. Any interface that is expected to experience Eulerian-Lagrangian contact must be located within the Eulerian mesh; once an Eulerian material passes beyond the boundaries of the Eulerian mesh, it is lost to the simulation and contact is not enforced. The overlapping of Lagrangian and Eulerian elements is acceptable because these two element types do not interact with each other. Lagrangian elements interact only with Eulerian material within the mesh. The initial position of this Eulerian material must be defined within the Eulerian mesh, as discussed in *Initial conditions* below.

Initial conditions

Because an Eulerian mesh is void of any material by default, the desired distribution of material within the Eulerian mesh must be specified using an initial condition. This distribution is defined using the concept of an Eulerian volume fraction, or the percentage of an element that is occupied by a given material. For each Eulerian element that initially contains material, an initial Eulerian volume fraction for that material must be specified. Abaqus evaluates all of the element volume fractions to determine the distribution and boundaries of Eulerian materials within the Eulerian mesh.

Abaqus/CAE provides tools that greatly simplify the definition of initial material volume fractions, particularly for complex geometries. The desired material geometry can be modeled as a separate part (the reference part), then instanced into the Eulerian mesh in a position that corresponds to the intended material position. The volume fraction tool performs a Boolean comparison between the Eulerian mesh and the reference part, then creates a discrete field that associates each element in the mesh with a percentage based on the amount of space occupied by the reference part within that element. This discrete field can be used in a material assignment predefined field to specify the appropriate Eulerian volume fractions for a particular material.

A part corresponding to the geometry of the water in the bottle is provided with this example (`cel_bottle_drop_watergeom.sat`). This part is imported into Abaqus/CAE, then instanced and oriented in the model assembly with the other part instances. The volume fraction tool uses the water geometry as the reference part instance within the Eulerian part instance (the Eulerian part must be meshed before using the tool). The resulting discrete field is used to create a water material assignment predefined field in the Load module. This material assignment distributes the water in the Eulerian mesh in a geometry corresponding to the water geometry part instance. The water geometry instance is subsequently suppressed to remove it from the model.

Although it is not used in the script for this example, a solid version of the bottle geometry is also available (`cel_bottle_drop_solidgeom.sat`). The water reference geometry can be created from this solid bottle using geometry cuts in the Assembly module (see *Performing Boolean operations on part instances*).

Predefined fields

The initial material assignment for the Eulerian mesh is also defined in Abaqus/CAE with a predefined field. A material assignment predefined field associates the discrete field that was created by the volume fraction tool (see *Initial conditions*) with the water material definition.

Interactions

The contact between the bottle and the Eulerian material initially positioned inside of the bottle is also modeled using the general contact algorithm.

Output requests

In addition to the field and history output requests listed above, the Eulerian volume fraction output variable (EVF) is requested as field output to visualize geometric results.

Case 2 SPH analysis

This case shows the use of a smoothed particle hydrodynamic (SPH) analysis to analyze the fluid-filled bottle drop in a purely Lagrangian environment.

Analysis types

The full simulation is conducted in a single explicit dynamic step lasting 0.05 s.

Analysis techniques

The water is modeled using continuum pseudo-particles. Internally, Abaqus/Explicit determines automatically every increment of the analysis which are the active neighbors associated with a given particle of interest in order to apply the SPH formulation. Since the internally determined connectivity is allowed to change every increment, the method robustly handles the severe deformations associated with the sloshing water. The other Lagrangian bodies (bottle, cap, and floor) are modeled the same as in the coupled Eulerian-Lagrangian method.

Mesh design

The pseudo-particles are modeled using PC3D elements that are spaced in a relatively uniform fashion every 5 mm in all directions. These one-node elements fill only the space initially occupied by the water just before the impact. Therefore, fewer PC3D elements are needed when compared to the number of EC3D8R elements in the coupled Eulerian-Lagrangian method.

Interactions

A node-based surface associated with the water pseudo-particle is included in the contact domain to model the interactions between the water and the bottle.

Case 3 Finite element conversion to SPH particles analysis

This case shows the use of the technique for converting reduced-integration continuum elements to SPH particles.

Analysis types

The full simulation is conducted in a single explicit dynamic step lasting 0.05 s.

Analysis techniques

The water is modeled using reduced-integration continuum C3D4 elements. A time-based criterion is used to trigger the conversion from a user-defined mesh to SPH particles (PC3D elements) at the beginning of the analysis. Upon conversion the continuum elements modeling the fluid will become inactive (deleted from the mesh) while the PC3D elements get activated. Since all particles get activated at the same time, the SPH functionality will model the fluid as described in Case 2.

Mesh design

The water is modeled using reduced-integration continuum C3D4 elements using regular preprocessing techniques; the mesh is shown in *Figure 4*. Thus, cumbersome meshing approaches based on unconnected particles are avoided.

Interactions

Contact between the fluid and the bottle is defined by including an element-based surface that includes the fluid's interior and its initial external surface in the general contact domain. Internally, Abaqus/Explicit generates a node-based surface for the generated SPH particles that will be also included in the contact domain.

Discussion of results and comparison of cases

The results of the drop test in the CEL analysis appear in *Figure 5*. (For tips on viewing the results of CEL analyses, see *Viewing output from Eulerian analyses*.) The water contributes significantly to the behavior of the bottle: the bottle flexes and bulges as the water sloshes. *Figure 6* shows the logarithmic strains in the bottle. Strains appear in the sides of the bottle as they bulge outward, but these strains are largely recovered when the water sloshes upward. The most significant strains occur on the bottom of the bottle, along the bottle-floor interface. Despite the deformation caused by the floor and the water, the damage criteria for the polyethylene is not met at any location on the bottle.

A comparison between the CEL method and the SPH method at different stages during impact is illustrated in *Figure 7* through *Figure 10*. The comparison between the two methods is quite reasonable.

The effect of sloshing water can also be seen in the reaction force on the floor surface. The reaction force in the z-direction is plotted in *Figure 11* using a Butterworth filter with a cutoff frequency of 2000. The reaction force peaks sharply at approximately 0.02 s, which corresponds roughly with the appearance of strains in the sides of the bottle. After this point, the water surges upward, offsetting the downward momentum of the bottle and reducing the reaction forces in the floor. Good agreement between the two methods used is observed.

The significant deformation of the fluid in this problem clearly influences the analysis results. The coupled Eulerian-Lagrangian and SPH techniques in Abaqus/Explicit provide an effective way to realistically capture the complexity of the fluid-structure dynamics.

Files

The basic system of units in this problem is kilograms (kg), millimeters (mm), and seconds (s). Under this system, the base unit of force is kg mm/s^2, or 10^{-3} newtons; values for all force-derived quantities are entered in the sample files using this base unit. Values for all force-derived quantities are reported in the documentation in units of standard newtons (kg m/s^2) and scaled appropriately.

Case 1 Coupled Eulerian-Lagrangrian analysis

cel_bottle_drop.py	Script to generate the model in Abaqus/CAE using the orphan mesh from cel_bottle_drop_mesh.inp and the geometry from cel_bottle_drop_watergeom.sat.
cel_bottle_drop_mesh.inp	Orphan mesh for the bottle.
cel_bottle_drop_watergeom.sat	Reference geometry for the water.
cel_bottle_drop_solidgeom.sat	Reference geometry for the solid bottle.
cel_bottle_drop.inp	Complete input file for the model.
cel_bottle_drop_massadjust.inp	Complete input file for the model with the target time increment maintained by mass adjustment instead of mass scaling.

Case 2 SPH analysis

sph_bottle_drop.inp Complete input file for the model.

Case 3 Finite element conversion to SPH particles analysis

sphconv_bottle_drop.inp Finite element conversion to SPH particles input file for the model.

Tables

Table 1: Isotropic plastic hardening data for polyethylene material.

Yield stress (N/mm^2)	Plastic strain
8.618	0.0
13.064	0.007
16.787	0.025
18.476	0.044
20.337	0.081
24.543	0.28
26.887	0.59

Table 2: Material parameters for water.

Parameter	Value
Density (ρ)	9.96×10^{-7} kg/mm^3
Viscosity (η)	1×10^{-8} N s/mm^2
c_0	1.45×10^6 mm/s
s	0
Γ_0	0

Figures

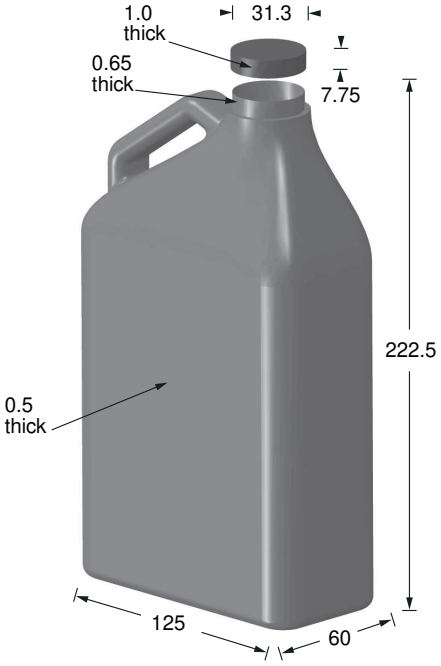


Figure 1: Bottle used in the drop test. All dimensions are in millimeters.



Figure 2: Assembly of the bottle drop model; the water is instanced inside of the bottle.

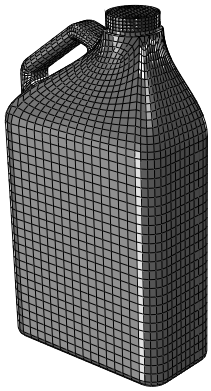


Figure 3: Mesh of the bottle and cap.

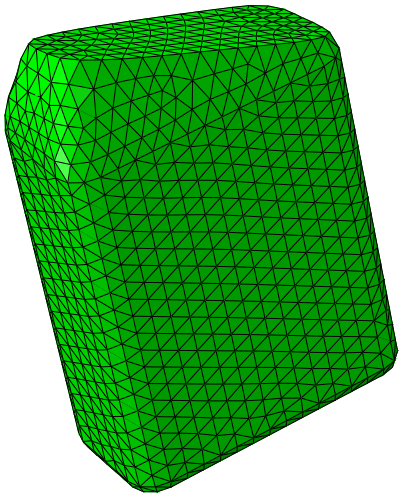


Figure 4: Tetrahedral mesh to model the fluid in the bottle in the undeformed configuration.

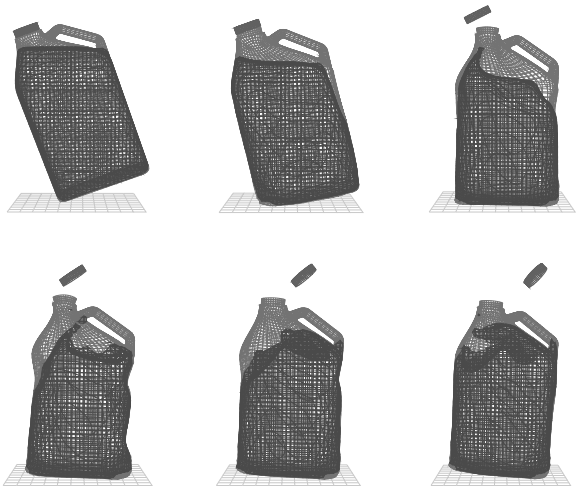


Figure 5: Deformed shape plots of the CEL model at 0.01 s intervals.

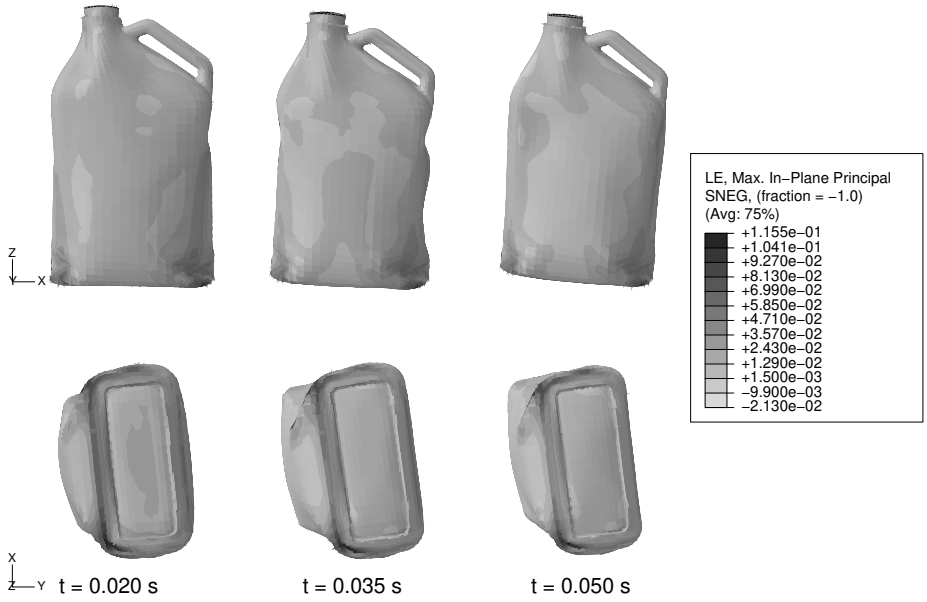


Figure 6: Logarithmic strains in the bottle during deformation in the CEL analysis.

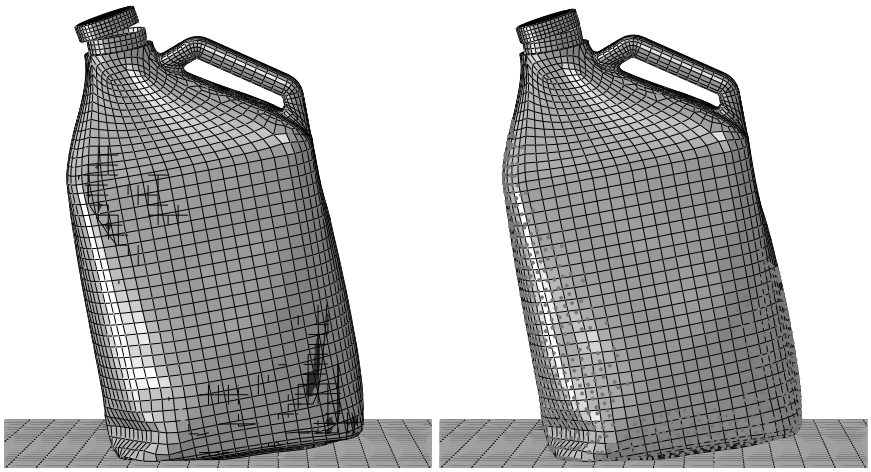


Figure 7: Comparison between coupled Eulerian-Lagrangian and SPH results shortly after the impact.

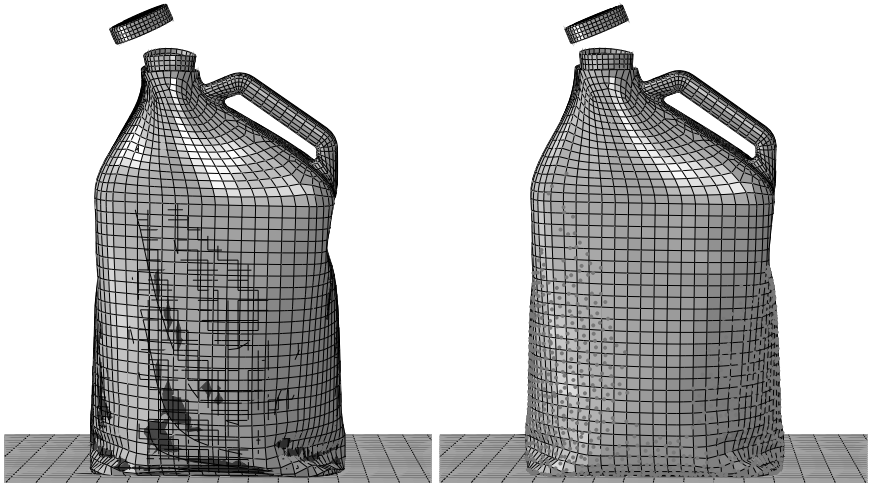


Figure 8: Comparison between coupled Eulerian-Lagrangian and SPH results after full contact with the floor.

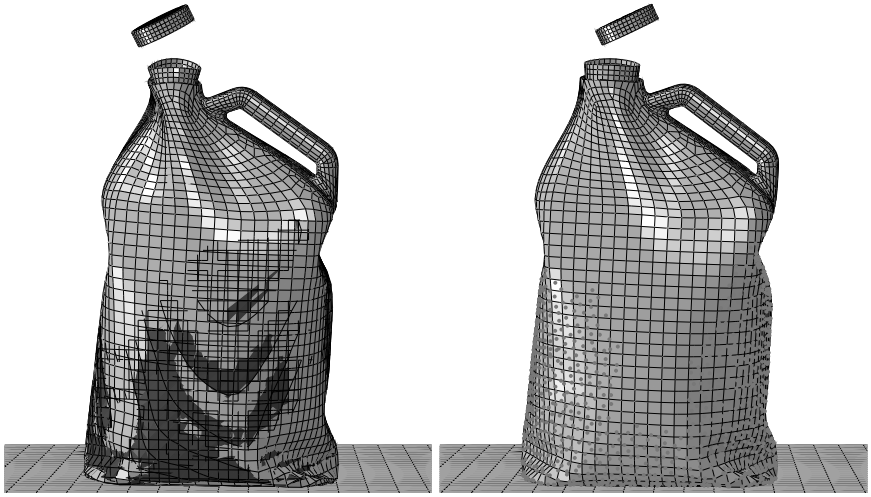


Figure 9: Comparison between coupled Eulerian-Lagrangian and SPH results in the middle stages of the analysis.

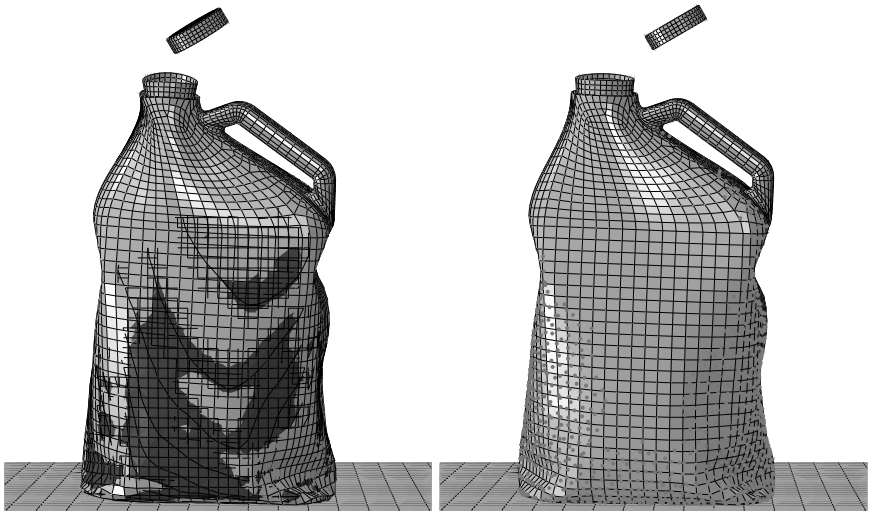


Figure 10: Comparison between coupled Eulerian-Lagrangian and SPH results towards the end of the analysis.

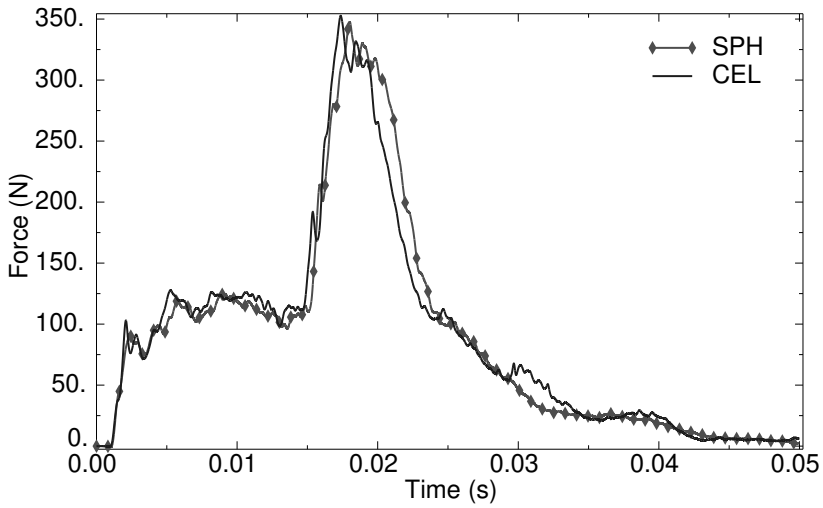


Figure 11: Reaction force (filtered) in the floor.

Dynamic impact of a scooter with a bump

Products: Abaqus/Standard Abaqus/Explicit

References:

- *About Co-Simulation*
- *Preparing an Abaqus Analysis for Co-Simulation*
- *Structural-to-Structural Co-Simulation*
- **CO-SIMULATION*
- **CO-SIMULATION REGION*
- *Abaqus/Standard to Abaqus/Explicit co-simulation*

Overview

This example demonstrates the combined use of Abaqus/Standard and Abaqus/Explicit to provide a more cost effective solution than by using either Abaqus/Standard or Abaqus/Explicit alone.

The following Abaqus features are demonstrated:

- Abaqus/Standard to Abaqus/Explicit co-simulation, where
- Abaqus/Standard employs substructuring, a feature not available in Abaqus/Explicit, to efficiently handle modeling of a component subjected to small strains, and
- Abaqus/Explicit is used to efficiently simulate high-speed contact interactions.

Application description

This example considers the impact of a recreational scooter with a bump. An analysis of the transient response of an interaction with a bump is used to determine the accelerations felt by the scooter operator. With this analysis a product designer can make informed design decisions by varying certain design parameters, such as the frame component cross-section properties, tire material, or inflation pressure and observing their influence on the operator acceleration. Effective use of this simulation technique requires that the simulation turnaround time be as quick as possible while retaining essential fidelity in the results.

Geometry

The scooter consists of an operator deck, a frame with handlebars, and two tires, as shown in *Figure 1*. The overall length of the scooter is 1200 mm, and the overall height is 800 mm. The handlebars are oriented straight.

Materials

The scooter deck and frame tubing are made of mild steel. The tires are a Butyl rubber material.

Initial conditions

The simulation begins with the scooter having inflated tires and traveling toward the bump at a speed of 3 m/s.

Boundary conditions and loading

Weight loading is due to the scooter and the operator.

Interactions

The scooter travels on a rough surface. This surface includes a bump that is 7.5 cm high and is positioned in front of the scooter.

Abaqus modeling approaches and simulation techniques

Several Abaqus analysis approaches can be used to simulate the transient behavior of the scooter: using only Abaqus/Standard, using only Abaqus/Explicit, or using Abaqus/Standard to Abaqus/Explicit co-simulation. To illustrate the computational cost savings of the co-simulation approach, the analysis cases that follow focus on comparing the co-simulation approach to an Abaqus/Explicit-only simulation approach.

The Abaqus/Standard model of the co-simulation analysis consists of the scooter deck (*Figure 2*) and frame (*Figure 3*). The deck is modeled using substructure techniques to further reduce the solution cost. The Abaqus/Explicit model of the co-simulation analysis consists of the tires and the road with the bump (*Figure 4*). Co-simulation regions across which data will be exchanged during the co-simulation analysis are identified on each model at the location of the wheel axles.

Summary of analysis cases

Case 1	A reference analysis performed using only Abaqus/Explicit.
Case 2	A co-simulation analysis using the subcyling coupling scheme.

Both analysis cases address the same transient simulation.

Analysis types

The tire transient response and contact with the road are simulated using the explicit dynamics procedure. In Case 1 the explicit dynamics procedure applies to the rest of the model as well. The scooter frame transient response is simulated using the implicit dynamics procedure.

Analysis Techniques

Static stabilization, substructuring, and import analysis techniques are used in this example.

Static stabilization

The inflation of the tires uses the Abaqus/Standard static stabilization option.

Substructuring

The scooter deck is expected to experience small strains during the analysis, allowing the use of a substructure to model the deck as a technique to lower computational cost.

Import

The tires are inflated in an Abaqus/Standard static procedure. The inflated state and configuration of the tires are imported into Abaqus/Explicit for the subsequent transient analysis.

Mesh design

The meshed models are shown in *Figure 5*. The frame components are modeled using truss and connector elements. The deck is modeled using S4R shell elements. The tires are modeled using C3D8I continuum elements. The operator is represented by a point mass connected to foot and hand-hold locations through a distributing coupling constraint definition.

Materials

The deck is modeled as a simple linear elastic material, with an elastic modulus of 5 GPa, a Poisson's ratio of 0.3, and a density of 5000 kg/m^3.

The tires are modeled as a hyperelastic material with viscoelastic properties. The tire material density is 1100 kg/m^3.

Hyperelastic material constants	
C_{10}	1 MPa
C_{01}	0.0 Pa
D_1	5.085×10^{-8} Pa

Viscoelastic material constants	
$\overline{g}_1^P$	0.3
$\overline{k}_1^P$	0
τ_1	0.1

Initial conditions

An initial tire pressure of 20 KPa is applied. The tires begin the analysis in a statically equilibrated footprint configuration as a result of import from an earlier static analysis. The scooter is traveling toward the bump at an initial velocity of 3 m/s.

Boundary conditions

The scooter steering assembly is fixed in a straight configuration.

Loads

Forces are applied consistent with a total scooter weight of 42.4 N and a total operator weight of 222 N. To simplify the analysis setup, gravity loading is not applied; instead, the weight forces are applied at the axle locations. The main consequence of this loading approach is that the static sag of the deck due to the operator is neglected.

Constraints

The scooter deck is attached to the frame using tie constraints.

Interactions

Contact interactions define the tire contact with the road and bump.

Case 1 Abaqus/Explicit reference analysis

This analysis case uses Abaqus/Explicit exclusively for the transient analysis and is provided as a reference for comparing the results and computational cost of the co-simulation solution.

Analysis types

Abaqus/Explicit is used for the transient analysis with the tire inflation occurring using the static procedure in Abaqus/Standard.

Analysis steps

The following analysis step types are used.

Static analysis

An Abaqus/Standard static procedure is used to inflate the tires and establish the tire footprint.

Explicit dynamics

The Abaqus/Explicit step does not employ mass scaling. The default bulk viscosity parameters are used.

Results and discussion

The simulation results show the scooter impacting the bump, the operator shifting forward slightly, and the deck flexing.

Case 2 Co-simulation analysis using subcycling

In this case co-simulation occurs between Abaqus/Standard and Abaqus/Explicit, with each program advancing its simulation time according to its own automatic incrementation scheme and exchanging data as needed. Co-simulation data are exchanged at each Abaqus/Explicit time increment.

Analysis types

Abaqus/Explicit is used for the transient analysis of the tires with the tire inflation occurring using the static procedure in Abaqus/Standard. Abaqus/Standard is used for the transient analysis of the scooter frame and deck.

Analysis steps

The following analysis step types are used.

Static analysis

An Abaqus/Standard static procedure is used to inflate the tires and establish the tire footprint.

Explicit dynamics

The Abaqus/Explicit step does not employ mass scaling. The default bulk viscosity parameters are used.

Substructure generation

The substructure generation of the scooter deck uses 100 retained eigmodes, obtained using the Lanczos eigensolver.

Implicit dynamics

The Abaqus/Standard transient dynamic steps use a half-increment residual limit setting of 200 N.

Solution controls

Co-simulation controls are used to specify a subcycling approach.

Run procedure

The analyses proceed in two stages. In the first stage a pair of jobs is run. One job creates a substructure matrix for the main structure:

```
abaqus job=scooter_subgen
```

A second job establishes the state of an inflated and seated tire:

```
abaqus job=scooter_tire_inflation
```

With the results from these two jobs available, the co-simulation job can be run, coupling the structure to two instances of the inflated tire. The coupling is achieved using the Abaqus co-simulation execution procedure (see *Abaqus/Standard, Abaqus/Explicit, and FMU Co-Simulation Execution*).

For example, use the following command to run the Abaqus/Standard and Abaqus/Explicit jobs:

```
abaqus cosimulation cosimjob=scooter
   job=scooter_cosim_std,scooter_cosim_xpl
   oldjob=NONE,scooter_tire_inflation
   configure=scooter_cosim_config
```

Results and discussion

The results for a co-simulation analysis appear in multiple output files. To make effective use of the Abaqus/Standard and Abaqus/Explicit output databases, you should use the Abaqus/Viewer overlay functionality to view the combined results. For more information, see *Overlaying multiple plots*.

Discussion of results and comparison of cases

The results show that Case 1 is more computationally expensive. Case 2 provides a significant cost improvement. Further, the results show that the solution fidelity, when compared to the reference solution, is not significantly affected by the use of the co-simulation technique.

Final configuration of the assembly

As the scooter collides with the bump, the entire assembly leaves the ground and flexes slightly, as shown in *Figure 6*.

Dynamic response of the operator

We consider measures of operator comfort to relate to the displacement, velocity, and acceleration magnitude histories at the operator node location. *Figure 7*, *Figure 8*, and *Figure 9* show these respective response histories, comparing the results for the Case 1 and Case 2 analyses. The results are plotted after applying the Butterworth filter with a cutoff frequency of 100 Hz (see *Applying Butterworth filtering to an X–Y data object*) and show very good agreement between the Case 1 and Case 2 workflows.

Computational cost

Table 1 lists the relative computation cost of the two simulation approaches and clearly shows the value of co-simulation in this analysis.

Files

Common files

scooter_tire_inflation.inp — Abaqus/Standard input file to inflate the tires and establish the static footprint due to the assembly weight.

scooter_parameters.inp — Job parameters common to all scooter analyses.

Case 1 Abaqus/Explicit reference analysis

scooter_xpl.inp — Abaqus/Explicit input file to model all components, importing the inflated tires from scooter_tire_inflation.inp results and simulating the transient impact with the bump.

Case 2 Co-simulation analysis using subcycling

Co-simulation configuration

scooter_cosim_config.xml — Co-simulation configuration file defining the subcycling algorithm.

Tire and road modeling

scooter_cosim_xpl.inp — Abaqus/Explicit input file to model the tires and road, importing the inflated tires from scooter_tire_inflation.inp results and simulating the transient impact with the bump through co-simulation coupling with scooter_cosim_std.inp.

Frame and deck modeling

scooter_subgen.inp — Abaqus/Standard input file modeling the deck and creating the substructure representation of the deck.

scooter_cosim_std.inp — Abaqus/Standard input file modeling the frame components, referring to the scooter_subgen.inp substructure definition and simulating the transient impact with the bump through co-simulation coupling with scooter_cosim_xpl.inp.

Tables

Table 1: Comparison of relative CPU times (normalized with respect to the CPU time for the Abaqus/Explicit analysis).

Analysis job	Relative CPU time	
	Co-simulation workflow	Abaqus/Explicit workflow
Substructure generation	0.007	N/A
Tire inflation and footprint	0.002	0.002
Co-simulation Abaqus/Explicit analysis	0.060	N/A
Co-simulation Abaqus/Standard analysis	0.057	N/A
Complete Abaqus/Explicit analysis	N/A	1.0
Total simulation cost	0.126	1.002

Figures

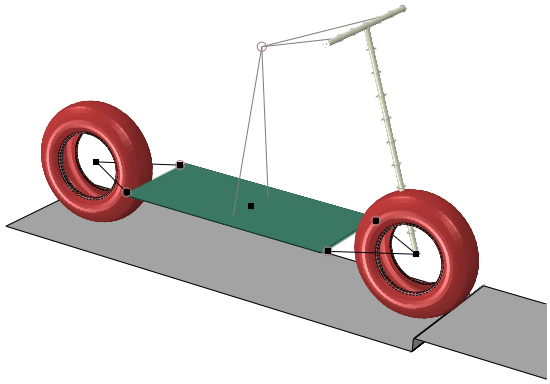


Figure 1: Scooter assembly geometry. The operator is modeled as a point mass.

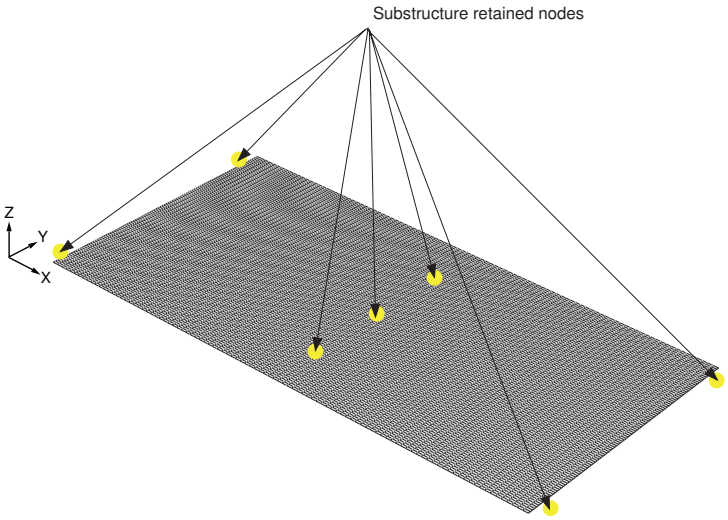


Figure 2: Deck analysis region. Substructure retained nodes represent the attachment locations to the scooter frame, the operator foot locations, and a center location used to constrain roll about the x-direction for the global model.

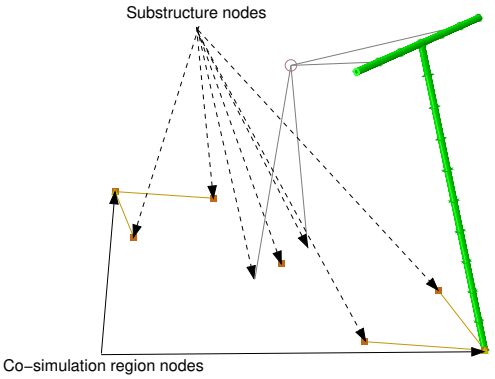


Figure 3: Frame analysis region.

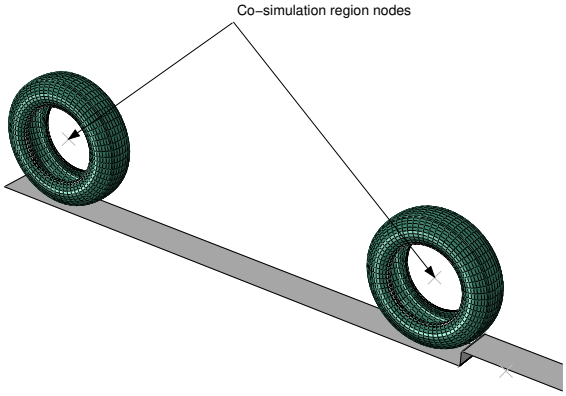


Figure 4: Tire and road analysis region.

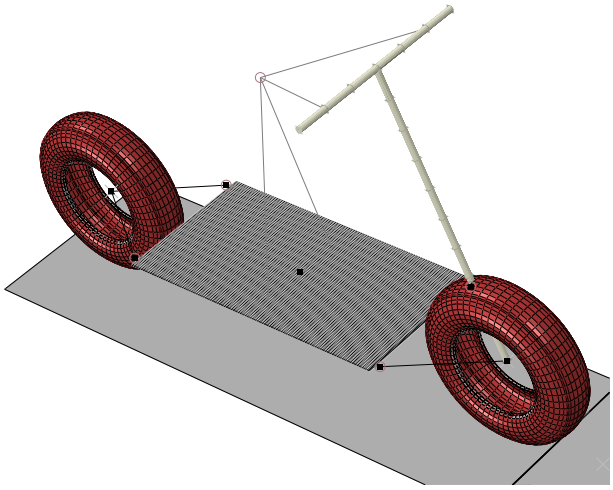


Figure 5: The complete scooter assembly mesh.

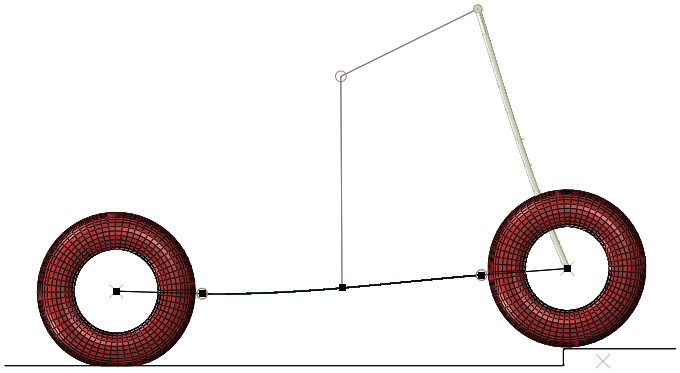


Figure 6: Deformed configuration after colliding with the bump.

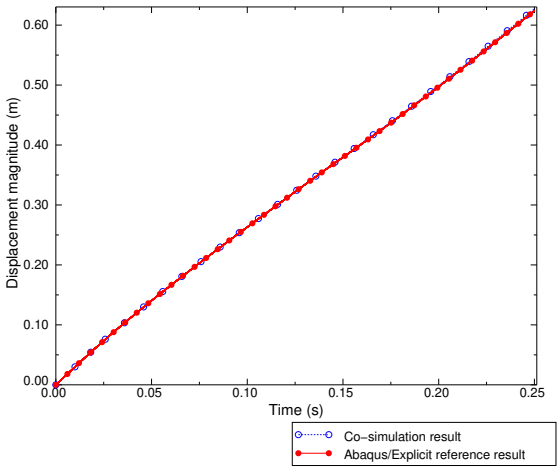


Figure 7: Operator displacement magnitude history.

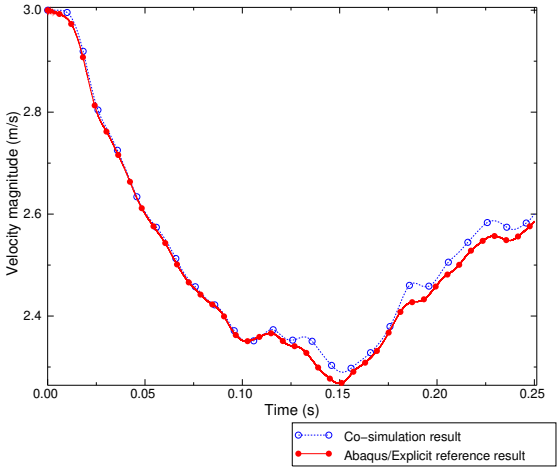


Figure 8: Operator velocity magnitude history.

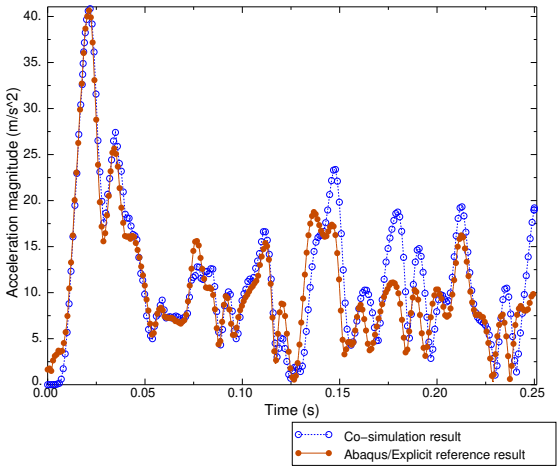


Figure 9: Operator acceleration magnitude history.

Tire and Vehicle Analyses

In this section:

- *Tire analyses*
- *Vehicle analyses*
- *Occupant safety analyses*

Tire analyses

In this section:

- *Symmetric results transfer for a static tire analysis*
- *Steady-state rolling analysis of a tire*
- *Subspace-based steady-state dynamic tire analysis*
- *Steady-state dynamic analysis of a tire substructure*
- *Coupled acoustic-structural analysis of a tire filled with air*
- *Import of a steady-state rolling tire*
- *Analysis of a solid disc with Mullins effect and permanent set*
- *Tread wear simulation in Abaqus/Standard*
- *Dynamic analysis of an air-filled tire with rolling transport effects*
- *Acoustics in a circular duct with flow*

Symmetric results transfer for a static tire analysis

Products: Abaqus/Standard

Overview

This example illustrates the use of symmetric results transfer and symmetric model generation to model the static interaction between a tire and a rigid surface.

Symmetric model generation (*Symmetric Model Generation*) can be used to create a three-dimensional model by revolving an axisymmetric model about its axis of revolution or by combining two parts of a symmetric model, where one part is the original model and the other part is the original model reflected through a line or a plane. Both model generation techniques are demonstrated in this example.

Symmetric results transfer (*Transferring Results from a Symmetric Mesh or a Partial Three-Dimensional Mesh to a Full Three-Dimensional Mesh*) allows the user to transfer the solution obtained from an axisymmetric analysis onto a three-dimensional model with the same geometry. It also allows the transfer of a symmetric three-dimensional solution to a full three-dimensional model. Both these results transfer features are demonstrated in this example. The results transfer capability can significantly reduce the analysis cost of structures that undergo symmetric deformation followed by nonsymmetric deformation later during the loading history.

The purpose of this example is to obtain the footprint solution of a 175 SR14 tire in contact with a flat rigid surface, subjected to an inflation pressure and a concentrated load on the axle. Input files modeling a tire in contact with a rigid drum are also included. These footprint solutions are used as the starting point in *Steady-state rolling analysis of a tire*, where the free rolling state of the tire rolling at 10 km/h is determined and in *Subspace-based steady-state dynamic tire analysis*, where a frequency response analysis is performed.

Problem description

The different components of the tire are shown in *Figure 1*. The tread and sidewalls are made of rubber, and the belts and carcass are constructed from fiber-reinforced rubber composites. The rubber is modeled as an incompressible hyperelastic material, and the fiber reinforcement is modeled as a linear elastic material. A small amount of skew symmetry is present in the geometry of the tire due to the placement and $\pm 20.0^\circ$ orientation of the reinforcing belts.

Two simulations are performed in this example. The first simulation exploits the symmetry in the tire model and utilizes the results transfer capability; the second simulation does not use the results transfer capability. Comparisons between the two methodologies are made for the case where the tire is in contact with a flat rigid surface. Input files modeling a tire in contact with a rigid drum are also included. The methodology used in the first analysis is applied in this simulation. Results for this case are presented in *Steady-state rolling analysis of a tire*.

The first simulation is broken down into three separate analyses. In the first analysis the inflation of the tire by a uniform internal pressure is modeled. Due to the anisotropic nature of the tire construction, the inflation loading gives rise to a circumferential component of deformation. The resulting stress field is fully three-dimensional, but the problem remains axisymmetric in the sense that the solution does not vary as a function of position along the circumference. Abaqus provides axisymmetric elements with twist (CGAX) for such situations. These elements are used to model the inflation loading. Only half the tire cross-section is needed for the inflation analysis due to a reflection symmetry through the vertical line that passes through the tire axle (see *Figure 2*). We refer to this model as the axisymmetric model.

The second part of the simulation entails the computation of the footprint solution, which represents the static deformed shape of the pressurized tire due to a vertical dead load (modeling the weight of a vehicle). A

three-dimensional model is needed for this analysis. The finite element mesh for this model is obtained by revolving the axisymmetric cross-section about the axis of revolution. A nonuniform discretization along the circumference is used as shown in *Figure 3*. In addition, the axisymmetric solution is transferred to the new mesh where it serves as the initial or base state in the footprint calculations. As with the axisymmetric model, only half of the cross-section is needed in this simulation, but skew-symmetric boundary conditions must be applied along the midplane of the cross-section to account for antisymmetric stresses that result from the inflation loading and the concentrated load on the axle. We refer to this model as the partial three-dimensional model.

In the last part of this analysis the footprint solution from the partial three-dimensional model is transferred to a full three-dimensional model and brought into equilibrium. This full three-dimensional model is used in the steady-state transport example that follows. The model is created by combining two parts of the partial three-dimensional model, where one part is the mesh used in the second analysis and the other part is the partial model reflected through a line. We refer to this model as the full three-dimensional model.

A second simulation is performed in which the same loading steps are repeated, except that the full three-dimensional model is used for the entire analysis. Besides being used to validate the results transfer solution, this second simulation allows us to demonstrate the computational advantage afforded by the Abaqus results transfer capability in problems with rotational and/or reflection symmetries.

Model definition

In the first simulation the inflation step is performed on the axisymmetric model and the results are stored in the results files (`.res`, `.mdl`, `.stt`, and `.prt`). The axisymmetric model is discretized with CGAX4H and CGAX3H elements. The belts and ply are modeled with rebar in surface elements embedded in continuum elements. The roundoff tolerance for the embedded element technique is used to adjust the positions of embedded element nodes such that they lie exactly on host element edges. This feature is useful in cases where embedded nodes are offset from host element edges by a small distance caused by numerical roundoff. Eliminating such gaps reduces the number of constraint equations used to embed the surface elements and, hence, improves performance. The axisymmetric results are read into the subsequent footprint analysis, and the partial three-dimensional model is generated by Abaqus by revolving the axisymmetric model cross-section about the rotational symmetry axis. The partial three-dimensional model is composed of four sectors of CCL12H and CCL9H cylindrical elements covering an angle of 320°, with the rest of the tire divided into 16 sectors of C3D8H and C3D6H linear elements. The linear elements are used in the footprint region. The use of cylindrical elements is recommended for regions where it is possible to cover large sectors around the circumference with a small number of elements. In the footprint region, where the desired resolution of the contact patch dictates the number of elements to be used, it is more cost-effective to use linear elements. The road (or drum) is defined as an analytical rigid surface in the partial three-dimensional model. The results of the footprint analysis are read into the final equilibrium analysis, and the full three-dimensional model is generated by reflecting the partial three-dimensional model through a vertical line. The line used in the reflection is the vertical line in the symmetry plane of the tire, which passes through the axis of rotation. The model is reflected through a symmetry line, as opposed to a symmetry plane, to take into account the skew symmetry of the tire. The analytical rigid surface as defined in the partial three-dimensional model is transferred to the full model without change. The three-dimensional finite element mesh of the full model is shown in *Figure 4*.

In the second simulation a **datacheck** analysis is performed to write the axisymmetric model information to the results files. The full tire cross-section is meshed in this model. No analysis is needed. The axisymmetric model information is read in a subsequent run, and a full three-dimensional model is generated by Abaqus by revolving the cross-section about the rotational symmetry axis. The road is defined in the full model. The three-dimensional finite element mesh of the full model is identical to the one generated in the first analysis. However, the inflation load and concentrated load on the axle are applied to the full model without making use of the results transfer capability.

The footprint calculations are performed with a friction coefficient of zero in anticipation of eventually performing a steady-state rolling analysis of the tire using steady-state transport, as explained in *Steady-state rolling analysis of a tire*.

Since the results from the static analyses performed in this example are used in a subsequent time-domain dynamic example, the input files include a hyperelastic material that models the rubber directly using the Prony series parameters. This approach enables us to model viscoelasticity in the steady-state transport example that follows. As a consequence of defining a time-domain viscoleastic material property, the elastic properties specified in the hyperelasticity material behavior define the long-term behavior of the rubber. In addition, all static steps are defined to ensure that the static solutions are based upon the long-term elastic moduli.

Loading

As discussed in the previous sections, the loading on the tire is applied over several steps. In the first simulation the inflation of the tire to a pressure of 200.0 kPa is modeled using the axisymmetric tire model (*tiretransfer_axi_half.inp*) with a static analysis procedure. The results from this axisymmetric analysis are then transferred to the partial three-dimensional model (*tiretransfer_symmetric.inp*) in which the footprint solution is computed in two sequential static steps. The first of these static steps establishes the initial contact between the road and the tire by prescribing a vertical displacement of 0.02 m on the rigid body reference node. Since this is a static analysis, it is recommended that contact be established with a prescribed displacement, as opposed to a prescribed load, to avoid potential convergence difficulties that might arise due to unbalanced forces. The prescribed boundary condition is removed in the second static step, and a vertical load of $\boldsymbol{N} =$ 1.65 kN is applied to the rigid body reference node. The 1.65 kN load in the partial three-dimensional model represents a 3.3 kN load in the full three-dimensional model. The transfer of the results from the axisymmetric model to the partial three-dimensional model is accomplished by using symmetric results transfer. Once the static footprint solution for the partial three-dimensional model has been established, symmetric results transfer is used to transfer the solution to the full three-dimensional model (*tiretransfer_full.inp*), where the footprint solution is brought into equilibrium in a single static increment. The results transfer sequence is illustrated in *Figure 5*.

Boundary conditions and loads are not transferred with the symmetric results transfer; they must be carefully redefined in the new analysis to match the loads and boundary conditions from the transferred solution. Due to numerical and modeling issues the element formulations for the two-dimensional and three-dimensional elements are not identical. As a result, there may be slight differences between the equilibrium solutions generated by the two- and three-dimensional models. In addition, small numerical differences may occur between the symmetric and full three-dimensional solutions because of the presence of symmetry boundary conditions in the symmetric model that are not used in the full model. Therefore, it is advised that in a results transfer simulation an initial step be performed where equilibrium is established between the transferred solution and loads that match the state of the model from which the results are transferred. It is recommended that an initial static step with the initial time increment set to the total step time be used to allow Abaqus/Standard to find the equilibrium in one increment.

In the second simulation identical inflation and footprint steps are repeated. The only difference is that the entire analysis is performed on the full three-dimensional model (*tiretransfer_full_footprint.inp*). The full three-dimensional model is generated using the restart information from a **datacheck** analysis of an axisymmetric model of the full tire cross-section (*tiretransfer_axi_full.inp*).

Contact modeling

The default contact pair formulation in the normal direction is hard contact, which gives strict enforcement of contact constraints. Some analyses are conducted with both hard and augmented Lagrangian contact to demonstrate that the default penalty stiffness chosen by the code does not affect stress results significantly. The augmented Lagrangian method can be invoked as part of the definition of the modified contact pressure-overclosure relationship. The hard and augmented Lagrangian contact algorithms are described in *Contact Constraint Enforcement Methods in Abaqus/Standard*.

Solution controls

Since the three-dimensional tire model has a small loaded area and, thus, rather localized forces, the default averaged flux values for the convergence criteria produce very tight tolerances and cause more iteration than is necessary for an accurate solution. To decrease the computational time required for the analysis, the solution controls can be used to override the default values for average forces and moments. The default controls are used in this example.

Results and discussion

The results from the first two simulations are essentially identical. The peak Mises stresses and displacement magnitudes in the two models agree within 0.3% and 0.2%, respectively. The final deformed shape of the tire is shown in *Figure 6*. The computational cost of each simulation is shown in *Table 1*. The simulation performed on the full three-dimensional model takes 2.5 times longer than the results transfer simulation, clearly demonstrating the computational advantage that can be attained by exploiting the symmetry in the model using the symmetric results transfer.

Input files

tiretransfer_axi_half.inp	Axisymmetric model, inflation analysis (simulation 1).
tiretransfer_symmetric.inp	Partial three-dimensional model, footprint analysis (simulation 1).
tiretransfer_symmetric_auglagr.inp	Partial three-dimensional model, footprint analysis using augmented Lagrangian contact (simulation 1).
tiretransfer_full.inp	Full three-dimensional model, final equilibrium analysis (simulation 1).
tiretransfer_full_auglagr.inp	Full three-dimensional model, final equilibrium analysis using augmented Lagrangian contact (simulation 1).
tiretransfer_axi_full.inp	Axisymmetric model, **datacheck** analysis (simulation 2).
tiretransfer_full_footprint.inp	Full three-dimensional model, complete analysis (simulation 2).
tiretransfer_symm_drum.inp	Partial three-dimensional model of a tire in contact with a rigid drum.
tiretransfer_full_drum.inp	Full three-dimensional model of a tire in contact with a rigid drum.
tiretransfer_node.inp	Nodal coordinates for the axisymmetric models.
tiretransfer_axi_half_ml.inp	Axisymmetric model, inflation analysis (simulation 1) with Marlow hyperelastic model.
tiretransfer_symmetric_ml.inp	Partial three-dimensional model, footprint analysis (simulation 1) with Marlow hyperelastic model.
tiretransfer_full_ml.inp	Full three-dimensional model, final equilibrium analysis (simulation 1) with Marlow hyperelastic model.

Tables

Table 1: Comparison of normalized CPU times for the footprint analysis (normalized with respect to the total "No results transfer" analysis).

	Use results transfer and symmetry conditions	No results transfer
Inflation	$0.005^{(a)}+0.040^{(b)}$	$0.347^{(e)}$
Footprint	$0.265^{(c)}+0.058^{(d)}$	$0.653^{(e)}$

	Use results transfer and symmetry conditions	No results transfer
Total	0.368	1.0
(a) axisymmetric model		
(b) equilibrium step in partial three-dimensional model		
(c) footprint analysis in partial three-dimensional model		
(d) equilibrium step in full three-dimensional model		
(e) full three-dimensional model		

Figures

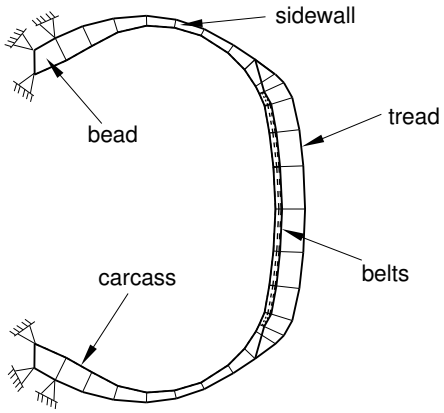


Figure 1: Tire cross-section.

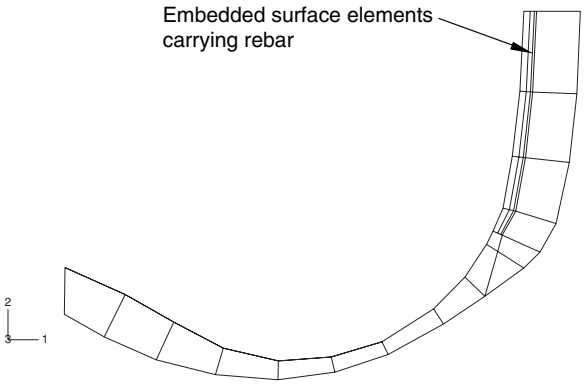


Figure 2: Axisymmetric tire mesh.

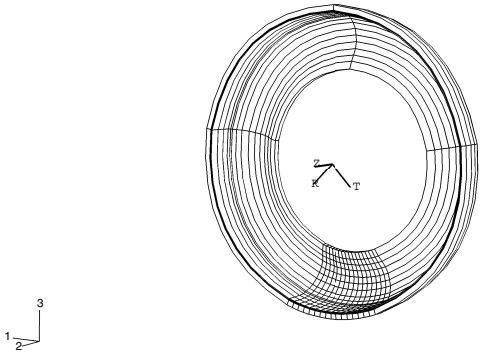


Figure 3: Partial three-dimensional tire mesh.

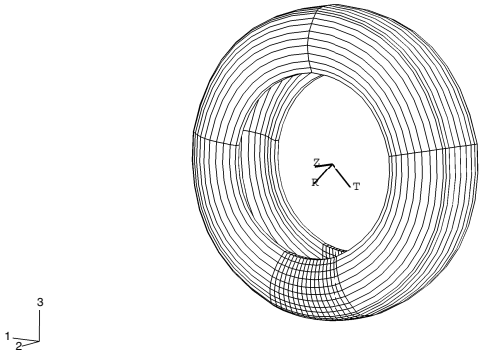


Figure 4: Full three-dimensional tire mesh.

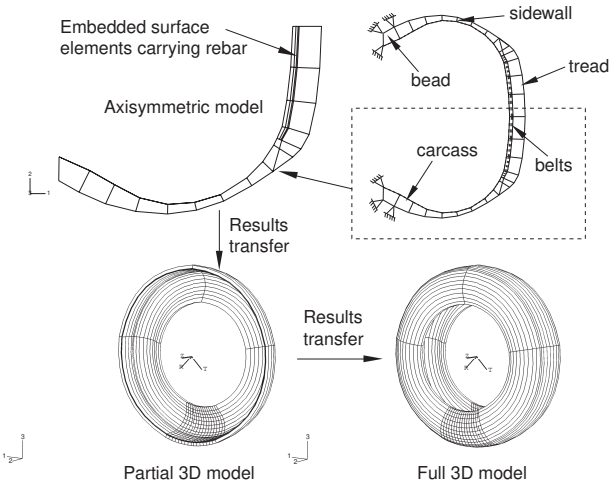


Figure 5: Results transfer analysis sequence.

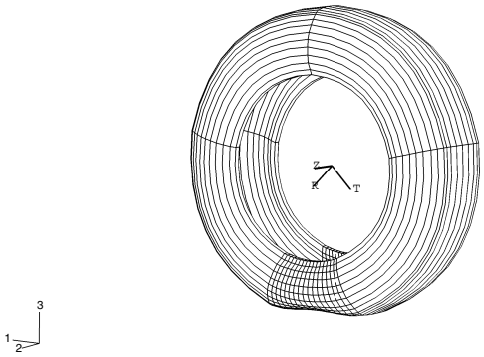


Figure 6: Deformed three-dimensional tire (deformations scaled by a factor of 2).

Steady-state rolling analysis of a tire

Products: Abaqus/Standard

References:

- *Steady-State Transport Analysis*

Overview

This example illustrates the use of steady-state transport in Abaqus to model the steady-state dynamic interaction between a rolling tire and a rigid surface.

A steady-state transport analysis uses a moving reference frame in which rigid body rotation is described in an Eulerian manner and the deformation is described in a Lagrangian manner. This kinematic description converts the steady moving contact problem into a pure spatially dependent simulation. Thus, the mesh need be refined only in the contact region—the steady motion transports the material through the mesh. Frictional effects, inertia effects, and history effects in the material can all be accounted for in a steady-state transport analysis.

The purpose of this analysis is to obtain free rolling equilibrium solutions of a 175 SR14 tire traveling at a ground velocity of 10.0 km/h (2.7778 m/s) at different slip angles on a flat rigid surface. The slip angle is the angle between the direction of travel and the plane normal to the axle of the tire. Straight line rolling occurs at a 0.0° slip angle. For comparison purposes we also consider an analysis of the tire spinning at a fixed position on a 1.5 m diameter rigid drum. The drum rotates at an angular velocity of 3.7 rad/s, so that a point on the surface of the drum travels with an instantaneous velocity of 10.0 km/h (2.7778 m/s). Another case presented examines the camber thrust arising from camber applied to a tire at free rolling conditions. This also enables us to calculate a camber thrust stiffness.

An equilibrium solution for the rolling tire problem that has zero torque, T, applied around the axle is referred to as a free rolling solution. An equilibrium solution with a nonzero torque is referred to as either a traction or a braking solution depending on the sense of T. Braking occurs when the angular velocity of the tire is small enough such that some or all of the contact points between the tire and the road are slipping and the resultant torque on the tire acts in an opposite sense from the angular velocity of the free rolling solution. Similarly, traction occurs when the angular velocity of the tire is large enough such that some or all of the contact points between the tire and the road are slipping and the resultant torque on the tire acts in the same sense as the angular velocity of the free rolling solution. Full braking or traction occurs when all the contact points between the tire and the road are slipping.

A wheel in free rolling, traction, or braking will spin at different angular velocities, $\boldsymbol{\omega}$, for the same ground velocity, $\mathbf{v_0}$. Usually the combination of $\boldsymbol{\omega}$ and $\mathbf{v_0}$ that results in free rolling is not known in advance. Since the steady-state transport analysis capability requires that both the rotational spinning velocity, $\boldsymbol{\omega}$, and the traveling ground velocity, $\mathbf{v_0}$, be prescribed, the free rolling solution must be found in an indirect manner. One such indirect approach is illustrated in this example. An alternate approach involves controlling the rotational spinning velocity using user subroutine *UMOTION* while monitoring the progress of the solution through a second user subroutine *URDFIL*. The *URDFIL* subroutine is used to obtain an estimate of the free rolling solution based on the values of the torque at the rim at the end of each increment. This approach is also illustrated in this example.

A finite element analysis of this problem, together with experimental results, has been published by Koishi et al. (1997).

Problem description and model definition

A description of the tire and finite element model is given in *Symmetric results transfer for a static tire analysis*. To take into account the effect of the skew symmetry of the actual tire in the dynamic analysis, the steady-state rolling analysis is performed on the full three-dimensional model, also referred to as the full model. Inertia effects are ignored since the rolling speed is low ($v_0 =$ 10 km/h).

As stated earlier, the steady-state transport capability in Abaqus uses a mixed Eulerian/Lagrangian approach in which, to an observer in the moving reference frame, the material appears to flow through a stationary mesh. The paths that the material points follow through the mesh are referred to as streamlines and must be computed before a steady-state transport analysis can be performed. As discussed in *Symmetric results transfer for a static tire analysis*, the streamlines needed for the steady-state transport analyses in this example are computed using the revolve functionality for symmetric model generation.

The incompressible hyperelastic material used to model the rubber in this example includes a time-domain viscoelastic component, which is specified directly using the Prony series parameters. A simple 1-term Prony series model is used. For an incompressible material a 1-term Prony series in Abaqus is defined by providing a single value for the shear relaxation modulus ratio, $\bar{g}_1^P$, and its associated relaxation time, τ_1. In this example $\bar{g}_1^P = 0.3$ and $\tau_1 = 0.1$. The viscoelastic—that is, material history—effects are included in a steady-state transport step unless you are investigating the long-term behavior of the material. See *Time Domain Viscoelasticity* for a more detailed discussion of modeling time-domain viscoelasticity in Abaqus.

Loading

As discussed in *Symmetric results transfer for a static tire analysis*, it is recommended that the footprint analyses be obtained with a friction coefficient of zero (so that no frictional forces are transmitted across the contact surface). The frictional stresses for a rolling tire are very different from the frictional stresses in a stationary tire, even if the tire is rolling at very low speed; therefore, discontinuities may arise in the solution between the last static analysis and the first steady-state transport analysis. Furthermore, varying the friction coefficient from zero at the beginning of the steady-state transport step to its final value at the end of the steady-state transport step ensures that the changes in frictional forces reduce with smaller load increments. This is important if Abaqus must take a smaller load increment to overcome convergence difficulties while trying to obtain the steady-state rolling solution.

Once the static footprint solution for the tire has been computed, the steady-state rolling contact problem can be solved using steady-state transport. The objective of the first simulation in this example is to obtain the straight line, steady-state rolling solutions, including full braking and full traction, at different spinning velocities. We also compute the straight line, free rolling solution. In the second simulation free rolling solutions at different slip angles are computed. In the first and second simulations material history effects are ignored by specifying that the long-term behavior of the material is to be used. The third simulation repeats a portion of the straight line, steady-state rolling analysis from the first simulation; however, material history effects are included if you do not specify a long-term material response. A steady ground velocity of 10.0 km/h is maintained for all the simulations. The objective of the fourth simulation is to obtain the free rolling solution of the tire in contact with a 1.5 m rigid drum rotating at 3.7 rad/s.

In the first simulation (*rollingtire_brake_trac.inp*) the full braking solution is obtained in the first steady-state transport step by setting the friction coefficient, μ, to its final value of 1.0 by changing friction properties and applying the translational ground velocity together with a spinning angular velocity that will result in full braking. An estimate of the angular velocity corresponding to full braking is obtained as follows. A free rolling tire generally travels farther in one revolution than determined by its center height, *H*, but less than determined by the free tire radius. In this example the free radius is 316.2 mm and the vertical deflection is approximately 20.0 mm, so $H =$ 296.2 mm. Using the free radius and the effective height, it is estimated that free rolling occurs at an angular velocity between $\omega =$ 8.78 rad/s and $\omega =$ 9.38 rad/s. Smaller angular velocities would result in braking, and larger angular velocities would result in traction. We use an angular velocity $\omega =$ 8.0 rad/s to ensure that the solution in the first steady-state transport step is a full braking solution (all contact points are slipping, so the magnitude of the total frictional force across the contact surface is μN).

In the second steady-state transport analysis step of the full model, the angular velocity is increased gradually to $\omega =$ 10.0 rad/s while the ground velocity is held constant. The solution at each load increment is a steady-state solution to the loads acting on the structure at that instant so that a series of steady-state solutions between full

braking and full traction is obtained. This analysis provides us with a preliminary estimate of the free rolling velocity. The second simulation (rollingtire_trac_res.inp) performs a refined search around the first estimate of free rolling conditions.

In the third simulation (*rollingtire_slipangles.inp*) the free rolling solutions at different slip angles are computed. The slip angle, θ, is the angle between the direction of travel and the plane normal to the axle of the tire. In the first step the straight line, free rolling solution from the first simulation is brought into equilibrium. This step is followed by a steady-state transport step where the slip angle is gradually increased from $\theta = 0.0°$ at the beginning of the step to $\theta = 3.0°$ at the end of the step, so a series of steady-state solutions at different slip angles is obtained. This is accomplished by prescribing a traveling velocity vector with components $v_x = v_0 \cos\theta$ and $v_y = v_0 \sin\theta$ in the prescribed translational motion, where $\theta = 0.0°$ in the first steady-state transport step and $\theta = 3.0°$ at the end of the second steady-state transport step.

The fourth simulation (*rollingtire_materialhistory.inp*) includes a series of steady-state solutions between full braking and full traction in which the material history effects are included.

The fifth simulation (rollingtire_camber.inp) analyzes the effect of camber angle on the lateral thrust at the contact patch under free rolling conditions.

The final simulation in this example (*rollingtire_drum.inp*) considers a tire in contact with a rigid rotating drum. The loading sequence is similar to the loading sequence used in the first simulation. However, in this simulation the translational velocity of the tire is zero, and a rotational angular velocity is applied to the reference node of the rigid drum. Since a prescribed load is applied to the rigid drum reference node to establish contact between the tire and drum, the rotation axis of the drum is unknown prior to the analysis. Abaqus automatically updates the rotation axis to its current position if the angular velocity is defined. The rotational velocity of the rigid surface can also be defined. In that case the position and orientation of the axis of revolution must be defined in the steady-state configuration and, therefore, must be known prior to the analysis. The position and orientation of the axis are applied at the beginning of the step and remain fixed during the step. When the drum radius is large compared to the axle displacement, as in this example, it is a reasonable approximation to define the axle in the original configuration without significantly affecting the accuracy of the results.

Results and discussion

Figure 1 and *Figure 2* show the reaction force parallel to the ground (referred to as rolling resistance) and the torque, *T*, on the tire axle at different spinning velocities. The figures compare the solutions obtained for a tire rolling on a flat rigid surface with those for a tire in contact with a rotating drum. The figures show that straight line free rolling, $T = 0.0$, occurs at a spinning velocity of approximately 9.0 rad/s. Full braking occurs at spinning velocities smaller than 8.0 rad/s, and full traction occurs at velocities larger than 9.75 rad/s. At these spinning velocities all contact points are slipping, and the rolling resistance reaches the limiting value μN.

Figure 3 and *Figure 4* show shear stress along the centerline of the tire surface in the free rolling and full traction states for the case where the tire is rolling along a flat rigid surface. The distance along the centerline is measured as an angle with respect to a plane parallel to the ground passing through the tire axle. The dashed line is the maximum or limiting shear stress, μp, that can be transmitted across the surface, where p is the contact pressure. The figures show that all contact points are slipping during full traction. During free rolling all points stick.

A better approximation to the angular velocity that corresponds to free rolling can be made by using the results generated by *rollingtire_brake_trac.inp* to refine the search about an angular velocity of 9.0 rad/s. The file *rollingtire_trac_res.inp* restarts the previous straight line rolling analysis from Step 3, Increment 8 (corresponding to an angular velocity of 8.938 rad/s) and performs a refined search up to 9.04 rad/s. *Figure 5* shows the torque, *T*, on the tire axle computed in the refined search, which leads to a more precise value for the free rolling angular velocity of approximately 9.022 rad/s. This result is used for the model where the free rolling solutions at different slip angles are computed.

Figure 6 shows the transverse force (force along the tire axle) measured at different slip angles. The figure compares the steady-state transport analysis prediction with the result obtained from a pure Lagrangian analysis. The Lagrangian solution is obtained by performing an explicit transient analysis using Abaqus/Explicit (discussed in *Import of a steady-state rolling tire*). With this analysis technique a prescribed constant traveling velocity is applied to the tire, which is free to roll along the rigid surface. Since more than one revolution is necessary to obtain a steady-state configuration, fine meshing is required along the full circumference; hence, the Lagrangian solution is much more costly than the steady-state solutions shown in this example. The figure shows good agreement between the results obtained from the two analysis techniques.

Figure 7 compares the free rolling solutions with and without material history effects included. The solid lines in the diagram represent the rolling resistance (force parallel to the ground along the traveling direction); and the broken lines, the torque (normalized with respect to the free radius) on the axle. The figure shows that free rolling occurs at a lower angular velocity when history effects are included. The influence of material history effects on a steady-state rolling solution is discussed in detail in *Steady-state spinning of a disk in contact with a foundation*.

Figure 8 shows the camber thrust as a function of camber angle. The lateral force at zero camber and zero slip is referred to as ply-steer and arises due to the asymmetry in the tire caused by the separation of the belts by the interply distance. Discretization of the contact patch is responsible for the nonsmooth nature of the curve, and an overall camber stiffness of 44 N/degree is reasonably close to expected levels.

Figure 9 shows the torque on the rim as the rotational velocity is applied with user subroutine *UMOTION*, based on the free rolling velocity predicted in user subroutine *URDFIL*. As the torque on the rim falls to within a user-specified tolerance of zero torque, the rotational velocity is held fixed and the step completed. Initially, when the free rolling rotational velocity estimates are beyond a user-specified tolerance of the current rotational velocity, only small increments of rotational velocity are applied. The message file contains information on the estimates of free rolling velocity and the incrementation as the solution progresses. The angular velocity thus found for free rolling conditions is 9.026 rad/s.

Acknowledgments

SIMULIA gratefully acknowledges Hankook Tire and Yokohama Rubber Company for their cooperation in developing the steady-state transport capability used in this example. SIMULIA thanks Dr. Koishi of Yokohama Rubber Company for supplying the geometry and material properties used in this example.

Input files

rollingtire_brake_trac.inp	Three-dimensional full model for the full braking and traction analyses.
rollingtire_trac_res.inp	Three-dimensional full model for the refined braking and traction analyses.
rollingtire_slipangles.inp	Three-dimensional full model for the slip angle analysis.
rollingtire_camber.inp	Three-dimensional full model for the camber analysis.
rollingtire_materialhistory.inp	Three-dimensional full model with material history effects.
rollingtire_drum.inp	Three-dimensional full model for the simulation of rolling on a rigid drum.
rollingtire_freeroll.inp	Three-dimensional full model for the direct approach to finding the free rolling solution.
rollingtire_freeroll.f	User subroutine file used to find the free rolling solution.

References

- Koishi, M., K. Kabe, and M. Shiratori, “Tire Cornering Simulation using Explicit Finite Element Analysis Code,” *16th annual conference of the Tire Society at the University of Akron*, 1997.

Figures

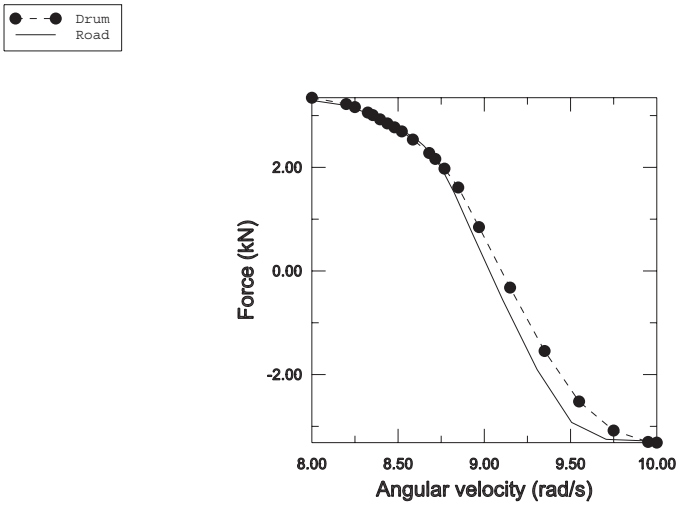


Figure 1: Rolling resistance at different angular velocities.

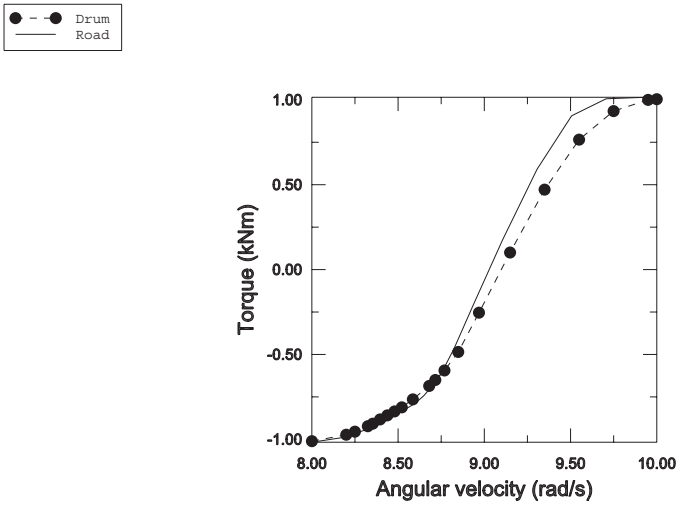


Figure 2: Torque at different angular velocities.

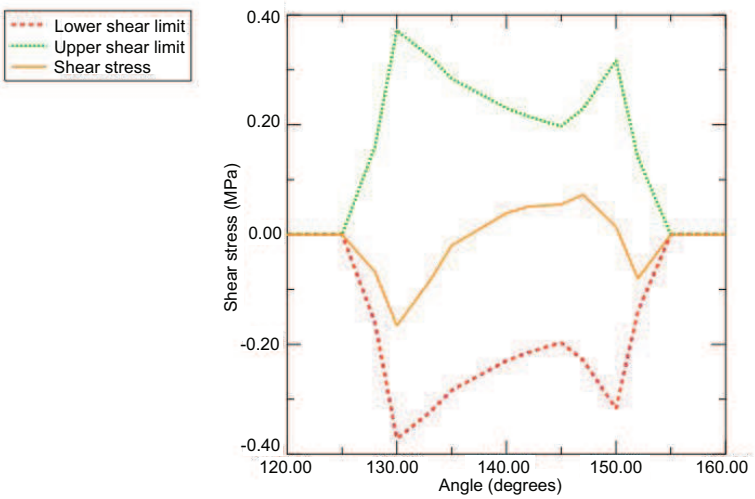


Figure 3: Shear stress along tire center (free rolling).

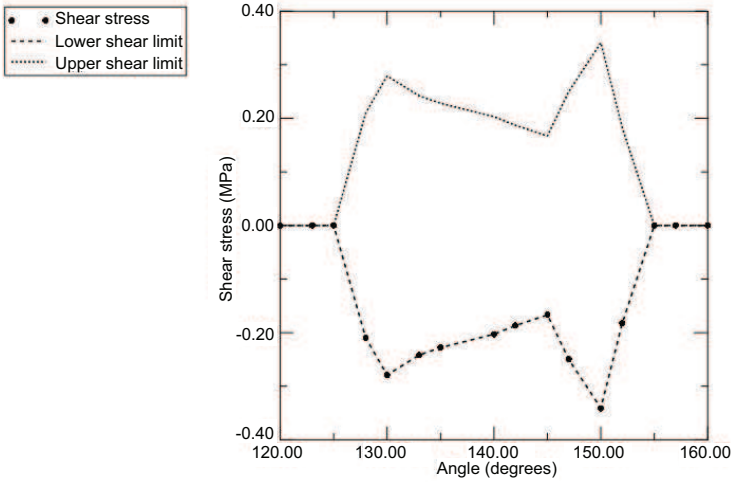


Figure 4: Shear stress along tire center (full traction).

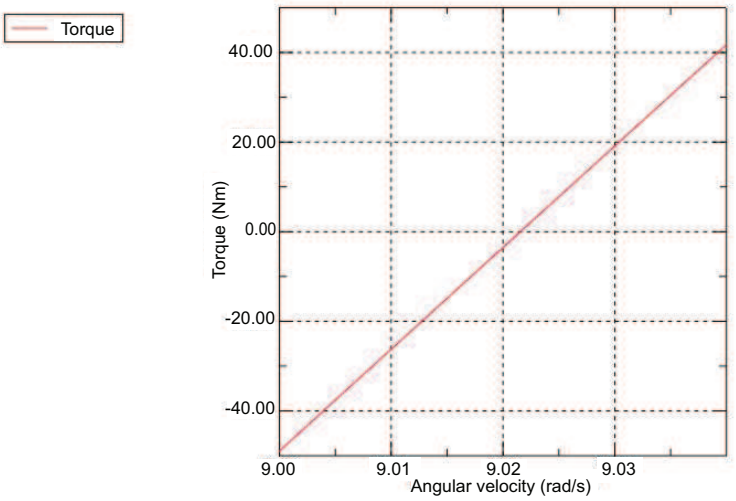


Figure 5: Torque at different angular velocities (refined search).

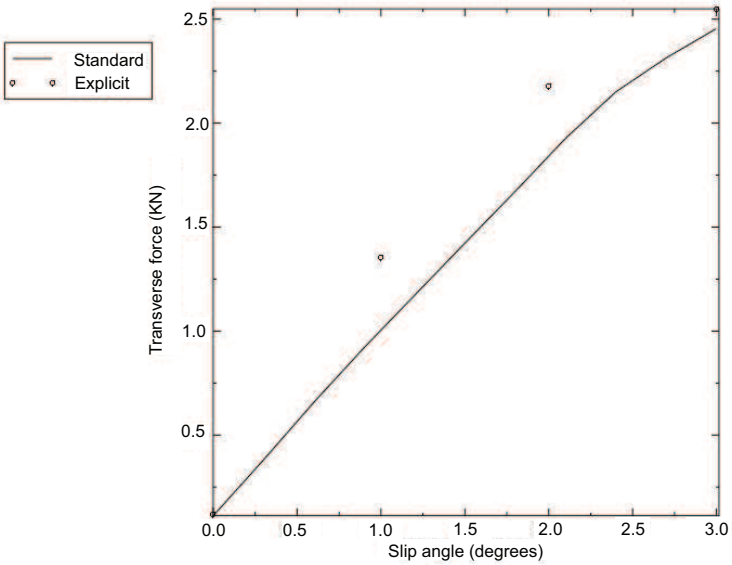


Figure 6: Transverse force as a function of slip angle.

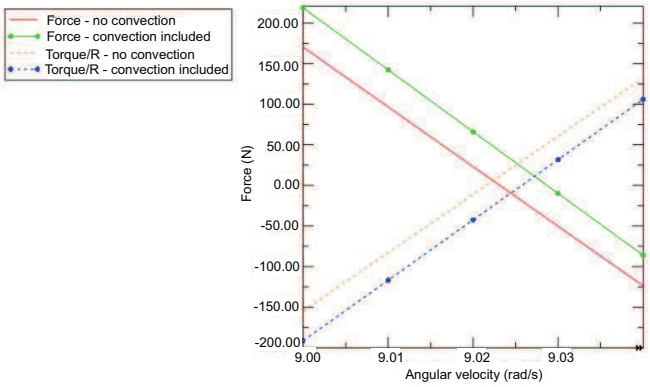


Figure 7: Rolling resistance and normalized torque as a function of angular velocity (R=0.3162 m).

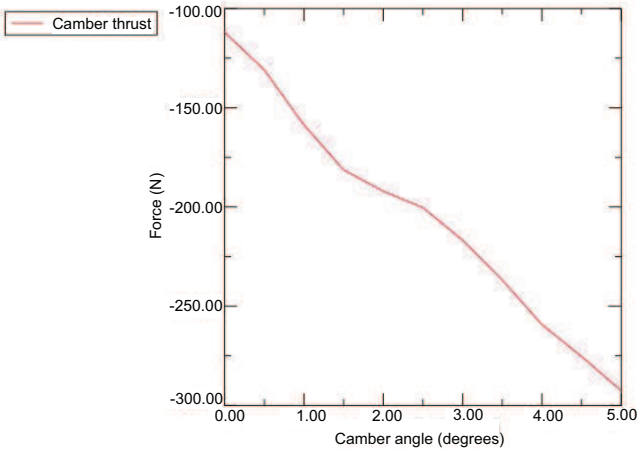


Figure 8: Camber thrust as a function of camber angle.

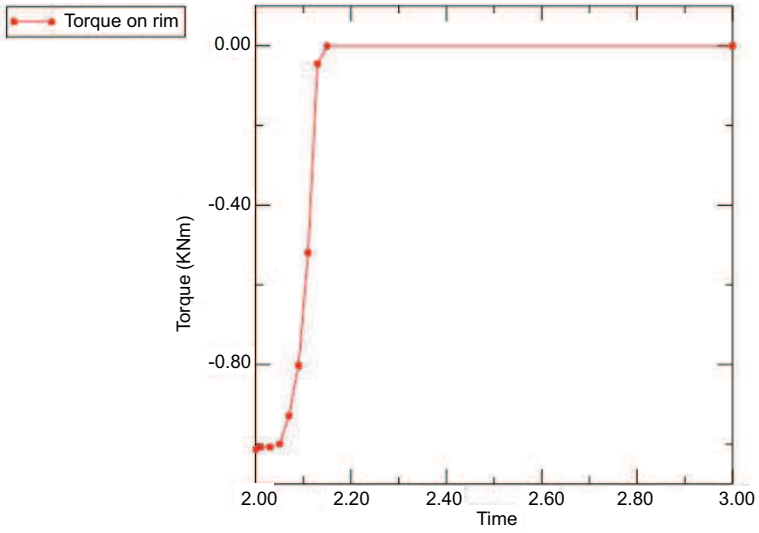


Figure 9: Torque on the rim for the direct approach to finding the free rolling solution.

Subspace-based steady-state dynamic tire analysis

Products: Abaqus/Standard

Overview

This example illustrates the use of subspace-based steady-state dynamic analysis to model the frequency response of a tire about a static footprint solution.

Subspace-based steady-state dynamic analysis (*Subspace-Based Steady-State Dynamic Analysis*) is an analysis procedure that can be used to calculate the steady-state dynamic response of a system subjected to harmonic excitation. It does so by the direct solution of the steady-state dynamic equations projected onto a reduced-dimensional subspace spanned by a set of eigenmodes of the undamped system. If the dimension of the subspace is small compared to the dimension of the original problem (i.e., if a relatively small number of eigenmodes is used), the subspace method can offer a very cost-effective alternative to a direct-solution steady-state analysis.

The purpose of this analysis is to obtain the frequency response of a 175 SR14 tire subjected to a harmonic load excitation about the footprint solution discussed in *Symmetric results transfer for a static tire analysis*. Symmetric results transfer and symmetric model generation are used to generate the footprint solution, which serves as the base state in the steady-state dynamics calculations.

Problem description

A description of the tire being modeled is given in *Symmetric results transfer for a static tire analysis*. In this example we exploit the symmetry in the tire model and utilize the results transfer capability in Abaqus to compute the footprint solution for the full three-dimensional model in a manner identical to that discussed in *Symmetric results transfer for a static tire analysis*.

Once the footprint solution has been computed, several steady-state dynamic steps are performed. Both direct-solution steady-state dynamic analysis and subspace-based steady-state dynamic analysis are used. Besides being used to validate the subspace projection results, the direct steady-state procedure allows us to demonstrate the computational advantage afforded by the subspace projection capability in Abaqus.

Model definition

The model used in this analysis is essentially identical to that used in the first simulation discussed in *Symmetric results transfer for a static tire analysis*, with CGAX4H and CGAX3H elements used in the axisymmetric model and rebar in the continuum elements for the belts and carcass. In addition, instead of using a nonuniform discretization about the circumference, the uniform discretization shown in *Figure 1* is used.

The incompressible hyperelastic material used to model the rubber includes a viscoelastic component described by a 1-term Prony series of the dimensionless shear relaxation modulus:

$$g_R(t) = 1 - \overline{g}_1^P \left(1 - e^{-t/\tau_1}\right),$$

with relaxation coefficient $\overline{g}_1^P = 0.3$ and relaxation time $\tau_1 = 0.1$. Since the material is incompressible, the volumetric behavior is time independent. This time domain description of the material must be written in the frequency domain to perform a steady-state dynamic analysis. By applying a Fourier transform, the expressions for the time-dependent shear modulus can be written in the frequency domain as follows:

$$G_s(\omega) = G_0\left[1 - g_1^P\right] + G_0 \frac{g_1^P \tau_1^2 \omega^2}{1+\tau_1^2\omega^2},$$

$$G_\ell(\omega) = G_0 \frac{g_1^P \tau_1 \omega}{1+\tau_1^2\omega^2},$$

where $G_s(\omega)$ is the storage modulus, $G_\ell(\omega)$ is the loss modulus, and ω is the angular frequency. Abaqus will perform the conversion from time domain to frequency domain automatically if the Prony series parameters are defined. See *Time Domain Viscoelasticity* for a more detailed discussion on frequency domain viscoelasticity.

Loading

The loading sequence for computing the footprint solution is identical to that discussed in *Symmetric results transfer for a static tire analysis*, with the axisymmetric model contained in *tiredynamic_axi_half.inp*, the partial three-dimensional model in *tiredynamic_symmetric.inp*, and the full three-dimensional model in *tiredynamic_freqresp.inp*. Since geometric nonlinearity is accounted for in the static steps used in computing the footprint solution, the steady-state dynamic analyses, which are linear perturbation procedures, are performed about the nonlinear deformed shape of the footprint solution.

The first frequency response analyses of the tire are performed using the subspace-based steady-state dynamic analysis. The excitation is due to a harmonic vertical load of 200 N, which is applied to the analytical rigid surface through its reference node. The frequency is swept from 80 Hz to 130 Hz. The rim of the tire is held fixed throughout the analysis. Prior to the subspace analysis being performed, the eigenmodes that are used for the subspace projection are computed in an eigenfrequency extraction step. In the frequency step the first 20 eigenpairs are extracted, for which the computed eigenvalues range from 50 to 185 Hz.

The accuracy of the subspace analysis can be improved by including some of the stiffness associated with frequency-dependent material properties—i.e., viscoelasticity—in the eigenmode extraction step. In general, if the material response does not vary significantly over the frequency range of interest, the frequency extraction can be set to evaluate properties at the frequency in the center of the frequency span. Otherwise, more accurate results will be obtained by running several separate frequency analyses over smaller frequency ranges. In this example a single frequency sweep is performed to evaluate properties at 105 Hz.

The main advantage that the subspace projection method offers over mode-based techniques (*Mode-Based Steady-State Dynamic Analysis*) is that it allows frequency-dependent material properties, such as viscoelasticity, to be included directly in the analysis. However, there is a cost involved in assembling the projected equations, and this cost must be taken into account when deciding between a subspace solution and a direct solution. Abaqus offers four methods to control how often the projected subspace equations are recomputed. These methods are computing the projected equations for every frequency in the analysis, recomputing projected equations only at the eigenfrequencies, recomputing projected equations when the stiffness and/or damping properties have changed by a user-specified percentage, and computing the projected equations only once at the center of the frequency range specified in the steady-state dynamic step. Computing the projected equations for every frequency is, in general, the most accurate option; however, the computational overhead associated with recomputing the projected equations at every frequency can significantly reduce the cost benefit of the subspace method versus a direct solution. Computing the projected equations only once at the center of the frequency range is the most inexpensive choice, but it should be chosen only when the material properties do not depend strongly on frequency. In general, the accuracy and cost associated with the four subspace projection methods are strongly problem dependent. In this example problem the results and computational expense for all four subspace projection methods are discussed.

The results from the various subspace analyses are compared to the results from a direct-solution steady-state dynamic analysis.

Results and discussion

Each of the subspace analyses utilizes all 20 modes extracted in the eigenfrequency extraction step. *Figure 2* shows the frequency response plots of the vertical displacements of the road's reference node for the direct solution along with the four subspace solutions using each of the subspace projection methods discussed above. Similarly, *Figure 3* shows the frequency response plots of the horizontal displacement of a node on the tire's

sidewall for the same five analyses. As illustrated in *Figure 2* and *Figure 3*, all four of the subspace projection methods yield almost identical solutions; except for small discrepancies in the vertical displacements at 92 and 120 Hz, the subspace projection solutions closely match the direct solution as well. Timing results shown in *Table 1* show that the subspace projection method results in savings in CPU time versus the direct solution.

Input files

tiredynamic_axi_half.inp	Axisymmetric model, inflation analysis.
tiredynamic_symmetric.inp	Partial three-dimensional model, footprint analysis.
tiredynamic_freqresp.inp	Full three-dimensional model, steady-state dynamic analyses.
tiredynamic_axi_half_ml.inp	Axisymmetric model, inflation analysis with Marlow hyperelastic model.
tiredynamic_symmetric_ml.inp	Partial three-dimensional model, footprint analysis with Marlow hyperelastic model.
tiredynamic_freqresp_ml.inp	Full three-dimensional model, steady-state dynamic analyses with Marlow hyperelastic model.
tiretransfer_node.inp	Nodal coordinates for axisymmetric model.

Tables

Table 1: Comparison of normalized CPU times (normalized with respect to the direct-solution analysis) for the frequency sweep from 80 Hz to 130 Hz and the eigenfrequency extraction step.

Direct-solution analysis method	Normalized CPU time
Subspace (projection at all frequencies)	0.89
Subspace (projection at eigenfrequencies)	0.54
Subspace (projection when properties change)	0.49
Subspace (projection computed once)	0.36
Direct	1.0
Eigenfrequency extraction	0.073

Figures

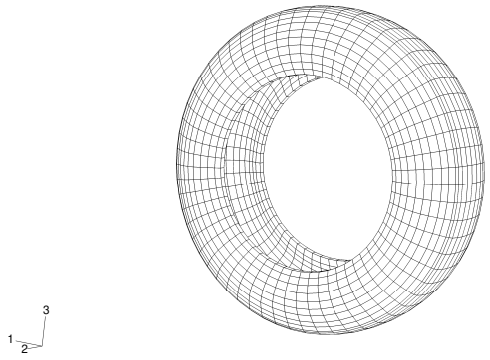


Figure 1: Uniform three-dimensional tire mesh.

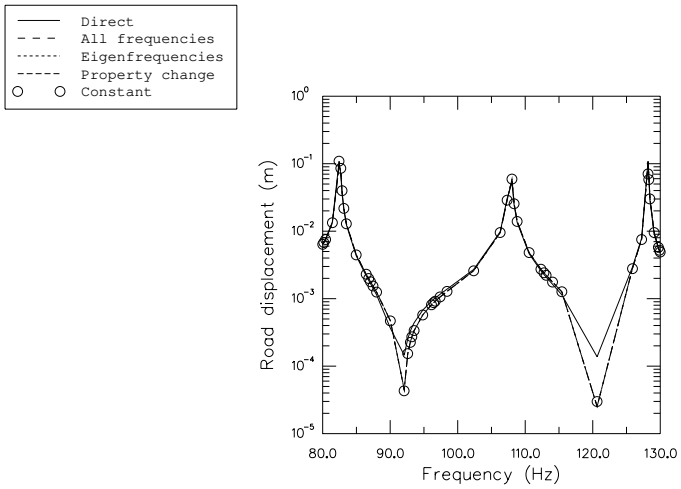


Figure 2: Frequency response of the vertical road displacement due to a vertical harmonic point load of 200 N applied to the reference node.

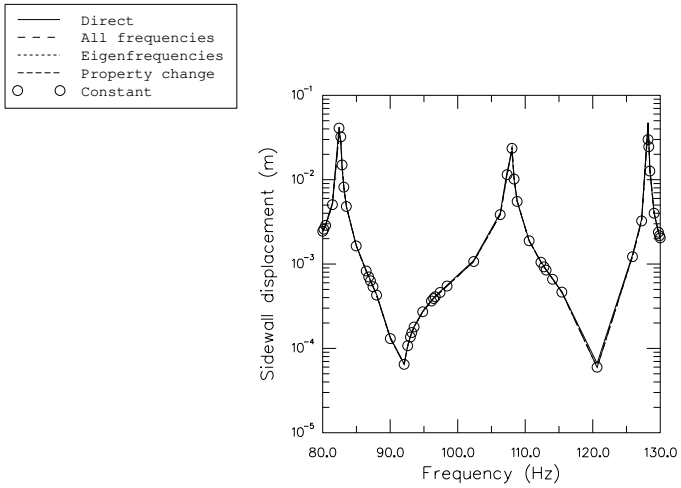


Figure 3: Frequency response of the horizontal sidewall displacement due to a vertical harmonic point load of 200 N applied to the reference node.

Steady-state dynamic analysis of a tire substructure

Products: Abaqus/Standard

References:

- *Generating Substructures*

Overview

This example illustrates the use of the substructuring capability in Abaqus to create a substructure from a tire under inflation and footprint loading.

Use of tire substructures is often seen in vehicle dynamic analyses where substantial cost savings are made using substructures instead of the whole tire model. Since tires behave very nonlinearly, it is essential that the change in response due to preloads is built into the substructure. Here the substructure must be generated in a preloaded state. Some special considerations for creating substructures with preloads involving contact are also discussed.

Problem description and model definition

A description of the tire model used is given in *Symmetric results transfer for a static tire analysis*. In this problem inflation and fooptprint preloads are applied in a series of general analysis steps identical to *Symmetric results transfer for a static tire analysis*. Symmetric model generation and symmetric results transfer are used to exploit the symmetric nature of the structure and loading. Nodes in the bead area are tied to the rigid body representing the rim.

The substructure's retained nodes include the rim node and the road node that is the reference node of the rigid body representing the road surface. To enhance the dynamic response of the substructure, these interfacial degrees of freedom are augmented with generalized degrees of freedom associated with the first 20 fixed interface eigenmodes. Depending on the nature of the loading, it may be necessary to increase the number of generalized degrees of freedom to cover a sufficient range of frequencies. The extra cost incurred due to the addition of the extra frequency extraction step is offset by the enhanced dynamic response of the substructure.

Loading

An inflation load of 200 kPa is applied in the axisymmetric half-tire model contained in *substructtire_axi_half.inp*. This is followed by a footprint load of 1650 N applied to the three-dimensional half-tire model given in *substructtire_symmetric.inp*; and, subsequently, results are transferred to the full tire model with the complete footprint load of 3300 N. All of these steps are run with the NLGEOM=YES parameter, so all preload effects including stress stiffening are taken into account when the substructure is generated.

To enhance the dynamic response of the substructure, several restrained eigenmodes are included as generalized degrees of freedom. These restrained eigenmodes are obtained from an eigenfrequency extraction step with all the retained degrees of freedom restrained. In this example the first 20 eigenmodes are computed. With one road node with six degrees of freedom, one rim node with six degrees of freedom, and 20 generalized degrees of freedom, the substructure has 32 degrees of freedom.

At the usage level the steady-state response of the substructure to harmonic footprint loading is analyzed over a range of frequencies from 40 to 130 Hz. Structural damping is applied in the material definitions of the finite element model to obtain finite displacements at the resonant frequencies. The generated condensed structural damping operator is used to apply damping in the substructure usage analysis.

Results and discussion

The results for the frequency sweep are shown in *Figure 1*, which compares the magnitude of the vertical displacement at the road node of the substructure to the response of the entire tire model. The substructure captures all resonances in the original finite element tire model. The baseline results for the finite element tire model were calculated using the direct and mode based steady-state dynamic solvers. The same two solvers were used to calculate the substructure response. To improve accuracy of the mode-based analysis the modal subspace was augmented with a residual mode. Results for the finite element tire model obtained using the mode-based steady-state dynamic analysis and results for the model using the substructure have the same level of accuracy. That was expected because dynamic substructures are modal reductions of the finite element models.

Input files

substructtire_axi_half.inp	Axisymmetric model, inflation analysis.
substructtire_symmetric.inp	Partial three-dimensional model, footprint analysis.
substructtire_full.inp	Full three-dimensional model, final equilibrium analysis.
substructtire_generate.inp	Substructure generation analysis.
substructtire_subuse_ssd.inp	Usage level model with steady-state dynamics analysis.
substructtire_femodel_ssd.inp	Steady-state dynamic analysis of the original finite element model of the tire.

Figures

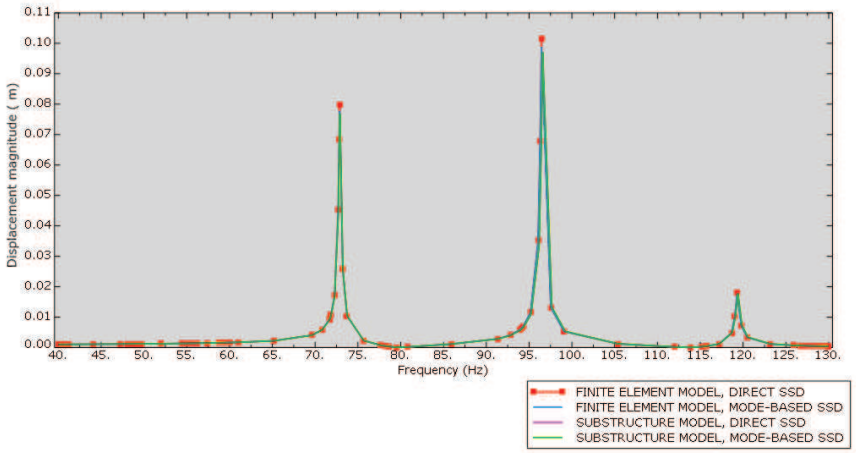


Figure 1: Vertical response of the road node due to unit vertical harmonic load.

Coupled acoustic-structural analysis of a tire filled with air

Products: Abaqus/Standard

Overview

This example studies the acoustic response of a tire and air cavity subjected to an inflation pressure and footprint load.

The air cavity resonance in a tire is often a significant contributor to the vehicle interior noise, particularly when the resonance of the tire couples with the cavity resonance. This example demonstrates how the ALE adaptive mesh domain can be used to update an acoustic mesh when structural deformation causes significant changes to the geometry of the acoustic domain. The effect of rolling motion is ignored; however, the rolling speed can have a significant influence on the coupled acoustic-structural response. This effect is investigated in detail in *Dynamic analysis of an air-filled tire with rolling transport effects*.

The acoustic elements in Abaqus model only small-strain dilatational behavior through pressure degrees of freedom and, therefore, cannot model the deformation of the fluid when the bounding structure undergoes large deformation. Abaqus solves the problem of computing the current configuration of the acoustic domain by periodically creating a new acoustic mesh. The new mesh uses the same topology (elements and connectivity) throughout the simulation, but the nodal locations are adjusted periodically so that the deformation of the structural-acoustic boundary does not lead to severe distortion of the acoustic elements. The calculation of the updated nodal locations is based on adaptive mesh smoothing.

Problem description

A detailed description of the tire model is provided in *Symmetric results transfer for a static tire analysis*. We model the rubber as an incompressible hyperelastic material and include damping in the structure by specifying a 1-term Prony series viscoelastic material model with a relaxation modulus of 0.3 and relaxation time of 0.1 s. We define the model in the frequency domain with a direct specification of the Prony series parameters since we want to include the damping effects in a steady-state dynamic simulation.

The air cavity in the model is defined as the space enclosed between the interior surface of the tire and a cylindrical surface of the same diameter as the diameter of the bead. A segment of the tire is shown in *Figure 1*. The values of the bulk modulus and the density of air are taken to be 426 kPa and 3.6 kg/m^3, respectively, and represent the properties of air at the tire inflation pressure.

The simulation assumes that both the road and rim are rigid. We further assume that the contact between the road and the tire is frictionless during the preloading analyses. However, we use a nonzero friction coefficient in the subsequent coupled acoustic-structural analyses.

Model definition

We use a tire model that is identical to that used in the simulation described in *Symmetric results transfer for a static tire analysis*. The air cavity is discretized using linear acoustic elements and is coupled to the structural mesh using a surface-based tie constraint with the secondary surface defined on the acoustic domain. We model the rigid rim by applying fixed boundary conditions to the nodes on the bead of the tire, while the interaction between the air cavity and rim is modeled by a traction-free surface; i.e., no boundary conditions are prescribed on the surface.

Symmetric model generation and symmetric results transfer, together with a static analysis procedure, are used to generate the preloading solution, which serves as the base state in the subsequent coupled acoustic-structural analyses.

In the first coupled analysis we compute the eigenvalues of the tire and air cavity system. This analysis is followed by a direct-solution and a subspace-based steady-state dynamic analysis in which we obtain the response of the tire-air system subjected to harmonic excitation of the spindle.

A coupled structural-acoustic substructure analysis is performed as well. Frequency-dependent viscoelastic material properties are evaluated at 230 Hz, which is the middle of the frequency range of the steady-state dynamic analysis. The viscoelastic material contributions to the stiffness and structural damping operator are taken into account. They are assumed to be constant with respect to frequency. A substructure is generated using mixed-interface dynamic modes (see *Defining the Retained Nodal Degrees of Freedom*). A retained degree of freedom that is loaded in the steady-state dynamic analysis is not constrained during the eigenmode extraction. Usually this modeling technique provides a more accurate solution than that obtained using the traditional Craig-Bampton substructures with the fixed-interface dynamic modes.

Loading

The loading sequence for computing the footprint solution is identical to that discussed in *Symmetric results transfer for a static tire analysis*. The simulation starts with an axisymmetric model, which includes the mesh for the air cavity. Only half the cross-section is modeled. The inflation pressure is applied to the structure using a static analysis. In this example the application of pressure does not cause significant changes to the geometry of the air cavity, so it is not necessary to update the acoustic mesh. However, we perform adaptive mesh smoothing after the pressure is applied to illustrate that the updated geometry of the acoustic domain is transferred to the three-dimensional model when symmetric results transfer is used.

The axisymmetric analysis is followed by a partial three-dimensional analysis in which the footprint solution is obtained. The footprint load is established over several load increments. The deformation during each load increment causes significant changes to the geometry of the air cavity. We update the acoustic mesh by performing 5 mesh sweeps after each converged structural load increment using an ALE adaptive mesh domain. At the end of this analysis sequence we activate friction between the tire and road by changing the friction properties. This footprint solution, which includes the updated acoustic domain, is transferred to a full three-dimensional model. This model is used to perform the coupled analysis. In the first coupled analysis we extract the eigenvalues of the undamped system, followed by a direct-solution steady-state dynamic analysis in which we apply a harmonic excitation to the reference node of the rigid surface that is used to model the road. We apply two load cases: one in which the excitation is applied normal to the road surface and one in which the excitation is applied parallel to the road surface along the rolling (fore-aft) direction.

We want to compute the response of the coupled system in the frequency range in which we expect the air cavity to contribute to the overall acoustic response. We consider the response near the first eigenfrequency of the air cavity only, which has a wavelength that is equal to the circumference of the tire. Using a value of 344 m/s for the speed of sound and an air cavity radius of 0.240 m (the average of the minimum and maximum radius of the air cavity), we estimate this frequency to be approximately 230 Hz. We extract the eigenvalues and perform the steady-state dynamic analysis in the frequency range between 200 Hz to 260 Hz. In the eigenvalue extraction analysis, stiffness contributions from the frequency-dependent viscoelasticity material model are evaluated at a frequency of 230 Hz. In the direct-solution and subspace-based steady-state dynamic analyses, frequency-dependent contributions to the stiffness and damping are evaluated at every frequency point.

The same inflation and loading steps were used for the substructure analysis. In the substructure analysis, the frequency-dependence of viscoelastic material properties cannot be taken into account. Only a vertical excitation load case is considered.

Results and discussion

Figure 1 shows the updated acoustic mesh near the footprint region. The geometric changes associated with the updated mesh are taken into account in the coupled acoustic-structural analyses.

The eigenvalues of the air cavity, the tire, and the coupled tire-air system are tabulated in *Table 1*. The resonant frequencies of the uncoupled air cavity are computed using the original configuration. We obtain two acoustic

modes at frequencies of 228.58 Hz and 230.17 Hz. These frequencies correspond to two identical modes rotated 90° with respect to each other, as shown in *Figure 2* and *Figure 3*; the magnitudes of the frequencies are different since we have used a nonuniform mesh along the circumferential direction. We refer to the two modes as the fore-aft mode and the vertical mode, respectively. These eigenfrequencies correspond very closely to our original estimate of 230 Hz. The table shows that these eigenfrequencies occur at almost the same magnitude in the coupled system, indicating that the coupling has a very small effect on the acoustic resonance. The difference between the two vertical modes is larger than the difference between the fore-aft modes. This can be attributed to the geometry changes associated with structural loading. The coupling has a much stronger influence on the structural modes than on the acoustic modes, but we expect the coupling to decrease as we move away from the 230 Hz range.

Figure 4 to *Figure 7* show the response of the structure to the spindle excitation. *Figure 4* and *Figure 5* compare the response of the coupled tire-air system to the response of a tire without the air cavity. *Figure 6* and *Figure 7* show the acoustic pressure measured in the crown and side of the air. We draw the following conclusions from these figures. The frequencies at which resonance is predicted by the steady-state dynamic analysis correspond closely to the eigenfrequencies. However, not all the eigenmodes are excited by the spindle excitations. For example, the fore-aft mode is not excited by vertical loading. Similarly, the vertical mode is not excited by fore-aft loading. In addition, only some of the structural modes are excited by the spindle loads, while others are suppressed by material damping. These figures further show that the air cavity resonance has a very strong influence on the behavior of the coupled system and that the structural resonance of the coupled tire-air system occurs at different frequencies than the resonance of the tire without air. As expected, this coupling effect decreases as we move further away from the cavity resonance frequency.

The eigenfrequencies obtained in the substructure analysis are identical to the eigenfrequencies obtained in the equivalent analysis without substructures. The reaction force obtained at the road reference node is also compared to the reaction force at the same node in the equivalent analysis without substructures. As shown in *Figure 8* the results for the two steady-state dynamics steps in the substructure analysis are virtually identical, and they compare well, in general, with the reaction force obtained in the nonsubstructure analysis. The observed small differences are due to modal truncation and the fact that constant material properties are used to generate the substructure.

Input files

tire_acoustic_axi.inp	Axisymmetric model, inflation analysis.
tire_acoustic_rev.inp	Partial three-dimensional model, footprint analysis.
tire_acoustic_refl.inp	Full three-dimensional model, eigenvalue extraction and frequency response analyses.
tire_substracous_gen.inp	Substructure generation analysis.
tire_substracous_dyn.inp	Substructure usage analysis (eigenvalue extraction and frequency response analyses).
tiretransfer_node.inp	Nodal coordinates for the axisymmetric tire mesh.
tire_acoustic_air.inp	Mesh data for the axisymmetric acoustic mesh.

Tables

Table 1: Eigenfrequencies in the 200–260 Hz frequency range.

Mode	Air cavity	Tire	Coupled tire-air
1		202.83	199.06
2		208.95	205.73

Mode	Air cavity	Tire	Coupled tire-air
3		212.55	210.67
4		220.53	218.64
5		221.30	219.24
6	228.58		228.24
7	230.17		232.18
8		233.42	232.24
9		236.49	234.19
10		244.63	243.12
11		254.84	252.84
12		257.56	255.51

Figures

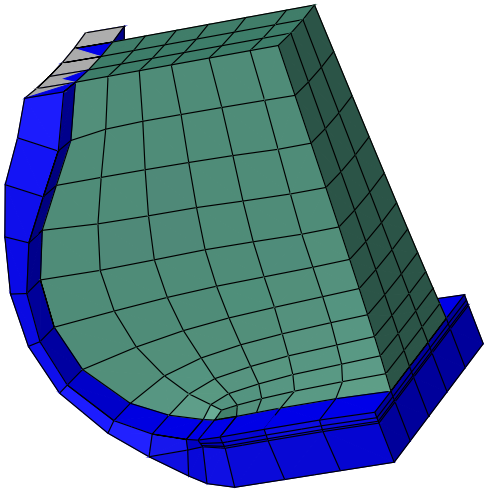


Figure 1: Deformation of tire and air in the footprint region.

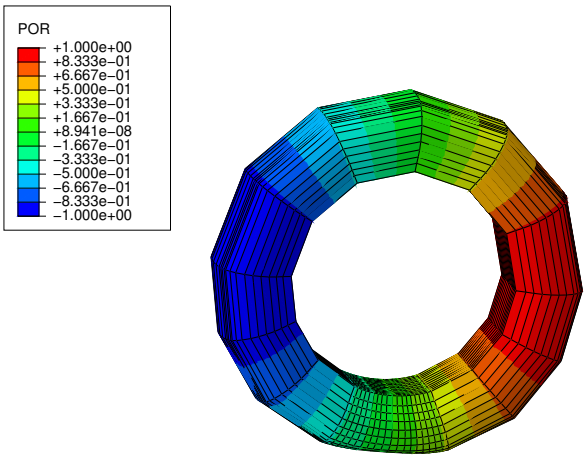


Figure 2: Fore-aft acoustic mode.

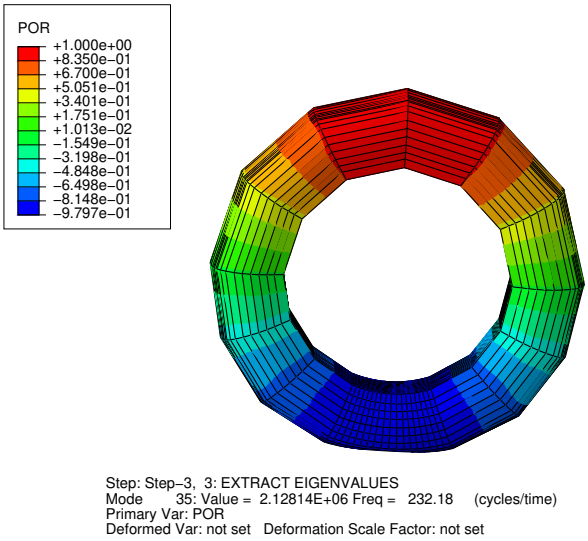


Figure 3: Vertical acoustic mode.

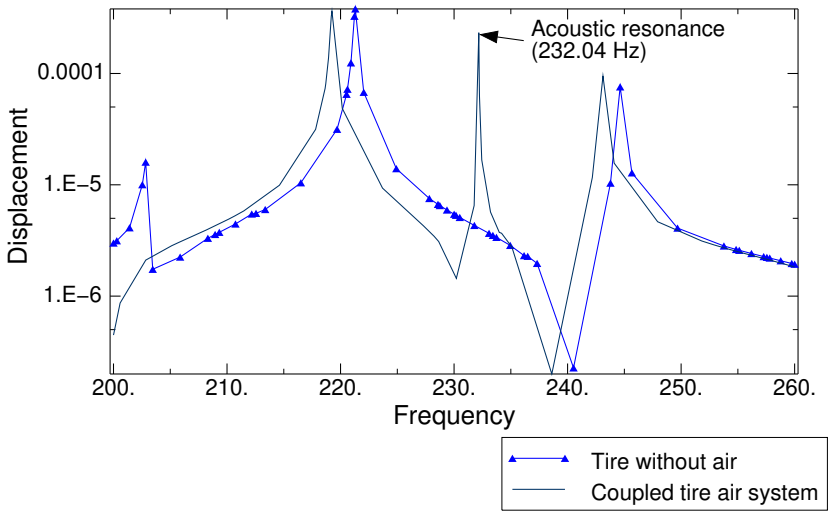


Figure 4: Vertical road displacement due to vertical excitation load.

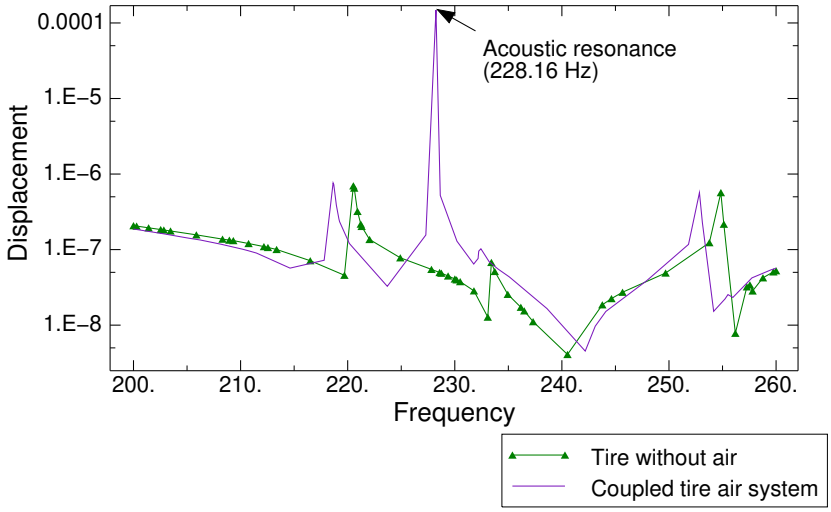


Figure 5: Horizontal road displacement due to horizontal excitation load.

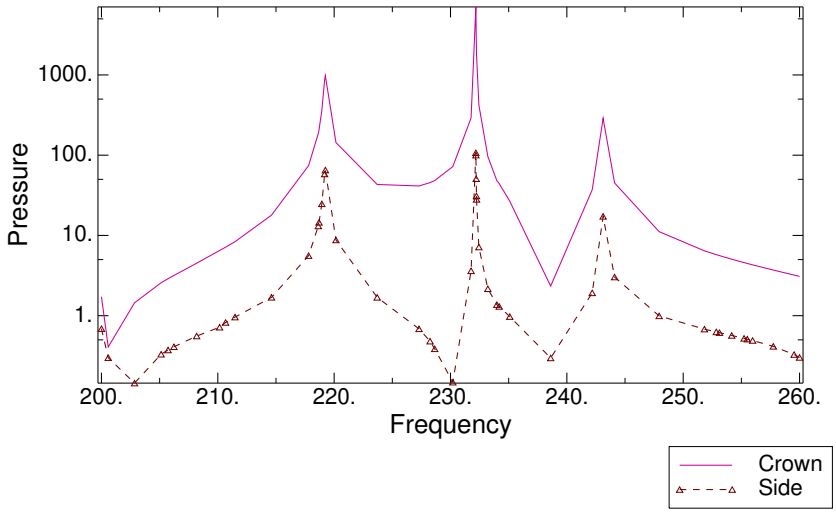


Figure 6: Acoustic pressure due to vertical excitation load.

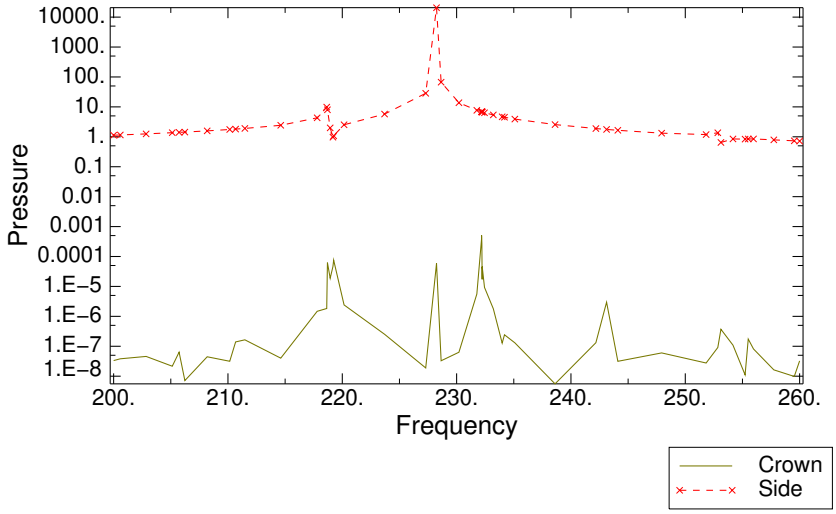


Figure 7: Acoustic pressure due to horizontal excitation load.

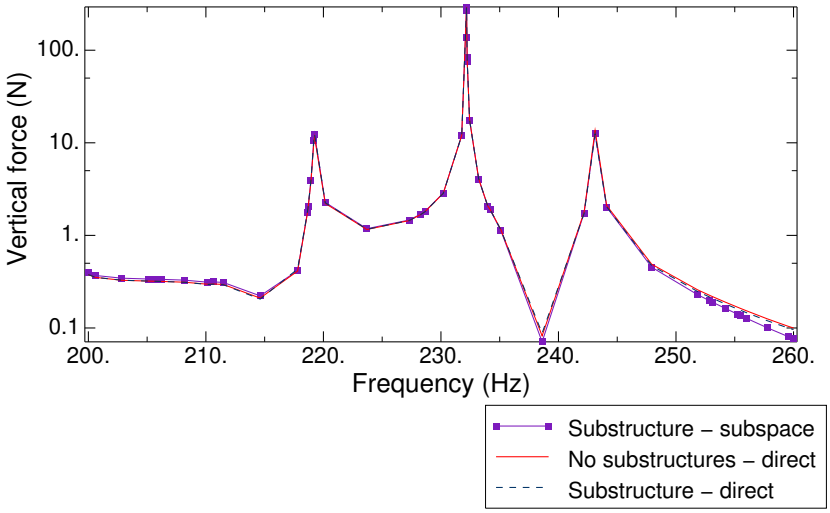


Figure 8: Vertical reaction force due to vertical excitation load.

Import of a steady-state rolling tire

Products: Abaqus/Explicit

References:

- *Transferring Results between Abaqus/Explicit and Abaqus/Standard*

Overview

This example illustrates the use of the results transfer capability in Abaqus to import results from an Abaqus/Standard steady-state transfer analysis to Abaqus/Explicit to simulate transient rolling.

Examples of loading during transient rolling include impact of a tire with an obstacle or vehicle acceleration. This problem analyzes the impact of a tire with a curb. Because of the small stable time increment necessary for the explicit dynamic procedure and the large time scales involved in simulating quasi-static and steady-state loading, simulating quasi-static inflation loading and steady-state rolling in Abaqus/Standard provides significant cost savings over performing these simulations in Abaqus/Explicit. Moreover, the cost to obtain a steady-state rolling simulation in Abaqus/Explicit increases with rolling speed, whereas in Abaqus/Standard the cost is independent of the magnitude of the rolling speed.

Problem description and model definition

The model used in this example differs slightly from that used in *Symmetric results transfer for a static tire analysis*. Since only elements common to both Abaqus/Standard and Abaqus/Explicit can be imported, reduced-integration solid elements are used in this example and nodes in the bead area are attached to rigid elements representing the rim. A plot of the tire before impact with the curb, a 0.025 m high step, is shown in *Figure 1*. The large rigid body rotation involved in a transient rolling analysis necessitates the use of the nondefault second-order-accurate kinematic formulation in the explicit dynamic analysis. In addition, it is recommended that the enhanced hourglass control scheme be used for both Abaqus/Standard and Abaqus/Explicit for all import analyses. Use section controls in the axisymmetric analysis to specify the enhanced hourglass control scheme.

The inflation and footprint preloads are applied in a series of general analysis steps identical to those described in *Symmetric results transfer for a static tire analysis*. Symmetric model generation and symmetric results transfer are used to exploit the symmetric nature of the structure and loading (see *Symmetric Model Generation* and *Transferring Results from a Symmetric Mesh or a Partial Three-Dimensional Mesh to a Full Three-Dimensional Mesh*).

The repeated dynamic impact of tire nodes as they come into contact with the road is an unavoidable source of the high frequency noise that can be seen in the reaction force at the rim. The viscoelastic property of the matrix provides sufficient damping to reduce such high frequency noise. No material damping (mass proportional or stiffness proportional) is necessary in this case. The viscoelastic stress contributions from the steady-state transport analysis are imported into the Abaqus/Explicit analysis.

Loading

An inflation load of 200 kPa is applied to the axisymmetric half-tire model in *importrolling_axi_half.inp*. This load is followed by a footprint load of 1650 N applied to the three-dimensional half-tire model in *importrolling_symmetric.inp*, and subsequently results are transferred to the full tire model with the complete footprint load of 3300 N.

The rolling analysis involves rolling the tire up to free rolling conditions. As in *Steady-state rolling analysis of a tire*, a translational velocity of 10 km/h is applied to the tire. The free rolling velocity of the tire is determined in an independent analysis similar to the one described in *Steady-state rolling analysis of a tire*. The rolling velocity corresponding to free rolling conditions is 8.9759 rad/s. Inertia loads are accounted for in this example, since the objective is to import results to a transient dynamic analysis in which inertial effects should be considered during impact with the curb. Viscoelastic effects in the rubber matrix are also included, since viscolasticity is the primary damping mechanism present in the tire.

During the transient dynamic analysis, the tire is moved forward with a prescribed velocity of 10 km/h and the vehicle load is applied to the rim reference node. The tire is allowed to rotate freely about the axle. All other degrees of freedom at the road and rim reference nodes are held fixed.

Results and discussion

Results are imported and the analysis begins at time $t = 3$ sec. A plot of the rotational velocity at the rim in *Figure 2* shows that the oscillations at the beginning of the analysis are well within acceptable limits. Impact with the curb is initiated after about a quarter rotation of the tire subsequent to import, at approximately $t = 3.13$ sec. The vertical response of the rim reference node can be seen in *Figure 3* and shows oscillations in the tire after impact. The contact patch and the stresses in the belts under steady-state rolling conditions in the transient dynamic solution compare well with the direct steady-state solution from Abaqus/Standard. The pressure in the footprint can be seen in *Figure 4*, and the stresses in the belts are plotted in *Figure 5*. A plot of the shear stress during impact in *Figure 6* shows that, as expected, the maximum stress is set up in the shoulder region.

Input files

importrolling_axi_half.inp	Axisymmetric model, inflation analysis.
importrolling_symmetric.inp	Half-symmetric three-dimensional model, inflation and footprint analysis.
importrolling_full.inp	Full three-dimensional model, inflation and footprint analysis.
importrolling_roll.inp	Steady-state, free rolling solution.
importrolling_xpl.inp	Import and transient dynamic analysis.

Figures

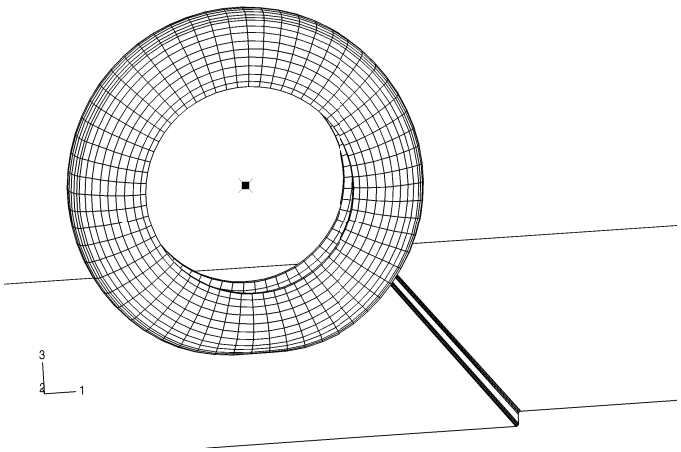


Figure 1: Tire before impact with the curb.

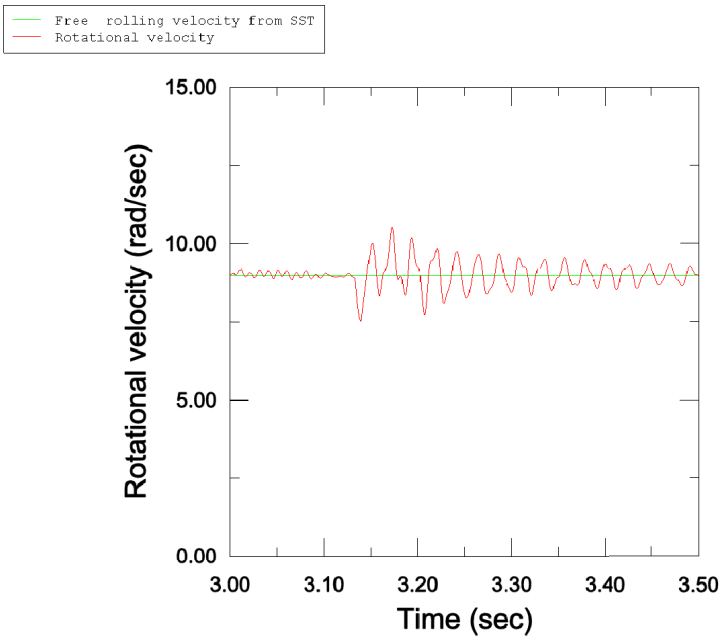


Figure 2: Rotational velocity at rim.

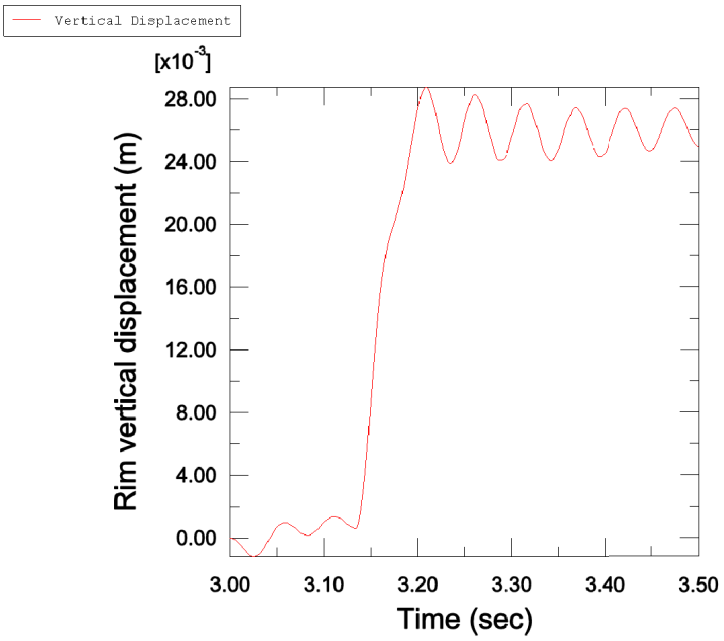


Figure 3: Vertical response of the rim reference node.

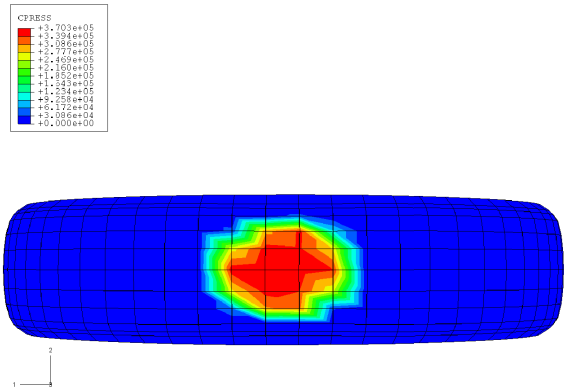


Figure 4: Pressure in the footprint patch.

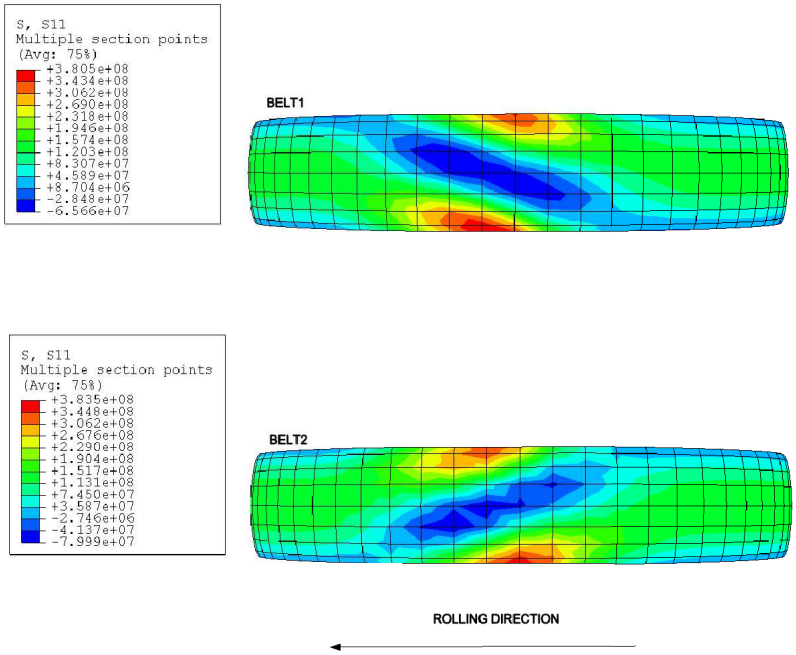


Figure 5: Stress in the belts.

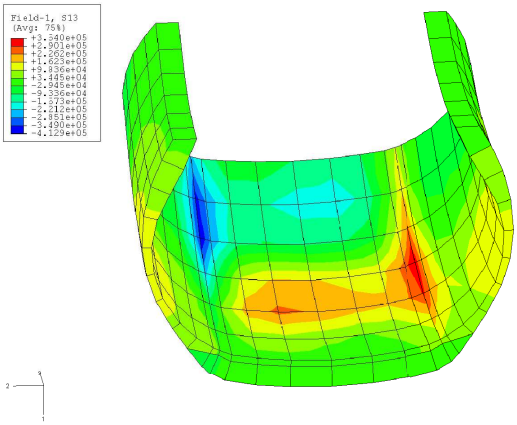


Figure 6: Shear stress in the shoulder region during impact with the curb.

Analysis of a solid disc with Mullins effect and permanent set

Products: Abaqus/Standard Abaqus/Explicit

References:

- *Stress Softening in Rubberlike Materials*
- *Permanent Set in Rubberlike Materials*

Overview

This example illustrates the use of the Mullins effect to model the static and steady-state rolling interaction between a solid rubber disc and a rigid surface.
The Mullins effect is used with a hyperelastic material model to model the phenomenon of stress softening upon unloading from a certain deformation level that is observed in certain filled elastomers. A variation of this example includes modeling of permanent set as well.

Problem description

This example is divided into three sections. The first section involves calibration of experimental data to determine the material coefficients for the Mullins effect. The second section describes the static response of a solid disc subjected to cyclic deformation induced by contact with a flat rigid surface that represents the road. This kind of test is commonly carried out in the tire industry to investigate the effect of stress softening on the load-deflection behavior of tires. The third section is a continuation of the problem described in the second section and studies the rolling solution of the deformed disc. The rolling is modeled using both a Lagrangian approach that involves rotation of the disc mesh and the steady-state transport capability in Abaqus/Standard that obtains the steady-state solution of the rolling disc using more of an Eulerian approach. This example also illustrates the enhanced hourglass control capability and modeling of permanent set in Abaqus.

Calibration of material properties

The first step in using the Mullins effect material model is calibration of test data. *Figure 1* shows a typical set of uniaxial tension test data for filled rubber. These data are representative of a class of rubber materials that is used in the tire industry and have been provided by Cooper Tire and Rubber Company. The bold curve labeled `exp_mono` represents the primary behavior of the rubber material that would be obtained from a monotonic test to a certain deformation level. These data are provided to Abaqus using a set of uniaxial test data and a customization to the hyperelastic material model specifying that material constants should be calculated from that test data. In this example the Yeoh material model is used for calibrating the primary material behavior. Reduced polynomial models, such as the Yeoh model, provide a better fit when limited test data are available, as is the case here. The unloading-reloading data, needed to calibrate the Mullins effect coefficients, are provided as stabilized loading/unloading cyclic data from three different maximum strain levels. Although the model for Mullins effect in Abaqus predicts unloading and reloading from a given maximum strain level to occur along a single curve, the real material often shows a cyclic behavior. In fact, the unloading/reloading cycles from a given maximum strain level also show evidence of progressive damage during the first few cycles. However, after a few cycles the behavior stabilizes. The unloading-reloading data can be provided to Abaqus either in the form of data points for all loading/unloading cycles from different maximum strain levels or in the form of data points for just the stabilized cycles. In this example the stabilized cycle has been chosen because the structure considered is expected to undergo repeated cyclic loading. The curves labeled `exp_unload1`, `exp_unload2`, and `exp_unload3` represent the stabilized unloading/reloading cycles for maximum nominal strain levels of 0.099, 0.26, and 0.51, respectively. The above data are input using three sets of uniaxial test data (one for each maximum strain level), along with Mullins effect defined by test data.

For models that include permanent set, the data are specified using a metal plasticity material definition so that a softer response (compared to a purely hyperelastic material) is obtained in loading. These models also include the Mullins effect material model as discussed above.

During design studies the analyst may wish to calibrate the Mullins effect parameters using the loading part of the stabilized cycle, the unloading part of the stabilized cycle, or an average of the two. This can be accomplished easily by creating three data files that include the loading, unloading, and both the loading and unloading parts of the stabilized cyclic data, respectively. Subsequently any one of these files can be referenced from the input file by using an external file reference. In this example, which considers cyclic loading instead of primarily monotonic loading, both loading and unloading data have been used for calibration; hence, an average behavior is studied.

Figure 2 shows the calibrated response of Abaqus along with the test data. The continuous bold line represents the numerical response for cyclic loading and unloading to the three maximum strain levels for which test data have been provided (the response at the lowest maximum strain level is not visible in the scale of this figure due to a relatively small amount of damage). As the figure illustrates, the model in Abaqus approximates the stabilized cycle at each strain level with a single curve. The figure also illustrates that the model does not capture the progressive damage during the first few cycles at any strain level. Thus, while the numerical results for unloading from a given strain level begin from the primary curve, the maximum stress for the test results for the stabilized cycle may be somewhat below the measured primary material behavior.

The numerical response can be obtained by carrying out a uniaxial loading/unloading test with a single element. Alternatively, the numerical response for both the primary and the unloading-reloading behavior can be obtained by using a model definition data request and simply carrying out a **datacheck** run. In the latter case the response computed by Abaqus is printed to the data (`.dat`) file along with the experimental data. These tabular data can be plotted in Abaqus/CAE for comparison and evaluation purposes. The primary material behavior can also be evaluated with the automated material evaluation tools available in Abaqus/CAE.

Static response of a solid disc to cyclic deformation

A solid disc made out of the rubber material described in the earlier section is subjected to cyclic deformation.

The coefficients for the Yeoh model determined during the calibration are used along with a value of $D_1 =$ 5.E−06 to introduce a small amount of compressibility in the material. This value of D_1 is obtained based on the measured value of the initial bulk modulus of the rubber. The Mullins effect model in Abaqus assumes that the damage is associated with the deviatoric behavior only. The disc has an outer diameter of 3 inches, an inner diameter of 1.75 inches, and a thickness of 0.7 inches. The inner surface of the disc is fully constrained. The outer surface is initially defined to be just touching a flat rigid surface. The coefficient of friction between the disc and the rigid surface is assumed to be zero. During the analysis the rigid surface is pushed up 0.15 inches, brought back to its original position, then again pushed up 0.22 inches before being pushed back to its original position. The above deformation history constitutes two displacement-controlled deformation cycles. The analysis is carried out using both Abaqus/Standard and Abaqus/Explicit. An axisymmetric model is created to define the geometry of the disc. Symmetric model generation is used to create a three-dimensional disc model using first-order reduced-integration bricks (C3D8R elements) with enhanced hourglass control. The Abaqus/Explicit model is created by importing the model definition from Abaqus/Standard. The first step in the Abaqus/Standard analysis is a do-nothing step, which is used to import the initial state into Abaqus/Explicit.

Figure 3 shows the force versus displacement at the rigid body reference node. The two cycles in this figure correspond to the two deformation cycles discussed earlier. During the first cycle the unloading response is softer compared to the loading response due to damage associated with the Mullins effect. During the second loading cycle the response is identical to the unloading segment of the first cycle until the displacement of 0.15 inches is reached. Beyond this point the response is a continuation of the original loading segment of the first cycle. Thus, the load-displacement behavior is consistent with the expected behavior due to the Mullins effect.

Figure 4 shows the time history of the energy dissipated in the whole model due to damage. The dissipation increases during the loading segment of the first cycle because the material undergoes more and more damage

as it is deformed. During the unloading segment of the first cycle and the loading segment of the second cycle up to a displacement of 0.15 inches, no additional damage occurs. As a result, the total dissipation remains constant. For additional displacement beyond 0.15 inches, more damage occurs. This results in further increase of the total damage energy. During the final unloading cycle the damage energy again remains constant.

The loading in the Abaqus/Explicit analysis is carried out using displacement boundary conditions with an amplitude that uses the smooth step definition to reduce the noise in the response. Thus, the time history of the displacement is different between the Abaqus/Standard and the Abaqus/Explicit simulations, although the total amount of displacement is identical in both analyses. As a result, the time history of the damage dissipation between the two analyses shows some differences in the slope of the response. However, the total dissipation at the end of each stage is identical in the two cases.

Rolling response of a solid disc

The geometry and material of the disc are identical to those described earlier except that the inner surface of the disc is not totally constrained, as it is for the static problem. Instead, all the nodes on the inner surface are attached to a node (axle node) located at the center of the disc using kinematic coupling constraints. This facilitates the application of angular velocity or displacement to the axle node to simulate rolling in the Lagrangian approach, as well as the measurement of reaction forces and moments at the axle. The mesh is more refined for the rolling problem compared to the static problem. In particular, two elements are used through the thickness of the disc. The first step is a do-nothing step to facilitate import of the initial state from Abaqus/Standard to Abaqus/Explicit. This step is followed by a static step in which the rigid surface is pushed against the disc a distance of 0.15 inches. The next step involves rolling of the loaded disc against the rigid surface, which is accomplished in two ways. The first is a Lagrangian analysis in which an angular velocity of 2.5 radians per second is applied to the axle node of the disc. In this example the structure reaches a steady state after one full revolution. No additional damage occurs in subsequent revolution cycles. Therefore, the total time is chosen such that the disc undergoes two revolutions. A variation of the Lagrangian analysis includes modeling of permanent set using a metal plasticity material definition. In the second analysis the rolling is simulated using the steady-state transport capability in Abaqus/Standard. Frictional and inertial effects are neglected in both cases.

The steady-state transport capability directly obtains the steady-state rolling solution of the disc on the rigid surface. Due to Mullins effect the stress state for the rolling solution can be quite different from the stress state for the static non-rolling solution. As a result, an attempt to obtain a steady-state rolling solution directly from a static non-rolling solution may lead to convergence problems in the Newton's scheme that is used to solve the overall nonlinear system of equations. Since the damage and, hence, the discontinuity in state are independent of the angular rolling speed, a time increment cutback during the steady-state transport step does not overcome the convergence difficulties. Such convergence difficulties can be resolved by introducing the damage gradually over an additional steady-state transport step preceding the actual analysis. In this example this is accomplished by following the static loading step with a steady-state transport step with a small rolling angular speed of 0.25 radians per second and with the Mullins effect ramped up over the time period of the step. This step is followed by another steady-state transport step at an angular speed of 2.5 radians per second with the MULLINS parameter set equal to STEP, which provides the solution we are interested in obtaining.

A Lagrangian simulation is also carried out in Abaqus/Explicit. The kinetic energy is monitored to ensure that the problem remains essentially quasi-static. The revolutions of the disc are carried out by applying a rotational displacement (corresponding to two full revolutions) at the axle node using an amplitude with a smooth step definition to reduce the noise in the response. The compressibility parameter, $\boldsymbol{D_1}$, is chosen to be 5×10^{-5}, an order of magnitude higher than the actual value, to obtain relatively higher time increments and, thus, a relatively lower run time.

Figure 5 compares the time history of the reaction force at the axle node for the Lagrangian and the steady-state rolling analyses in Abaqus/Standard. The results for the Lagrangian problem without permanent set (curve labeled `lagr`) indicate that the reaction force increases during the loading step and decreases during the first revolution of the disc. The decrease in the reaction force is a result of lower overall stresses due to damage in the material. During the second revolution of the disc the reaction force remains constant as no additional damage occurs.

The results of the Lagrangian problem including permanent set (curve labeled `fefp`) show a softer behavior compared to the former problem and clearly indicate the presence of permanent set in the footprint area as the solid disc rolls against the rigid surface. The steady-state rolling results (labeled `sst`) show a gradual transition of the reaction force during the first steady-state transport step that ramps up the Mullins effect. During the second steady-state transport step the reaction force remains constant at the value reached at the end of the prior step. This curve also illustrates that the damage associated with the Mullins effect is independent of the angular speed of rotation. The reaction force remains the same at angular speeds of both 0.25 and 2.5 radians per second. If the Mullins effect is not applied gradually over the first steady-state transport step, the discontinuity between the rolling and the static states may lead to convergence difficulties.

The same results can be observed from a different viewpoint in *Figure 6*, which shows the reaction force as a function of the number of revolutions for both Abaqus/Standard and Abaqus/Explicit. The reaction force decreases during the first revolution and remains steady (except for the noise in the Abaqus/Explicit analysis) during the second revolution.

Figure 7 shows a comparison of the time histories of the reaction moment at the axle node between the Lagrangian and the steady-state rolling solutions. The Lagrangian results are labeled `m-lagr`, while the steady-state rolling results are labeled `m-sst`. If the material were purely hyperelastic (without damage), the contact forces would be symmetric about a plane normal to the rigid surface and containing the axle; hence, no torque would be required to rotate the disc. However, as a result of the damage associated with Mullins effect, the contact forces are not symmetrical because material particles transition through the contact area during the very first revolution. This leads to the reaction moment during the first revolution, as shown in the results for the Lagrangian analysis. The moment reduces to zero during the second revolution. The steady-state rolling results do not include the transient solution of the first revolution; hence, they show a zero moment at all times. *Figure 8* shows the same results from a different viewpoint. In this figure the reaction moment is plotted as a function of the number of revolutions for both Abaqus/Standard and Abaqus/Explicit.

Figure 9 shows a contour plot of the damage energy dissipated at material points at an instant of time that corresponds to about three-quarters of the way into the first revolution of the disc. The figure indicates damage in the material that has already passed through the contact area and no damage in the material that is yet to pass through the contact area. This corresponds to damage in about three-quarters of the disc material. The remaining quarter is still undamaged, as it has not undergone any deformation yet. The full disc will be damaged at the end of the first revolution, and the damage state remains unchanged during the second revolution.

Results and discussion

The results are discussed in the individual sections above and clearly demonstrate the different effects of damage in the material.

Input files

mullins_calibrate.inp	Unit element test to calibrate the material model.
mullins_axi_tire.inp	Axisymmetric model for the static non-rolling problem.
mullins_full_tire.inp	Full three-dimensional model for the static non-rolling problem.
mullins_full_tire_xpl.inp	Full three-dimensional model for the static non-rolling problem (quasi-static simulation using Abaqus/Explicit).
mullins_axi_tire_ref.inp	Refined axisymmetric model for the rolling problem.
mullins_full_tire_roll_lag.inp	Full three-dimensional model for the Lagrangian rolling problem.
mullins_full_tire_roll_lag_xpl.inp	Full three-dimensional model for the Lagrangian rolling problem (quasi-static simulation using Abaqus/Explicit).
mullins_full_tire_roll_sst.inp	Full three-dimensional model for the steady-state rolling problem.
mullins_calibrate_testdata.inp	Uniaxial test data for calibrating the Mullins effect coefficients.

mullins_hepl_axi_tire_ref.inp	Axisymmetric reference model for rolling problem with permanent set.
mullins_hepl_full_tire_roll_lag.inp	Full three-dimensional model for the Lagrangian rolling problem with permanent set.

Figures

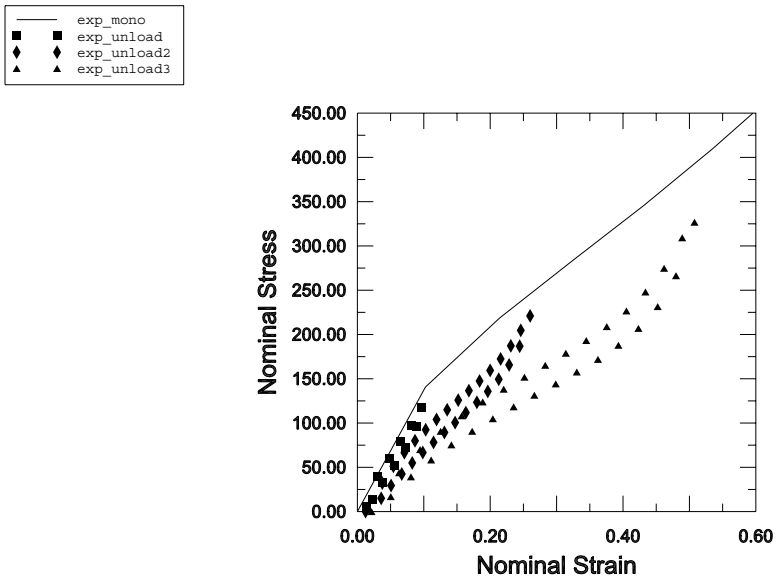


Figure 1: Test data for calibrating the Mullins effect.

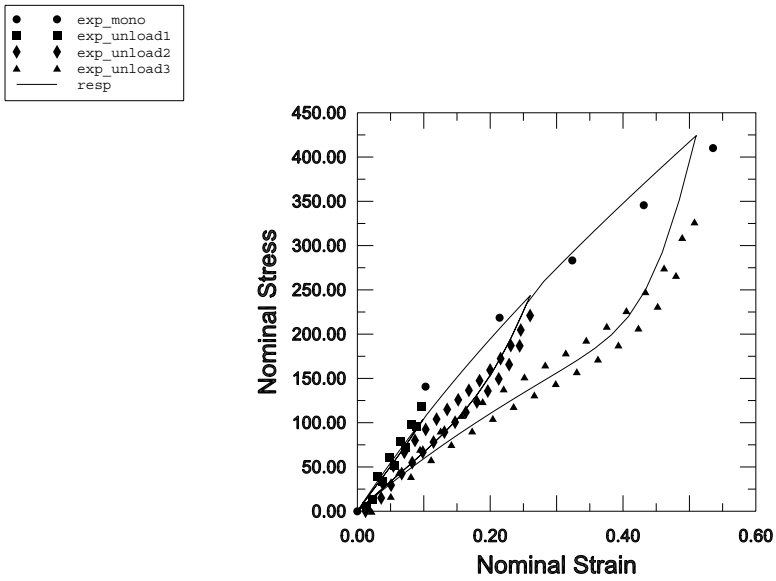


Figure 2: Calibration of Mullins effect.

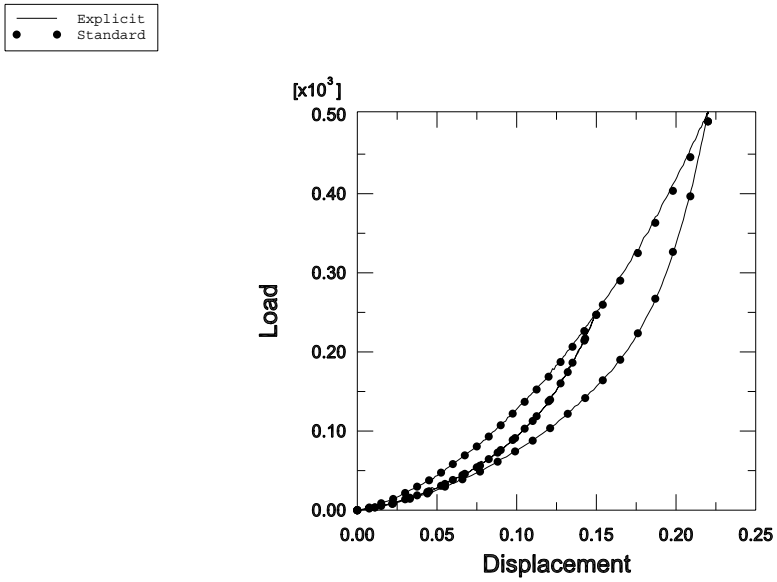


Figure 3: Force versus displacement for the static non-rolling solution.

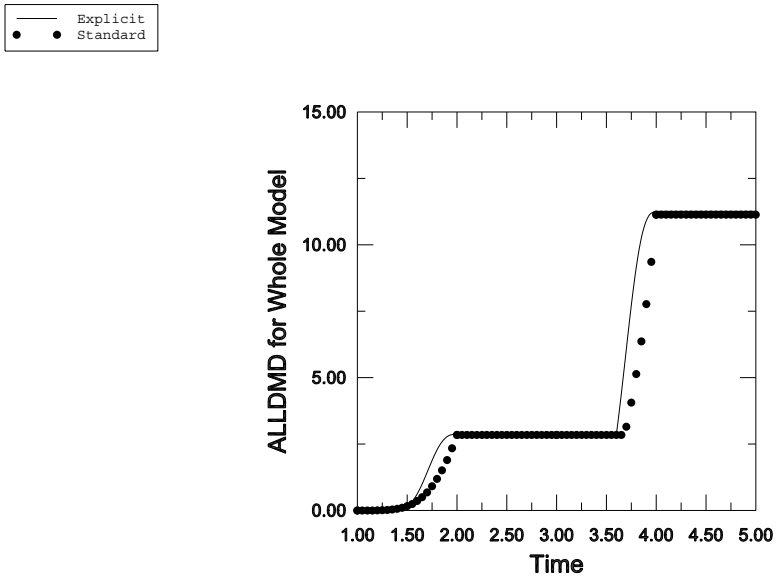


Figure 4: Whole model damage energy history for the static non-rolling solution.

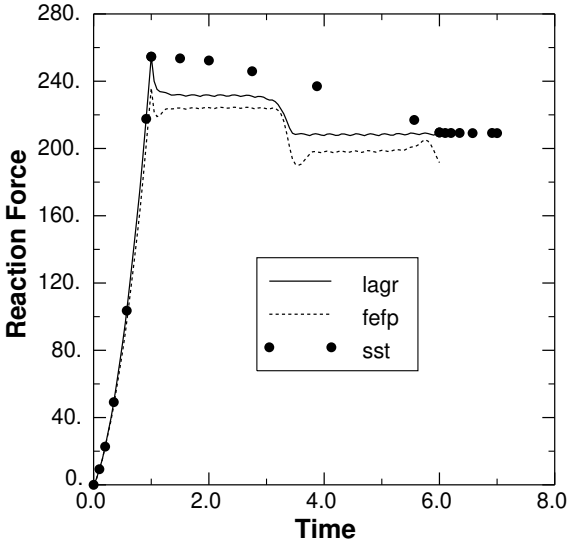


Figure 5: Time history of reaction force for the rolling solution.

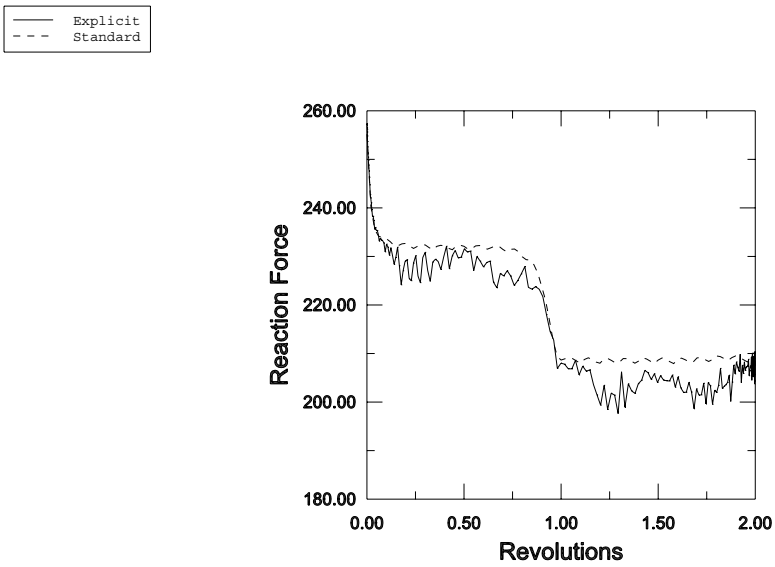


Figure 6: Reaction force versus number of revolutions.

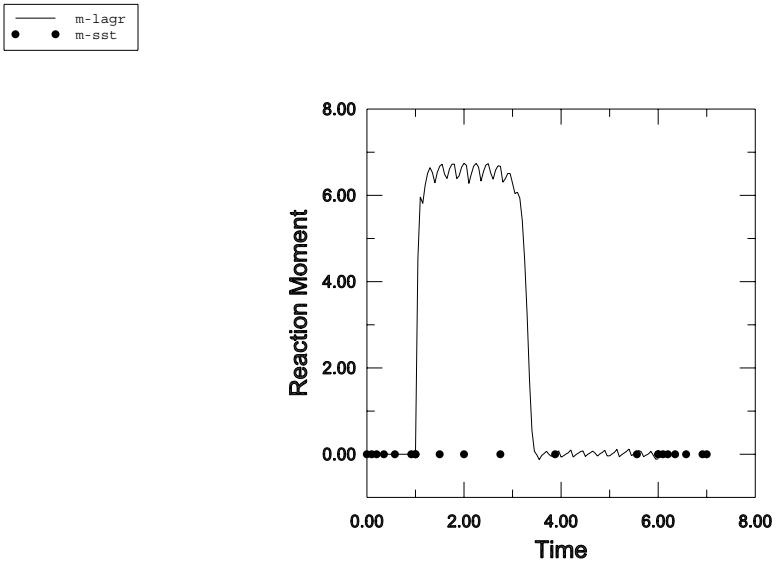


Figure 7: Time history of reaction moment for the rolling solution.

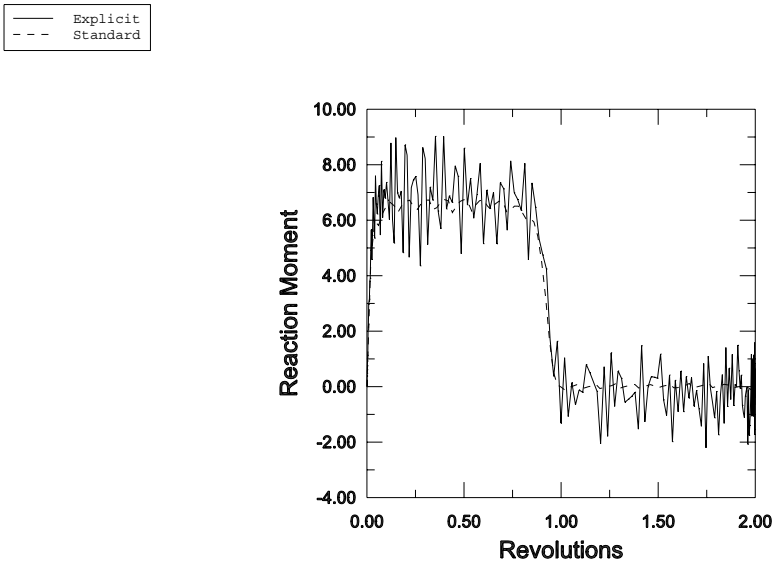


Figure 8: Reaction moment versus number of revolutions.

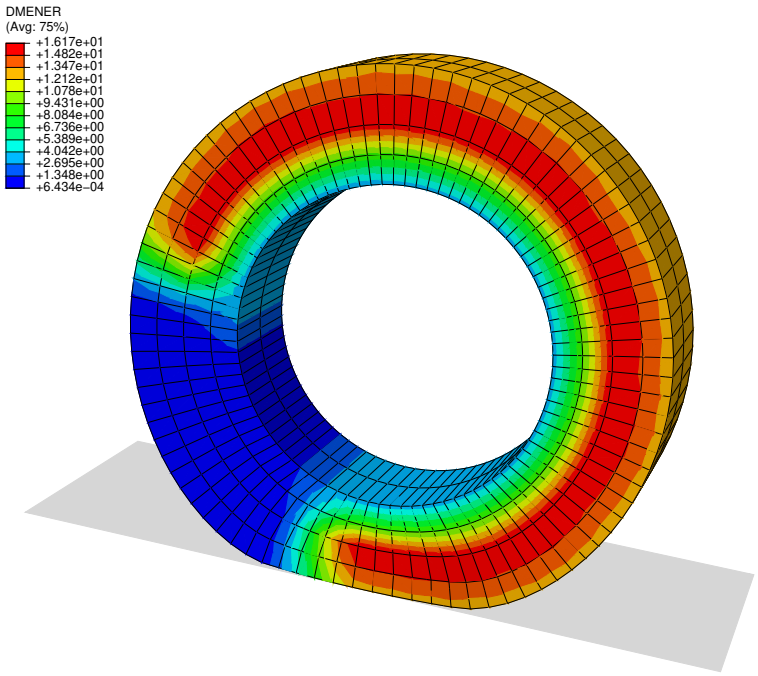


Figure 9: Contour plot of damage energy.

Tread wear simulation in Abaqus/Standard

Products: Abaqus/Standard

Overview

This example illustrates techniques to model tread wear in a steady rolling tire in Abaqus/Standard.

The analysis closely follows the techniques used in *Steady-state rolling analysis of a tire* to establish first the footprint and then the state of the steady rolling tire. These steps are then followed by a steady-state transport step in which a wear rate is calculated and extrapolated over the duration of the step, providing an approximate consideration of the transient process of wear in this steady-state procedure.

Two approaches are presented:

- In the first approach, which is simpler, surface wear properties are specified for wear rate calculations and the resulting total wear distances are computed natively for the steady-state transport procedure. The wear distances are treated as nodal offsets for contact penetration calculations without modifying the nodal positions of the underlying elements (*Accounting for Wear with Nodal Offsets in Contact Calculations*) and can be output for visualization.
- In the second approach, an adaptive meshing user subroutine is utilized to do detailed wear calculations along nodal streamlines for simultaneously moving nodal positions on underlying elements. The user subroutine approach can be rather cumbersome because you have to carefully define node sets for individual streamlines and implement wear calculations to effect adaptive mesh motion from wear increments.

Problem description and model definition

With some exceptions, noted here, the description of the tire and finite element model is the same as that given in *Import of a steady-state rolling tire*. Since the focus of this analysis is tread wear, the tread is modeled in more detail. In addition, a linear elastic material model is used in the tread region to avoid difficulties advecting the hyperelastic material state during the adaptive meshing procedure.

The axisymmetric half-model of the 175 SR14 tire is shown in *Figure 1*. The rubber matrix is modeled with CGAX4 and CGAX3 elements. The reinforcement is modeled with SFMGAX1 elements that carry rebar layers. An embedded element constraint is used to embed the reinforcement layers in the rubber matrix. The tread is modeled with an elastic material of elastic modulus 6 MPa and Poisson's ratio 0.49. The rest of the tire is modeled with the hyperelastic material model. The polynomial strain energy potential is used with coefficients $C_{10}=10^6$, $C_{01}=0.0$, and $D_1=2\times10^{-8}$. The rebar layers used to model the carcass fibers are oriented at 0° to the radial direction and have an elastic modulus of 9.87 GPa. The modulus in compression is set to 1/100th of the modulus in tension. The Marlow hyperelastic model is used to specify the nominal stress-nominal strain data for such a material definition. The elastic modulus in tension of the material of the belt fibers is 172.2 GPa. The modulus in compression is set to 1/100th of the modulus in tension. The fibers in the belts are oriented at +20° and −20° with respect to the hoop (circumferential) direction.

The three-dimensional model is created by first revolving the axisymmetric half-model, using symmetric model generation, by 360° to generate the partial three-dimensional model shown in *Figure 2*. A focused mesh is applied at the footprint region. The partial three-dimensional model is then reflected about a line to generate the full three-dimensional model. The results are then transferred from the end of the footprint simulation for the partial three-dimensional model (see *Figure 2*).

Loading

The analysis is conducted in five stages, beginning with an axisymmetric model and ending with a full three-dimensional model created using symmetric model generation. The first four steps follow closely those used in *Steady-state rolling analysis of a tire*.

1. Axisymmetric inflation: An inflation pressure of 200 kPa is applied to the interior of the tire, while symmetry conditions are imposed at the midplane.
2. Three-dimensional footprint analysis of the partial model: The axisymmetric half-model is revolved around the axle axis.
3. Three-dimensional footprint analysis of the whole model: The partial three-dimensional model is reflected about a line to generate the full three-dimensional model.
4. Steady-state transport: The generated full model is then subjected to a steady-state transport analysis at 30 km/h. An angular velocity of 25 rad/s is specified for the tire. These conditions correspond to a state of braking. Inertia effects and viscoelasticity are taken into account during this step.
5. Tread wear simulation: The tread wear simulation is conducted in the final step, in which the velocity of the tire is held constant and the wear is computed from the frictional energy dissipated and applied around the periphery of the tire. Inertia effects and viscoelasticity are taken into account during this simulation as well. This step is run for a duration of 3.6×10^6s, simulating 30,000 kilometers of travel of the tire at 30 km/h in both the approaches for wear calculations mentioned earlier. Steady-state rolling along a straight line and a 2^o side-slip are considered for wear calculations.

The final step uses a wear model, which predicts a wear distance, or a surface ablation rate based on the steady rolling of the tire. We are interested in predicting tire configuration changes as a result of this wear rate; hence, we must introduce some modeling assumptions that enable modeling of a transient effect in a steady-state procedure.

The basic assumption made is that the steady-state transport step time can be interpreted as a real-time duration of rolling at the current angular velocity. We consider that the configuration changes due to wear have only a minor effect on the rolling tire solution at any time; hence, the results remain valid in a steady-state sense at each time through the step. With this assumption we can simultaneously consider effects at two disparate time scales: the shorter tire revolution time scale and the longer tire life time scale.

Wear model

Steady-state transport uses an Eulerian formulation where the tire material is convected through a stationary mesh. The following sections describe how a cumulative wear distance is computed at a material point on the tire tread as it traverses along a streamline during tire revolutions.

Wear rate calculation

For tire rubber we assume a wear coefficient $\tilde{\varphi}_{wear} = 1.11\times 10^{-4}$ and a material hardness $H = 2$ GPa based on the Archard's wear model (see *Wear Rate Models*).

The goal of the following development is an expression for a material wear distance, or ablation, rate $\dot{w}$, which can be applied to a node to simulate wear. First, consider a ribbon around the tire, where the centerline of this ribbon is defined by a sequence of nodes comprising one of the surface streamlines on the tire's tread. This centerline is then bounded on either side by the tributary region of the surface associated with each node. The combination of all such "stream ribbons" then comprises the total surface of the tire involved in tire-to-road contact interactions. We expect the wear to occur uniformly over this stream ribbon; hence, we express a volumetric wear rate $\dot{q}$ for the entire ribbon:

$$\dot{q}(t) = \frac{\tilde{\varphi}_{wear}}{H} \int_{\text{ribbon}} P(\mathbf{x},t)\,\dot{\gamma}(\mathbf{x},t)\,dA,$$

where t is the time and $\mathbf{x}$ is the current configuration position. Since we are using the Eulerian steady-state transport procedure, we can now rewrite this expression in a time-independent form:

$$\dot{q} = \frac{\tilde{\varphi}_{wear}}{H} \int_s P(s)\,\dot{\gamma}(s)\,T(s)\,ds,$$

where s is a position along the streamline and $T(s)$ is the width of the stream ribbon at position s. We can also express $\dot{q}$ as a function of the local material recession rate:

$$\dot{q} = \int_s \dot{w}(s)\,T(s)\,ds.$$

Equating these two expressions in a discrete form results in the following expression, summed over a streamline:

$$\sum_{i=1}^{N} \dot{w}_i A_i = \frac{\tilde{\varphi}_{wear}}{H} \sum_{i=1}^{N} P_i \dot{\gamma}_i A_i,$$

where $\dot{w}_i$ is the magnitude of nodal ablation velocity and A_i is the nodal contact area. This equation implies that $\dot{w}_i$ is generally not uniform along the streamline, which follows as a consequence of the stream ribbon width varying as it enters and leaves the tire footprint. Since, however, we are ablating nodes away from the footprint region solely to maintain a reasonable general shape of the worn tire configuration, we will accept the assumption of a uniform nodal ablation velocity. This enables the following expression for $\dot{w}$:

$$\dot{w} = \frac{\tilde{\varphi}_{wear} \sum_{i=1}^{N} P_i \dot{\gamma}_i A_i}{H \sum_{i=1}^{N} A_i}.$$

Using again the assumption that the variation in stream ribbon width can be neglected, $T_i = T$, and recognizing that the nodal contact area $A_i = T\Delta S_i$ enables a simpler expression that does not require the use of contact areas. This nodal wear rate applies equally to all the nodes belonging to a streamline:

$$\dot{w} = \frac{\tilde{\varphi}_{wear} \sum_{i=1}^{N} P_i \dot{\gamma}_i \Delta S_i}{H \sum_{i=1}^{N} \Delta S_i}.$$

Wear process implementation

The wear computation described in the previous section is similar to what is used in the built-in approach in a steady-state transport procedure. You can visualize wear distance vectors by requesting the contact output variable CWEAR.

For the second approach, the wear rate that is the surface ablation velocity is computed as a mesh motion velocity within user subroutine *UMESHMOTION* and applied at the nodes that are on the exterior surface of the tire. *UMESHMOTION* is used in conjunction with adaptive meshing, a mesh smoothing technique applied at the end of each converged increment. The ablation velocities specified through the user subroutine are applied at the tread surface nodes, and adaptive mesh smoothing adjusts nodes in the interior tread region to maintain a well-shaped mesh.

To accumulate wear quantities around each tread streamline, the nodal numbering scheme along the streamlines must be recorded in the user subroutine. This record is made in a set of common block variables. The common block records nodes that belong to node set `NADAPT` (*Figure 4*) and that lie at the reference cross section (0°) of the full model. The common block variables also include the node numbering offsets specified for symmetric model generation, which, together with the reference cross section definitions, completely describe the tread surface node numbering. The following variables need to be defined in the external common block:

- `nStreamlines`: The number of nodes at the reference section (full model) at which wear is applied.
- `nGenElem`: The number of sectors or element divisions along the streamline in the model.
- `nRevOffset`: The node offset specified as part of the definition of a revolved symmetric model.

- `nReflOffset`: The node offset specified as part of the definition of a reflected symmetric model. (Set this parameter equal to zero if the model is not reflected).
- `jslnodes`: The array that contains the required nodal information for all nodes potentially undergoing wear at the reference section. This is an array of size (2, `nStreamlines`). For each streamline the first component is the node number of the "root node" (node *a* in the following discussion), which is the node on that particular streamline at the reference section. The second component is the node that provides the wear direction (node *b* in the following discussion). This second component is necessary only for the tread corner nodes. Set it equal to the number of the node at the reference section that defines the wear direction. For nodes that are not located at tread corners, set the second array component equal to zero. The wear will be applied opposite to the local 3-direction for these nodes located away from tread corners.

The variables in the expression for wear are accessed from the analysis database using the utility routines `GETVRN` and `GETVRMAVGATNODE`. P is accessed from output variable CSTRESS; $\dot{\boldsymbol{\gamma}}$ from variable CDISP; and the $\boldsymbol{\Delta S}$ are determined from streamline nodal coordinates, accessed from variable COORD.

The following limitations apply when adaptive meshing is used for wear simulation in a steady-state transport procedure:

- Cylindrical elements are not supported with adaptive meshing and are not used in this model.
- Adaptive meshing generally performs poorly with hyperelastic material models due to the inaccurate advection of deformation gradient state variables. The tread, therefore, is modeled with an elastic material definition.
- Embedded elements including rebar layers cannot be used within the adaptive mesh domain.
- Adaptive mesh smoothing on free surfaces occurs in directions determined by features of the element geometry, which might not always be consistent with or easily enable description of wear directions. As a result, as discussed below, you will generally have to do additional work to explicitly describe the direction of wear.

Wear motion directions for the adaptive meshing user subroutine approach

The wear rate, $\dot{\boldsymbol{w}}$, is then applied as the components of the mesh constraint vector variable `ULOCAL`. This variable is passed into the user subroutine with default mesh smoothing motions defined in a local coordinate system `ALOCAL`, which reflects a measure of the surface normal at the current node. The 3-direction is defined as the direction of the outward normal and is based on an average of element facet normals near the node. Under most circumstances it is sufficient to describe wear as resulting in ablation, or nodal recession, opposite to this direction. At the tread corners, however, this average normal does not provide an accurate wear direction. The appropriate normal is shown in *Figure 5* and is computed as follows: Suppose *a* is a corner node on the tread. It is possible to identify a node *b* that lies along the edge of the tread. In this case the wear direction is given by the vector *ab*. By knowing the coordinates of *a* and *b*, the wear can be calculated in the global coordinate system and rotated into the local coordinate system (`ALOCAL`) directions.

Results and discussion

The tire model is run for a duration of 3.6×10^6 s, or 1000 hours, the equivalent of 30,000 kilometers of operation at 30 km/h. Based on the first approach, the CWEAR contours are shown in *Figure 6*. In the straight rolling case, the CWEAR contours are nearly symmetric and a maximum wear distance of 1.46 x 10^{-3}m is accumulated, which is close to the values calculated using adaptive meshing. In the side-slip case, a larger maximum wear distance of 1.69 x 10^{-3}

m is accumulated on one side of the tire due to higher contact stresses, and an uneven wear pattern is seen as expected.

Using adaptive meshing, *Figure 7* shows the resulting tread profile including the effects of wear. *Figure 8* shows the footprint distributions of the contact pressure and the contact pressure error indicator for a new tire and one in the as-worn configuration.

Input files

treadcwear_axi.inp	Axisymmetric model, inflation analysis with wear properties specified on the tread surface for the built-in approach.
treadcwear_rev.inp	Partial 3D model, footprint analysis where the wear properties are transferred from the previous axisymmetric analysis.
treadcwear_refl.inp	Full 3D model, footprint analysis.
treadcwear_roll.inp	Full 3D model, steady rolling along a straight line and with a 2^o side-slip angle.
treadcwear_wear_straight.inp	Full 3D model; wear accumulation for steady rolling of a tire along a straight line with CWEAR output.
treadcwear_wear_slip.inp	Full 3D model; wear accumulation for steady rolling of a tire subject to 2^o side-slip angle with CWEAR output.
treadwear_axi.inp	Axisymmetric model, inflation analysis. The sequence of input files starting with the current one are used with the adaptive meshing approach.
treadwear_rev.inp	Partial 3D model, footprint analysis.
treadwear_refl.inp	Full 3D model, footprint analysis.
treadwear_roll.inp	Full 3D model, steady rolling analysis.
treadwear_wear_straight.inp	Full 3D model, steady rolling wear analysis.
treadwear_wear_slip.inp	Full 3D model, steady rolling wear analysis with slip.
treadwear.f	`UMESHMOTION` user subroutine.

References

- Archard, J. F., “Contact and Rubbing of Flat Surfaces,” *Journal of Applied Physics*, vol. 24, pp. 981–988, 1953.

Figures

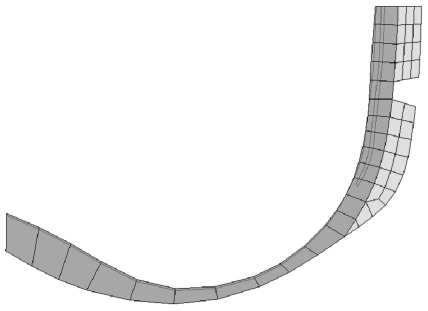


Figure 1: Axisymmetric cross section of the tire.

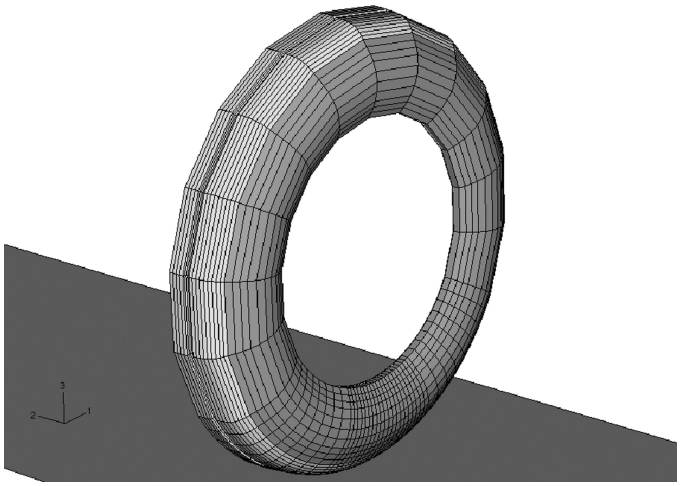


Figure 2: Partial three-dimensional model.

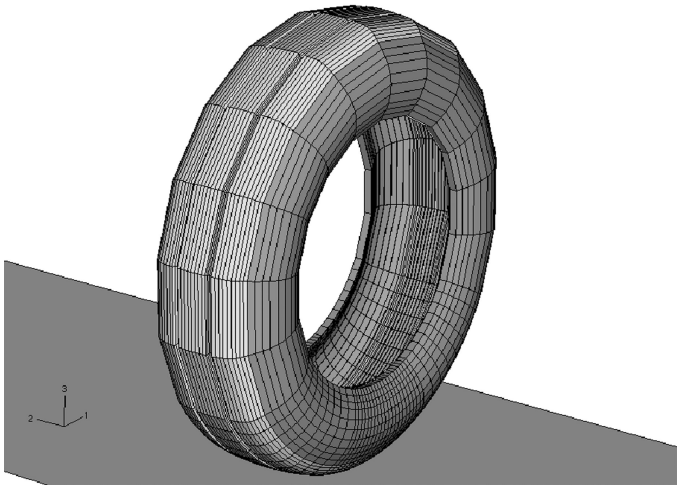


Figure 3: Full three-dimensional model.

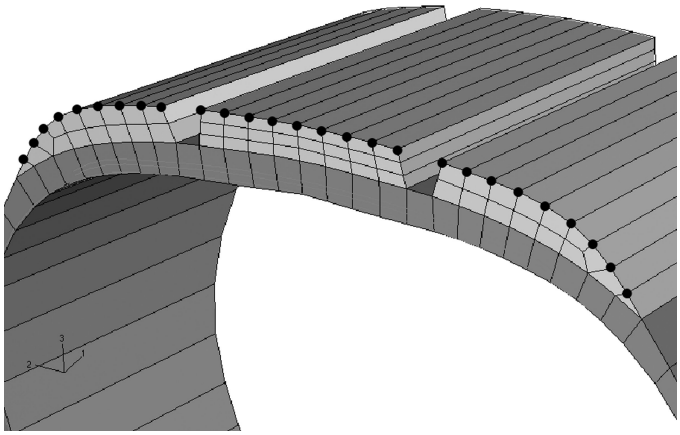


Figure 4: Nodes belonging to node set `NADAPT` at a particular sector (`NADAPT` includes all such nodes from all the sectors).

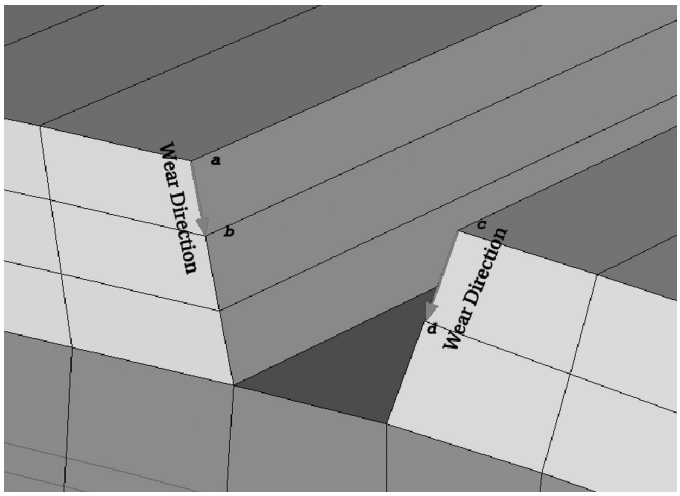


Figure 5: Wear directions at tread corners.

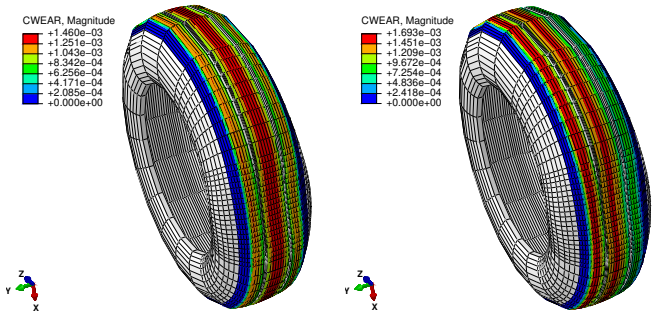


Figure 6: Wear distance contours for steady-state rolling (left) and steady-state rolling with side-slip(right)

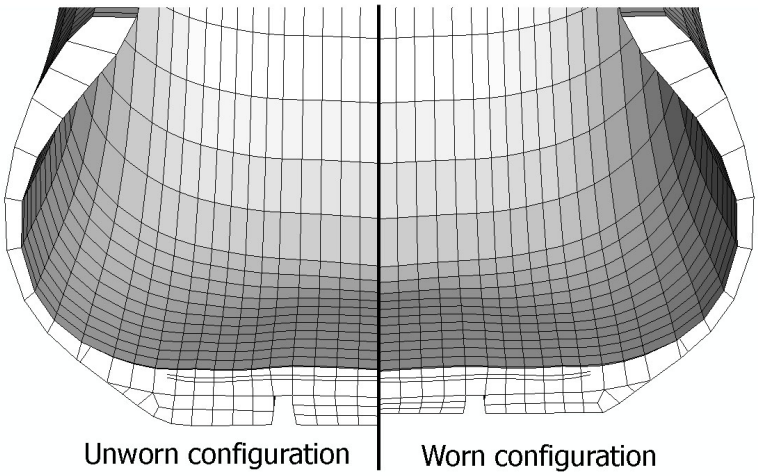


Figure 7: Tread profile in the unworn and worn configurations using the ALE approach.

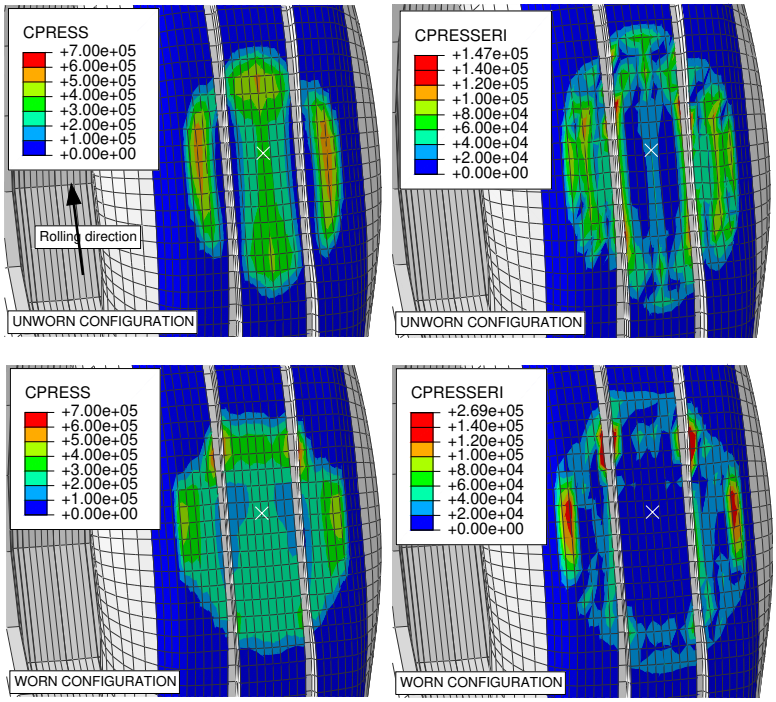


Figure 8: Footprint contact pressure and contact pressure error indicator, in the unworn and worn configurations.

Dynamic analysis of an air-filled tire with rolling transport effects

Products: Abaqus/Standard

Overview

This example examines the effect of steady-state rolling transport on the acoustic response of a tire and its associated air cavity after it has been subjected to the inflation pressure and footprint load.

The air cavity resonance in a tire is often a significant contributor to the vehicle interior noise, particularly when the resonance of the tire couples with the cavity resonance. This coupled resonance phenomenon, however, is affected by the rotating motion in the fluid and the solid. This example extends the analyses of *Coupled acoustic-structural analysis of a tire filled with air* to include rolling transport effects in the tire and air. The acoustic cavity is modeled as part of an axisymmetric model, which is inflated, revolved, reflected, and deformed to obtain a footprint in a manner consistent with the aforementioned example.

Problem description

A detailed description of the tire model is provided in *Symmetric results transfer for a static tire analysis*. We model the rubber as an incompressible hyperelastic material. Viscoelasticity in the material is ignored in this example.

The air cavity in the model is defined as the space enclosed between the interior surface of the tire and a cylindrical surface of the same diameter as the diameter of the bead. A cross-section of the tire model is shown in *Figure 1*. The values of the bulk modulus and the density of air are taken to be 426 kPa and 3.6 kg/m^3, respectively, and represent the properties of air at the tire inflation pressure.

The simulation assumes that both the road and rim are rigid. We further assume that the contact between the road and the tire is frictionless during the preloading analyses. However, we use a nonzero friction coefficient in the subsequent coupled acoustic-structural analyses.

To assess the effect of rolling motion on the dynamics of the coupled tire-air system, we first generate dynamic results for the stationary tire. In a subsequent analysis the tire and air are set into rolling motion, and corresponding dynamic results are obtained.

Model definition

We use a tire cross-section that is identical to that used in the simulation described in *Symmetric results transfer for a static tire analysis*. The air cavity is discretized using linear acoustic elements and is coupled to the structural mesh using a surface-based tie constraint with the secondary surface defined on the acoustic domain. We model the rigid rim by applying fixed boundary conditions to the nodes on the bead of the tire, while the interaction between the air cavity and rim is modeled by a traction-free surface; that is, no boundary conditions are prescribed on the surface.

We first create an axisymmetric mesh of half of the cross-section of the tire and air, then revolve it into half-symmetry. Symmetric model generation and symmetric results transfer, together with a static analysis procedure, are used to generate the preloading solution, which serves as the base state in the subsequent coupled acoustic-structural analyses.

In the first coupled analysis we reflect the revolved tire and air model into a full three-dimensional configuration. We then compute the real eigenvalues of the stationary tire and air cavity system. During frequency extraction steps, fixed boundary conditions are imposed automatically on the tire-road interface in the contact normal direction. Fixed boundary conditions are also applied in the tangential direction for points that are sticking.

Points that are slipping are free to move in the tangential direction. This analysis is followed by a direct steady-state dynamic analysis in which we obtain the response of the tire-air system subjected to imposed harmonic motion of the road.

In the second coupled analysis the reflected model is restarted from the first step, after the footprint equilibrium configuration had been established. The steady-state transport procedure is used to obtain the free-rolling state of the tire at 60 km/h. The corresponding magnitude of the transport velocity is determined independently in a separate analysis. In this step the acoustic flow velocity signifies that the acoustic medium is also in rotational motion. It is assumed that the air inside the tire rotates with the same angular velocity as the tire. The direct steady-state analysis, with similar parameters to those used in the stationary case, is repeated.

Additional analyses using a substructure that was generated with the effect of acoustic flow velocity are also included.

Loading

The loading sequence for computing the footprint solution is identical to that discussed in *Symmetric results transfer for a static tire analysis*. The simulation starts with an axisymmetric model, which includes the mesh for the air cavity. Only half the cross-section is modeled. The inflation pressure is applied to the structure using a static analysis. In this example the application of pressure does not cause significant changes to the geometry of the air cavity, so it is not necessary to update the acoustic mesh. However, we perform adaptive mesh smoothing after the pressure is applied to illustrate that the updated geometry of the acoustic domain is transferred to the three-dimensional model when symmetric results transfer is used.

The axisymmetric analysis is followed by a reflection—symmetric three-dimensional analysis in which the footprint solution is obtained. The footprint load is established over several load increments. The deformation during each load increment causes significant changes to the geometry of the air cavity. We update the acoustic mesh by performing five mesh sweeps after each converged structural load increment using the adaptive mesh domain. At the end of this analysis sequence we activate friction between the tire and road using a change to friction properties. This footprint solution, which includes the updated acoustic domain, is transferred to a full three-dimensional model. This model is used to perform the coupled analysis. In the first coupled analysis we extract the eigenvalues of the undamped system, followed by a direct-solution steady-state dynamic analysis in which we apply a harmonic excitation to the reference node of the rigid surface that is used to model the road.

In the stationary and rolling analyses we compute the response of the coupled system in the same frequency range used in *Coupled acoustic-structural analysis of a tire filled with air* (200 to 260 Hz). This band includes the natural frequencies of the fore-aft and vertical acoustic modes, at 225.67 Hz and 230.94 Hz, respectively.

The model is excited by a boundary condition specified at the road reference node in the direct-solution steady-state dynamic step. A small amount of stiffness proportional damping is applied to the rubber to avoid computing unbounded response at the eigenfrequencies.

Results and discussion

The characteristic frequencies of the coupled tire-air system are affected by the rolling motion. Generally, we expect a mode observed in the stationary coupled tire-air system to convert to a pair of modes, corresponding to forward and backward wave travel. This does not always occur in complex systems, because the stationary modes are not all affected by the rolling motion to a similar degree. However, the mode split described above can be observed for several of the structural modes and the fundamental acoustic modes of the cavity. For an observer in a nonrotating reference frame attached to the axle of the tire, similar to the reference frame used by the steady-state transport procedure, these modes appear as waves traveling clockwise and counterclockwise along the circumference of the tire. The mode corresponding to forward wave travel increases in frequency, whereas the mode corresponding to backward wave travel decreases in frequency. The modes of a stationary tire appear as static vibrations. The resonant frequencies of the stationary case are computed using the real-valued frequency extraction procedure. The complex frequency procedure will yield almost identical results for the stationary case, since the damping used in this example is relatively low. However, for the rolling analysis the

complex frequency procedure must be used to obtain accurate results using all the element contributions due to rotation. In *Table 1* an example is shown of a pair of structural and acoustic modes splitting into a pair of corresponding modes in the rolling case. The structural mode is a radial mode of circumferential order two. If there is no footprint loading, the stationary case will predict identical frequencies for these modes; however, the split would still be observed for a rotating tire. The modes are shown in *Figure 2*, *Figure 3*, and *Figure 4*. Similar behavior is observed for radial modes of higher order.

Figure 5, *Figure 6*, and *Figure 7* show the response of the structure to the imposed vertical motion at the spindle. *Figure 5* compares the acoustic response, at the crown, of the coupled tire-air system for the stationary case and the rolling case at 60 km/h. *Figure 6* shows the fore-aft reaction force, and *Figure 7* shows the vertical reaction force.

These figures further show that the rolling motion of the solid has a very strong influence on the behavior of the coupled system and that the rolling motion of the air exerts a similarly strong effect in the frequency range observed here. In particular, the resonances affecting the reaction force occur at different frequencies for the stationary and rolling cases. The resonances observed in the reaction force frequency response diagram also proliferate as rolling is introduced since waves traveling against and with the direction of rolling propagate at different speeds with respect to the observer. For a stationary tire, a vertical excitation to the road produces negligible fore-aft reaction forces. However, the fore-aft reaction forces due to a vertical excitation are significant in the case of a rolling tire.

The same effect can be shown when using a substructure generated with the acoustic flow velocity. The tire model with a coarser mesh is used following the same pattern of actions: axisymmetric model followed by the revolution, reflection, and the steady-state transport analysis for a rolling tire. The substructure is used within the frequency range of 200–250 Hz using a direct steady-state dynamic procedure. The same reaction force split resonance can also be observed for the full finite element model used in place of the substructure.

Input files

sst_acoustic_axi.inp	Axisymmetric model, inflation analysis.
sst_acoustic_rev.inp	Partial three-dimensional model, footprint analysis.
sst_acoustic_refl.inp	Full three-dimensional model, coupled structural-acoustic analyses, no transport effects.
sst_acoustic_roll100.inp	Full three-dimensional model, coupled structural-acoustic analyses, transport effects at moderate speed.
tiretransfer_node.inp	Nodal coordinates for the axisymmetric tire mesh.
tire_acoustic_air.inp	Mesh data for the axisymmetric acoustic mesh.
sst_acoustic_axi_sm.inp	Coarse axisymmetric model, inflation analysis.
sst_acoustic_rev_sm.inp	Coarse partial three-dimensional model.
sst_acoustic_refl_sm.inp	Coarse full three-dimensional model, coupled structural-acoustic analyses, no transport effects.
sst_acoustic_roll_sm.inp	Coarse full three-dimensional model, coupled structural-acoustic analyses, transport effects at moderate speed.
P_tire_acoustic_air.inp	Coarse mesh data for the axisymmetric acoustic mesh.
substracous_afv_sm_gen.inp	Frequency extraction and substructure generation, including acoustic flow velocity.
substracous_afv_sm_use.inp	Direct steady-state analysis with substructure.

Tables

Table 1: Example of eigenvalue splitting.

Mode description	Stationary coupled air-tire	Rolling coupled air-tire
Structural (radial, circumferential order 2)	95.467 Hz, 99.474 Hz	86.69 Hz, 103.77 Hz
Acoustic (fundamental modes of the acoustic cavity)	225.69 Hz, 230.96 Hz	220.38 Hz, 237.34 Hz

Figures

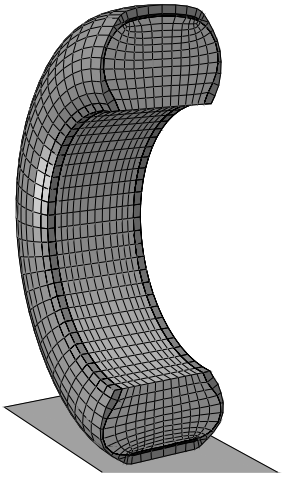


Figure 1: Cross-section of tire and air.

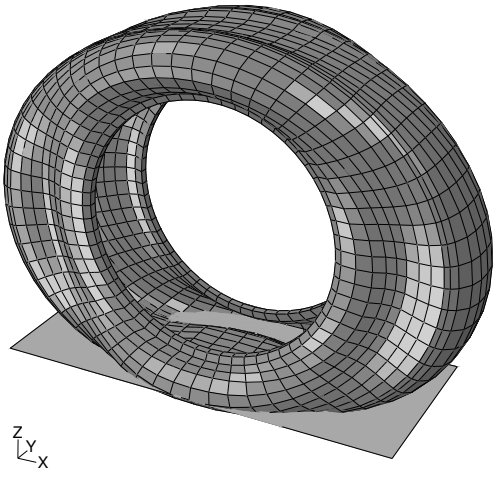


Figure 2: Radial mode, stationary case, at 95.46 Hz.

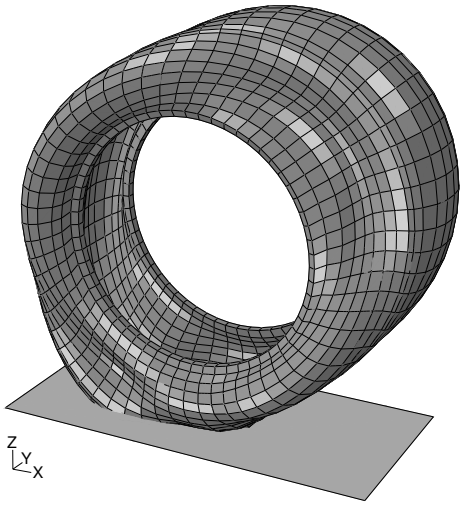


Figure 3: Radial mode, rolling case, backward, at 86.7 Hz (value at angle 90°).

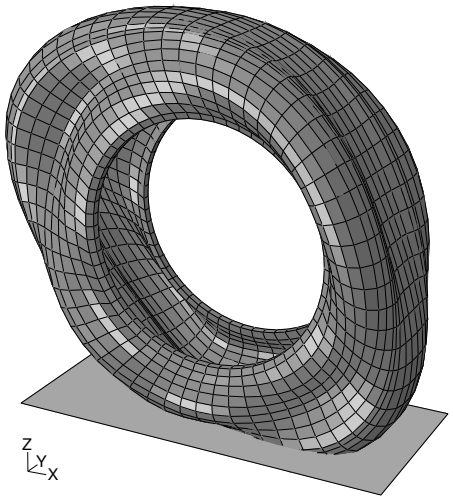


Figure 4: Radial mode, rolling case, forward, at 103.8 Hz (value at angle 90°).

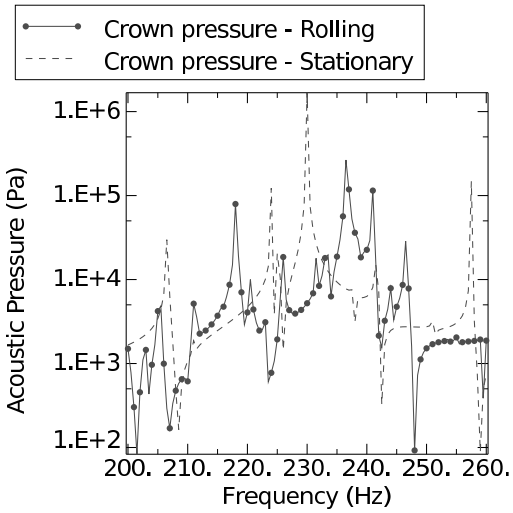


Figure 5: Acoustic pressure at crown due to imposed vertical motion.

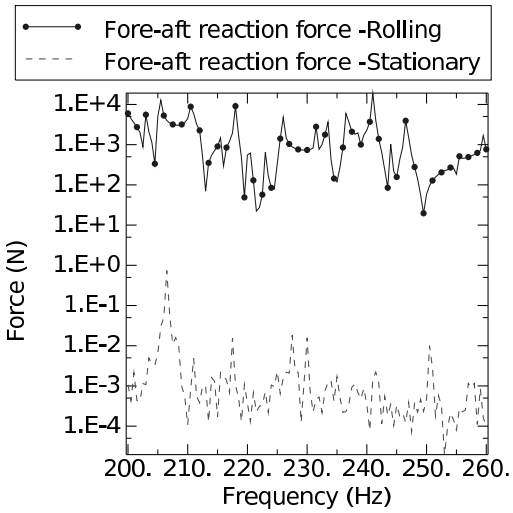


Figure 6: Fore-aft reaction force due to road displacement excitation.

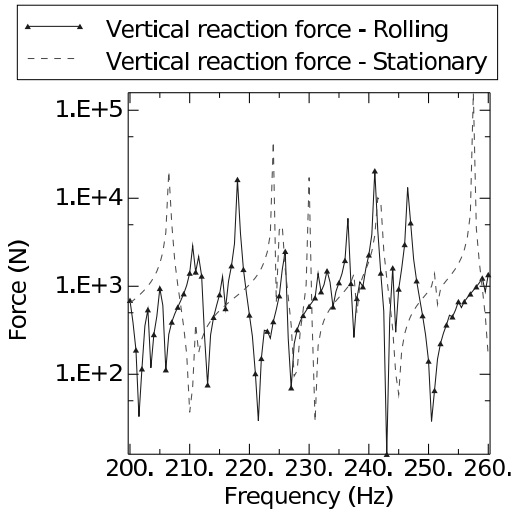


Figure 7: Vertical reaction force due to road displacement excitation.

Acoustics in a circular duct with flow

Products: Abaqus/Standard

Overview

This example illustrates the use of the symmetric model generation and symmetric results transfer capabilities to create an acoustic model.
In addition, it illustrates the use of the steady-state transport capability in Abaqus to specify a rotational velocity in acoustic media and to analyze the acoustic field subject to the effects of rotational flow. A tire-shaped acoustic cavity is analyzed.

Problem description and model definition

The model is a simplified version of the acoustic cavity in a vehicle tire. The cross-section of the cavity is established using axisymmetric elements, and these elements are assigned material properties corresponding to air. A dummy step is used to establish the various files used in the subsequent three-dimensional analysis steps.

The complete toroidal cavity is created using a revolved symmetric model definition to form a circumferentially uniform mesh of 60 segments around the circle. Steady-state dynamic, eigenfrequency extraction, and complex frequency extraction procedures use the acoustic flow velocity to include the effects of the rotational flow.

The air cavity is analyzed at rest, at 28.0 radians per second (corresponding to 100 kilometers per hour), and at 320 radians per second (corresponding to Mach 0.3 at the outer edge of the cavity).

The steady-state dynamic results are computed at frequencies between 150 and 370 Hz. For the complex analyses, modes are requested between 45 and 1000 Hz.

Loading

In the time-harmonic dynamic analysis, a unit imaginary concentrated volumetric acceleration acoustic load is applied to a pair of nodes at the base of the torus. No boundary conditions are applied.

In the real and complex frequency analyses, no acoustic loads or boundary conditions are applied.

Results and discussion

Results from the steady-state dynamic analysis show the effect of the rotating flow on the acoustic pressure field. Strong effects in the field develop as speed increases.

Results from the real frequency analysis show some effect of the rotating flow on the acoustic pressure modes and the frequencies. However, real frequency analysis ignores the important convective term in the formulation, which is complex-valued. Consequently, real frequency analysis of rotating acoustic media should be regarded as a preliminary step for the complex frequency computation, and the results of the real frequency analysis should not be assigned much physical significance. Complex frequency analysis uses the entire, unsymmetric complex acoustic element formulation. The complex contributions are, however, antisymmetric; therefore, the resulting resonant frequencies computed by the complex frequency procedure are pure imaginary; that is, there is no real part of the eigenvalues associated with gain or loss of energy. This is consistent with the physics of acoustics in rotating media. The effect of the flow on the eigenvalues is evident in the apparent splitting of the resonances as speed increases.

Input files

exa_acrotflowaxi.inp	Axisymmetric model.
exa_acrotflow3dssdd.inp	Full three-dimensional model, steady-state dynamic analysis.
exa_acrotflow3dfreq.inp	Full three-dimensional model, real and complex frequency analysis.

Figures

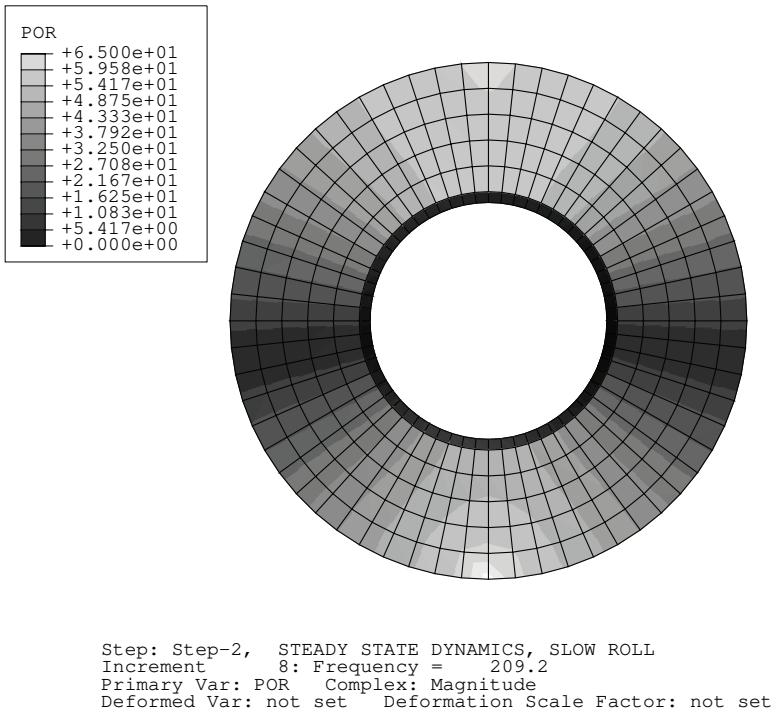


Figure 1: Steady-state dynamic response at 209 Hz, very low flow velocity.

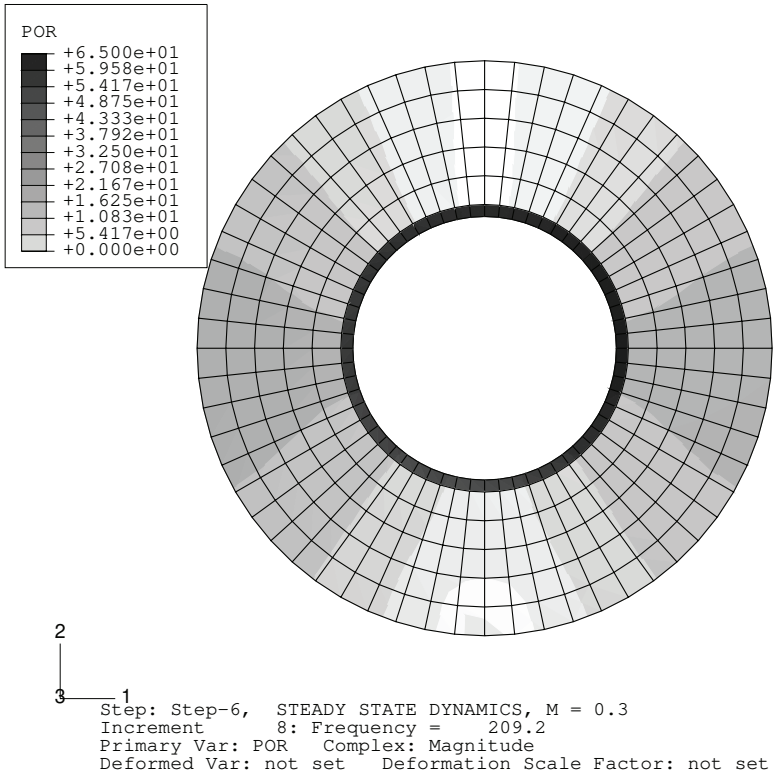


Figure 2: Steady-state dynamic response at 209 Hz, flow velocity of Mach 0.3.

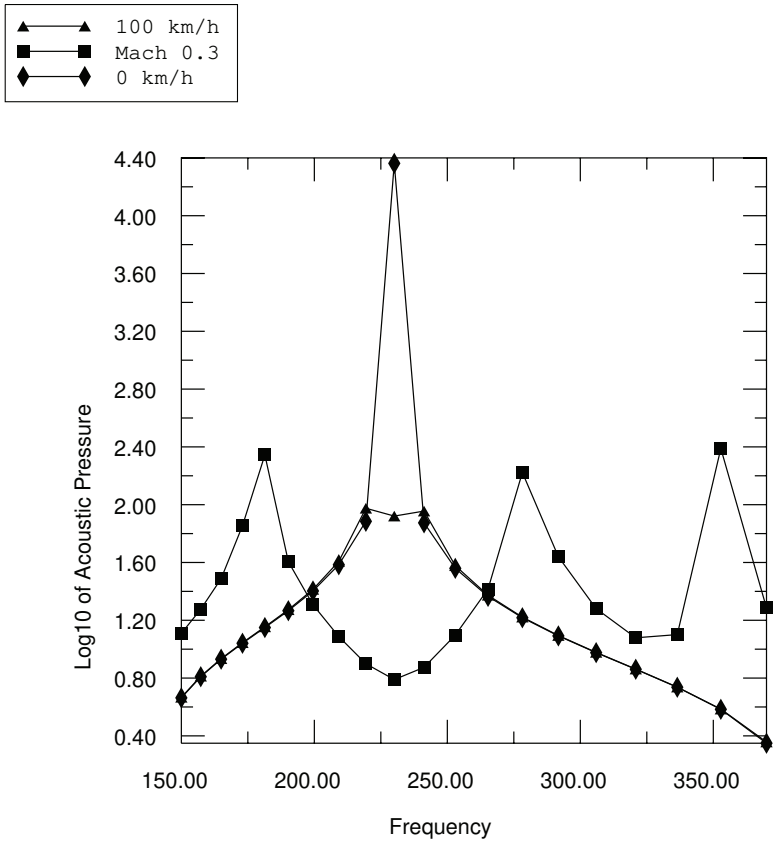


Figure 3: Steady-state dynamic response at node 13204, flow velocities from slow to Mach 0.3.

Vehicle analyses

In this section:

- *Inertia relief in a pick-up truck*
- *Substructure analysis of a pick-up truck model*
- *Display body analysis of a pick-up truck model*
- *Continuum modeling of automotive spot welds*

Inertia relief in a pick-up truck

Products: Abaqus/Standard

References:

- *Inertia Relief*

Overview

This example shows how to perform inertia relief in a static analysis in Abaqus/Standard.
The example involves stopping a pick-up truck, moving with an initial velocity of 50.0 km/h (13.89 m/s), by applying braking loads. Inertia relief is used here to supply inertia forces in a static analysis that oppose the braking loads specified in the model. The solution provides the rigid body deceleration and the static stresses in the pick-up truck. For comparison purposes a dynamic analysis is performed with the same initial velocity and braking loads.

Problem description and model definition

A 1994 Chevrolet C1500 pick-up truck (see *Figure 1*) is modeled using approximately 55,000 elements. The model was obtained from the Public Finite Element Model Archive of the National Crash Analysis Center at George Washington University. The finite element model was converted into an Abaqus/Standard input file, and several missing constraints were added to carry out the analyses. The model consists of various parts—such as cabin, truck bed, doors, etc.—which are meshed with shell elements, three-dimensional beam elements, and three-dimensional solid elements. The parts are attached with connector elements, coupling elements, and multi-point constraints.

The materials used in the truck model are idealized as elastic or elastic-plastic. Suitable adjustments are made to the material properties to account for unmodeled features of various parts such as the internal details of the engine, gearbox, etc. A summary of the material properties and the parts for which they are used is given in *Table 1* and *Table 2*.

Rigid body definitions are used for brakes and brake assemblies to take advantage of the high stiffness of these parts relative to other parts. Connector elements are used to model kinematic constraints governing relative motions between various parts (see *Substructure analysis of a pick-up truck model* for details).

The finite element model of the truck is oriented such that the positive 1-direction goes from the rear to the front of the truck, the positive 2-direction goes from the passenger (right-hand) side to the driver (left-hand) side, and the positive 3-direction is upward. In this system the braking loads are applied at the respective wheel spindles as concentrated forces in the negative 1-direction.

To simplify the analysis, normal contact between the tires and the road surface is modeled through spring elements that have one node connected to the wheel spindle and the other node fixed against displacement in the 3-direction and kinematically constrained to the wheel spindle in the other directions. Friction between the tires and the road surface is assumed to be nonexistent. This allows the truck to translate freely in the 1- and 2-directions and rotate freely about the 3-direction; the constraints on the spring nodes prevent translation in the 3-direction and rotation about the 1- and 2-directions.

Loading

A separate static analysis is performed to obtain the correct initial configuration and stress distribution under the applied gravity load. The details of this analysis are explained in *Substructure analysis of a pick-up truck model*. This gives us the base state for the analysis of interest.

The total braking load for the truck moving at 13.89 m/s is computed by assuming the truck to be a rigid body that comes to rest over a distance of 20 m after the brakes are applied. This gives a deceleration of 4.82 m/s^2 in the 1-direction for the truck. The total mass of the truck as computed from the finite element analysis is 1.72×10^3 kg, which gives the total inertial force resisted by the brakes (or braking load) as 8.30 kN. Assuming that the front brakes provide 75% of the total resistance and the rear brakes provide the remaining 25%, the braking load for each of the front wheels is 3.11 kN and the braking load for each of the rear wheels is 1.04 kN. The four braking loads applied to the truck are balanced in a static analysis with an inertia relief load. The inertia relief load represents the dynamic effects (not modeled otherwise in a static analysis) of a constant deceleration from the truck's travel velocity to a complete stop.

Since the truck is free to translate in the 1- and 2-directions and rotate about the 3-direction, inertia relief is performed in these three directions. The other directions are constrained by boundary conditions as explained in the previous section.

For comparison purposes a transient dynamic analysis is also performed (after the initial static equilibrium under gravity load) in which the truck is accelerated from zero velocity to the final uniform velocity of 13.89 m/s. This dynamic analysis step is followed by another dynamic analysis step in which the braking loads are applied to bring the truck to a complete stop. The braking loads are ramped up smoothly from zero to the maximum value over 0.5 seconds and then kept constant for 2.88 seconds—the time required to bring the truck to rest from the initial velocity of 13.89 m/s with an average deceleration of 4.82 m/s^2. To minimize the analysis time, substructures are used in the dynamic analysis for all deformable parts except the chassis and suspension components, which are modeled as fully deformable since they are the parts that show significant stresses.

Results and discussion

The results for inertia relief in the pick-up truck model with braking loads show that the truck decelerates at 4.83 m/s^2 in the 1-direction. The truck has an angular acceleration of 0.01 rad/s^2 about the 3-direction at the center of mass due to asymmetry in the distribution of mass. The vertical displacements at the wheel spindles (the front wheel spindles dip about 0.7 mm, and the rear wheel spindles rise about 0.7 mm without loss of contact between tires and the road surface) indicate that the truck pitches forward due to the braking action. A plot of the von Mises stress shown in *Figure 2* indicates that the largest stresses occur in the suspension components and the regions where the suspension components are connected to the chassis. Plots of the active yield flag and equivalent plastic strains (not shown) indicate that there is no plastic yielding in any part of the truck.

The results for the transient dynamic analysis indicate that the average deceleration after the full braking load has been applied is around 4.94 m/s^2 in the 1-direction and the average angular acceleration about the 3-direction is 0.03 rad/s^2. The truck pitches forward with the front wheel spindles dipping about 0.7 mm and the rear wheel spindles rising about 0.7 mm in the braking load step. The von Mises stress for the dynamic analysis, shown in *Figure 3*, shows a distribution similar to that obtained for inertia relief. There is no plastic yielding in the chassis or suspension components.

Inertia relief relies on the assumption that the body undergoing loading is free to translate and rotate as a rigid body. Therefore, no external or internal constraints are allowed in the free directions (with the exception of the case where statically determinant boundary conditions are applied and all available directions are considered inertia relief directions). In a complex model like the pick-up truck, with various kinematic constraints and large geometry changes, it is necessary to ensure that the base state for the step including inertia relief is converged to a tight residual tolerance. If it is not converged to a tight tolerance, the out-of-balance forces and moments in the base state will act as internal constraints on rigid body motions. Hence, such unequilibrated forces and moments may prevent a geometrically linear or nonlinear analysis from converging. In this example the solution controls tighten the convergence tolerance in the gravity load step preceding the inertia relief step.

The comparison of results for inertia relief and dynamic analysis of the truck shows that inertia relief is an inexpensive alternative to dynamic analysis for obtaining the steady-state response of a dynamic system for certain loading situations. In this braking analysis, for example, the static analysis with inertia relief runs about 10 times faster than the general deformable transient dynamic simulation.

Input files

irltr_brake_irl.inp	Static equilibrium analysis for gravity load and initial stresses followed by inertia relief with braking loads.
irltr_brake_dyn.inp	Static equilibrium analysis for gravity load and initial stresses followed by dynamic acceleration to uniform velocity and deceleration with braking loads.

The model data are contained in multiple smaller files and referenced in the main input files. The referenced file names are given in *Substructure analysis of a pick-up truck model*.

Tables

Table 1: Properties for elastic-plastic materials used in the truck model.

Elastic-Plastic Material Name	Properties				Part Name
	E (N/m^2)	ν	σ_y (N/m^2)	ρ (kg/m^3)	
Steel	2.1×10^{11}	0.3	2.7×10^{8}	7.89×10^{3}	rail (chassis), engine oil box, radiator mounting, fenders, wheel housings, cabin, bed, fan center, fuel tank, rear rim, steering support, battery tray, seat track, radiator outer
Steel	2.1×10^{11}	0.3	3.5×10^{8}	7.89×10^{3}	engine mountings, radiator mountings, radiator, fender mountings, hood, doors, cabin hinges
Plastic	2.8×10^{9}	0.3	4.5×10^{7}	1.2×10^{3}	fan cover
Glass	7.6×10^{10}	0.3	1.38×10^{8}	2.5×10^{3}	windows, windshield
Plastic	3.4×10^{9}	0.3	1.0×10^{8}	1.1×10^{3}	radiator side block
Plastic	3.4×10^{9}	0.3	1.0×10^{8}	7.1×10^{3}	dashboard interior

Table 2: Properties for elastic materials used in the truck model.

Elastic Material Name	Properties			Part Name
	E (N/m^2)	ν	ρ (kg/m^3)	
Steel	2.1×10^{11}	0.3	7.89×10^{3}	A-arm mountings, fan, door lock beams, headrest connector beams, radiator mounting beams, oil pan beams, rear axle, drive shaft, steering, A-arm-rim connectors, A-arm-rail connectors, bed-rail connector, dashboard support, steering column, rail connector, brakes, gearbox CV joint, front rim
Steel	1.2×10^{11}	0.3	3.89×10^{3}	engine gearbox
Steel	2.1×10^{10}	0.3	1.82×10^{3}	engine front
Steel	2.1×10^{11}	0.3	2.5×10^{3}	door lock
Rubber	2.461×10^{9}	0.323	8.0598×10^{3}	tires
Steel	2.1×10^{11}	0.3	3.5765×10^{3}	rear suspension

Elastic Material Name	Properties			Part Name
	E (N/m^2)	ν	ρ (kg/m^3)	
Steel	2.1×10^{11}	0.3	2.089×10^{4}	brake assembly
Steel	2.1×10^{11}	0.3	6.911×10^{3}	brake assembly
Rubber-Metal Composite	2.1×10^{11}	0.3	1.96×10^{3}	battery
Foam	2.0×10^{9}	0.3	2.527×10^{2}	seat bottom
Foam	2.0×10^{9}	0.3	7.55×10^{2}	seat top
Foam	2.0×10^{9}	0.3	1.69×10^{2}	seat headrest

Figures

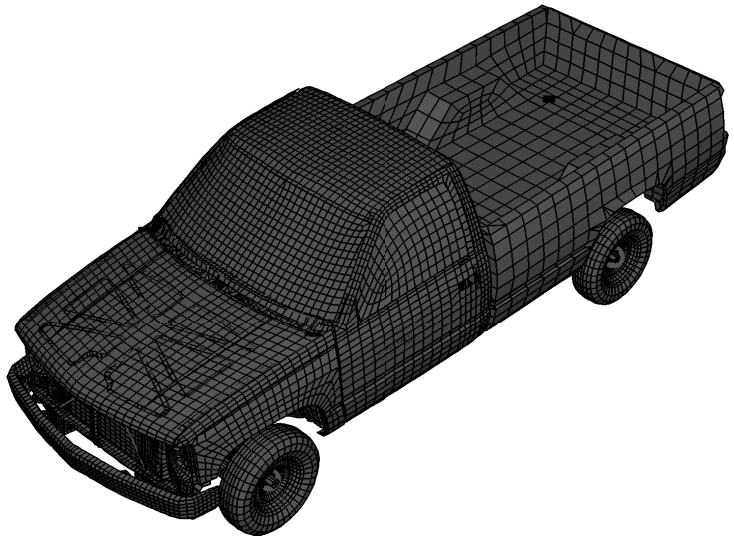


Figure 1: Finite element model of 1994 Chevrolet C1500 pick-up truck.

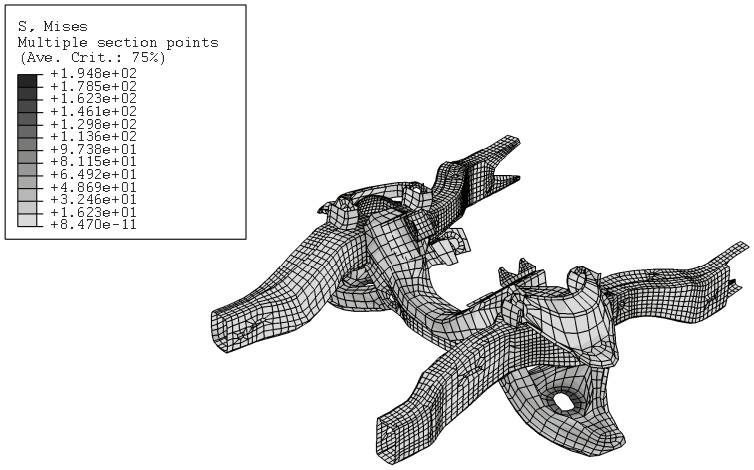


Figure 2: Von Mises stress in front chassis and A-arm components for inertia relief.

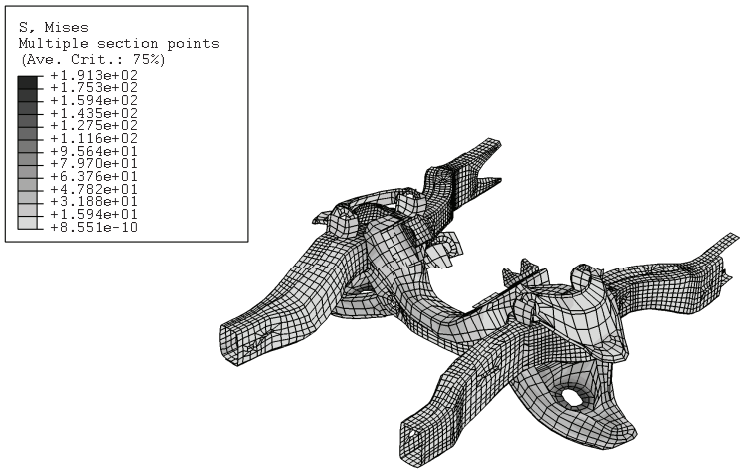


Figure 3: Von Mises stress in front chassis and A-arm components for dynamic analysis.

Substructure analysis of a pick-up truck model

Products: Abaqus/Standard

Overview

This example illustrates the use of the substructure capability in Abaqus to efficiently simulate the vehicle dynamics of a detailed pick-up truck model traveling over road bumps.

The pick-up truck model geometry described in *Inertia relief in a pick-up truck* is used in this example. The model is organized as a collection of individual parts that are connected together. Twenty substructures are then created, one for each part that may undergo large motions but for which it is reasonable to assume small-strain elastic deformation (for example, the chassis). Several parts that may deform nonlinearly (for example, leaf springs for the rear suspension or the stabilizer bar in the front) are modeled using the usual general nonlinear modeling options. Connection points are created for each part using a coupling constraint. The parts are then attached together using appropriate connector elements. A simplified CALSPAN tire model is used (*UEL*) to model the radial forces in the tires. The vehicle is loaded statically by gravity, accelerated in a dynamic step on a flat road, and run over bumps. Without stress recovery in the substructures the substructure analysis runs an estimated 120 times faster than an equivalent analysis without substructures.

Geometry and materials

The pick-up truck model (1994 Chevrolet C1500) discussed here is depicted in *Figure 1* riding over antisymmetric bumps. The model geometry, element connectivity, and material properties are obtained from the Public Finite Element Model Archive of the National Crash Analysis Center at George Washington University. The materials used are described in *Inertia relief in a pick-up truck*.

The model is organized as a collection of individual parts connected together. Most parts that undergo only small deformations in addition to a large rigid body motion are defined as substructures. Substructures are created for the following parts: the chassis, each of the four A-arms for the front suspension, each of the four wheels, the rear axle, the driveshaft, the engine/transmission, the cabin, each of the two doors, the hood, the seat, the front bumper, the truck bed, and the fuel tank.

The number of retained nodes for each substructure is determined primarily by its connection points with neighboring parts as illustrated in *Figure 2* for the cabin substructure. There are twenty points associated with this substructure that are used to connect the cabin to other parts in the model. There are six retained nodes on the cabin bottom (connections to the chassis); three retained nodes for the hood connections (two hinges and the hood lock); three retained nodes for each of the two door connections (two hinges in the front and the door lock in the back); four retained nodes for the seat connections; and one node retained at the center of mass of the vehicle used for yaw, pitch, and roll measurement purposes.

Several parts deform too much to be considered substructures and are modeled using regular elements. The leaf springs in the back and the stabilizer bar in the front are both modeled with beam elements. The front brake assemblies are modeled as rigid bodies.

The connections between parts are modeled using connector elements. JOIN and REVOLUTE connectors are used to model the hinges between each of the following parts: the A-arms and chassis, the doors and cabin, the hood and cabin, the wheels and knuckles, and the leaf springs and chassis. CARTESIAN and CARDAN connections with appropriately defined connector elasticity, connector damping, and connector friction behaviors are used to define some of the bushing connections (e.g., engine mounts). Two UNIVERSAL connectors are used to model the driveshaft connections to the transmission in the front and to the differential in the back. BEAM connectors are used to model rigid connections between parts. The available connector components of relative motion applied to an AXIAL connector are used to lock (or open) the doors and the hood. The available

connector components of relative motion applied to a SLOT connector are used to specify the steering by moving the steering rack. The struts are modeled using an AXIAL connection by specifying approximate nonlinear elasticity and damping. Several of the suspension-related parts are shown in *Figure 3*.

The radial forces in the tires are modeled approximately using a simplified CALSPAN tire model (Frik, Leister, and Schwartz, 1993) implemented via user subroutine *UEL*. A radial stiffness of 600 N/mm is considered.

Models

For all analyses the gravity-loaded static equilibrium configuration is found first. Since the given mesh geometry corresponds to the gravity-loaded equilibrium position and data are not available for the pre-stress in the suspension springs and tires, the pre-stress forces have to be computed. To achieve this end, a separate static stress analysis with artificial properties for the suspension springs and artificial boundary conditions is first performed, as follows. The vehicle is supported with boundary conditions in the vertical direction at the four wheel spindles (where the tire *UEL*s will be connected) and fixed at the center of mass to prevent in-plane rigid body motion (degrees of freedom 1, 2, and 6). The stiffnesses of the suspension springs are increased artificially by a thousand times in this independent analysis to minimize deformation. The gravity load is then applied to obtain equilibrium stresses in the suspension components and reaction forces at the wheel spindles.

In the analysis of interest (with realistic properties and boundary conditions), the stresses and reaction forces obtained from the artificial static step are used as initial stresses in the suspension spring components and as pre-stress forces in the tires, respectively. A static gravity loading step is run to obtain an equilibrium configuration. This equilibrium configuration differs only slightly from the given initial geometry (the wheel spindles move laterally about two millimeters). Thus, the initial stress state in the suspension springs and tires accurately represents static equilibrium, and the vehicle is ready for dynamic loading.

The vehicle model is prescribed an initial velocity and then accelerated (0.5 g) to the desired velocity (5 m/sec or 7 m/sec) in a direct-integration implicit dynamic step. Once the “cruise” velocity is achieved, the truck model is run over symmetric or antisymmetric bumps (0.2 m high and 5.0 m long).

Results and discussion

In *Figure 4* a snapshot of the truck moving forward with a velocity of 7 m/sec (25.2 km/h) and “jumping” over symmetric bumps is shown. The wheels lose contact with the ground and then land again on the road (not shown).

More results are presented for the case when the truck is riding over antisymmetric bumps (see *Figure 1*). Stresses are recovered for the lower left A-arm substructure and shown in *Figure 5* when the front wheels have traveled 3.2 m over the bumps (for information on how to display results on the substructure, see “A script to visualize the element results for beams and shells within substructure in Abaqus” in the Dassault Systèmes Knowledge Base at *http://support.3ds.com/knowledge-base/*). The radial forces on the tires are shown in *Figure 6* beginning from the moment when the front tires are about to go over the bumps. A zero radial force indicates that the tire is out of contact. The yaw, pitch, and roll angles recorded using a CARDAN connector attached to the model at its center of mass are shown in *Figure 7*.

The advantage of using substructures instead of regular deformable elements becomes obvious when the total times needed to complete these types of analyses are compared. A full analysis using regular elements has not been performed since approximately five CPU days would be necessary to complete either of the two analyses discussed above. This total time was estimated by running a few increments, estimating the time needed per iteration, and then multiplying the time per iteration by the total number of iterations needed to complete the analysis. Using these estimates, the substructure analysis is up to 120 times faster than the regular mesh analysis, depending on the amount of recovery performed for each substructure.

The **abaqus substructurecombine** execution procedure can combine model and results data from two substructure output databases into a single output database. For more information, see *Combining Output from Substructures*. To combine the substructure output databases, at least one frame of field output must be requested.

Input files

tr_entire_truck_in_phase.inp	Substructure global analysis of the truck going over in-phase bumps.
tr_entire_truck_anti_phase.inp	Substructure global analysis of the truck going over antiphase bumps.
tr_road_antiphase.inp	Definition of antiphase road profile.
tr_road_inphase.inp	Definition of in-phase road profile.
tr_all_nodes.inp	All node definitions.
tr_parameters.inp	**PARAMETER* definitions.
tr_materials.inp	Material definitions.
tr_materials_plastic_irl.inp	Material definitions.
tr_initial_stress.inp	Initial stress definition for the leaf springs.
tr_lock_doors_and_hood.inp	**CONNECTOR MOTION* that keeps the doors and the hood locked.
tr_substruct_recovery.inp	Output definitions for the substructures.
tr_brake_front_left.inp	**RIGID BODY* definition of the front left brake assembly.
tr_brake_front_right.inp	**RIGID BODY* definition of the front right brake assembly.
tr_parameters_inphase.inp	**PARAMETER* definitions for the in-phase analysis.
tr_parameters_antiphase.inp	**PARAMETER* definitions for the antiphase analysis.
tr_readme.inp	Description of procedure used to run the jobs.

User subroutine

exa_tr_radial_uel.f	*UEL* used to define the tire model.

Substructure generation files

tr_chassis_gen.inp	Chassis.
tr_retained_chassis.inp	Retained nodes for the chassis.
tr_susp_lower_arm_left_gen.inp	Lower left A-arm.
tr_susp_lower_arm_right_gen.inp	Lower right A-arm.
tr_susp_upper_arm_left_gen.inp	Upper left A-arm.
tr_susp_upper_arm_right_gen.inp	Upper right A-arm.
tr_rear_axle_gen.inp	Rear axle.
tr_retained_rear_axle.inp	Retained nodes for the rear axle.
tr_engine_gen.inp	Engine and transmission.
tr_driveshaft_gen.inp	Driveshaft.
tr_cabin_gen.inp	Cabin and front fenders.
tr_retained_cabin.inp	Retained nodes for the cabin.
tr_hood_gen.inp	Hood.
tr_door_left_gen.inp	Left door.
tr_door_right_gen.inp	Right door.
tr_seat_gen.inp	Seat.
tr_bed_gen.inp	Truck bed.

tr_fuel_tank_gen.inp	Fuel tank.
tr_bumper_gen.inp	Front bumper.
tr_wheel_back_left_gen.inp	Rear left wheel.
tr_wheel_back_right_gen.inp	Rear right wheel.
tr_wheel_front_left_gen.inp	Front left wheel.
tr_wheel_front_right_gen.inp	Front right wheel.

Element definitions

tr_rear_susp_leaf_springs.inp	Rear leaf spring suspension.
tr_stabilizer_elts.inp	Front stabilizer bar.
tr_steering_rods_elts.inp	Steering rods and steering rack.
tr_chassis_elts.inp	Chassis.
tr_susp_lower_arm_left_elts.inp	Lower left A-arm.
tr_susp_lower_arm_right_elts.inp	Lower right A-arm.
tr_susp_upper_arm_left_elts.inp	Upper left A-arm.
tr_susp_upper_arm_right_elts.inp	Upper right A-arm.
tr_rear_axle_elts.inp	Rear axle.
tr_engine_elts.inp	Engine and transmission.
tr_driveshaft_elts.inp	Driveshaft.
tr_cabin_elts.inp	Cabin and front fenders.
tr_hood_elts.inp	Hood.
tr_door_left_elts.inp	Left door.
tr_door_right_elts.inp	Right door.
tr_seat_elts.inp	Seat.
tr_bed_elts.inp	Truck bed.
tr_fuel_tank_elts.inp	Fuel tank.
tr_bumper_elts.inp	Front bumper.
tr_wheel_back_left_elts.inp	Rear left wheel.
tr_wheel_back_right_elts.inp	Rear right wheel.
tr_wheel_front_left_elts.inp	Front left wheel.
tr_wheel_front_right_elts.inp	Front right wheel.

Multi-point constraint definitions

tr_chassis_mpc.inp	Chassis.
tr_engine_mpc.inp	Engine and transmission.
tr_cabin_mpc.inp	Cabin and front fenders.
tr_hood_mpc.inp	Hood.
tr_door_left_mpc.inp	Left door.
tr_door_right_mpc.inp	Right door.
tr_seat_mpc.inp	Seat.
tr_fuel_tank_mpc.inp	Fuel tank.
tr_bumper_mpc.inp	Front bumper.

Coupling definitions

tr_chassis_coup.inp	Chassis.
tr_susp_lower_arm_left_coup.inp	Lower left A-arm.
tr_susp_lower_arm_right_coup.inp	Lower right A-arm.
tr_susp_upper_arm_left_coup.inp	Upper left A-arm.
tr_susp_upper_arm_right_coup.inp	Upper right A-arm.
tr_rear_axle_coup.inp	Rear axle.
tr_engine_coup.inp	Engine and transmission.
tr_driveshaft_coup.inp	Driveshaft.
tr_cabin_coup.inp	Cabin and front fenders.
tr_hood_coup.inp	Hood.
tr_door_left_coup.inp	Left door.
tr_door_right_coup.inp	Right door.
tr_seat_coup.inp	Seat.
tr_bed_coup.inp	Truck bed.
tr_fuel_tank_coup.inp	Fuel tank.
tr_bumper_coup.inp	Front bumper.
tr_wheel_back_left_coup.inp	Rear left wheel.
tr_wheel_back_right_coup.inp	Rear right wheel.
tr_wheel_front_left_coup.inp	Front left wheel.
tr_wheel_front_right_coup.inp	Front right wheel.

Connector definitions

tr_conn_aarms_left.inp	Left A-arms.
tr_conn_aarms_right.inp	Right A-arms.
tr_conn_brake_left.inp	Left front brake assembly.
tr_conn_brake_right.inp	Right front brake assembly.
tr_conn_steering_rods.inp	Steering rods.
tr_conn_stabilizer.inp	Stabilizer bar.
tr_conn_leaf_springs.inp	Leaf springs.
tr_conn_engine.inp	Engine and transmission.
tr_conn_driveshaft.inp	Driveshaft.
tr_conn_cabin_to_chassis.inp	Cabin to chassis.
tr_conn_hood.inp	Hood.
tr_conn_door_left.inp	Left door.
tr_conn_door_right.inp	Right door.
tr_conn_seat.inp	Seat.
tr_conn_bed.inp	Truck bed.
tr_conn_fuel_tank.inp	Fuel tank.
tr_conn_bumper.inp	Front bumper.
tr_conn_wheels_back.inp	Rear wheels.
tr_conn_wheels_front.inp	Front wheels.

References

- Frik, S., G. Leister, and W. Schwartz, “Simulation of the IAVSD Road Vehicle Benchmark Bombardier Iltis with FASIM, MEDYNA, NEWEUL, and SIMPACK,” in Multibody Computer Codes in Vehicle System Dynamics, Ed. W. Kortum and R. S. Sharp, February 1993.

Figures

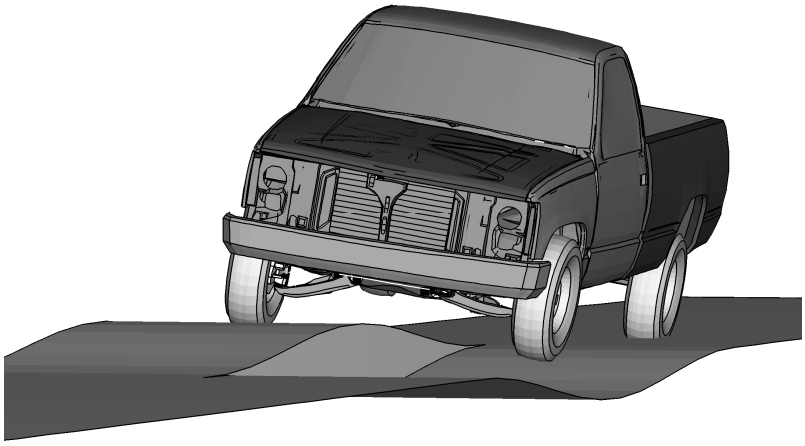


Figure 1: Substructure truck model going over antisymmetric (left-right) bumps.

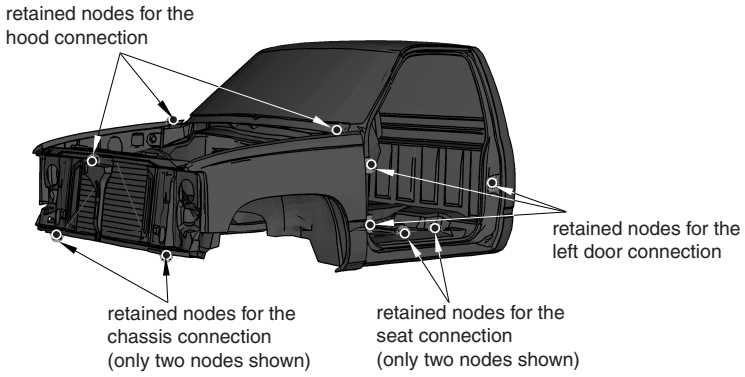


Figure 2: Substructure mesh for the cabin.

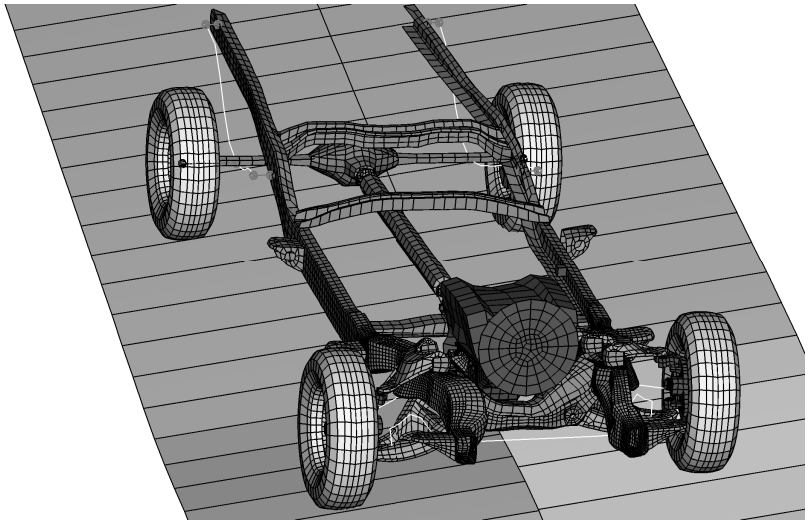


Figure 3: Chassis, suspension, and power-train-related parts.

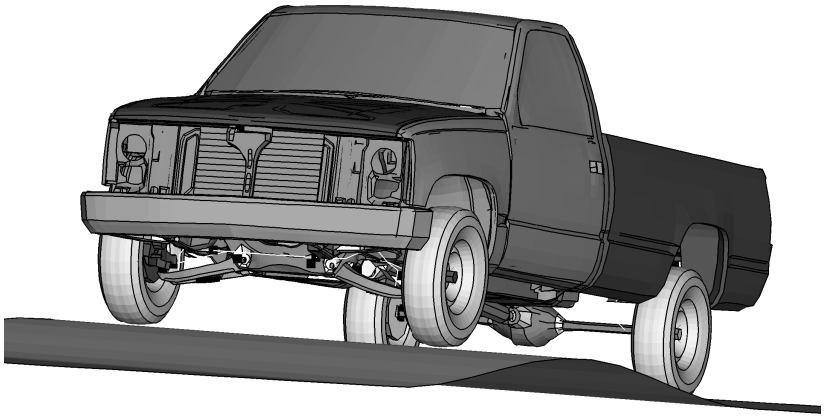


Figure 4: Substructure truck model going over symmetric (left-right) bumps.

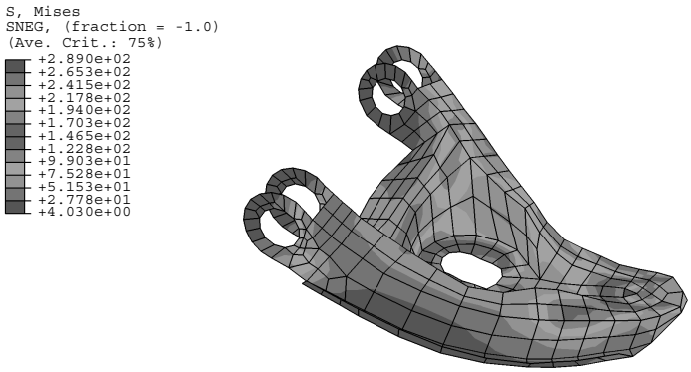


Figure 5: Recovered stresses in lower left A-arm.

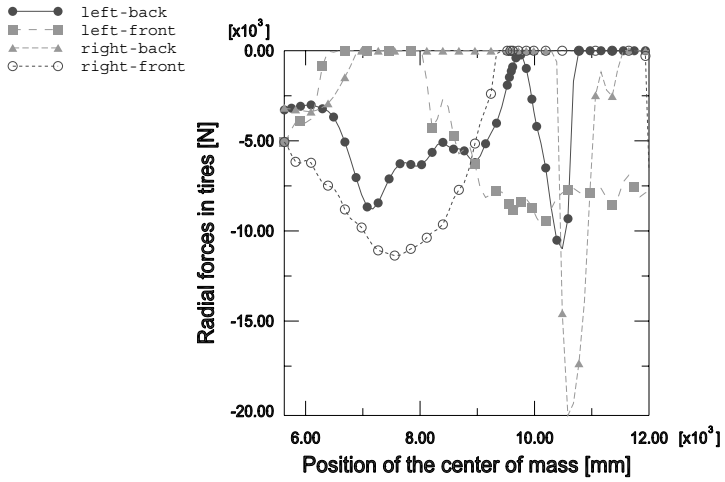


Figure 6: Tire radial forces for the truck going over antisymmetric bumps.

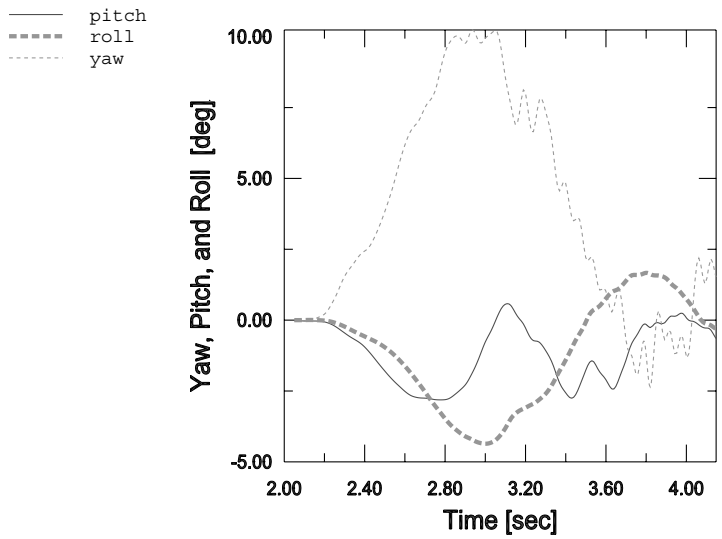


Figure 7: Yaw, pitch, and roll for the truck going over antisymmetric bumps.

Display body analysis of a pick-up truck model

Products: Abaqus/Standard

Overview

This example illustrates the use of the display body capability in Abaqus to simulate efficiently the vehicle dynamics of a detailed pick-up truck model driving over a symmetric road bump.

The pick-up truck model geometry is described in *Inertia relief in a pick-up truck*. The model is organized as a collection of several parts, twenty-two of which are converted to display bodies. The display bodies are connected using connectors. Several parts that may deform nonlinearly (e.g., the leaf springs for the rear suspension and the stabilizer bar in the front) are modeled using beam elements. The mass, center of mass, and rotary inertia are computed for every part by running a datacheck analysis on separate models, each containing one part. A mass and a rotary inertia element are then created at the reference point of the display body. The reference point of each display body is connected to its attachment points using connector elements. A simplified CALSPAN tire model is used to model the radial forces in the tires via user subroutine *UEL*. The vehicle is loaded statically by gravity, accelerated in a dynamic step on a flat road, and run over bumps. The results are then compared to corresponding analyses with substructures as in *Substructure analysis of a pick-up truck model*.

Geometry and materials

The problem discussed here is a model of a pick-up truck (1994 Chevrolet C1500). The model geometry, element connectivity, and material properties are obtained from the Public Finite Element Model Archive of the National Crash Analysis Center at George Washington University. The materials used are described in *Inertia relief in a pick-up truck*.

The model is organized as a collection of individual parts connected together. Most parts that undergo only small deformations in addition to a large rigid body motion are defined as display bodies; their deformation can be neglected since only the overall motion of the vehicle is of interest. A display body is created for each of the following parts: the chassis, each of the four A-arms of the front suspension, each of the four wheels, the rear axle, the driveshaft, the engine/transmission, the cabin, each of the two doors, the hood, the seat, the front bumper, the truck bed, and the fuel tank.

The number of attachment points for each display body is determined by its connection points with neighboring parts. The display bodies of the front-wheel assembly are shown connected at attachment nodes in *Figure 1*.

Several parts deform nonlinearly and are modeled using regular elements. For example, the leaf springs in the rear and the stabilizer bar in the front are modeled with beam elements.

To create the display bodies with appropriate attachment points and realistic material properties, the following steps are taken:

1. A datacheck analysis is run on every part with deformable elements to determine its mass, rotary inertia, and center of mass.
2. An instance of this part is used to create a display body.
3. A reference node created at the center of mass for this part is also used as the reference node for the corresponding display body.
4. A mass element and a rotary inertia element are created at the reference node.
5. The reference node is then connected to the attachment points of the part using either BEAM or CARTESIAN and CARDAN connectors with stiff elastic properties.

The connections between the attachment points of two display bodies are modeled using connector elements. JOIN and REVOLUTE connectors are used to model the hinges between each of the following parts: the A-arms and chassis, the doors and cabin, the hood and cabin, the wheels and knuckles, and the leaf springs and chassis. CARTESIAN and CARDAN connections with connector elasticity, connector damping, and connector friction behaviors are used to define some of the bushing connections (e.g., the engine mounts). Two UNIVERSAL connectors are used to model the driveshaft connections to the transmission in the front and to the differential in the back. BEAM connectors are used to model rigid connections between parts. In cases where the BEAM connectors form a closed loop, the model becomes overconstrained. In such situations CARTESIAN and CARDAN connectors with elasticity are used instead of some of the BEAM connectors. A connector motion is applied to an AXIAL connector to lock (or open) the doors and the hood. For steering the vehicle, a connector motion is applied to a SLOT connector and is used to specify the steering for the steering rack. The struts are modeled using an AXIAL connection by specifying approximate nonlinear elasticity and damping.

The radial forces in the tires are modeled approximately using a simplified CALSPAN tire model (Frik, Leister, and Schwartz, 1993) implemented via user subroutine *UEL*. A radial stiffness of 600 N/mm is considered. Friction is not accounted for in this user subroutine.

Models

The analysis is performed using the technique described in *Substructure analysis of a pick-up truck model*.

Results and discussion

In *Figure 2* a snapshot of the truck moving forward with a velocity of 7 m/sec (25.2 km/h) and going over a symmetric bump is shown. The wheels lose contact with the ground and then land again on the road (not shown).

The pitch and roll angles recorded using a CARDAN connector attached to the model at its center of mass are shown in *Figure 3*. The pitch angles are almost the same for the display body model and the substructure model (see *Substructure analysis of a pick-up truck model*) of a truck going over a symmetric bump with a difference of 1 degree. The difference is due to the fact that the display body model uses several CARTESIAN and CARDAN connectors with elasticity to remove overconstraints. These connectors introduce some compliance. The roll angles are almost zero for both models.

The advantage of using substructures or display bodies instead of regular deformable elements becomes obvious when the total times needed to complete these types of analyses are compared. It is estimated that a full analysis using regular elements would take approximately 5 days to complete. It is estimated that the substructure analysis is up to 120 times faster than the regular mesh analysis, depending on the amount of recovery performed for each substructure. If the total substructure generation and analysis time for the substructure model is compared to the analysis time for the display body model, the display body model runs 5 times faster than the substructure model. If only the analysis times are compared for both models, the display body model runs 2.5 times faster than the substructure model.

Input files

tr_entire_truck_in_phase_db.inp	Display body model of the truck going over in-phase bumps.
tr_parameters_db.inp	Parameter definitions.
tr_materials_db.inp	All material definitions.
tr_initial_stress_db.inp	Initial stress definition for the leaf springs.
tr_lock_doors_and_hood_db.inp	Connector motion that keeps the doors and the hood locked.
exa_tr_radial_uel_db.f	*UEL* used to define the tire model.

References

- Frik, S., G. Leister, and W. Schwartz, “Simulation of the IAVSD Road Vehicle Benchmark Bombardier Iltis with FASIM, MEDYNA, NEWEUL, and SIMPACK,” in Multibody Computer Codes in Vehicle System Dynamics, Ed. W. Kortum and R. S. Sharp, February 1993.

Figures

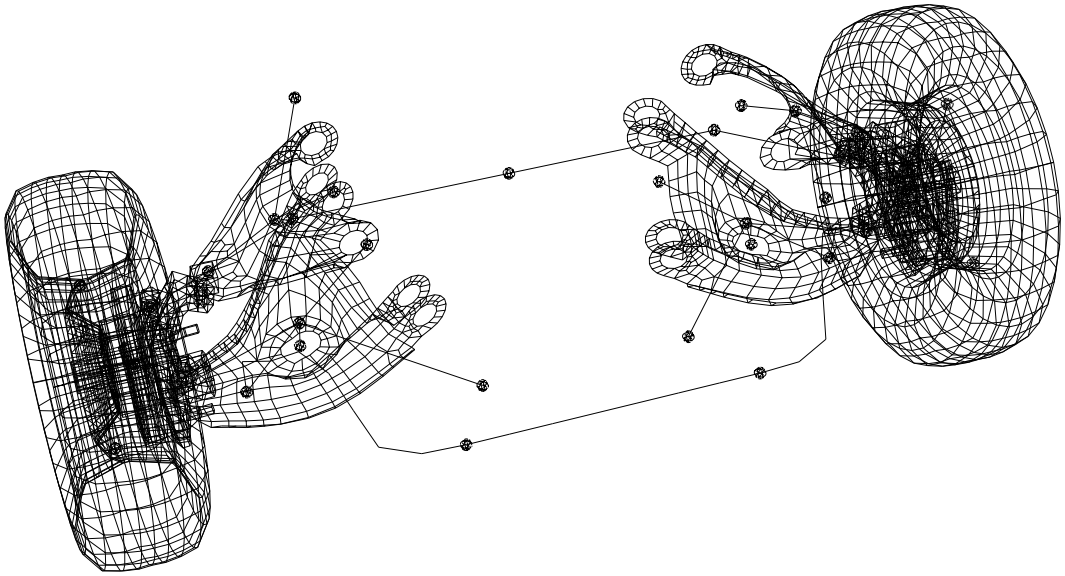


Figure 1: Display bodies of the front-wheel assembly connected at attachment nodes.

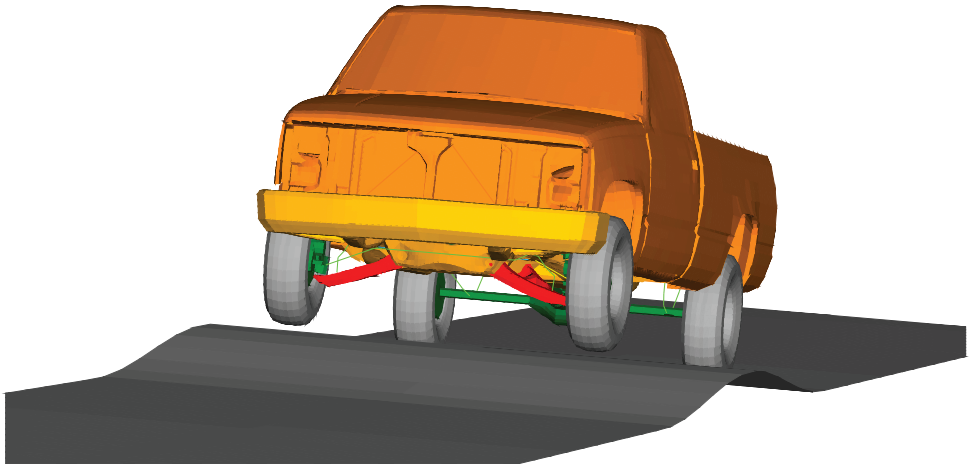


Figure 2: Display body truck model going over symmetric (left-right) bumps.

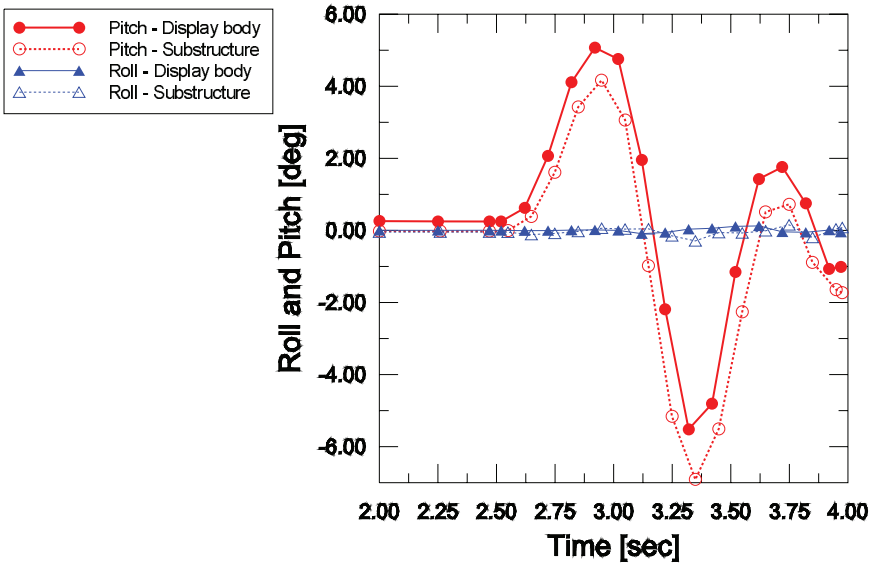


Figure 3: Comparison of the pitch and roll angles for the display body and substructure models of the truck going over symmetric bumps.

Continuum modeling of automotive spot welds

Products: Abaqus/Explicit

References:

- *Mass Scaling*
- *About Progressive Damage and Failure*
- **DAMAGE INITIATION*
- **DAMAGE EVOLUTION*
- *Progressive damage and failure of ductile metals*

Overview

This example models a range of failure mechanisms typically observed in spot welds. These virtual experiments are compared with laboratory-obtained data (courtesy of BMW).

The following Abaqus features are demonstrated:

- using three-dimensional continuum elements and intricate material models (elastic-plastic and damage constitutive behavior) to reproduce experimentally observed load-displacement curves (courtesy of BMW) of tested spot weld specimens; and
- demonstrating how virtual experiments allow for the generation of load-displacement data of structural components from readily available geometric and material data. The load-displacement curves can be used subsequently in calibrating connector behavior (not discussed in this section) for efficient use in large-scale models, such as full-vehicle analyses.

Since the material data are highly proprietary, the input files provided below contain fictitious material data. The material data used in the input files were obtained from the actual material data by subjecting it to a number of transformations that preserve the trends in the overall shape of the curves without revealing the exact material behavior. Moreover, the stiffness of the testing machine referenced in this section is fictitious. However, the results published in this example use the actual data for comparison with the physical tests. For these reasons, you will not obtain the force-deflection curves or deformed configurations published in this section when you run the associated input files.

Application description

The use of spot welds for the bonding of metal sheets is an extremely common practice in the automobile industry. The number of such bonds in a typical vehicle is on the order of several thousand. The use of Abaqus connector elements to model spot welds in full-vehicle analyses leads to efficient finite element models that are able to capture the structural response of these local features with optimal computational effort; however, load-displacement curves required for the modeling of spot welds may not be readily available. Furthermore, the number of experimental tests required for the proper calibration of a complete set of spot weld pairs in a vehicle can be prohibitive since the mechanical response of these local mechanisms is dependent on both the geometric data, such as the thickness of the metal plates and the radius of the spot weld, and the material properties of the plates being welded. Virtual testing can generate the necessary modeling parameters when experimental data are not available.

Geometry

The geometry of the patented test specimens used (Hahn et al., 1996, and Hahn and Rohde, 2004) is shown in *Figure 1* and *Figure 2*. A single spot weld of radius 2.65 mm connects two steel plates that are 1.4 mm thick. These steel plates are 50 mm long and are bent over radii of 4.0 mm.

Materials

All specimens used in this study are made of galvanized high-strength steel H340LAD+Z100. This material behaves in an elastic-plastic manner during the initial loading stages. As the material is further loaded, it can either display a ductile damage response (caused by growth and coalescence of voids) or it can display a shear failure mechanism (caused by the formation and growth of cracks within shear bands). For confidentiality reasons, the material data published in the input files associated with this section are fictitious. The data were obtained by transforming the actual material test data to preserve the overall trends in the behavior without revealing the true material behavior.

Boundary conditions and loading

A photo of the testing machine (Hahn et al., 2000) is shown in *Figure 3*. In the physical tests the vertical sides of the specimens were longer than those shown in *Figure 1* and *Figure 2*, and they were rigidly clamped in the clamping device (not shown) of the tensile testing machine. In the analysis these clamping conditions are modeled by fixing the bottom edges of the specimens and imposing a constant velocity of 0.15 m/s along the top edges in the vertical global 3-direction. For the physical tests and analysis, a hinge is located 400 mm from the top edges of the specimens along the vertical global 3-direction allowing the fixed edges to rotate about the horizontal axis. During the pull and peeling tests the specimens are aligned so the top edges are initially orthogonal to the vertical direction, whereas during the shear test the edges are initially parallel to the vertical direction. The combined stiffness of the loading piston and restraints used in the analysis is 50 kN/mm in pull and peeling tests and 35 kN/mm in shear tests. For confidentiality reasons both values are fictitious, but they approximate the compliance of the actual testing machine used in the physical tests.

Abaqus modeling approaches and simulation techniques

A total of 18 different simulations were performed corresponding to the three test cases (pull test, shear test, and peeling test). Each of the simulations was solved with a coarse mesh and a fine mesh using three different scaling factors for the thermal influence modeling, as discussed below.

Summary of analysis cases

Case 1	Pull test.
Case 2	Shear test.
Case 3	Peeling test.

The sections that follow discuss analysis considerations that apply to all cases, except where noted otherwise.

Mesh design

All simulations were performed with 8-node, linear brick, reduced integration elements (C3D8R). The density of the meshes increases toward the center of the plate where most of the deformation occurs. Each test was performed with a coarse mesh using four elements through the thickness of each plate and a fine mesh using six elements through the thickness of each plate. The coarse mesh and fine mesh used for the pull and shear tests are shown in *Figure 4* and *Figure 5*, respectively. *Figure 6* shows the initial undeformed configuration of the coarse mesh used in the peeling test simulation, while *Figure 7* shows the initial configuration of the fine mesh. The same meshes are used for the pull and shear tests since they are based on the same model geometry.

Materials

For confidentiality reasons, some but not all details of the Abaqus models used for constitutive behavior and progressive damage analysis are discussed below. For guidelines on obtaining the material parameters from experimental data, see *Progressive failure analysis of thin-wall aluminum extrusion under quasi-static and dynamic loads*.

Elastic-plastic behavior

Werner et al. (2004) have shown that correct modeling of the elastic-plastic deformation of spot welds is a prerequisite for realistic predictions of subsequent failure mechanisms. The authors showed that by taking into consideration material property changes in the weld nugget they could obtain different failure modes involving peeling or shearing failure of the spot weld. Furthermore, they suggested the use of hardness measurements as a possible indicator for the change in properties of the welded material. The change in hardness between the center of a spot weld and at a distance far away from the center depends on the material grades joined, as well as their thicknesses.

In this example we assume both the elastic and the plastic behavior to be isotropic with the yield surface described by a von Mises yield function (see *Inelastic Behavior*). Different hardening curves are considered to encapsulate thermal effects near the spot weld. For simplicity, the specimen is partitioned into three zones corresponding to the different thermal exposures observed in the vicinity of the spot weld during the welding process. Different scaling factors for the stress-strain curves are used in the three zones as derived from hardness measurements. The geometry of each zone can be prescribed according to the welding process parameters. The scaling of the yield curve is accomplished with the use of a field variable defined as constant throughout each region, and we test three scaling magnitudes for each test. For confidentiality reasons the scaling factors given below are fictitious, but they reflect the trends of the elastic-plastic behavior near the spot weld:

- A baseline configuration, where the original material properties (no scaling) are assigned to the specimens in all three zones. While this choice is not realistic, it provides an extreme solution for comparison purposes.
- A second configuration uses a scaling of 1.2 of the yield curve in Zone 1 and a scaling of 1.1 in Zone 2.
- A third configuration uses a scaling of 1.4 of the yield curve in Zone 1 and a scaling of 1.2 in Zone 2.

These three scaling factor configurations help us to understand the effect of the thermal influence zone in capturing the correct behavior, as discussed below.

Damage initiation and evolution

The failure of aluminum-alloy sheets and thin-walled extrusions results from one or more of the following mechanisms (Hooputra et al., 2004): nucleation, growth, and coalescence of voids; shear bands; and necking. Damage due to initiation, growth, and coalescence of voids leads to ductile failure in metals; the formation of cracks within shear bands leads to shear failure. Abaqus offers phenomenological damage initiation criteria for both of these mechanisms. The ductile criterion is specified by providing the equivalent plastic strain at the onset of ductile damage as a function of stress triaxiality and strain rate. Similarly, the shear criterion is specified by providing the equivalent plastic strain at the onset of shear damage as a function of shear stress ratio and strain rate (see *Damage Initiation for Ductile Metals*). The actual damage initiation criterion data were provided by BMW but for confidentiality reasons, the data were transformed to preserve the overall trends without revealing the actual material behavior.

Damage evolution occurs once the damage initiation criteria are satisfied and further loading is applied. A plastic displacement–based linear damage evolution law is used for each damage initiation criterion. The value of the plastic displacement at which the damage variable reaches 1.0 (complete degradation) is taken as 0.1, based on data from independent base material testing. The default maximum degradation rule is used, and the elements are removed from the mesh once the maximum degradation has occurred (see *Maximum Degradation and Choice*

of Element Removal). Damage initiation and evolution are assumed to be the same in all three thermal influence zones described above, a simplifying modeling assumption.

Initial conditions

As discussed in the material model section above, a field variable is used to scale the yield surface and capture the thermal effects of the welding process on the yield strength.

Boundary conditions

The specimens are loaded by fixing their bottom edges and imposing a constant velocity of 0.15 m/s along the top edges in the vertical global 3-direction.

Constraints

The top and bottom edges are constrained by kinematic couplings to model the rigid clamping in the testing devices. A TRANSLATOR connection is used to model the stiffness of the loading apparatus.

Interactions

General frictionless contact is defined between all surfaces, an appropriate approximation since friction forces are small when compared with the forces in the weld nugget.

Analysis steps

All simulations consist of one explicit dynamic step. All analyses consider geometric nonlinearity and utilize mass scaling to model quasi-static loading conditions (see *Mass Scaling*).

Output requests

Field output requests include the following quantities: displacement, stress, strain, element status, and damage initiation criteria variables. The history output request consists of displacement, velocity, acceleration, and reaction force at the reference points of the kinematic coupling constraints. Energy output variables are requested for the entire model.

Discussion of results and comparison of cases

The results presented in this example compare the actual material data with the physical test data. For confidentiality reasons, the actual material data are not published in the associated input files. Hence, you will not obtain the results below when you run the input files provided.

The von Mises stress contour and final deformed shapes of the pull, shear, and peeling tests with the coarse mesh and baseline material (no scaling of the yield curve) are depicted in *Figure 8*, *Figure 9*, and *Figure 10*, respectively. The final deformed shape in pull and peeling tests show good qualitative agreement with experimental results provided by BMW. Shear test results predict failure of the spot weld rather than on the surrounding plate, regardless of the amount of scaling that was applied to the yield curve. This behavior was seen in some but not all experimental results, and it does not seem to influence the overall bearing capacity of the structure in shear.

The load-displacement history obtained from the simulations is compared with the experimental results in *Figure 11* to *Figure 16*. A good match is observed for the pull and shear tests with the yield curve scaled by 1.2 in Zone 1. Results obtained with the baseline material underpredict the peak load capacity, and results obtained with the 1.4 scaling overpredict the peak load of the structures. The thermal zone scaling does not have a significant impact on the qualitative character of the loading curves. Good mesh convergence is also observed, indicating that acceptable results can be obtained even with the coarse mesh discretization.

Results obtained for the peeling test do not show such good agreement with the experimental results provided to SIMULIA by BMW; even the baseline material simulation results overpredict the peak load capacity. This overstiff behavior is shown even in the purely elastic levels of deformation (in very early deformation stages), which indicates that modeling of the loading apparatus for this loading configuration may be inexact (data not available).

In conclusion, the results from both the quasi-static spot weld connector simulations match the experimental pull and shear data very well. Changes in the material properties induced by the welding process are not essential to the capture of the peak loads during the spot weld failure.

Input files

Case 1a: Pull test, coarse mesh

spotweld_pu_cm.inp	Input file to create and analyze the model (baseline material).
spotweld_pu_cm_1p2.inp	Input file to create and analyze the model (spot weld yield surface scaled by a factor of 1.2).
spotweld_pu_cm_1p4.inp	Input file to create and analyze the model (spot weld yield surface scaled by a factor of 1.4).

Case 1b: Pull test, fine mesh

spotweld_pu_fm.inp	Input file to create and analyze the model (baseline material).
spotweld_pu_fm_1p2.inp	Input file to create and analyze the model (spot weld yield surface scaled by a factor of 1.2).
spotweld_pu_fm_1p4.inp	Input file to create and analyze the model (spot weld yield surface scaled by a factor of 1.4).

Case 2a: Shear test, coarse mesh

spotweld_sh_cm.inp	Input file to create and analyze the model (baseline material).
spotweld_sh_cm_1p2.inp	Input file to create and analyze the model (spot weld yield surface scaled by a factor of 1.2).
spotweld_sh_cm_1p4.inp	Input file to create and analyze the model (spot weld yield surface scaled by a factor of 1.4).

Case 2b: Shear test, fine mesh

spotweld_sh_fm.inp	Input file to create and analyze the model (baseline material).
spotweld_sh_fm_1p2.inp	Input file to create and analyze the model (spot weld yield surface scaled by a factor of 1.2).
spotweld_sh_fm_1p4.inp	Input file to create and analyze the model (spot weld yield surface scaled by a factor of 1.4).

Case 3a: Peeling test, coarse mesh

spotweld_pe_cm.inp	Input file to create and analyze the model (baseline material).
spotweld_pe_cm_1p2.inp	Input file to create and analyze the model (spot weld yield surface scaled by a factor of 1.2).
spotweld_pe_cm_1p4.inp	Input file to create and analyze the model (spot weld yield surface scaled by a factor of 1.4).

Case 3b: Peeling test, fine mesh

spotweld_pe_fm.inp	Input file to create and analyze the model (baseline material).
spotweld_pe_fm_1p2.inp	Input file to create and analyze the model (spot weld yield surface scaled by a factor of 1.2).
spotweld_pe_fm_1p4.inp	Input file to create and analyze the model (spot weld yield surface scaled by a factor of 1.4).

Auxiliary files

spotweld_material.inp	Input file with material model data.
spotweld_cm_part.inp	Input file with part data for Case 1a and Case 2a.
spotweld_cm_assembl.inp	Input file with assembly data for Case 1a and Case 2a.
spotweld_fm_part.inp	Input file with part data for Case 1b and Case 2b.
spotweld_fm_assembl.inp	Input file with assembly data for Case 1b and Case 2b.
spotweld_pe_cm_part.inp	Input file with part data for Case 3a.
spotweld_pe_cm_assembl.inp	Input file with assembly data for Case 3a.
spotweld_pe_fm_part.inp	Input file with part data for Case 3b.
spotweld_pe_fm_assembl.inp	Input file with assembly data for Case 3b.

References

- Hahn, O., J.R. Kurzok, and M. Oeter, “Test specification for KS-2 specimen,” Laboratory of Materials and Joint Technology, University of Paderborn, 2000.
- Hahn, O., and A. Rohde, “Procedures to manufacture specimen and specimen clamping device,” Patent Nr. 19522247 B4, April 15, 2004.
- Hahn, O., A. Rohde, and D. Gieske, “Specimen and specimen clamping device for use in tensile testing machines,” Patent Nr. 19510366 C1, August 22, 1996.
- Hooputra, H., H. Gese, H. Dell, and H. Werner, “A Comprehensive Failure Model for Crashworthiness Simulation of Aluminium Extrusions,” International Journal of Crashworthiness, vol. 9, pp. 449–463, 2004.
- Werner, H., H. Hooputra, H. Dell, and H. Gese, “A Phenomenological Failure Model for Sheet Metals and Extrusions,” Annual Review Meeting and Workshop, Impact and Crashworthiness Laboratory, Massachusetts Institute of Technology, 2004.

Figures

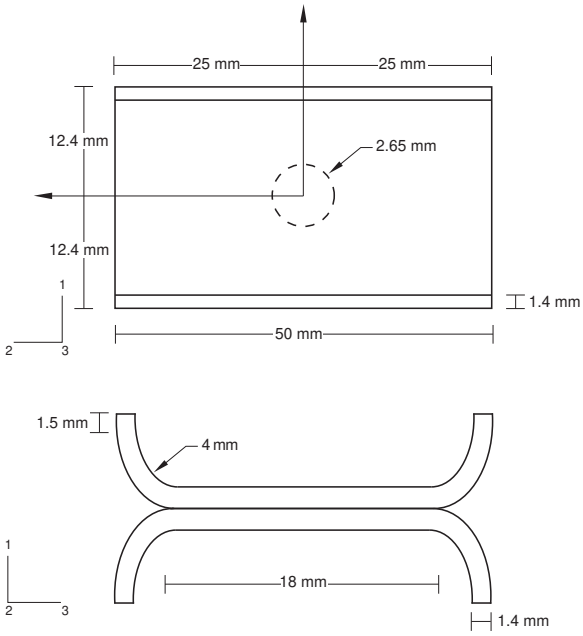


Figure 1: Pull and shear test specimen geometry.

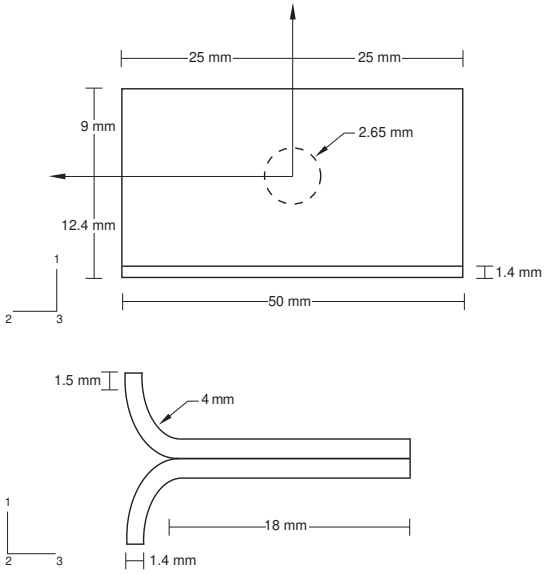


Figure 2: Peeling test specimen geometry.

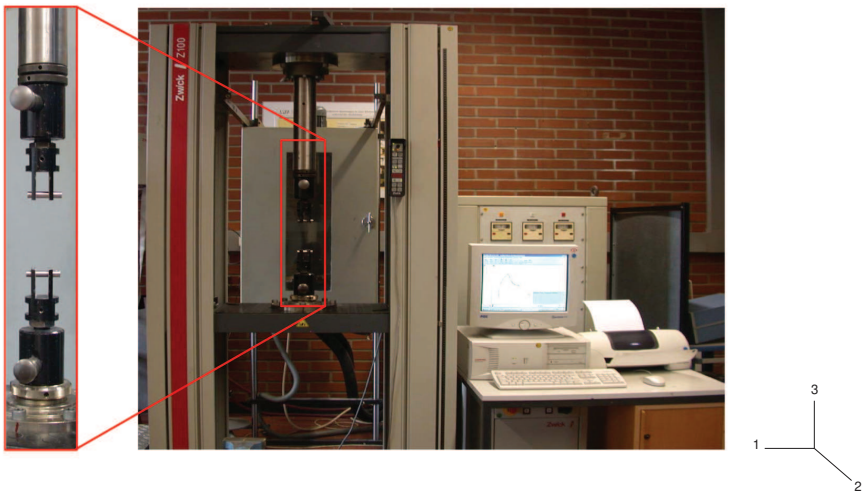


Figure 3: Experimental setup.

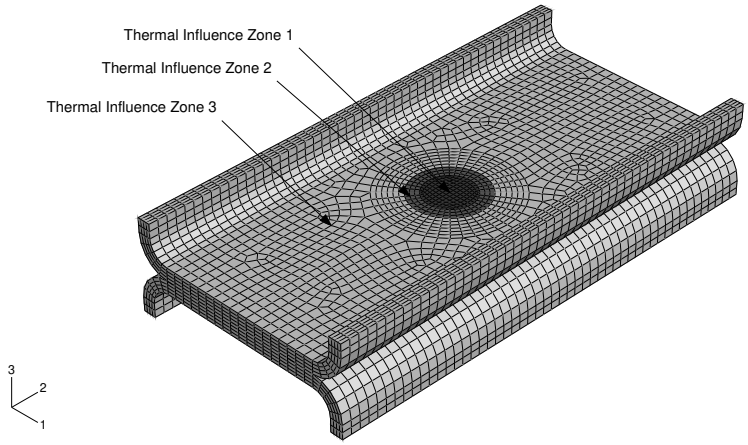


Figure 4: Pull and shear tests: coarse mesh and thermal influence zones.

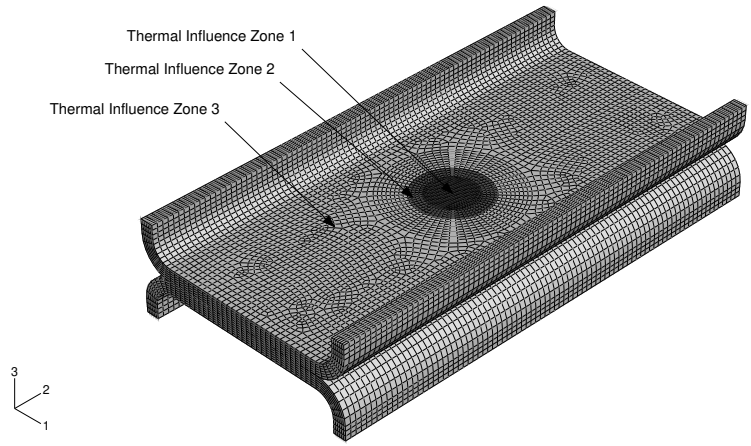


Figure 5: Pull and shear tests: fine mesh and thermal influence zones.

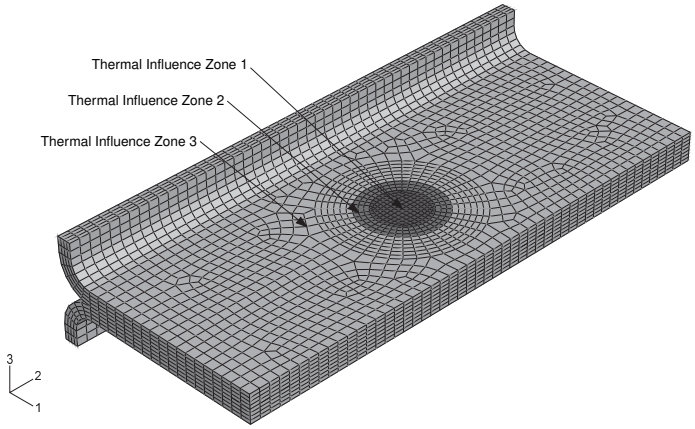


Figure 6: Peeling test: coarse mesh and thermal influence zones.

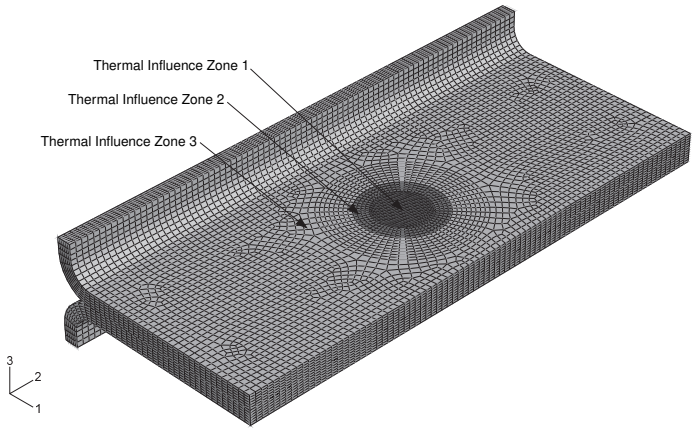


Figure 7: Peeling test: fine mesh and thermal influence zones.

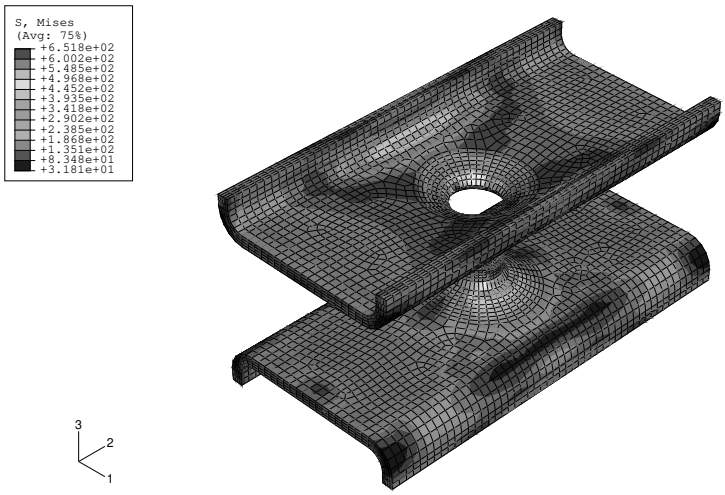


Figure 8: Pull test, coarse mesh, baseline material (no yield curve scaling): final deformed shape of specimen and von Mises stress contours.

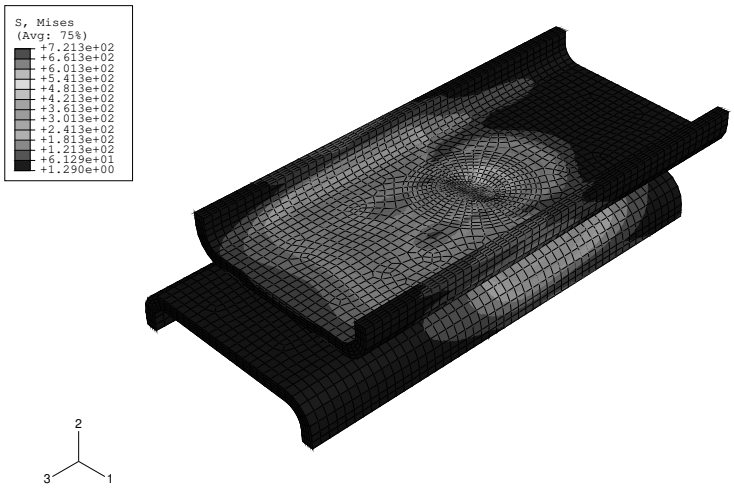


Figure 9: Shear test, coarse mesh, baseline material (no yield curve scaling): final deformed shape of specimen and von Mises stress contours.

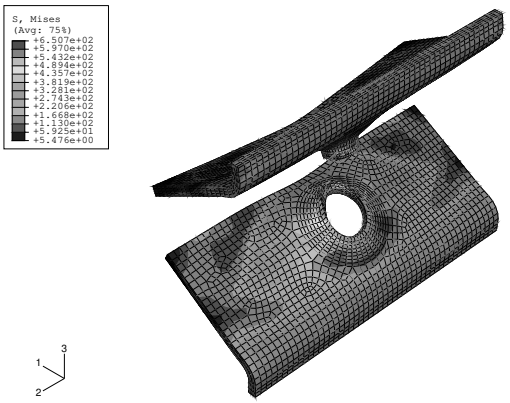


Figure 10: Peeling test, coarse mesh, baseline material (no yield curve scaling): final deformed shape of specimen and von Mises stress contours.

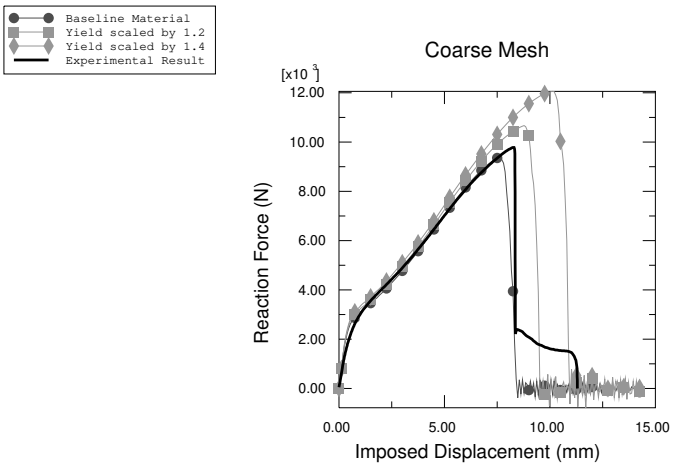


Figure 11: Pull test with coarse mesh: comparison of reaction force versus imposed displacement for different yield surface scaling. Experimental result courtesy of BMW.

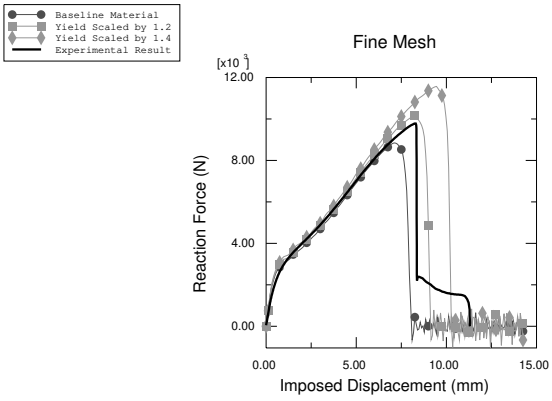


Figure 12: Pull test with fine mesh: comparison of reaction force versus imposed displacement for different yield surface scaling. Experimental result courtesy of BMW.

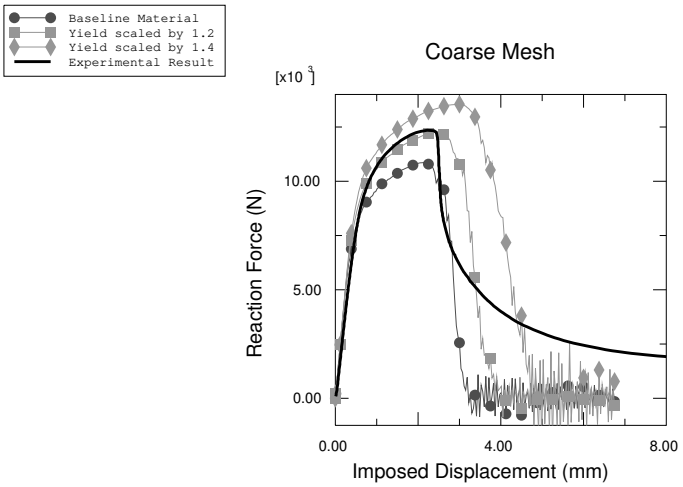


Figure 13: Shear test with coarse mesh: comparison of reaction force versus imposed displacement for different yield surface scaling. Experimental result courtesy of BMW.

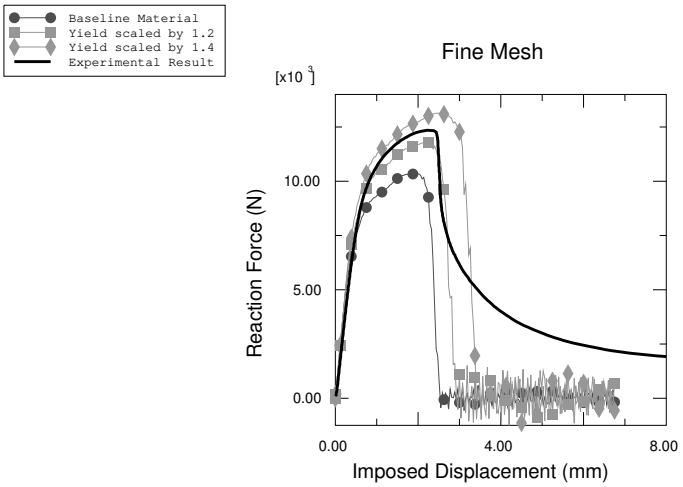


Figure 14: Shear test with fine mesh: comparison of reaction force versus imposed displacement for different yield surface scaling. Experimental result courtesy of BMW.

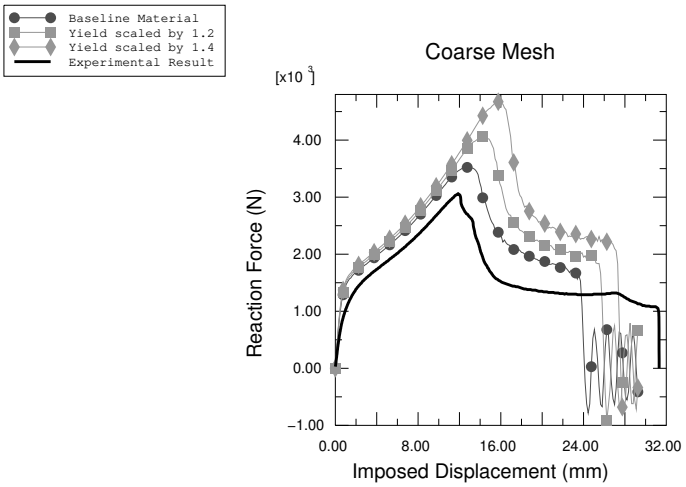


Figure 15: Peeling test with coarse mesh: comparison of reaction force versus imposed displacement for different yield surface scaling. Experimental result courtesy of BMW.

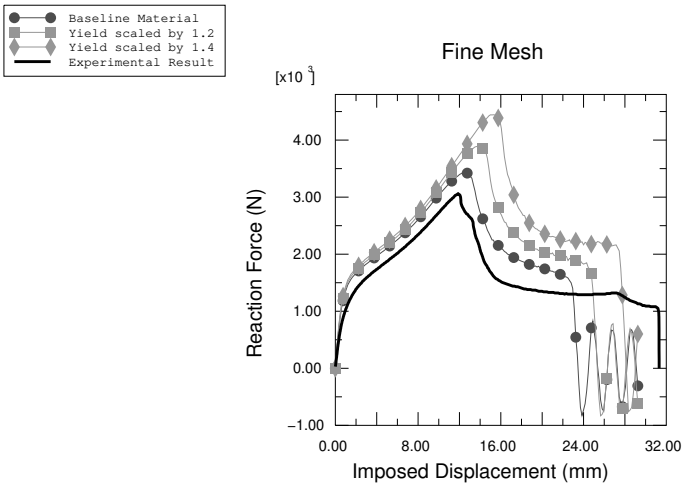


Figure 16: Peeling test with fine mesh: comparison of reaction force versus imposed displacement for different yield surface scaling. Experimental result courtesy of BMW.

Occupant safety analyses

In this section:

- *Seat belt analysis of a simplified crash dummy*
- *Side curtain airbag impactor test*
- *Side curtain airbag analysis using the lumped kinetic molecular method*

Seat belt analysis of a simplified crash dummy

Products: Abaqus/Explicit

Overview

This example shows the use of Abaqus in analyzing a simplified crash dummy restrained by a complete seat belt system as a moving vehicle comes to a sudden stop.

Automotive seat belts dramatically reduce the risk of vehicle occupant injury in the case of collision. The seat belt system includes:

- the webbing sliding through D-rings attached to the car frame;
- one retractor that models spool locking, tension preloading, and spool effects; and
- one pretensioner that tightens the seat belt.

Several SLIPRING connectors strung together are used to model the belt webbing passing through the rings. The RETRACTOR and HINGE connection types are used to model the retractor and pretensioner.

Geometry and materials

As shown in *Figure 1*, the model consists of several distinct entities: the dummy, the seat belt, the retractor, and the seat. The modeling strategy consists of applying an initial velocity for the dummy of approximately 45 miles per hour while the seat and the seat belt attachment points to the car frame are held fixed with boundary conditions, thus emulating a vehicle that comes to a sudden stop.

Dummy

A very simple dummy model is used (none of the limbs are modeled). The dummy model has two distinct parts, the lower torso and the upper torso, that are connected together through a rigid connector. The lower torso is modeled with rigid surface elements. The upper torso is modeled in the same fashion except in the chest area. The human body compliance is modeled approximately by meshing a region in the chest area using deformable shell elements. In addition, four CARTESIAN and CARDAN connectors with nonlinear elastic and damping behavior are inserted between four nodes around the chest area and four nodes belonging to the rigid back of the upper torso. The upper and lower torsos overlap over a small region around the waist area, but contact is excluded to avoid the interaction between them. The approximate mass of the dummy is 35 kilograms.

Seat

The seat modeling is minimal since the focus of this example is to illustrate the seat belt modeling technique. Only one solid element with crushable foam material properties is used to model the lower part of the seat. The back support is not modeled.

Seat belt webbing passing through rings

The seat belt is modeled primarily using several SLIPRING connectors strung together. To model the contact interactions between the belt and the chest and lap areas accurately, membrane elements are used to model short portions of the belt in these regions. *Figure 2* shows the seat belt arrangement. The node numbers associated with the connector elements are also illustrated. The seat belt is defined starting from the right side of the figure (adjacent to the B-pillar in the car) and moving to the chest area, the waist-level click-in buckle, the lap area, and finally the attachment to the car bottom floor, as follows:

- Three SLIPRING connectors are used between nodes 8300228, 8300237, 8300243, and 8300247. In this order, the four nodes correspond to the exit point from the retractor at the bottom of the B-pillar, the trim exit point along the B-pillar, the shoulder-level ring at the top of the B-pillar, and the connection point with the membrane-mesh region of the belt at the top of the chest area.

  The three SLIPRING connectors model the belt as it flows and stretches through these points. As outlined in *Connector Element Library*, the SLIPRING connection type activates the nodal material flow degree of freedom (10) at these four nodes. Material can flow freely between the three elements but cannot flow through the connection point with the membrane-meshed area at node 8300247. Hence, a boundary condition on this degree of freedom is specified at this node.
- The membrane-meshed area between nodes 8300247 and 8300248 models the contact with the chest area of the dummy accurately.
- Two SLIPRING connectors are used between nodes 8300248, 8300251, and 8300253. In this order, the three nodes correspond to the connection point with the membrane-mesh region of the belt at the bottom of the chest area, the ring of the waist-level click-in buckle, and the connection point with the membrane-meshed region of the belt at the left side of the lap area.

  Since no material flow occurs at the connection points with the membrane-meshed regions, boundary conditions on degree of freedom 10 are specified at nodes 8300248 and 8300253.
- The membrane-meshed area between nodes 8300253 and 8300254 models the contact with the lap area of the dummy accurately.
- One SLIPRING connector is used between nodes 8300254 and 8301398. These two nodes correspond to the connection point with the membrane-mesh region of the belt at the right side of the lap area and the attachment point to the car frame.

  Boundary conditions on degree of freedom 10 are specified at both nodes.

The stretching of the belt is governed by the specified nonlinear elastic connector behavior.

Retractor

The retractor device is located at the bottom of the B-pillar. It is modeled using a RETRACTOR connector in parallel with several HINGE connectors as illustrated in *Figure 3*. The connections are as follows:

- The RETRACTOR connector is connected to the first SLIPRING connector along the B-pillar at node 8300228. It converts the material flow at this node into a rotation about a local axis oriented along the longitudinal direction of the car.
- Several HINGE connectors model the various mechanisms in the retractor device. Their axes are all parallel to each other and are oriented along the longitudinal direction of the car.
  - The preload HINGE applies a small pretension elastic force typically given by a weak torsional elastic spring. Its purpose is to eliminate the slack in the belt.
  - The spool lock HINGE uses a connector lock definition to lock the almost-free rotation of the spool's axle if the velocity of the belt material exiting the retractor exceeds a certain threshold.
  - The spool effect HINGE is attached to the car frame and uses a connector plasticity definition to model the compression effects of the spool. This is accomplished via a tension-versus-spooled-out-material curve.

The spool effect is inactive until the spool lock connector locks; hence, the two HINGE connectors are placed in series. The preload is applied at all times and, therefore, is placed in parallel with the two other HINGE connectors.

Pretensioner

The pretensioner device is attached to the car frame in the vicinity of the waist-level click-in buckle. It is modeled using a SLIPRING connector and a RETRACTOR and a HINGE connector in parallel as illustrated in *Figure 4*. The connections are as follows:

- The pretensioner spool (HINGE) connector is attached to the car frame. It uses an amplitude-delayed connector motion definition to specify the rate at which the belt material flows into the pretensioner device.
- The RETRACTOR connector converts the specified motion in the HINGE connector into a material flow that is expelled from the following SLIPRING connector.
- The SLIPRING connector models the stiff cable to which the waist buckle is attached. Its length shortens as the pretensioner device is being triggered.
- A PIN-type MPC is used to connect the node associated with the waist-level click-in ring that slides over the seat belt (node 8300251) with the buckle it clicks into (node 8300351). By using the additional node 8300351, the material flows associated with the pretensioner SLIPRING and the SLIPRINGs of the adjacent seat belt segments are prevented from interacting with each other.

Models

Two analyses are studied: one analysis that considers the friction in the connector elements and one analysis without friction in the connector elements. In addition, a comparison between isotropic and anisotropic friction in the contact interaction between the belt and the dummy is analyzed.

Isotropic frictional model

Isotropic friction coefficient: $\mu_{isotropic} = 0.3$.

Anisotropic frictional model

Equivalent friction coefficients: $\mu_1 = 1.0$ (the 1-direction is transverse to the belt, as shown in *Figure 5*) and $\mu_2 = 0.3$ (the 2-direction is longitudinal to the belt).

To obtain these coefficients, the seat belt fabric is considered to contribute to the directional preference of friction while the dummy surface does not contribute; therefore, the frictional directional preference factors ϵ and the average friction coefficient are set as follows:

$$\epsilon_{belt} = 0.5384$$
$$\epsilon_{dummy} = 0$$
$$\mu_{iso} = 0.65$$

The maximum-epsilon-dominant weighting method for combining surface-based directional preferences for the anisotropic friction model is used (**COMBINATORIAL RULE*, ANISOTROPY WEIGHTING=DOMINANT).

Results and discussion

The undeformed and deformed shapes (t=0.0215 seconds) for the model are shown in *Figure 6* and *Figure 7*, respectively. Results for belt tensions and material flow in and out of the shoulder level slipring are shown in *Figure 8* and *Figure 9*. At this junction SLIPRING connector elements 8888803 and 8888804 share a common node 8300243. For the frictionless analysis, the normalized belt tensions are shown in *Figure 8*. As expected, the two tension histories are the same. For the frictional case, the ratio of the belt tension in adjacent belt segments is shown in *Figure 9*. For the case when the belt is slipping, the ratio of the belt tension is given by $t_2/t_1 = e^{\mu\alpha}$, where t_2 and t_1 are the tensions in the adjacent SLIPRING connector elements, μ is the coefficient of friction,

and is the angle between the two adjacent sliprings. For the seat belt model with friction where μ=0.1 and

=1.718132 radians, the ratio of the belt tension is $t_2/t_1 = 1.187$. As shown in *Figure 9*, the ratio agrees well with the analytical result. Near the end of the analysis (19 milliseconds) the ratio of the belt tension drops from this value. *Figure 9* shows that the normalized accumulated slip remains constant for the remainder of the analysis; hence, we can conclude that the ratio drops because the belt starts sticking. *Figure 10* shows that the material flows across node 8300243, which is the second node of connector element 8888803 and the first node of connector element 8888804, are identical as expected.

As expected, activating anisotropic friction with a higher friction coefficient in the transverse direction of the belt with respect to the longitudinal direction causes the belt to have a smaller lateral motion, as shown in *Figure 11*.

Input files

seatbelt_xpl.inp	Analysis of a frictionless seat belt–restrained simplified dummy using Abaqus/Explicit.
seatbelt_fric_xpl.inp	Analysis of a frictional seat belt–restrained simplified dummy using Abaqus/Explicit.
seatbelt_anisfric_xpl.inp	Analysis of a frictional seat belt–restrained simplified dummy using Abaqus/Explicit. Anisotropic friction is considered in the contact between the belt and the dummy.

Figures

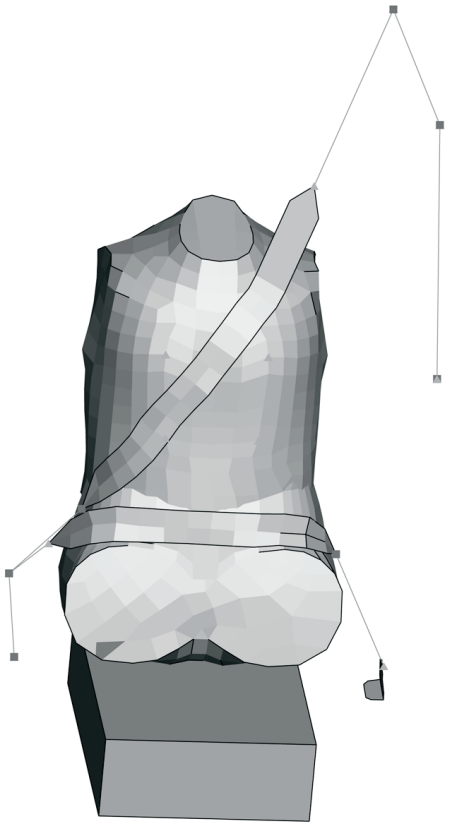


Figure 1: Seat belt and dummy arrangement.

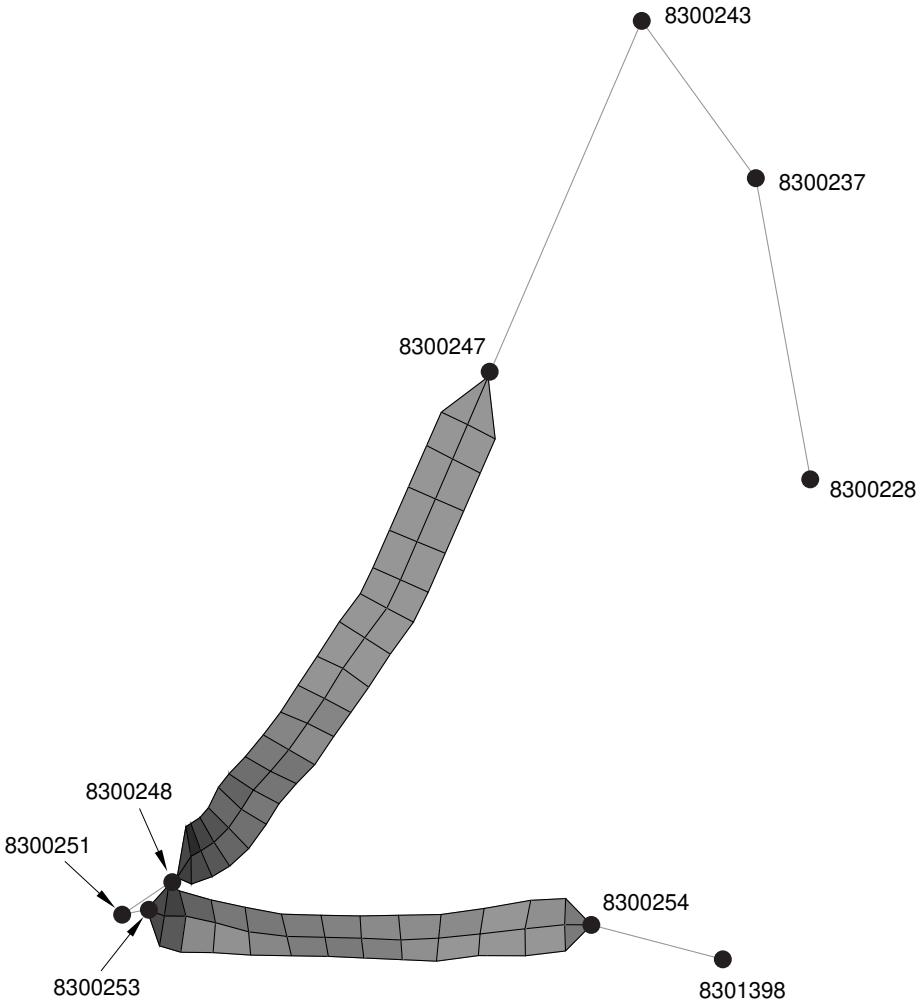


Figure 2: Seat belt arrangement.

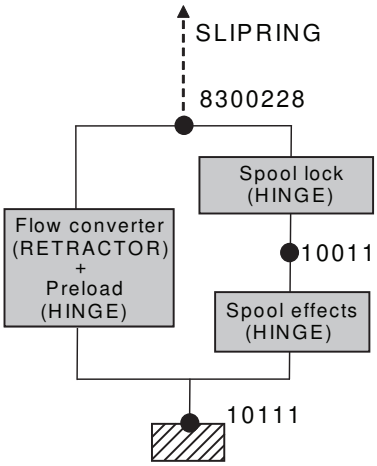


Figure 3: Retractor model.

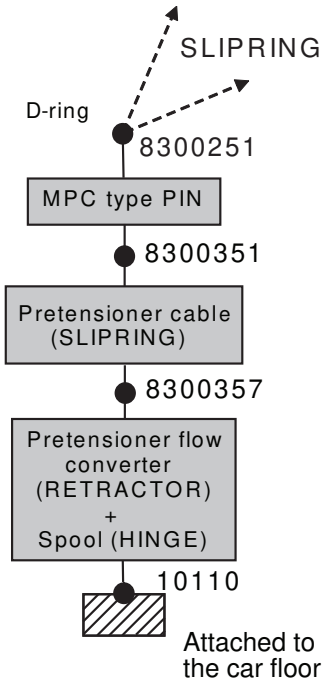


Figure 4: Pretensioner model.

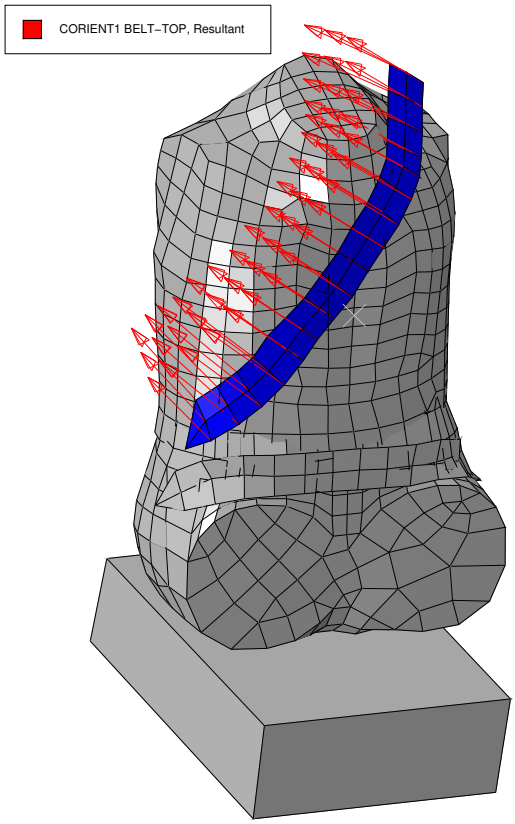


Figure 5: Directional preference 1 (the most resistance to slipping) for the anisotropic frictional model.

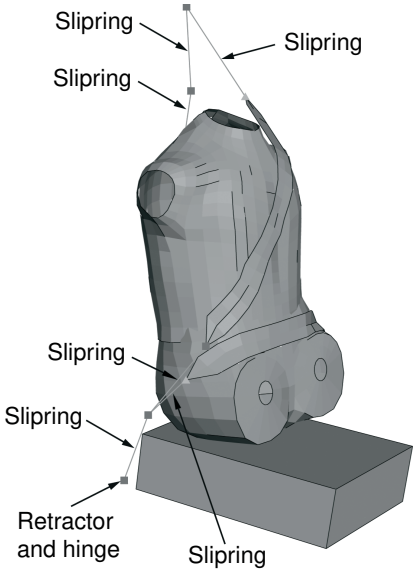


Figure 6: Dummy and seat belt system in the undeformed configuration.

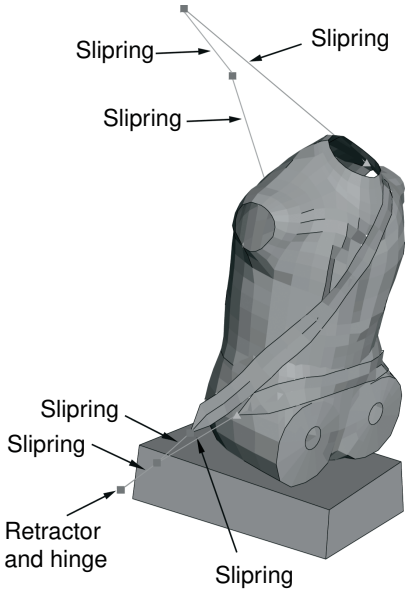


Figure 7: Dummy and seat belt system in the deformed configuration.

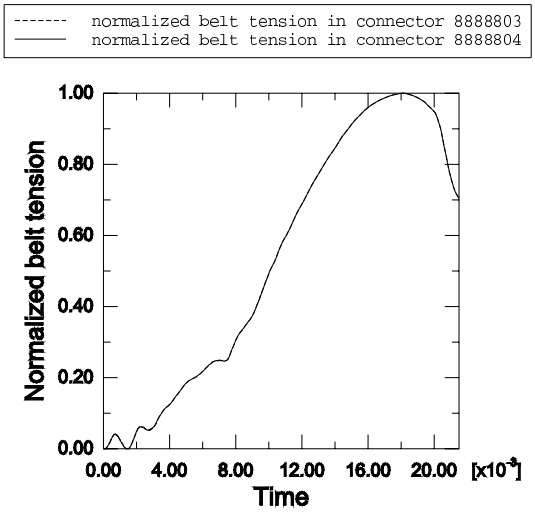


Figure 8: Normalized belt tension of adjacent frictionless sliprings.

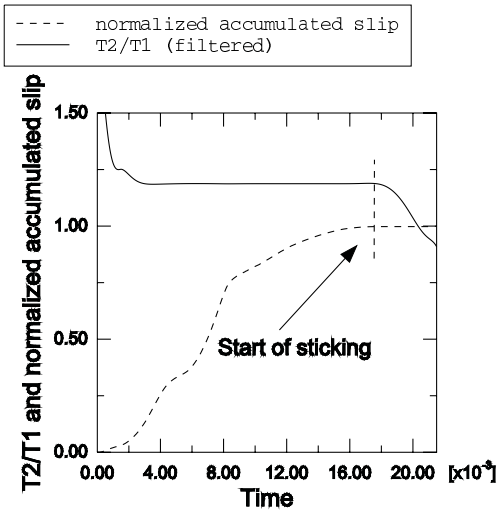


Figure 9: Ratio of belt tension of adjacent sliprings with friction and the normalized accumulated slip.

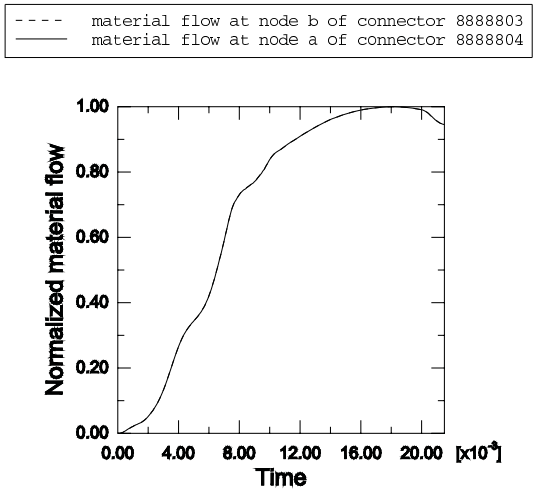


Figure 10: Normalized material flow across node between adjacent sliprings.

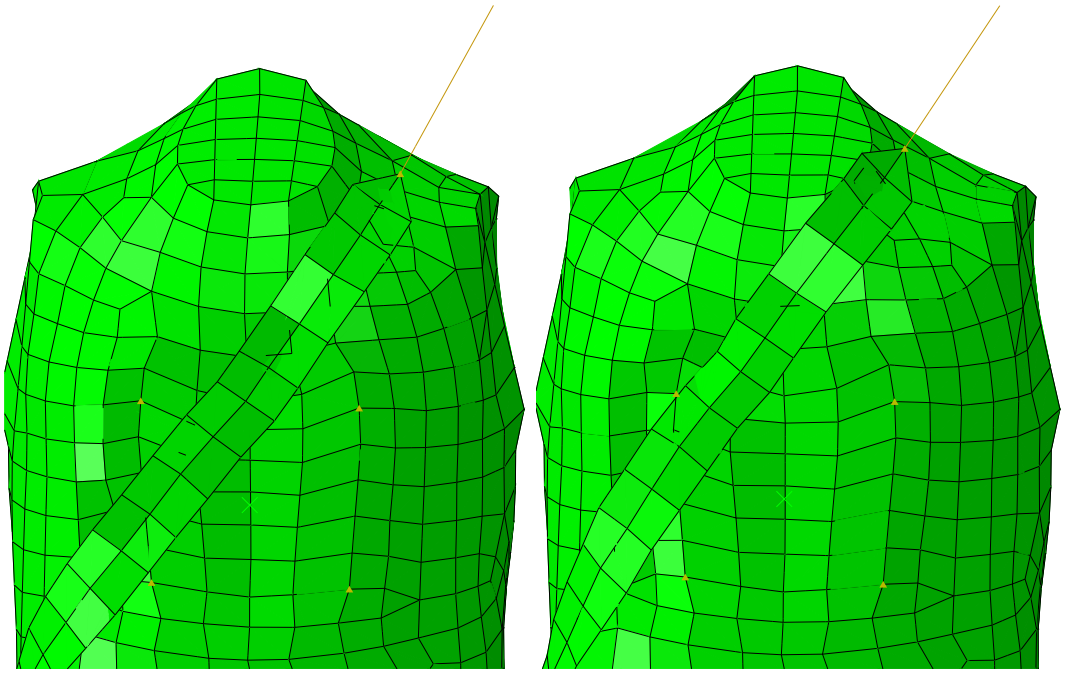


Figure 11: Position of the belt system in the deformed configuration (with anisotropic friction, left; without anisotropic friction, right).

Side curtain airbag impactor test

Products: Abaqus/Explicit

Overview

This example problem illustrates the use of surface-based fluid cavities in a multi-chambered automotive airbag inflation analysis using Abaqus/Explicit.
Airbags are supplemental safety devices that minimize the chance of an occupant striking the interior of a vehicle during a crash. The proper deployment of the airbag within a fraction of a second after a collision is detected is crucial to ensure the safety of the occupant. This example describes the modeling and validation of a side curtain airbag impactor test simulation against experimental results.

Geometry and model

The side curtain airbag model described in this example was provided by Autoliv GmbH. The model setup is shown in *Figure 1*, which shows an impactor hitting the deployed side curtain airbag. The airbag is divided into 18 chambers to approximate the nonuniform distribution of the gases across the airbag and the gradients in pressure observed during the complex unfolding of the airbag. The temperature and pressure are considered to be uniform in each chamber at any point of time. Each airbag chamber is modeled as a surface-based fluid cavity. The surface-based fluid cavity capability provides a coupling between the pressure exerted by the contained gas on the chamber boundary and the deformation of the gas-filled chamber. The individual chambers are separated by fictitious partitions modeled with surface elements, which represent the area of fluid exchange between the different chambers. The airbag has three inflators located in chambers 1, 6, and 9 that are used to inflate the airbag. The inflators are activated at different times using inflation time amplitude definitions associated with the activation of fluid inflator definitions. The inflators inject a mixture of 6 gas species, with the molecular fraction, mass flow rate, and temperature of the individual species defined as functions of time. The gas species behave as ideal gases with an adiabatic temperature update. Leakage of the gases to the environment through the airbag fabric is modeled by specifying a discharge coefficient (ratio of the effective leakage area to the total airbag surface area) of 0.0003 for fluid exchange from each airbag chamber. In addition, the loss of heat energy to the environment is also considered by providing the heat energy flow rate as a function of the temperature difference across the airbag fabric.

The external skin of the airbag is represented by 3-node membrane elements (M3D3) and uses a nonlinear elastic definition of the fabric material model. The fabric material model is calibrated by specifying the uniaxial behavior of the fibers in the warp and weft directions as well as the shear behavior of the fabric. The behavior of the fibers has been defined such that they offer negligible resistance to compression until a certain compressive strain limit beyond which they begin to offer finite resistance to compression to prevent excessive deformations of the fabric elements. Bilinear shear behavior is defined reflecting an initial shear modulus of G_0=9.44 N/mm^2 and a modulus of G_1=47.2 N/mm^2 at angles greater than 60° to prevent severe shear deformation. In addition, a small amount of stiffness proportional damping is applied. A reference mesh (initial metric) is defined to specify the unfolded stress-free configuration of the airbag.

The unstressed reference configuration of the airbag is defined using the initial reference mesh for membrane elements. The initial folded configuration and the reference configuration are used to account for the wrinkles that arise from the airbag folding process. The reference mesh and the folded airbag configuration are shown in *Figure 2* and *Figure 3*, respectively. The fabric material is defined such that any initial compressive strains along the yarn directions will cause compressive stresses in the yarn. Alternatively, the initial compressive strains along the yarn directions can be recovered stress free.

The general contact algorithm is used to consider self-contact between the airbag fabric layers as well as contact with surrounding structures during the inflation of the airbag. All edges of the airbag fabric are activated to improve the quality in the solution.

Since activating all edges for contact can be computationally expensive, the same model is also studied using the dynamic feature edge criteria, where edges are activated or deactivated automatically as needed during the analysis.

This approach is more computationally efficient than considering all edges for edge-to-edge contact and is more accurate than only considering edges associated with initial feature angles.

The impactor is modeled as a rigid body with a total mass of 7.2 kg and is given an initial velocity of 4500 mm/s towards the deploying airbag. The mass and velocity of the impactor reflect human head mass and velocities typically observed in vehicle side impacts, respectively.

Results and discussion

The airbag at different stages of deployment is shown in *Figure 4* (8 ms) and *Figure 5* (16 ms). *Figure 6*, *Figure 7*, and *Figure 8* show the time history of displacement, velocity, and acceleration of the impactor as it impacts the airbag, along with comparisons to experimental results. These plots demonstrate the close correlation between Abaqus/Explicit simulation results and physical test results.

Results with the dynamic feature edge criteria for contact are in good agreement with the default model where all the edges of the airbag are activated, as shown in *Figure 9*. As expected, performance is improved, as shown in *Table 1*.

Input files

sidecurtain_airbag_fabric.inp	Side curtain airbag impactor test.
sidecurtain_airbag_fabric_dynedge.inp	Side curtain airbag impactor test using the dynamic feature edge criteria for contact.
sidecurtain_airbag_refmesh.inp	Reference mesh for both side curtain airbags.

Tables

Table 1: Performance improvements obtained with dynamic feature edge criteria.

# CPUs	Normalized CPU time	
	Default Edge Contact	Dynamic Edge contact
1	1.00	0.85
2	1.00	0.82
4	1.00	0.85
8	1.00	0.81

Figures

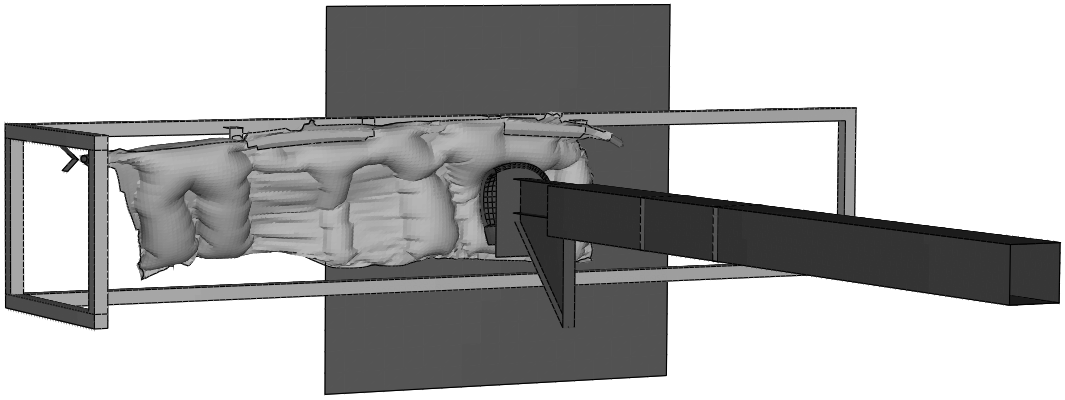


Figure 1: Model setup showing the fully deployed airbag (*t*=80 ms).

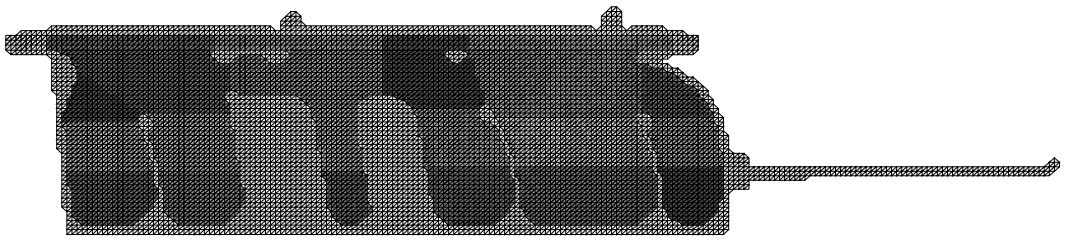


Figure 2: Airbag reference configuration.

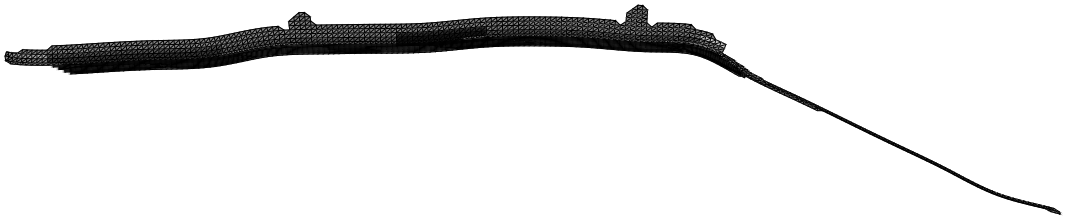


Figure 3: Folded airbag configuration.

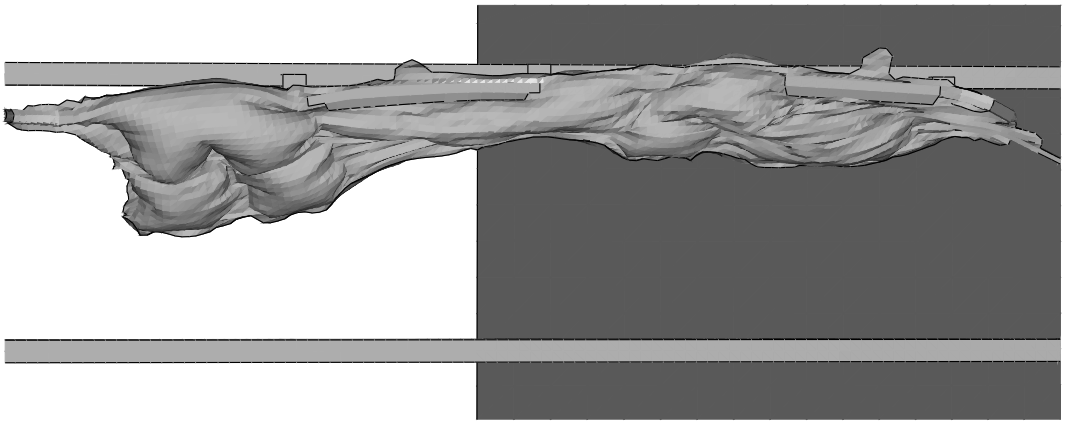


Figure 4: Deformed configuration at 8 ms.

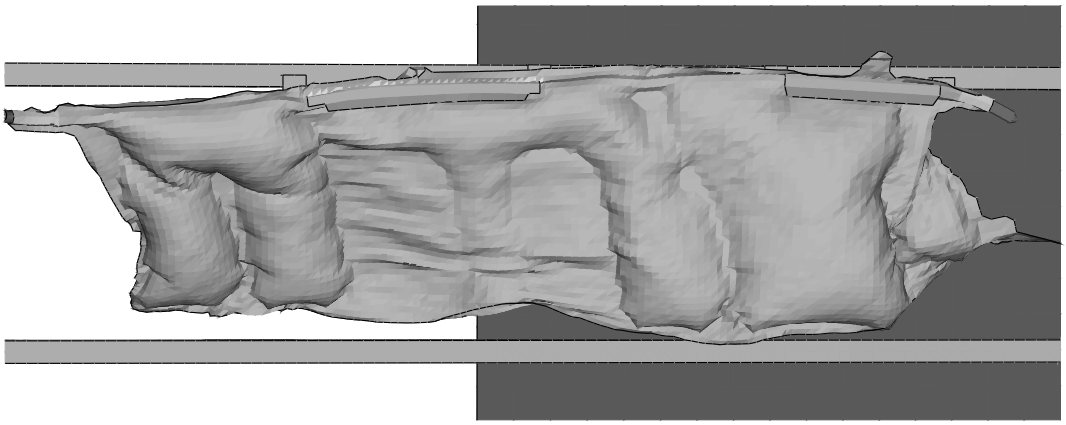


Figure 5: Deformed configuration at 16 ms.

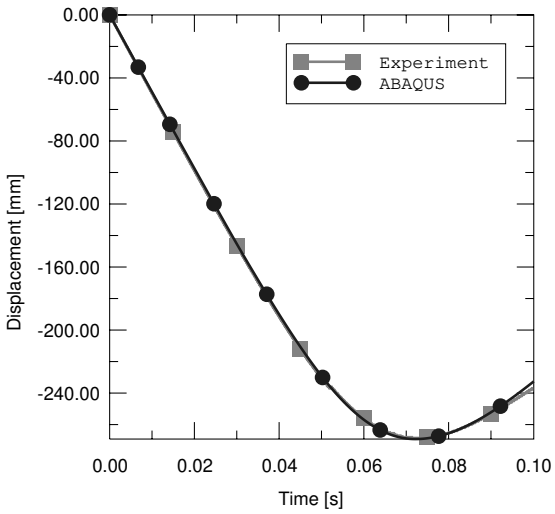


Figure 6: Time history of displacement of the impactor.

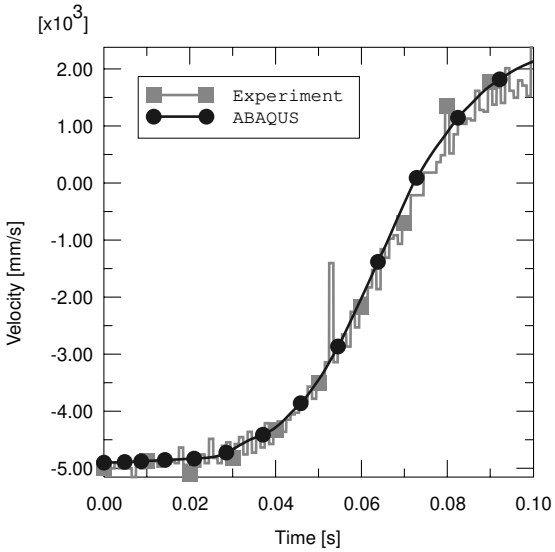


Figure 7: Time history of velocity of the impactor.

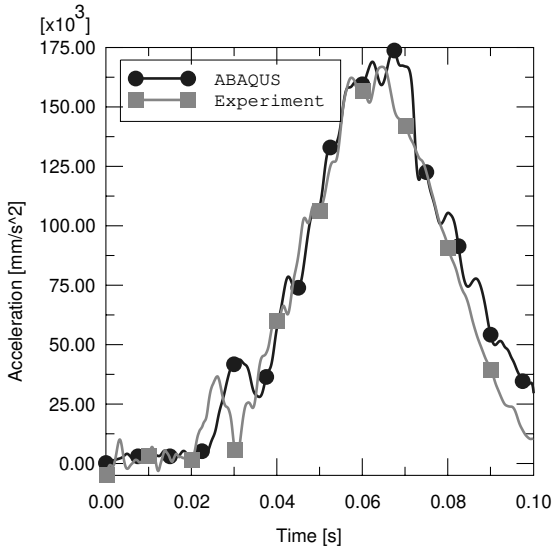


Figure 8: Time history of acceleration of the impactor.

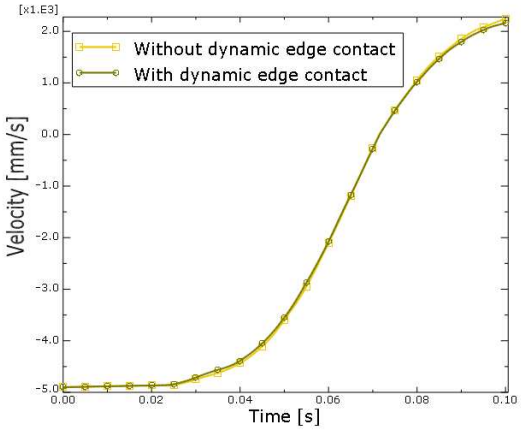


Figure 9: Time history of velocity of the impactor with and without dynamic edge.

Side curtain airbag analysis using the lumped kinetic molecular method

Products: Abaqus/Explicit

Overview

This example illustrates the use of the lumped kinetic molecular method in an automotive airbag deployment analysis.

This example demonstrates the following features and techniques:

- using an LKM-type pressure-overclosure contact interaction,
- using the particle generator to work as the airbag inflator,
- using PD3D elements to model lumped gas molecules, and
- switching from the LKM method to the uniform pressure method during the analysis.

Application description

Geometry

The side curtain airbag model is based on the airbag model described in *Side curtain airbag impactor test*. The geometry of the airbag and the rigid head form are the same as described in that example problem. The inflator gas is modeled using the lumped kinetic molecular (LKM) method. Because the LKM method is able to capture the nonuniformity of pressure in the airbag, only a single fluid cavity spanning the entire airbag is used.

Abaqus modeling approaches and simulation techniques

Mesh design

The airbag is discretized using 3-node M3D3 membrane elements. A single fluid cavity is associated with the airbag. In contrast to the model used in *Side curtain airbag impactor test* (where three inflators are used to inject gas into three different chambers of the airbag) a single particle generator is used to model the inflator device; it injects gas into the single fluid cavity associated with the airbag. The inlet of the particle generator consists of a single rectangular facet. The inlet facet is located at one end of a rigid fill tube and generates the gas particles inside the tube. The rigid inlet and fill tube are meshed with SFM3D4R surface elements. The gas particles generated inside the fill tube enter the airbag through three openings, shown in *Figure 1*. The placement of the three openings in the fill tube is based on the mid-location of the three chambers into which the three inflators in *Side curtain airbag impactor test* inject the gas.This example uses 200,000 PD3D discrete elements.

Materials

In *Side curtain airbag impactor test* six gases are injected by the inflators. Because the molar fractions for the six gases are nearly constant during the analysis, a single equivalent gas is used in this model to replace the six gases. The molecular weight of the equivalent gas is obtained from the molecular weights and molar fractions of the six gases used in *Side curtain airbag impactor test*. The molecular weight of the equivalent gas is 3.6614*e*–05 tonnes/mol. The polynomial forms of the constant-pressure molar heat capacity for each individual gas are evaluated at various temperatures. The evaluated values are combined using the mass fractions of individual gases to compute a tabular form of the constant-pressure heat capacity of the equivalent gas. The mass flow rate for the single gas species generated by the particle generator is obtained by combining the mass flow rate data for the three inflators used in *Side curtain airbag impactor test*. The airbag, as well as the rigid impactor mesh and material, are the same as described in *Side curtain airbag impactor test*.

Initial conditions

The unstressed reference configuration of the airbag is the same as in *Side curtain airbag impactor test*. Because there is no direct way to model the initial gas in the lumped kinetic molecular method, the particle generator generates the initial mass of gas at the initial temperature into the airbag over a very short duration of time at the start of the analysis.

Loads

An ambient pressure of 0.101329 MPa is considered outside the airbag. Gravity loading is applied to the model.

Interactions

An LKM-type pressure-overclosure contact interaction is defined between the gas particles, as well as between the gas particles and the airbag.

Solution controls

The discrete section associated with the lumped gas particles refers to a section control that establishes a bounding box within which all gas particle interactions occur. Particles that reach this bounding box are deactivated.

At 25 milliseconds, the lumped kinetic molecular method is deactivated and the uniform pressure method (UPM) is activated. The LKM method is used in the early deployment phase to capture the nonuniform nature of the pressure within the bag. In the later phase of deployment, the airbag is near full inflation, and the pressure is relatively uniform. Switching from the LKM method to UPM reduces the computational expense and allows for modeling of the inflator gas leaking to the environment via vents or permeable fabric and seams. Once the switch occurs, all the gas particles are deactivated.

Output requests

The filtered displacement, velocity, and acceleration of the head form in the vertical direction are requested. The history of the generated mass of gas (PDMASS), average gas temperature (PDTEMP), and average gas pressure (PDPAVG) is obtained via an integrated output request for the entire element set of gas particles. These three output variables are valid until deactivation of the lumped kinetic molecular method. After the activation of the uniform pressure method, output variables CMASS, CTEMP, and PCAV must be used for the generated gas mass, temperature, and pressure, respectively. Output variable CVOL is requested for history of the airbag volume.

Results and discussion

The analysis is performed in domain parallel mode using double precision. *Figure 2*, *Figure 3*, and *Figure 4* show the deformed shape of the airbag at different stages of deployment. *Figure 5*, *Figure 6*, and *Figure 7* show the results from the Abaqus/Explicit analysis (displacement, velocity, and acceleration of the head form), along with experimental results. These plots demonstrate the close correlation between the Abaqus/Explicit simulation results and the physical test results.

Input files

sidecurtain_airbag_lkm_switch_upm.inp	Side curtain airbag impactor test.
sidecurtain_airbag_refmesh_lkm.inp	Initial mesh for airbag.

References

- Simulation of Airbag Deployment Using the Coupled Eulerian-Lagrangian Method in Abaqus/Explicit. Abaqus Technology Brief, June 2011.

Figures

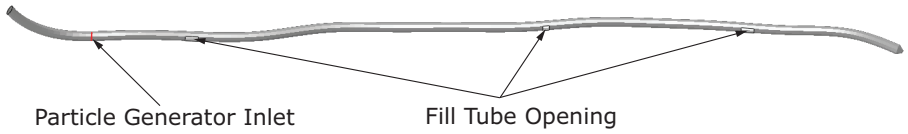


Figure 1: Particle generator inlet and fill tube with openings.



Figure 2: Deformed configuration at 8.75 ms.

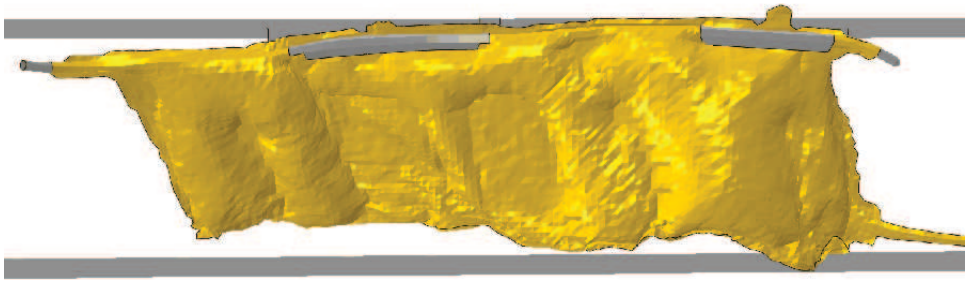


Figure 3: Deformed configuration at 16.25 ms.

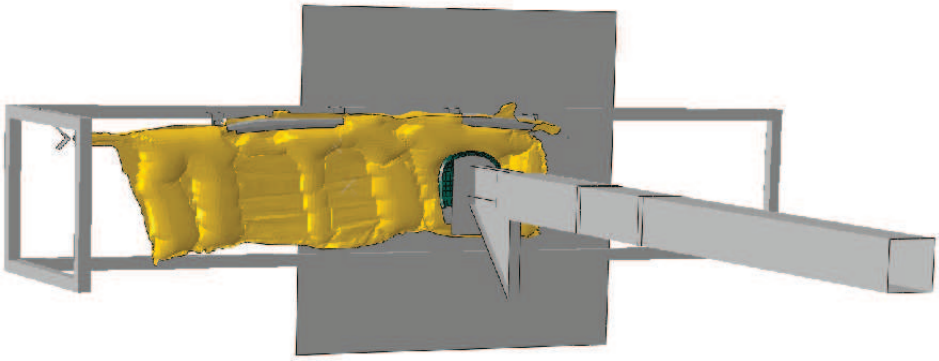


Figure 4: Deformed configuration at 80 ms.

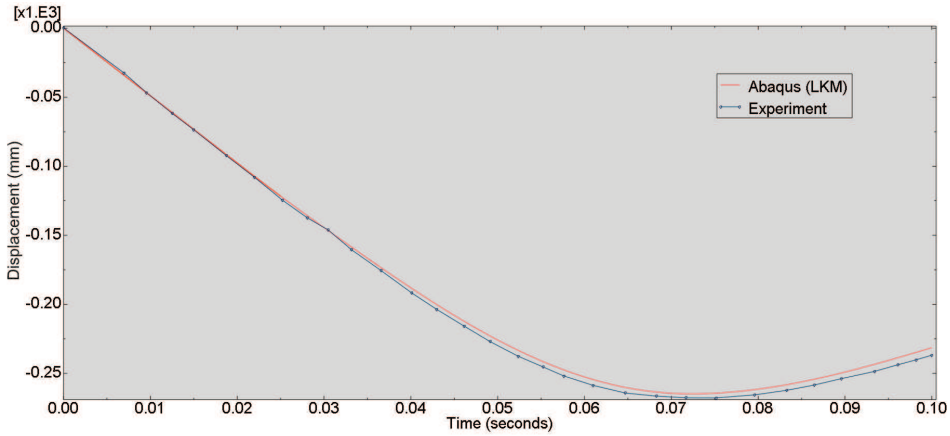


Figure 5: Impactor displacement.

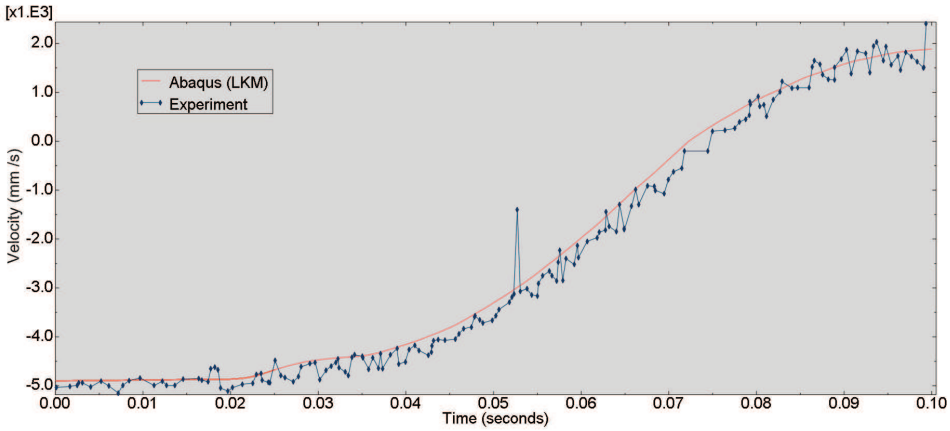


Figure 6: Impactor velocity.

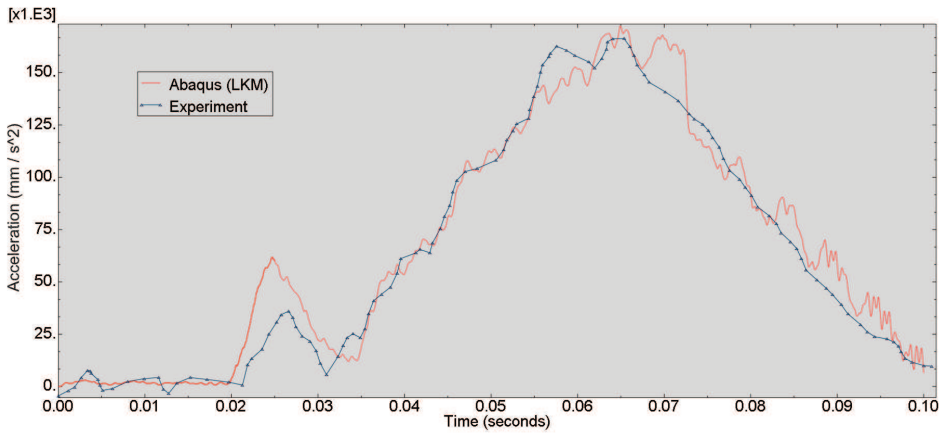


Figure 7: Impactor filtered acceleration.

Mechanism Analyses

In this section:

- *Resolving overconstraints in a multi-body mechanism model*
- *Crank mechanism*
- *Snubber-arm mechanism*
- *Flap mechanism*
- *Tail-skid mechanism*
- *Cylinder-cam mechanism*
- *Driveshaft mechanism*
- *Geneva mechanism*
- *Trailing edge flap mechanism*
- *Substructure analysis of a one-piston engine model*
- *Application of bushing connectors in the analysis of a three-point linkage*
- *Gear assemblies*

Resolving overconstraints in a multi-body mechanism model

Products: Abaqus/Standard

Overview

This example shows how you can resolve overconstraints in a multi-body mechanism.

An overconstraint occurs when multiple consistent or inconsistent kinematic constraints are applied to the same degree of freedom. Overconstraints may lead to inaccurate solutions or prevent convergence. A large number of overconstraint situations are detected and eventually resolved automatically either in the preprocessor or during an Abaqus/Standard analysis (see *Overconstraint Checks*). The vast majority of the overconstraints that are not resolved by the preprocessor are detected by the equation solver. The following symptoms identify such overconstrained models in Abaqus/Standard:

- Zero-pivot warning messages issued in the message (`.msg`) file indicating that the system of equations is rank deficient.
- Unreasonably large reaction forces.
- Very large time average forces in the message file.
- A displacement solution that violates the imposed constraints.

By default, overconstraint checks are performed continuously by the equation solver during the analysis. Abaqus/Standard does not resolve these overconstraints. Instead, detailed messages describing the modeling features that generated the overconstraint are issued to help the user resolve the problems. The message first identifies the nodes involved in either a consistent or an inconsistent overconstraint by using zero pivot information from the Gauss elimination in the solver (see *Direct Linear Equation Solver*). A detailed message containing constraint information is then issued.

Geometry and model

This example deals with resolving overconstraints in the multi-body mechanism model shown in *Figure 1*. The model consists of nine rigid bodies interconnected with connector elements (*About Connectors*). The bodies named in the figure are connected as follows:

- `LINK` is connected to both `CRANK` and `DISH` using two CVJOINT (JOIN + CONSTANT VELOCITY) connector elements. Each of these rigid bodies spins about its own axis.
- `TRANS CONNECT` is connected to `DISH` using a JOIN connector element, which acts like a pin connection. `TRANS CONNECT` is also constrained to translate along the direction defined by `TRANS GUIDE` using a TRANSLATOR (SLOT + ALIGN) connector element between the two. In addition, `TRANS CONNECT` is attached to `SLIDE BLOCK` using a HINGE (JOIN + REVOLUTE) connection with the hinge axis oriented along the global *Z*-direction.
- `SLIDE BLOCK` in turn is constrained to slide along `SLIDE GUIDE` using a TRANSLATOR (SLOT + ALIGN) connector element.
- `TRANS GUIDE` and `ROD GUIDE` are connected using a CYLINDRICAL (SLOT + REVOLUTE) connector element.
- `ROD` is allowed to slide in `ROD GUIDE` using a TRANSLATOR (SLOT + ALIGN) connector element.

These connections enable the deformed configuration shown in *Figure 2*. See *Crank mechanism* for another example using this model.

Loading and boundary conditions

Reference nodes 10009 of rigid body ROD and 10006 of rigid body SLIDE GUIDE are fixed completely. In addition, translations and rotations along the global *X*- and *Y*-directions are constrained at reference nodes 10001 of rigid body DISH and 10003 of rigid body CRANK. The mechanism is actuated using a boundary condition to prescribe a rotation of 360° about the global *Z*-direction at reference node 10001 (rigid body DISH) in a static step.

Understanding overconstraint messages

When Abaqus/Standard attempts to find a solution for this model, two zero pivots are identified in the first increment of the analysis suggesting that there are two overconstraints in the model. These overconstraints have to be identified and removed to render the model properly constrained. One way to identify possible overconstraints in the case of simple models is to count the number of degrees of freedom and constraints. There are nine rigid bodies in the model with a total of 54 degrees of freedom. There are 21 constraints specified using a boundary condition. The connector elements enforce additional constraints: three TRANSLATOR connection types enforce 5 constraints each, two CVJOINT connection types enforce 4 constraints each, one CYLINDRICAL connection type enforces 4 constraints, one HINGE connection type enforces 5 constraints, and one JOIN connection type enforces 3 constraints. Thus, the number of constraints enforced by connector elements is 35. Consequently, there are two (21 + 35 – 54) constraints too many in the model, corresponding to the number of zero pivots identified by the equation solver.

To help the user identify the constraints that should be removed, the following message is produced in the message file outlining the chains of constraints that generated the first overconstraint:

```
***WARNING: SOLVER PROBLEM. ZERO PIVOT WHEN PROCESSING ELEMENT 20025
INTERNAL NODE 1 D.O.F. 4

OVERCONSTRAINT CHECKS: An overconstraint was detected at one of the
Lagrange multipliers associated with element 20025. There are
multiple constraints applied directly or chained constraints that
are applied indirectly to this element. The following is a list of
nodes and chained constraints between these nodes that most likely
lead to the detected overconstraint.

LAGRANGE MULTIPLIER: 2321 <-> 863: connector element 20025 type
                     SLOT ALIGN constraining 2 translations and
                     3 rotations
..2321 -> 10007: *RIGID BODY (or *COUPLING-KINEMATIC)
....10007 -> 3159: *RIGID BODY (or *COUPLING-KINEMATIC)
......3159 -> 3031: connector element 20030 type SLOT REVOLUTE
                    constraining 2 translations and 2 rotations
........3031 -> 10008: *RIGID BODY (or *COUPLING-KINEMATIC)
..........10008 -> 3134: *RIGID BODY (or *COUPLING-KINEMATIC)
............3134 -> 2824: connector element 20035 type SLOT ALIGN
                          constraining 2 translations and
                          3 rotations
..............2824 -> 10009: *RIGID BODY (or *COUPLING-KINEMATIC)
................10009 -> *BOUNDARY in degrees of freedom
                          1  2  3  4  5  6
..863 -> 10004: *RIGID BODY (or *COUPLING-KINEMATIC)
....10004 -> 427: *RIGID BODY (or *COUPLING-KINEMATIC)
......427 -> 3157: connector element 20010 type JOIN constraining
                   3 translations
........3157 -> 10001: *RIGID BODY (or *COUPLING-KINEMATIC)
..........10001 -> 780: *RIGID BODY (or *COUPLING-KINEMATIC)
............780 -> 3156: connector element 20005 type
                         JOIN CONSTANT VELOCITY constraining
```

```
                          3 translations and 1 rotations
..............3156 -> 10002: *RIGID BODY (or *COUPLING-KINEMATIC)
................10002 -> 781: *RIGID BODY (or *COUPLING-KINEMATIC)
..................781 -> 3155: connector element 20001 type
                               JOIN CONSTANT VELOCITY constraining
                               3 translations and 1 rotations
.....................3155 -> 10003: *RIGID BODY
                                     (or *COUPLING-KINEMATIC)
.......................10003 -> *BOUNDARY in degrees of freedom
                                 1  2  4  5  6
..........10001 -> *BOUNDARY in degrees of freedom  1  2  4  5
....10004 -> 3158: *RIGID BODY (or *COUPLING-KINEMATIC)
......3158 -> 1539: connector element 20015 type JOIN REVOLUTE
                    constraining 3 translations and 2 rotations
........1539 -> 10005: *RIGID BODY (or *COUPLING-KINEMATIC)
..........10005 -> 1575: *RIGID BODY (or *COUPLING-KINEMATIC)
............1575 -> 2027: connector element 20020 type SLOT ALIGN
                          constraining 2 translations and
                          3 rotations
..............2027 -> 10006: *RIGID BODY (or *COUPLING-KINEMATIC)
................10006 -> *BOUNDARY in degrees of freedom
                          1  2  3  4  5  6

Please analyze these constraint loops and remove unnecessary
constraints.
```

The zero pivot warning message identifies an internal node (Lagrange multiplier) associated with the identified zero pivot. A typical line contains information pertaining to one constraint. The following line from the output:

```
LAGRANGE MULTIPLIER: 2321 <-> 863: connector element 20025 type
                     SLOT ALIGN constraining 2 translations and
                     3 rotations
```

identifies that the Lagrange multiplier associated with the zero pivot enforces one of the five constraints (SLOT and ALIGN) associated with connector element 20025 between user-defined nodes 2321 and 863. Each of the subsequent lines conveys information related to one constraint in the chains of constraints originating at the zero pivot node or in chains adjacent to them. For example, the line

```
....10007 -> 3159: *RIGID BODY (or *COUPLING-KINEMATIC)
```

informs the user that there is a rigid body constraint between nodes 10007 and 3159, while the line

```
................10009 -> *BOUNDARY in degrees of freedom
                          1  2  3  4  5  6
```

states that there is a boundary condition fixing degrees of freedom 1 through 6 at node 10009.

Indentation levels are used to help in identifying the links in a chain of constraints. A detailed explanation of the chains is printed at the first occurrence of an overconstraint in the message file. Using this methodology, the following chains of constraints starting from the two nodes involved in the Lagrange multiplier constraint are identified:

```
Lagrange multiplier: 2321 -> 10007 -> 3159 -> 3031 -> 10008 -> 3134
-> 2824 -> 10009 -> *BOUNDARY

Lagrange multiplier: 863 -> 10004 -> 427 -> 3157 -> 10001 -> 780
-> 3156 -> 10002 -> 781 -> 3155 -> 10003 -> *BOUNDARY

Lagrange multiplier: 863 -> 10004 -> 427 -> 3157 -> 10001
-> *BOUNDARY

Lagrange multiplier: 863 -> 10004 -> 3158 -> 1539 -> 10005 -> 1575
-> 2027 -> 10006 -> *BOUNDARY
```

If any of the chains terminates in a free end (meaning the chain does not form a closed loop or end in a constraint), the chain does not have any contribution in generating the overconstraint. In the example above, all the identified chains terminate in a constraint and, therefore, may contribute to the overconstraint.

A second zero pivot is generated by the same Lagrange multiplier associated with internal node 1 of connector element 20025 at degree of freedom 5. The chains associated with the zero pivot caused at degree of freedom 5 are identical to the ones at degree of freedom 4 and are not repeated in the message file.

Correcting the overconstrained model

A node set containing all the nodes in the chains of constraints associated with a particular zero pivot is generated automatically and can be displayed in the Visualization module.

In most overconstrained models there are many ways to resolve the overconstraints. The most obvious solution in the example above is to eliminate the unnecessary connector constraints. Upon investigation we see that two rotation constraints associated with the `TRANS CONNECT` rigid body are enforced by the SLOT + ALIGN connector element between the `TRANS CONNECT` and `TRANS GUIDE` bodies (nodes 2321 and 863) as well as by the JOIN + REVOLUTE connector element between the `TRANS CONNECT` and `SLIDE BLOCK` bodies (nodes 3158 and 1539), which renders the model overconstrained. Without affecting the intended kinematic behavior of the system, the JOIN + REVOLUTE connector can be replaced by a JOIN connector, enforcing only three displacement constraints and removing the two extra constraints on these degrees of freedom.

It is important to analyze the chains of constraints carefully and remove constraints properly rather than relax any two arbitrary constraints. For example, removing any two boundary conditions would neither produce the desired kinematic behavior nor remove the overconstraints.

An alternative solution is to add flexibility to some of the rigid bodies or constraints, which can be achieved by either making the `TRANS CONNECT` rigid body elastic or using appropriate combinations of CARTESIAN and CARDAN (EULER or ROTATION as well) connectors together with flexible connections (specified using connector elasticity behavior) to enforce some of the kinematic constraints in an approximate manner.

Results and discussion

As mentioned earlier, an additional method of identifying overconstraints is to plot the reaction forces at the constrained degrees of freedom. These forces are unreasonably large in the overconstrained model even though the displacement solution looks acceptable. In this model a plot of connector reaction forces (CRF) for element 20025 (TRANSLATOR connector between nodes 863 and 2321 of rigid bodies `TRANS GUIDE` and `TRANS CONNECT`, respectively) shows that they are unreasonably large. Once the overconstraints are resolved and the analysis is rerun, there are no zero pivots generated in the message file and the connector reaction forces become very small (approximately 10^{-6} magnitude) as expected. Comparisons of the connector reaction forces shown in *Figure 3* and *Figure 4* suggest that the model is properly constrained.

One other option for resolving the overconstraints is to relax some of the constraints associated with the TRANSLATOR (SLOT + ALIGN) connector for which zero pivots were identified. A CARDAN connection type with connector elasticity behavior could be used instead of the ALIGN connector. Alternatively, the HINGE constraint attached to the `SLIDE BLOCK` rigid body could be relaxed by using JOIN and CARDAN connection types with connector elasticity behavior for the CARDAN components.

Input files

mbmech_overconst.inp	Overconstrained multi-body mechanism model.
mbmech_overconst_crig.inp	Overconstrained multi-body mechanism model using the **CONNECTOR ELASTICITY*, RIGID option.
mbmech_resolved.inp	Multi-body mechanism model with overconstraints resolved.

mbmech_resolved_crig.inp	Multi-body mechanism model with overconstraints resolved using the **CONNECTOR ELASTICITY*, RIGID option.
mbmech_nodedefs.inp	Node definitions used in the overconstrained and resolved models.
mbmech_elemdefs.inp	Element definitions used in the overconstrained and resolved models.
mbmech_rigbdefs.inp	Rigid body definitions used in the overconstrained and resolved models.
mbmech_conndefs_oc.inp	Connector element definitions used in the overconstrained model.
mbmech_conndefs_res.inp	Connector element definitions used in the resolved model.
mbmech_cardan_elas.inp	Multi-body mechanism model with the HINGE connector element replaced by a JOIN + CARDAN connection and the **CONNECTOR ELASTICITY* option. This model uses parts and assemblies; the connector elements are defined at the assembly level.
mbmech_cardan_elas_parts.inp	Node, element, and rigid body definitions for all parts used in mbmech_cardan_elas.inp.

Figures

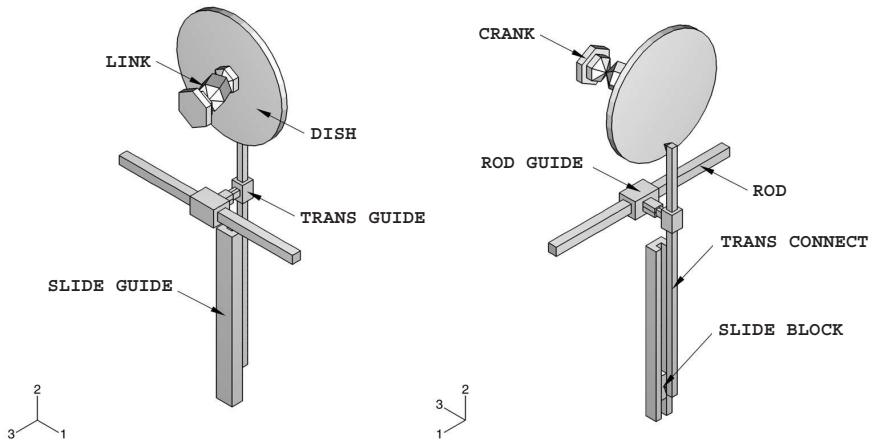


Figure 1: Model of the multi-body mechanism.

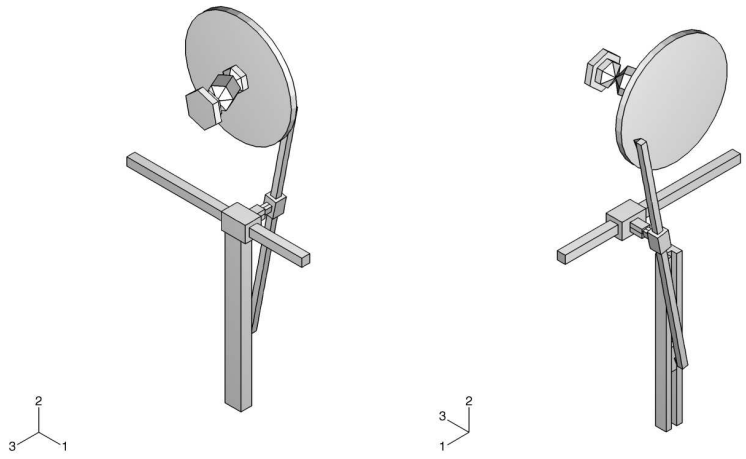


Figure 2: Displaced configuration of the mechanism.

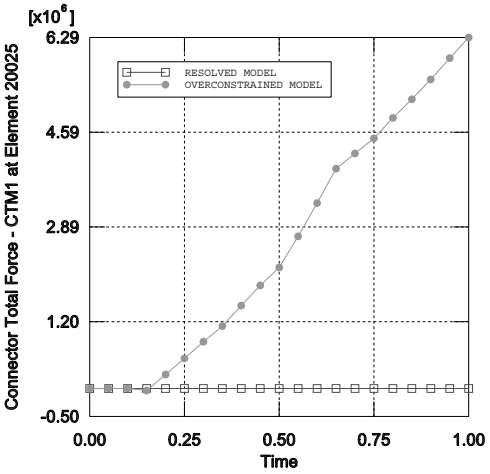


Figure 3: Comparison between time histories of CTM1 in element 20025 for the overconstrained and corrected models.

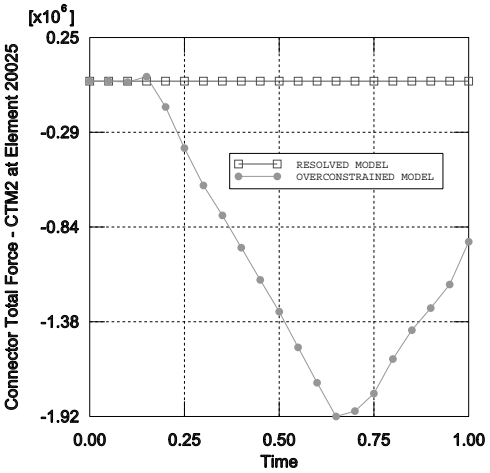


Figure 4: Comparison between time histories of CTM2 in element 20025 for the overconstrained and corrected models.

Crank mechanism

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example illustrates the use of connector elements to model kinematic constraints between rigid bodies in a multi-body mechanism.

Problem description

The crank mechanism considered here transmits a rotational motion through two universal joints and then converts the rotation into translational motion of two slides. The mechanism is modeled using nine rigid components attached with eight connector elements. The various kinematic constraints modeled with connector elements include TRANSLATOR, which allows relative translation along a line but no rotations; HINGE, which allows one relative rotation and fixes relative translations; CYLINDRICAL, which allows relative translation along a line and relative rotation about that line; JOIN, which fixes relative translations but leaves the rotations free; PLANAR, which keeps a point on a plane and allows only relative rotations about the normal to that plane; and UJOINT, which fixes the relative translations and enforces a universal constraint on the relative rotations. The complete model is shown in *Figure 1*.

The axes of rotation of the small and large disks are parallel but offset. A constant angular velocity of the small disk is specified about its axis with a velocity boundary condition on its rigid body reference node. All other degrees of freedom of the rigid body reference node are fixed. The rotational motion of the small disk is transmitted to the large disk through two UJOINT connections and a rigid link. A UJOINT connection, or a universal rotation constraint with shared translational degrees of freedom, between two nonaligned shafts will not transmit constant angular velocity. However, two symmetrically placed universal constraints, as here, will produce constant angular velocity coupling between the two disks. The large disk is connected to a rigid circular rod with a JOIN connection. A JOIN connection is equivalent to a ball-and-socket or a spherical joint. The circular rod connects through a sleeve to a flat block. The rod and sleeve constraint is modeled with a CYLINDRICAL connection, which allows the sleeve to translate along and rotate about the rod. The attachment of the circular rod to the flat block is a HINGE connection, which allows only a single relative rotation about the shared hinge axis. The flat block, in turn, is assumed to slide between two fixed parallel plates. This sliding constraint is modeled with a PLANAR (SLIDE-PLANE and REVOLUTE) connection. The sleeve on the circular rod is connected to a square-section sleeve on the square rod with a HINGE connection. The square rod is fixed in space. The square-section sleeve slides along the square bar without rotating. This sliding constraint is modeled with a TRANSLATOR connection.

A Python script is included that reproduces the model using the Scripting Interface in Abaqus/CAE. The script imports the parts from an ACIS file and creates the constraints and connectors that define the dynamics of the mechanism. The script creates both an Abaqus/Standard and an Abaqus/Explicit model that are ready to be submitted for analysis from the Job module.

Models with frictional interactions, plasticity, and damage in connectors are created by editing the input files without friction to introduce the friction, plasticity, or damage definitions.

A model that includes sensors and actuation via user subroutine *VUAMP* is also included to illustrate a means to model control engineering aspects in multibody analyses. Assuming that an electric motor provides the rotational input to the system, the intent is to shut off this motor (and complete the step) as the horizontal traveling head completes one full cycle. The amplitude properties can be defined in the input file or directly within the user subroutine.

Results and discussion

Figure 2 shows the position of the mechanism at various times. By visual inspection it can be observed that the connector elements are enforcing the correct kinematic constraints.

In this model there are nine rigid bodies with 6 degrees of freedom each, accounting for 54 rigid body degrees of freedom. The eight connector elements eliminate 33 rigid body degrees of freedom through kinematic constraints (enforced via Lagrange multipliers) as itemized in *Table 1*. Hence, the model has 21 rigid body degrees of freedom to be specified as boundary conditions or determined by the solution. In this case all remaining rigid body degrees of freedom are specified as boundary conditions, with the z-component of angular velocity specified for the small disk and 20 additional fixed boundary conditions used.

Files

rigmultimech_std.inp	Abaqus/Standard analysis.
rigmultimech_std_sensor.inp	Abaqus/Standard analysis with sensors and actuation via user subroutine *UAMP* in uamp_rigmultimech.f.
uamp_rigmultimech.f	Abaqus/Standard user subroutine *UAMP* to specify amplitudes.
rigmultimech_exp.inp	Abaqus/Explicit analysis.
rigmultimech_std_fric.inp	Abaqus/Standard analysis with friction.
rigmultimech_exp_fric.inp	Abaqus/Explicit analysis with friction.
rigmultimech_std_plas.inp	Abaqus/Standard analysis with plasticity.
rigmultimech_exp_plas.inp	Abaqus/Explicit analysis with plasticity.
rigmultimech_exp_dam.inp	Abaqus/Explicit analysis with damage.
rigmultimech_exp_sensor.inp	Abaqus/Explicit analysis with sensors and actuation via user subroutine *VUAMP* in vuamp_rigmultimech.f. The amplitude properties are defined in the user subroutine.
vuamp_rigmultimech.f	Abaqus/Explicit user subroutine *VUAMP* to specify amplitudes for rigmultimech_exp_sensor.inp.
rigmultimech_exp_sensor_props.inp	Abaqus/Explicit analysis with sensors and actuation via user subroutine *VUAMP* in vuamp_rigmultimech_props.f.
vuamp_rigmultimech_props.f	Abaqus/Explicit user subroutine *VUAMP* to specify amplitudes for rigmultimech_exp_sensor_props.inp. The amplitude properties are defined in the input file.
rigmultimech_bulk.inp	Node and element bulk data for the rigid bodies.
rigmultimech.py	Python script that creates both an Abaqus/Standard and an Abaqus/Explicit model using Abaqus/CAE. The script imports the parts from an ACIS file named `rigmultimech.sat`.
rigmultimech.sat	ACIS file containing the geometry of the model.

Tables

Table 1: Rigid body degrees of freedom eliminated by kinematic constraints.

Connection type	Number of kinematic constraints	Total rigid body DOFs eliminated from model
UJOINT (2)	4	8
JOIN	3	3

Connection type	Number of kinematic constraints	Total rigid body DOFs eliminated from model
CYLINDRICAL	4	4
HINGE (2)	5	10
SLIDE-PLANE and REVOLUTE	3	3
TRANSLATOR	5	5
		Total eliminated: 33

Figures

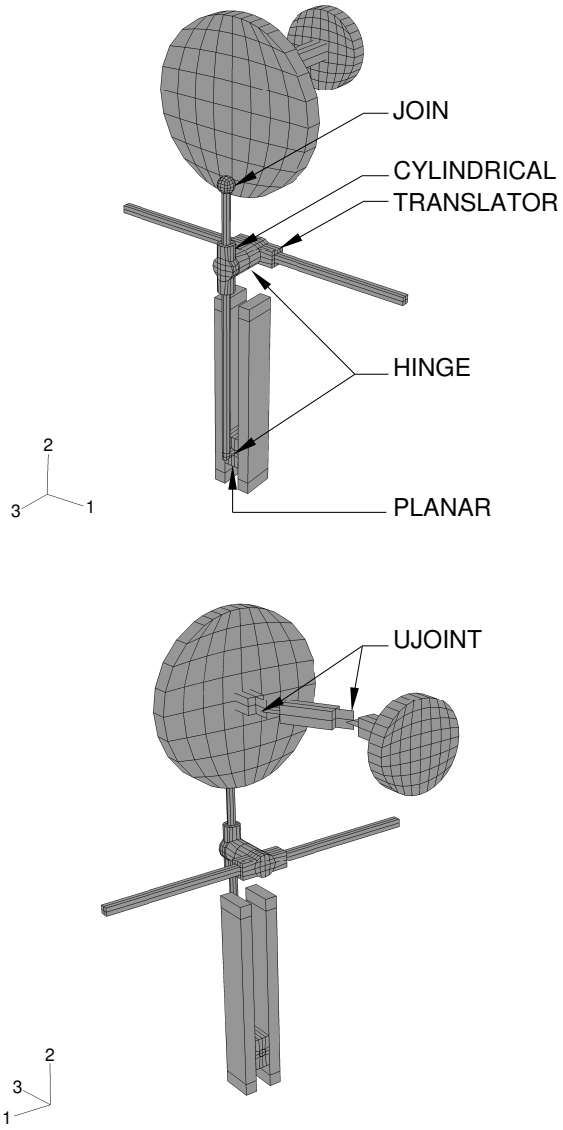


Figure 1: Rigid mechanism model.

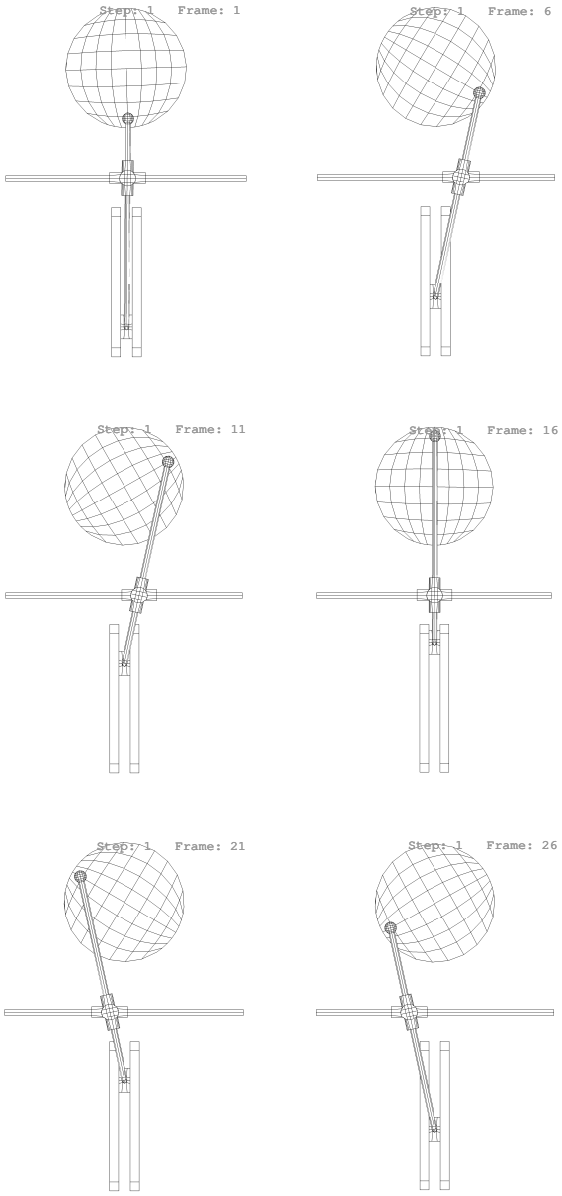


Figure 2: Time history of the motion of the mechanism during the first revolution.

Snubber-arm mechanism

Products: Abaqus/Standard

Overview

This example illustrates the use of connector elements to model a snubbing mechanism (i.e., two solids coming into contact).

Geometry and model

A snubber is used to cushion the closing of a cowling on an aircraft. The cowling opens and closes by applying a torque about the *Z*-axis at a fixed point (the origin, Q, in *Figure 1*). The snubber-arm assembly is attached to the cowling at point C. The snubber arm is constrained at point A by a spring and is allowed to rotate relative to the cowling about point C. The other end of the spring is attached to the cowling at point E. A shoe is attached to the snubber arm at point B and contacts with a block as the cover closes, as shown in detail in *Figure 2*. The shoe is allowed to pivot about point B. The spring retains the snubber arm against a stop on the cowling with an initial force of 50 lbs when the cover is open. The shoe contacts the block as the cover closes.

A small rotational spring (10 lb/rad) keeps the shoe from rotating relative to the snubber arm while the shoe is not in contact with the block. The block is fixed to the ground. As the shoe comes into contact, the two surfaces slide relative to each other. The coefficient of friction between the block and the shoe is 0.5.

The closing speed of the cowling is slow enough so that inertia effects can be neglected. *Figure 1* shows the position of the mechanism when the shoe and the block make initial contact. The cowling rotates five more degrees about point Q after the block and the shoe make initial contact.

Model interactions

The bodies named in *Figure 1* and *Figure 2* are connected as follows:

- `ARM` is connected to `COWLING` at point C using JOIN and ROTATION connector elements. The rotation is constrained using a connector stop to prevent `ARM` from rotating past the stop.
- The spring connecting `ARM` to `COWLING` is modeled using an AXIAL connector element defined between points A and E. The pre-tension in the spring is modeled using reference lengths and angles to be used in specifying connector constitutive behavior.
- `SHOE` is connected to `ARM` at point B using JOIN and ROTATION connector elements. The small rotational spring is modeled using connector elasticity behavior.

All bodies in the model are modeled using display bodies connected to the relevant connector nodes. `SHOE` (deformable elements) and `BLOCK` (rigid surface) make exception to this rule to allow contact interaction definition between these bodies. Some models include friction and plasticity effects in the connectors.

Results and discussion

Figure 3 shows the displaced configurations of the mechanism as the cowling is rotated. As the cowling opens, the spring pulls on the arm at point C until the arm is stopped. After contact is established, when the cowling is closing, the rotation of the arm increases the tension in the spring. During a flight this mechanism allows the absorption of vibrations between the cowling and the aircraft.

Such analyses allow one to study not only the dynamics of the snubber-arm mechanism but also the contact interaction between the block and the shoe. *Figure 4* shows the contact pressure and frictional shear stress at a node located at the center of the shoe contact surface. Over time the coefficient of friction between the shoe and the block will decrease. Knowledge of the contact characteristics between the block and the shoe is a critical component of the design of the mechanism.

Input files

snubber_model.py	Python replay file for constructing the snubber-arm mechanism model in Abaqus/CAE.
snubber.inp	Snubber-arm mechanism model.
snubber_surf.inp	Snubber-arm mechanism model with surface-to-surface contact.
snubber_fric.inp	Snubber-arm mechanism model with friction.
snubber_plas.inp	Snubber-arm mechanism model with connector plasticity.

Figures

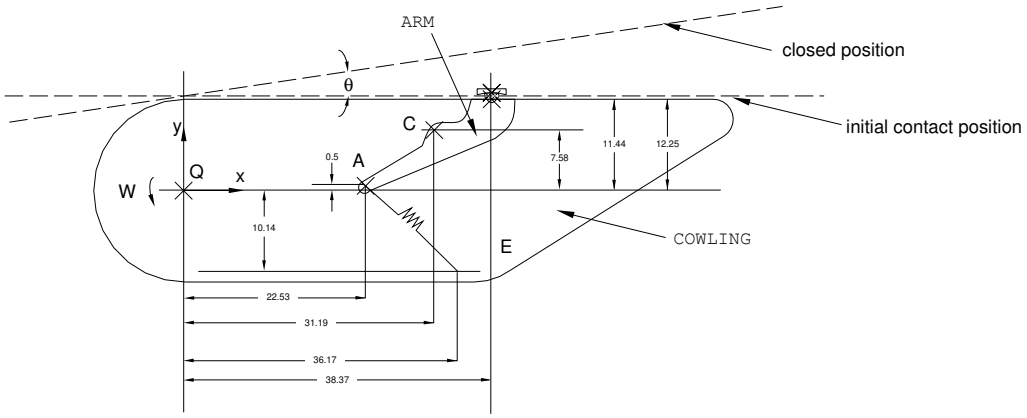


Figure 1: Undeformed configuration of the snubber-arm mechanism.

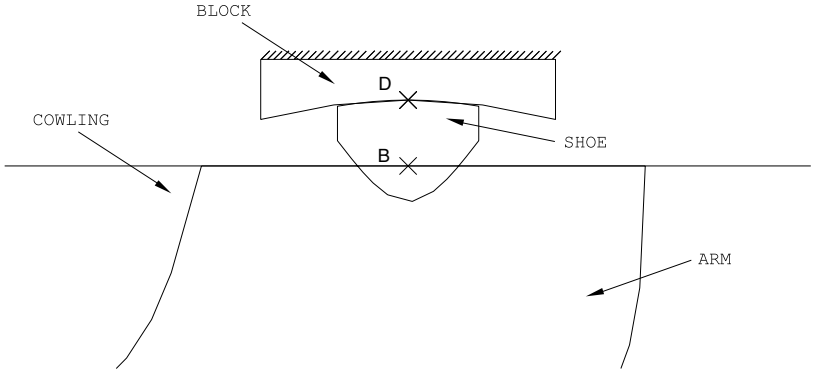


Figure 2: Undeformed configuration of the shoe and the block.

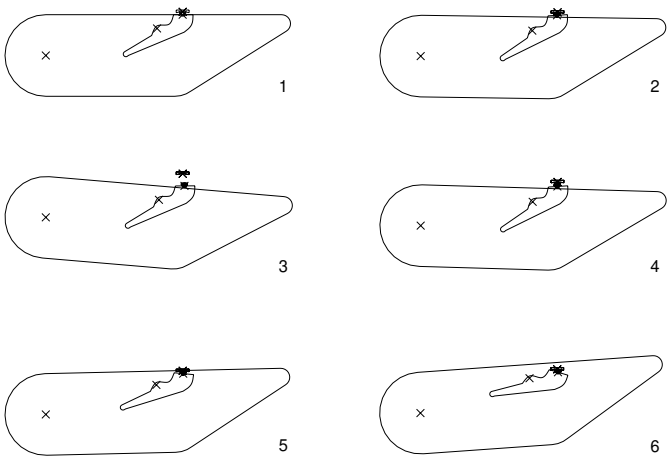


Figure 3: Displaced configurations of the mechanism.

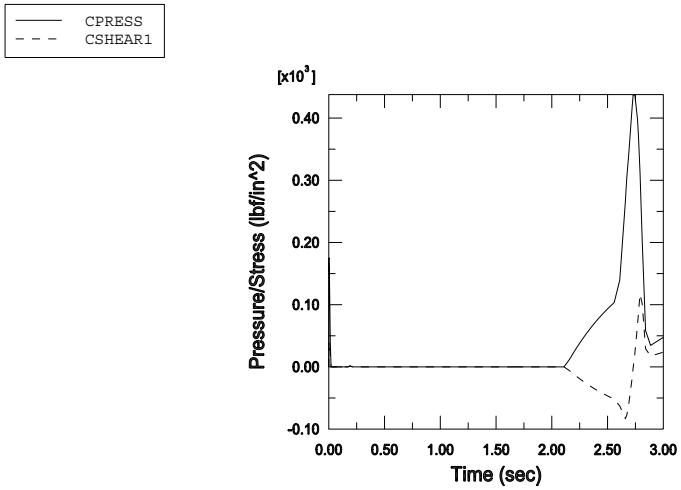


Figure 4: Contact pressure and frictional shear stress at the center of the shoe contact surface.

Flap mechanism

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example illustrates the use of connector elements to model a three-dimensional trailing edge mechanism.

Geometry and model

The complete model of the flap is shown in *Figure 1*. An actuator rotates a bell crank through the deployment of an actuator arm. The bell crank pushes and pulls a connecting rod that attaches to the arm of the flap.

The flap is connected to a rigid shaft on the aircraft wing structure at points E and I. An arm on the shaft is attached to the rod at point F. The other end of the rod attaches to a bell crank at point D. The bell crank is attached to the airplane so that it can rotate about point B. The axis of rotation of the bell crank passes through point B and is parallel to the global *Z*-axis. The rotation of the bell crank is driven by the deployment of the actuator arm. The actuator system attaches to the bell crank at point C. The other end of the actuator system is attached to the aircarft structure at point A and is allowed to compensate for the change of angle caused by the rotation of the bell crank.

A load is applied at the center of gravity of the flap so that it is colinear and oriented along the global *Z*-axis.

Model interactions

The bodies named in *Figure 1* are connected as follows:

- `ACTUATOR` is connected to the ground at point A using a HINGE connector element. `ACTUATOR` and `ACTUATOR ARM` are connected using a TRANSLATOR connector element. The available component of relative motion in the translated connector element is used to modify the configuration of the actuator system as a function of time.
- `BELL CRANK` is physically attached to the ground with a hinge connection at point B. The axis of rotation of the hinge connection is parallel to the global *Z*-axis. However, using a HINGE connector element to attach `BELL CRANK` to the ground would overconstrain the model. Because of the connections used between point A and point C, point C is already constrained to travel in the global *X*–*Y* plane. Because the position of point B has to remain fixed in space, the rotation of `BELL CRANK` about the $\boldsymbol{BD}$-axis is already constrained. As a result, only three translations and the rotation of `BELL CRANK` about the $\boldsymbol{BC}$-axis need to be constrained to realize the hinge connection. `BELL CRANK` is, thus, attached to the ground using JOIN and UNIVERSAL connector elements at point B. The UNIVERSAL connection is used to constrain the relative rotation of `BELL CRANK` with respect to the ground about the $\boldsymbol{BC}$-axis.
- `ROD` is connected to `BELL CRANK` at point D and to `ARM` at point F using JOIN connector elements. A CARDAN connector element is added at point F between `ROD` and `ARM` to introduce an elastic behavior to prevent the free rotation of `ROD` around its axis.
- `FLAP` and `ARM` are connected by a WELD connector element at point E. `FLAP` is attached to the ground using a HINGE connection at point I.

All bodies in the model are visualized using display bodies connected to the relevant connector nodes. Separate models in Abaqus/Standard and Abaqus/Explicit include friction, plasticity, and damage in the connectors.

Results and discussion

The amplitude curve used to drive the deployment of the actuator arm is shown in *Figure 3*. *Figure 2* shows the configuration of the flap mechanism at intermediate instants as it is actuated. As the actuator system is deployed, the flap rotates around the EI-axis to modify the aerodynamics of the wing.

Input files

flap_model.py	Python replay file for constructing the flap mechanism model in Abaqus/CAE.
flap.inp	Abaqus/Standard flap mechanism model.
flap_fric.inp	Abaqus/Standard flap mechanism model with friction.
flap_exp_fric.inp	Abaqus/Explicit flap mechanism model with friction.
flap_plas.inp	Abaqus/Standard flap mechanism model with plasticity.
flap_exp_plas.inp	Abaqus/Explicit flap mechanism model with plasticity.
flap_exp_plas_dam.inp	Abaqus/Explicit flap mechanism model with plasticity and damage.

Figures

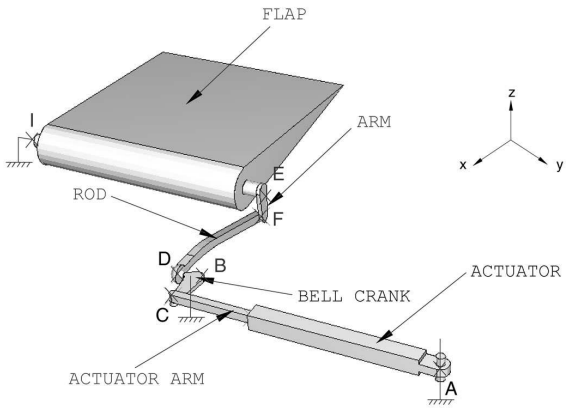


Figure 1: Undeformed configuration of the flap mechanism.

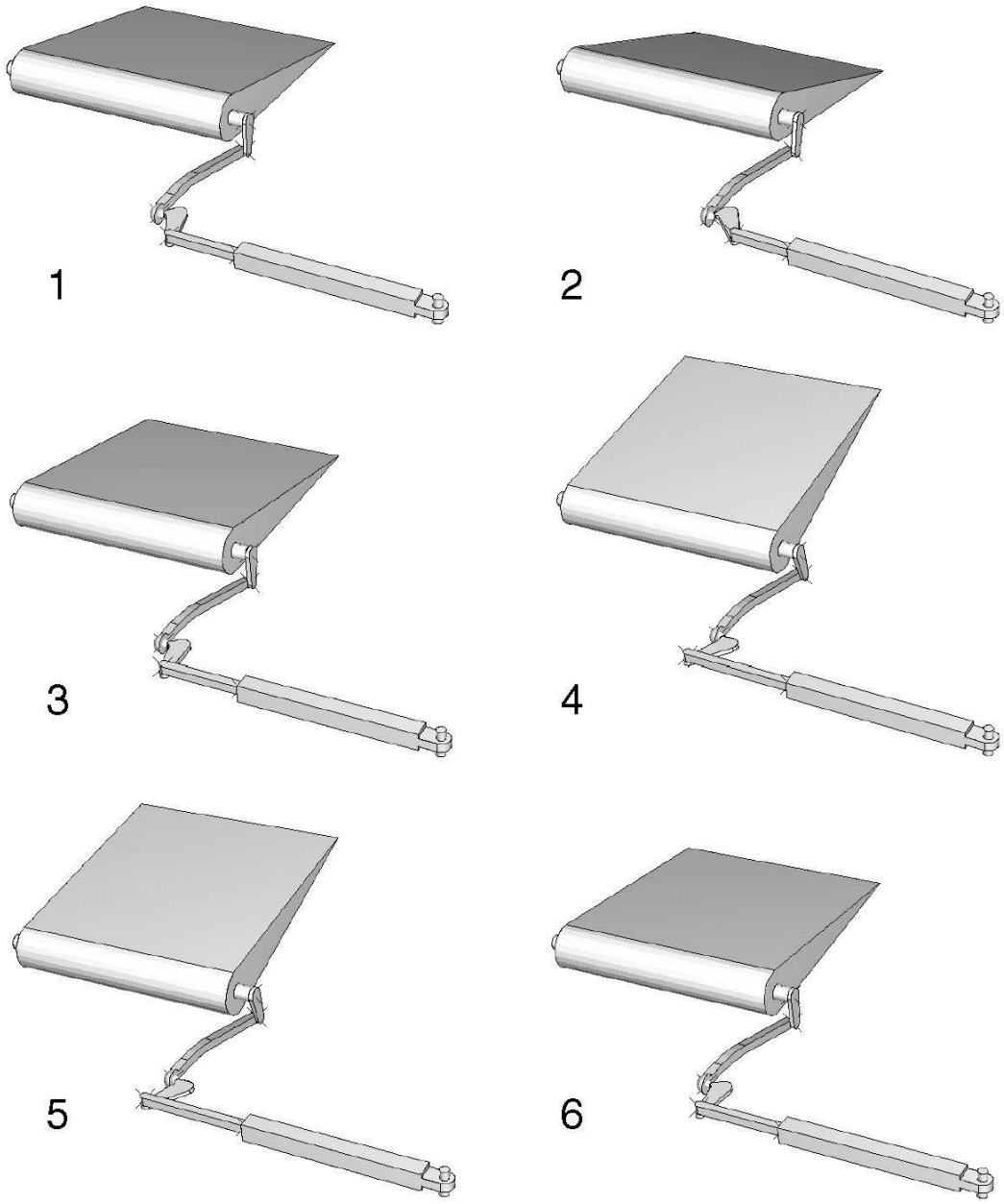


Figure 2: Deformed configurations of the flap mechanism.

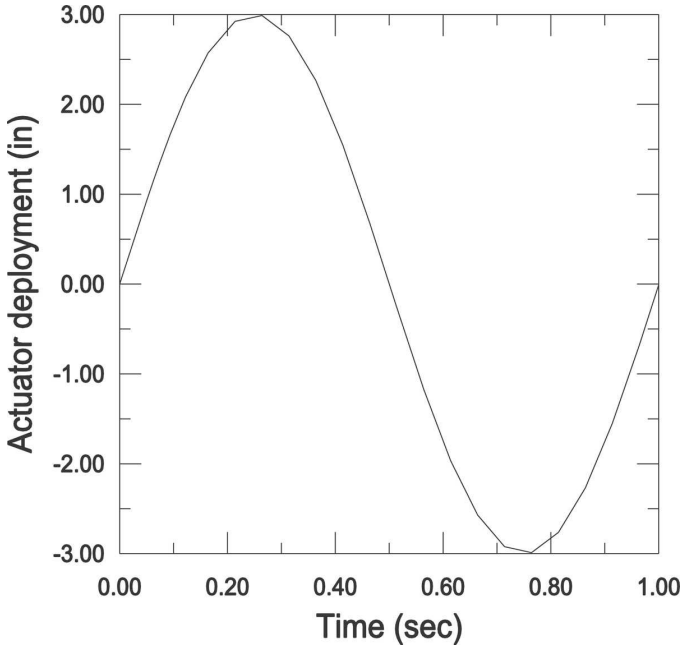


Figure 3: Translator connector motion.

Tail-skid mechanism

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example illustrates the use of connector elements to model the tail-skid mechanism of an airplane.

Geometry and model

The complete model of the tail-skid mechanism is shown in *Figure 1*. It consists of a bell crank, an actuator, a tail-skid arm, and a large cylinder. The bell crank is connected to the actuator at point A and to the large cylinder at point B. The tail-skid arm is connected to the actuator at point D and to the large cylinder at point E. The large cylinder is a compressible single-use cartridge designed to absorb energy in the unlikely event of a tail strike. As such, it behaves like a compression-only linear spring with a stiffness K=875600 N/m (K=5000 lb/in) and a damping coefficient C=175100 N/m sec (C=1000 lb/in sec). The bell crank pivots about point C to move the large cylinder. The whole mechanism is attached to the aircraft frame through points C and D. All point locations are listed in *Table 1*.

Two types of analyses are of interest: the deployment and stowage kinematics analysis and the tail strike analysis. In the kinematics analysis the actuator contracts a distance of 145.8 mm (5.74 in), which will rotate the bell crank and move the mechanism from the deployed to the stowage position. The large cylinder does not compress during stowage or deployment. The tail strike analysis simulates the over-rotation of an aircraft during takeoff. In this analysis the tail-skid arm strikes the ground at a speed of 1.839 m/s (6.0 ft/s) at point F when it is at the deployed position. The strike creates a force acting at an angle of 10 degrees with respect to the 2-axis as a result of friction between the airplane and the runway. During the strike the large cylinder is compressed a maximum distance of 101.6 mm (4.0 in) to absorb energy before it bottoms out. The mass and inertia of the tail-skid components are assumed to be negligible. The body of the airplane is assumed to be rigid, has a mass of 6193 kg (270000 lb), and a mass moment of inertia of 2220 kg m^2 (1.5e8 lb in^2) relative to the 3-axis. The center of gravity of the airplane is located at –18.06 m, –1.422 m, 0.0 m (–711.0 in, –56 in, 0.0 in); it is not shown in the model because the dimension of the aircraft is much larger than the mechanism.

Model interaction

The correct behavior of the tail-skid mechanism is modeled by defining appropriate connectors between the discrete points in *Figure 1*. For visualization purposes, display bodies are attached to those points to model the bell crank, actuator, tail-skid arm, and large cylinder.

- The actuator behavior is modeled using an AXIAL connector between points A and C.
- The large cylinder behavior is modeled by the combination of an AXIAL connector and a SLIDE-PLANE connector between points E and G.
- The bell crank is connected to the large cylinder by a HINGE (REVOLUTE+JOIN) connector and to the actuator by a JOIN connector.
- The tail-skid arm is connected to the actuator and the large cylinder at points D and E, respectively, using two HINGE connectors.

The tail-skid mechanism is then connected to the aircraft frame at points C and D using two HINGE connectors. The air frame is modeled with two BEAM connectors connecting points C and D to the center of gravity of the airplane. Since both the actuator and the large cylinder consist of two display bodies, additional connectors are

necessary to constrain the relative motion between them. For this purpose a SLIDE-PLANE connector element and an ALIGN connector element are defined between points A and D, and an ALIGN connector element is defined between points D and G.

In the kinematics analysis the position of the aircraft is fixed; the contraction of the actuator, which is realized through prescribed connector displacement of the AXIAL connector, moves the mechanism from the deployed to the stowage position. In the tail strike analysis the over-rotation of the airplane is modeled by allowing the airplane to be in free motion and applying an initial rotating velocity to it. During the strike the direction of the reaction of the ground to the airplane is assumed to remain fixed. As a result the reaction force is modeled by applying a fixed boundary condition at point F in a local coordinate system that will generate a reaction at an angle of 10 degrees with the 2-axis. The actuator is fixed during the strike, while the large cylinder is compressed. The large cylinder will absorb energy until it stops after being compressed 4 inches. This physical behavior of the large cylinder is modeled using the connector elasticity, connector damping, and connector stop behaviors. Separate models include friction, plasticity, and damage in the connectors.

In Abaqus/Explicit the tail strike analysis is repeated with a user-defined element in addition to the axial connector between points A and D. The element is modeled to mimic the elastic behavior of an axial connector; the damping and locking behavior are still defined through the axial connector acting in parallel with the user-defined element.

Results and discussion

A sequence of the deformed tail-skid mechanisms in the kinematics analysis is shown in *Figure 2*. By visual inspection it can be observed that the connector elements are enforcing the correct kinematic constraints. Some of the numerical results for the tail strike analysis are shown in *Figure 3* and *Figure 4*. *Figure 3* shows the total, elastic, and viscous force of the AXIAL connector that models the mechanical behavior of the large cylinder. *Figure 4* shows the relative distance between points G and D. The sudden changes of the connector force and displacement shown in both figures are due to the large cylinder reaching its maximum contraction (modeled using connector stops) during tail strike.

Input files

tail_kinematics.inp	Kinematics analysis.
tail_strike.inp	Tail strike analysis.
tail_strike_fric.inp	Tail strike analysis with friction.
tail_strike_exp_fric.inp	Abaqus/Explicit tail strike analysis with friction.
tail_strike_exp_fric_plas_dam.inp	Abaqus/Explicit tail strike analysis with friction, plasticity, and damage.
tail_kinematics_model.py	Python replay file for constructing the kinematics model using Abaqus/CAE.
tail_strike_model.py	Python replay file for constructing the tail strike model using Abaqus/CAE.
vuel_tail_strike_exp_fric.inp	Abaqus/Explicit tail strike analysis with a user-defined element modeling the elastic behavior of one of the axial connectors. The element is defined in user subroutine vuel_axial_connector3d.f.
vuel_axial_connector3d.f	User subroutine *VUEL* to model the axial connector.

Tables

Table 1: Point locations in the deployed position.

Point	X	Y	Z
A	14.95	19.62	0.00
B	14.15	10.24	0.00
C	15.29	16.13	0.00
D	0.00	0.00	0.00
E	13.69	−3.49	0.00
F	24.72	−12.24	0.00

Figures

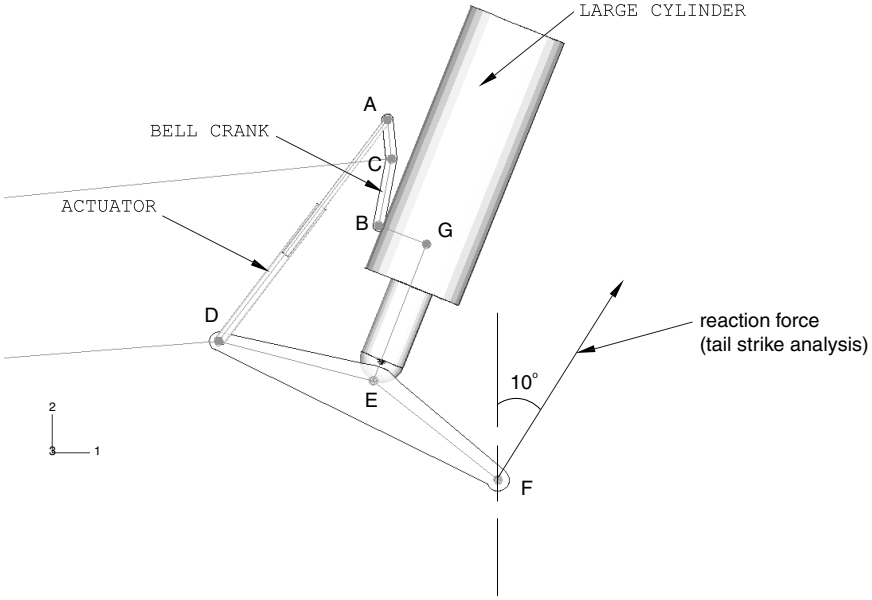


Figure 1: Tail-skid mechanism model (the deployed position).

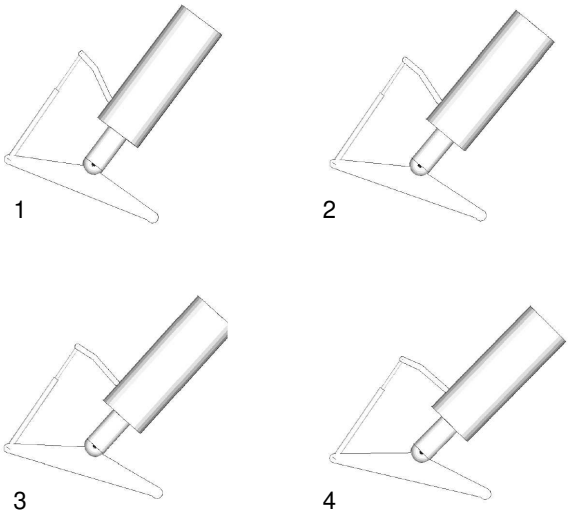


Figure 2: Intermediate positions of the tail-skid mechanism during tail strike.

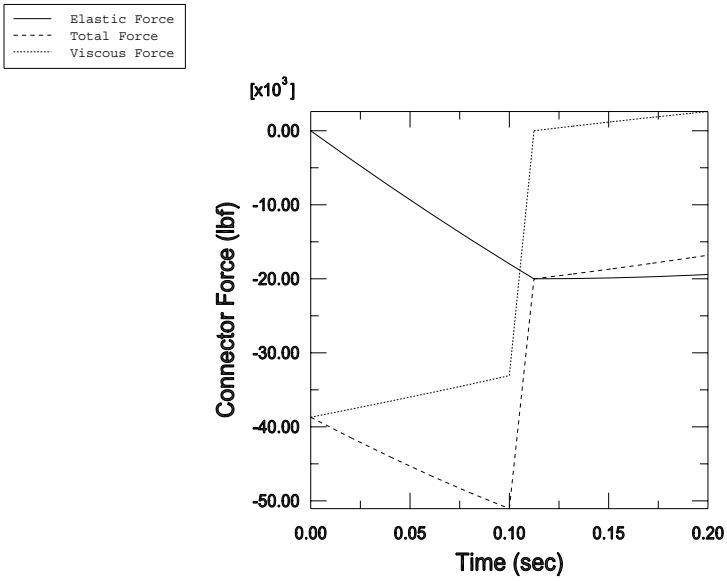


Figure 3: Elastic and damping forces of the connector modeling the large cylinder.

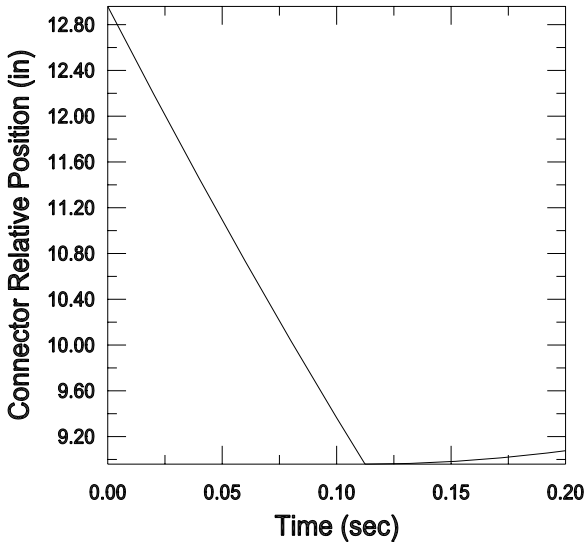


Figure 4: Relative displacement of the connector modeling the large cylinder.

Cylinder-cam mechanism

Products: Abaqus/Explicit

Overview

This example illustrates the use of connector elements to model a cylinder-cam mechanism.

Geometry and model

The cylinder-cam mechanism is shown in *Figure 1*. The solid aluminum cylinder is lying on the *Z*-axis and has a slot around its circumference. The centerline of the slot is defined by a plane intersecting the cylinder at 45 degrees. The slot has a radius of 7.62 mm (0.30 in) and a depth of 7.62 mm (0.30 in). A pin with a spherical end of radius 6.35 mm (0.25 in) is set into the slot. The pin is constrained to be parallel to the *X*-axis and remains in the *X*–*Z* plane. Because of the difference in radii, there is a 1.27 mm (0.05 in) gap between the slot and the pin head. As the cylinder rotates about the *Z*-axis, the pin travels in a track that is parallel to the cylinder. The pin has a weight of 13.34 N (3.0 lbf).

The cylinder rotates with a speed of 30 rpm, which drives the pin back and forth in the track. Resistance to the motion is caused by the friction between the pin and the track. The tolerance mismatch due to the pin head being smaller than the slot is considered in the model.

Model interactions

As shown in *Figure 1*, the track, the pin, and the cylinder are modeled using display bodies. The bodies in *Figure 1* are connected as follows:

- MASS and ROTARYI elements are attached to each display body through BEAM connector elements to account for the inertia of each part in the model.
- The interaction between `PIN` and `TRACK` is modeled using ALIGN+SLOT connectors between point D and point E. The friction dissipation effects between `PIN` and `TRACK` are taken into account using connector damping behavior.
- The interaction between `PIN` and `CYLINDER` is modeled by defining a CARTESIAN connector element between point B and point C. The 1.27 mm (0.05 in) gap between the pin and the slot is modeled using a connector stop. The local coordinate system of the CARTESIAN connector element (used to measure the displacement of point C relative to point B) is attached at point B to `CYLINDER` and has its 2–3 plane lying in the plane of the slot. The connector stop is defined along component 1 of the connector element. A displacement of 1.27 mm (0.05 in) of the pin relative to the center of the slot along the *Z*-axis corresponds to a displacement of point C relative to point B of 0.898 mm (0.035 in) in the connector local coordinate system. The connector stop is, thus, defined at a distance of 0.898 mm (0.035 in) in the positive and negative 1-directions of the CARTESIAN connector element.

Results and discussion

Figure 2 shows the positions of the mechanism as the cylinder is rotated. The time histories of the displacement of point C in the global coordinate system and the displacement of point C relative to point B in the CARTESIAN connector local coordinate system are shown in *Figure 3*. In the connector local coordinate system, point C travels a maximum distance of 0.035 in relative to point B. This corresponds to the value given in the connector stop definition. When `PIN` and `CYLINDER` are in contact, the displacement of point C relative to point B in the

connector local coordinate system remains constant, and the cylinder forces the pin to translate. When contact is lost, the connector relative displacement varies with time, and the motion of the pin stops as a result of damping.

Files

cylcammech.inp	Abaqus/Explicit analysis.
cylcammech.py	Python script that creates an Abaqus/Explicit model using Abaqus/CAE. The script imports the parts from an ACIS file named `cylcammech.sat`.
cylcammech.sat	ACIS file containing the geometry of the model.

Figures

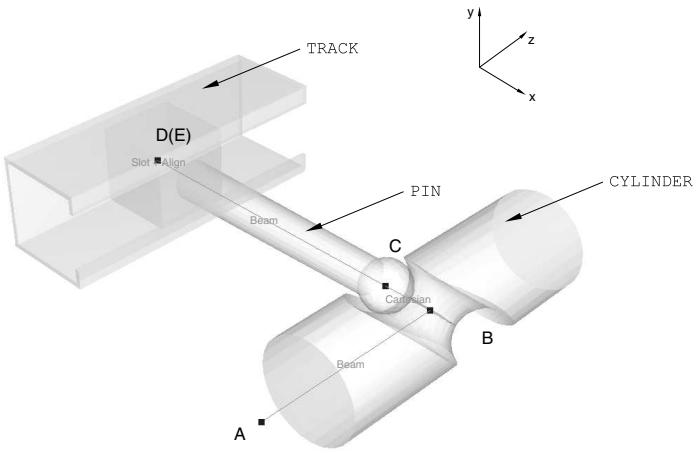


Figure 1: Cylinder-cam mechanism.

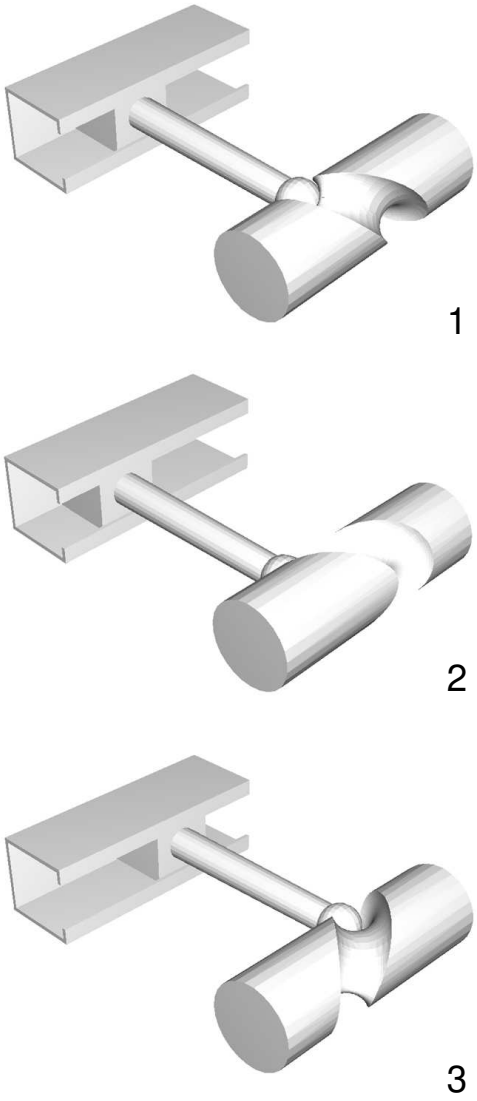


Figure 2: Displaced positions of the mechanism.

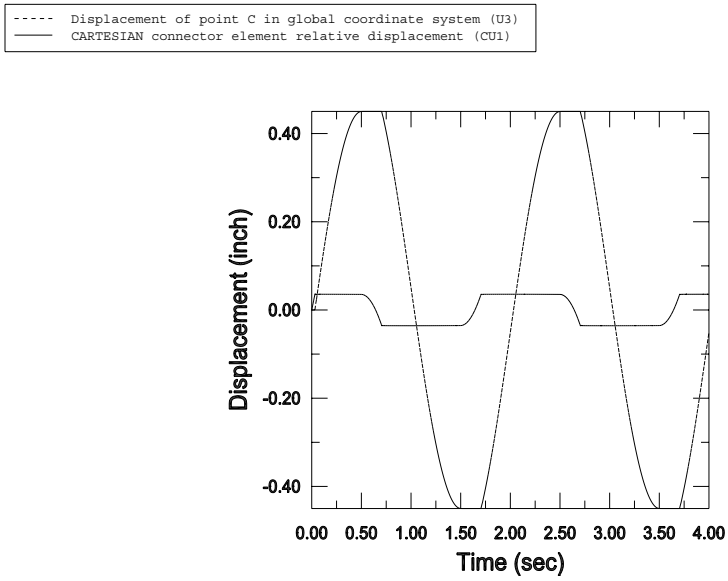


Figure 3: Time histories of the displacement of the pin head (Point C).

Driveshaft mechanism

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example illustrates the use of connectors to model a driveshaft mechanism.

Geometry and model

Driveshaft mechanisms are important components of motor vehicles. These mechanisms often contain universal joints that are used to carry motion from one shaft to another where the two shafts are not perfectly aligned. A typical universal joint assembly is shown in *Figure 1*. The joint consists of a cross that carries needle roller bearings at the four extremities. The bearing cups are attached to yokes, two on the input shaft and two on the output shaft. The universal joint is not a constant velocity joint. If the input and output shafts are not aligned, the uniform rotation of the input shaft will result in a nonuniform rotation of the output shaft. When the input shaft is rotated at constant velocity, the output shaft will accelerate and decelerate twice per revolution (a pulsation effect).

The driveshaft mechanism in this example problem consists of three shafts, `SHAFT AB` (the input shaft), `SHAFT CD` (the center shaft), and `SHAFT EF` (the output shaft). `SHAFT AB` is connected to `SHAFT CD` through a universal joint that connects B and C. `SHAFT CD` is connected to `SHAFT EF` through a universal joint that connects D and E. B31 beam elements are used to model the shafts. `SHAFT AB` and `SHAFT EF` have a length of 7 units each, and `SHAFT CD` is 10 units long. `SHAFT AB` and `SHAFT EF` are coplanar and parallel to each other. `SHAFT CD` is also coplanar to `SHAFT AB` and `SHAFT EF` but is inclined to `SHAFT AB` and `SHAFT EF` at an angle of 20 degrees. `SHAFT AB`, `SHAFT CD`, and `SHAFT EF` are circular in cross-section; and the radius of their cross-section is 0.1 units. The universal joints connecting B and C and connecting D and E are modeled using CONN3D2 elements.

The variation of angular motion between the input and output shafts can be avoided. The pulsation effect between the input shaft and the center shaft can be compensated entirely by the pulsation effect between the center shaft and the output shaft if the universal joints are oriented properly with respect to each other. A sufficient condition for the input/output velocity ratio to be constant is that the directions normal to the crosses of the universal joints be colinear. The configuration of the shafts is illustrated in *Figure 2*. In this figure we can see that the crosses lie in parallel planes. This will guarantee a constant input/output velocity ratio.

Model interactions

All the degrees of freedom except the rotational degree of freedom about the beam axis are fixed at A and F. A prescribed rotation of 360 degrees is specified at A. No other boundary conditions are specified. The bodies in *Figure 2* are connected as follows:

- `SHAFT AB` is connected to `SHAFT CD` with a UJOINT connector element (Joint 1).
- `SHAFT CD` is connected to `SHAFT EF` with a UJOINT connector element (Joint 2).

Because the rotary motion of one shaft is to be transmitted to another shaft that is not colinear, separate coordinate orientation systems are created and used to connect the connector nodes for each UJOINT connector connecting the shafts. The coordinate systems used to define Joint 2 are the systems used to define Joint 1 rotated by 90 degrees about the global 1-axis. This ensures proper orientation of the joint crosses and a constant input/output velocity ratio. Separate models for Abaqus/Standard and Abaqus/Explicit include friction in the connectors.

Results and discussion

The rotation of `SHAFT AB` about its axis leads to a rotation of `SHAFT EF` about its axis. The rotation of the input and output shafts about their axis is plotted in *Figure 3*. We can see that the motion of these shafts is perfectly synchronized.

Input files

driveshaft_model.py	Python replay file for constructing the driveshaft mechanism model in Abaqus/CAE.
driveshaft.inp	Abaqus/Standard driveshaft mechanism model.
driveshaft_fric.inp	Abaqus/Standard driveshaft mechanism model with friction.
driveshaft_exp_fric.inp	Abaqus/Explicit driveshaft mechanism model with friction.

Figures

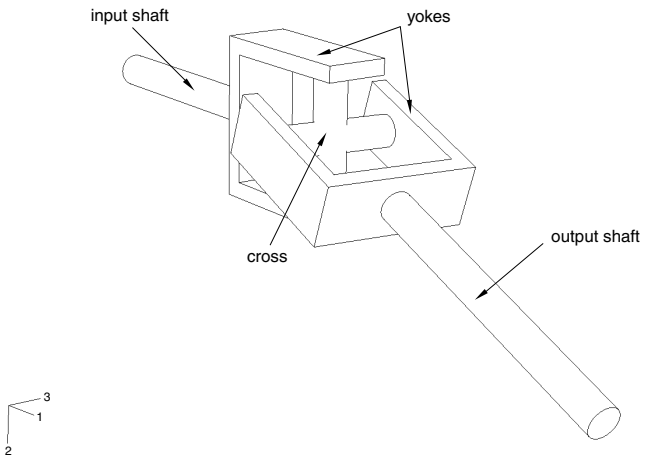


Figure 1: Universal joint.

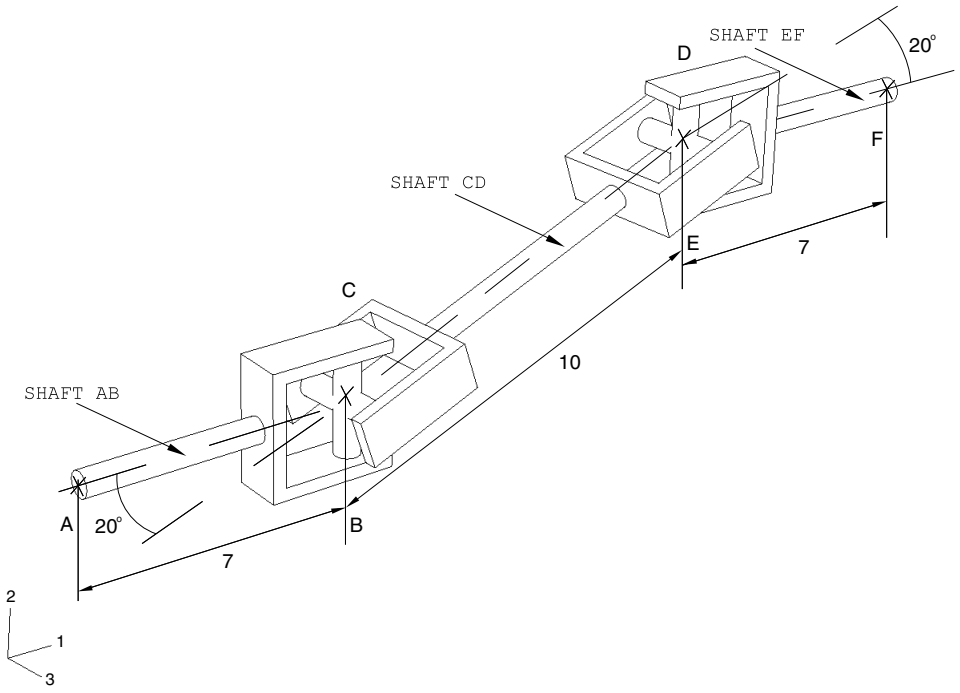


Figure 2: Configuration of the driveshaft mechanism.

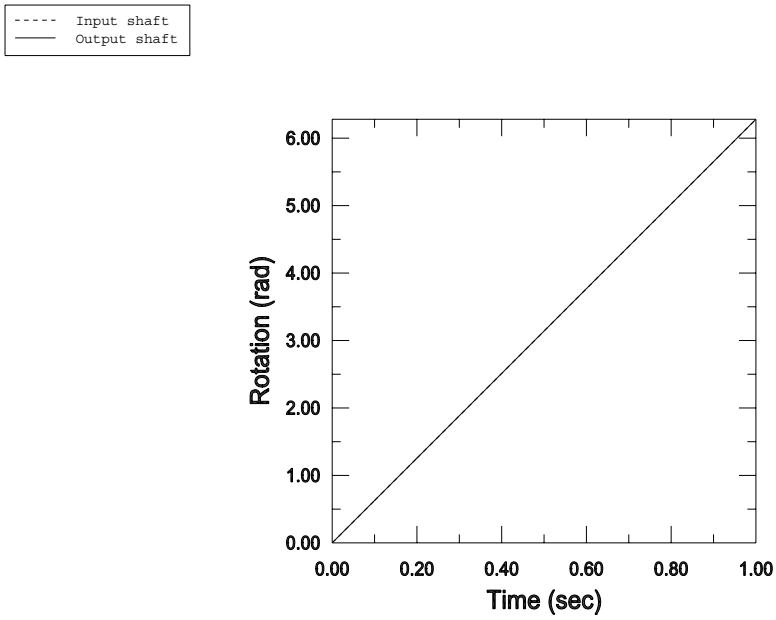


Figure 3: Rotation of the input and output shafts about the global 1-axis.

Geneva mechanism

Products: Abaqus/Standard

Overview

This example illustrates the use of connector elements to model a Geneva mechanism, which converts continuous rotary motion into intermittent rotary motion.

The Geneva mechanism is essentially a timing device. It is used in counting instruments and other applications where a continuous rotary motion needs to be converted to an intermittent rotary motion. For example, it is used in clocks to limit the number of winding rotations of the clock spring and in movie film projectors to move the film frame by frame. The Geneva mechanism consists of a rotating body with a protruding pin and another rotating body with slots into which the pin slides.

Geometry and model

In this example problem the mechanism consists of three bodies named `PIN`, `SLOT1`, and `SLOT2`, as illustrated in *Figure 1*. `SLOT1` is an analytical rigid surface that overlays `SLOT2`. `SLOT2` is modeled as a display body. `SLOT1` and `SLOT2` are rigidly joined to each other at point A so as to allow them to rotate in unison about A. `SLOT2` has a radius of 3.0 units and is 1.0 units thick. The pin is constrained to be rigid. The thickness of the rigid part of `PIN` is 0.5 units. The rigid portion of `PIN` has a reference point B, and `PIN` is allowed to rotate about B. Distance AB is 4.24264 units in the model. The protruding portion of `PIN` is located at a distance of 3.0 units from reference point B, and its length is 1.0 units. The rigid portion of `PIN` has a radius of 3.0 units.

Model interactions

The contact between `PIN` and `SLOT1` takes place through a single secondary node located at C, the center of the protruding portion of `PIN`. As a result, the slots in `SLOT1` (used for contact evaluation) are 0.05 units wide, whereas the slots in `SLOT2` (used for display purposes only) are 0.25 units wide. The slots have a length of about 2.0 units to allow for the pin to slide. This contact is considered to be frictionless.

All the degrees of freedom at A and B, except the rotational degrees of freedom about the 3-axis, are fixed. A rotation of 720 degrees about the 3-axis is prescribed at B over three steps.

An EULER connector is constructed connecting A, the reference point of `SLOT1`, to the ground. The connector damping and connector friction behaviors are used to introduce damping and friction in the EULER connector. Damping and friction will prevent rigid body motion of `SLOT1` and `SLOT2` after the pin has left the slot.

Results and discussion

The protruding portion of `PIN` enters the slots in `SLOT1` as `PIN` is rotated. This interaction between `PIN` and `SLOT1` results in the rotation of `SLOT1` and `SLOT2`, and they rotate 90 degrees for every complete 360 degree rotation of `PIN`. *Figure 2* shows the configuration of the bodies of the Geneva mechanism at some intermediate instants during the analysis. *Figure 3* shows a plot of the U3 rotation of `SLOT1` at A as a function of the U3 rotation of `PIN` at B.

Input files

geneva_model.py	Python replay file for constructing the Geneva mechanism model in Abaqus/CAE.
geneva.inp	Geneva mechanism model.

Figures

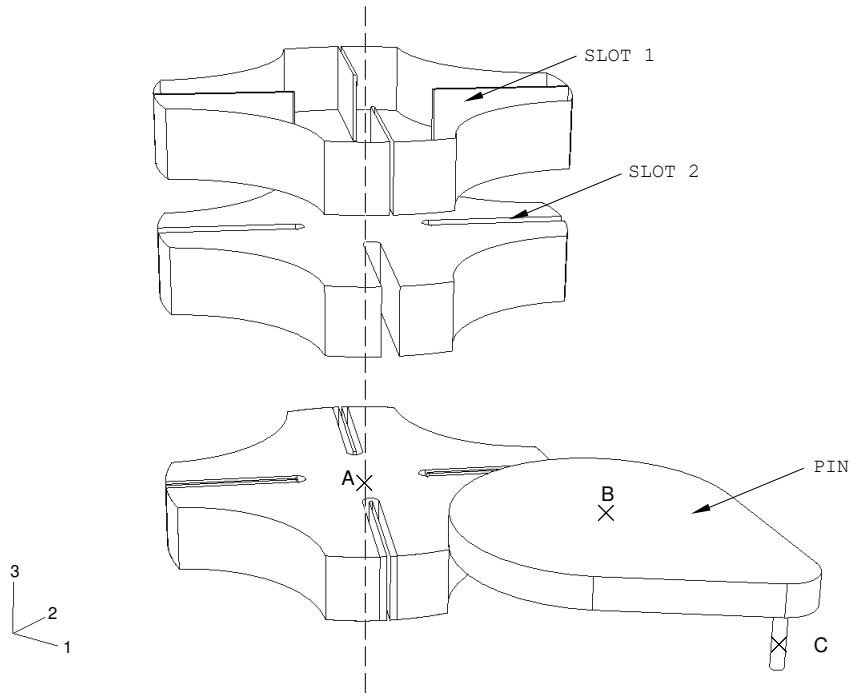


Figure 1: Exploded view of the Geneva mechanism.

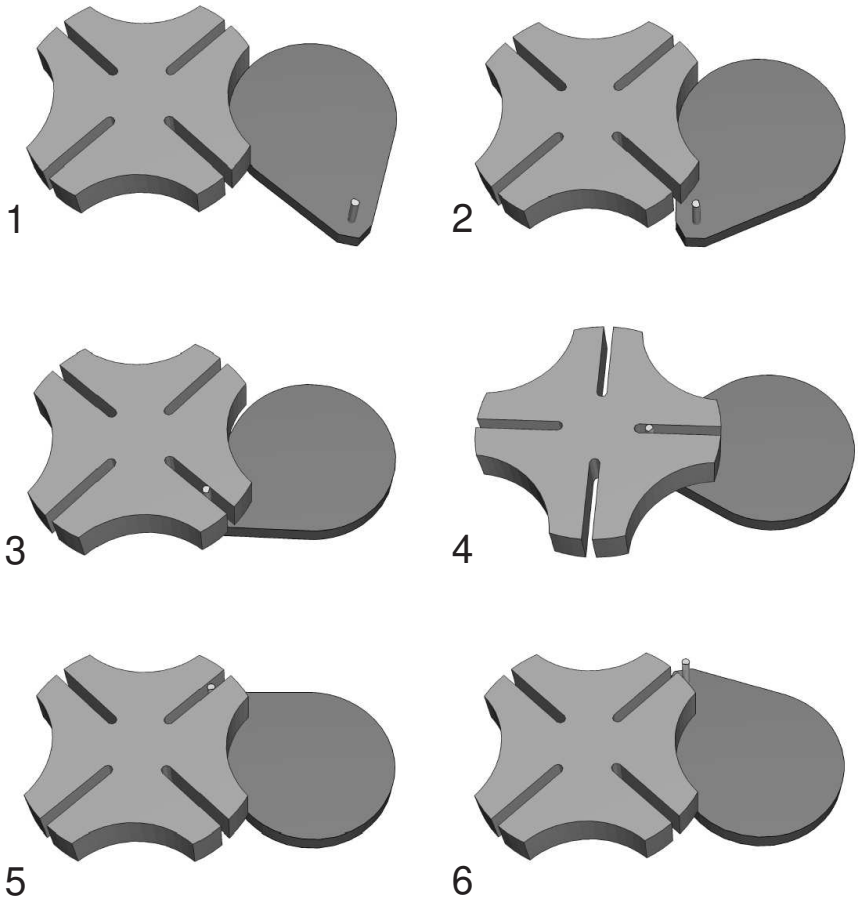


Figure 2: Configurations of the Geneva mechanism for increasing values of `PIN` rotation.

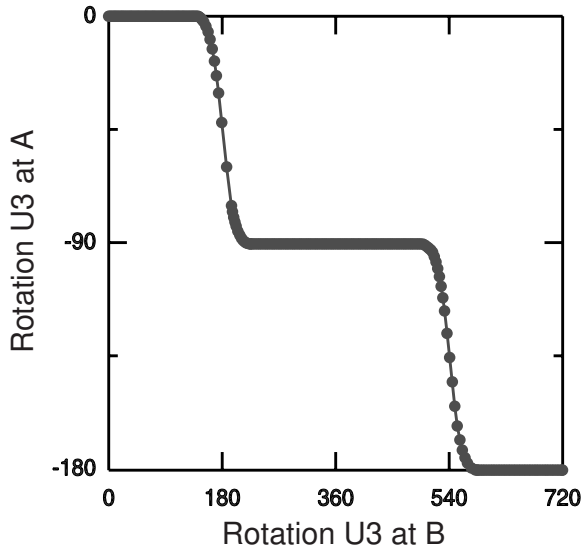


Figure 3: Plot of the U3 rotation of `SLOT1` at A as a function of the U3 rotation of `PIN` at B. Rotation values are in degrees.

Trailing edge flap mechanism

Products: Abaqus/Standard

Overview

This example illustrates the use of connectors to model the deployment of a trailing edge flap mechanism in an aircraft.

Geometry and model

The trailing edge flap mechanism is a critical component of an aircraft. The shape and positioning of the trailing edge flaps are important determinants of the aircraft's lift and aerodynamic behavior.

The trailing edge flap structure in this example problem consists of three flaps, `BASE`, `MIDFLAP`, and `ENDFLAP`; and these flaps are connected to each other by arms, which are rigid links pinned to the flaps at different points. The configuration of the flaps is illustrated in *Figure 1*. There are nine arms used in the model: `ARM AB`, `ARM BF`, `ARM DEFG`, `ARM EI`, `ARM HIM`, `ARM JK`, `ARM MLJ`, `ARM NK`, and `ARM PO`. `BASE`, `MIDFLAP`, and `ENDFLAP` are modeled as display bodies; and the arms are modeled as rigid truss members. `MIDFLAP` and `ENDFLAP` are deployed by rotating `ARM AB` pinned on `BASE` at point A.

Model interactions

The bodies in *Figure 1* are connected as follows:

- JOIN connector elements are used to connect the arms at their endpoints to `BASE`, `MIDFLAP`, and `ENDFLAP`. The endpoints of some arms are connected to other arms instead of being connected to the flaps. `ARM AB` is connected to `ARM BF` at B. `ARM BF` and `ARM EI` are connected to `ARM DEFG` at F and E, respectively. `ARM EI` and `ARM MLJ` are connected to `ARM HIM` at I and M, respectively. `ARM MLJ` is connected to `ARM JK` at J, and `ARM MLJ` is connected to `MIDFLAP` at L, which is not one of the endpoints of `ARM MLJ`.
- LINK connector elements are used to rigidly fix the positions of the pivot points on the flaps. These link connectors are called `LINK GL`, `LINK GN`, `LINK GP`, `LINK KO`, `LINK LN`, and `LINK LP`. These additional link connectors are necessary to support `BASE`, `MIDFLAP`, and `ENDFLAP` because these flaps are modeled as display bodies and not as deformable or rigid bodies. `LINK GL`, `LINK GN`, `LINK GP`, `LINK LN`, and `LINK LP` are used to position points G, L, N, and P on `MIDFLAP`; and `LINK KO` is used to position points K and O on `ENDFLAP`.

To deploy the flap, we fix `BASE` in space by fixing points A, D, and H and rotate `ARM AB` 90 degrees about point A. One analysis models plasticity effects in some of the connectors.

Results and discussion

The rotation of `ARM AB` results in the deployment of `MIDFLAP` and `ENDFLAP`. *Figure 2* contains a series of illustrations showing the positions of the flaps at the beginning and end of the analysis and also at some intermediate instants.

Input files

edgeflap_model.py	Python replay file for constructing the trailing edge flap mechanism model in Abaqus/CAE.

edgeflap.inp	Trailing edge flap mechanism model.
edgeflap_plas.inp	Trailing edge flap mechanism model with connector plasticity.

Figures

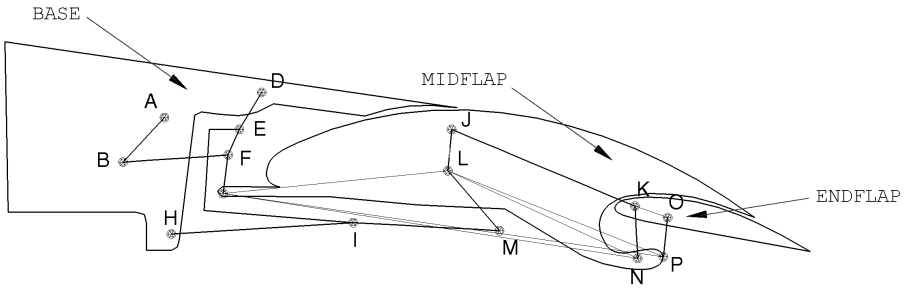


Figure 1: Undeformed configuration of the trailing edge flap mechanism.

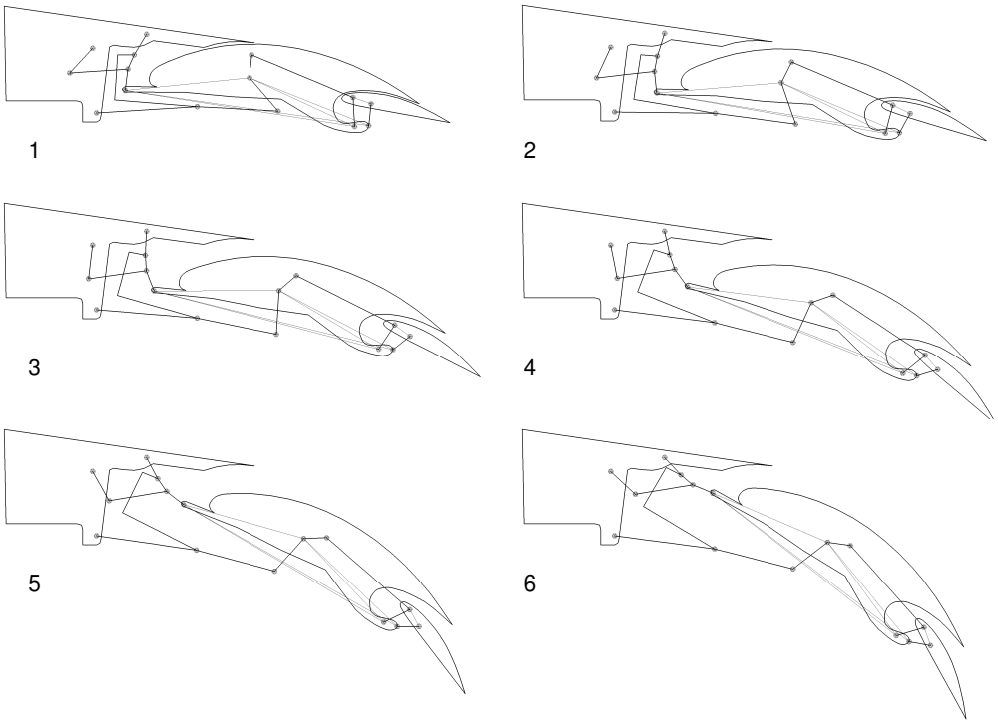


Figure 2: Deformed configurations of the trailing edge flap mechanism.

Substructure analysis of a one-piston engine model

Products: Abaqus/Standard

Overview

This example illustrates the use of the substructure capability in Abaqus to model efficiently multi-body systems that undergo large motions but exhibit only small linear deformations.
The example illustrates how to switch between a full-mesh representation of a part, a substructure representation, and a rigid body representation of the same part depending on the modeling needs. Both static and dynamic analyses are performed, and the results are compared between the various models.

Geometry and material

The multi-body system discussed here is the simplified one-piston engine model depicted in *Figure 1* and in the detail shown in *Figure 2*. The model consists of several parts that are constrained together using connector elements. Each part is modeled either as a fully deformable regular mesh, a rigid body, or a substructure. The connections between parts are idealized using coupling constraints and connector elements. On each part a coupling constraint is used to constrain a few nodes near the connection points so that the connection forces get distributed to a finite area, much like in the physical model. The reference nodes in the coupling constraint are then used to define appropriate connector elements between neighboring parts that enforce the desired kinematic linkage between bodies.

The two parts of the crankshaft are rigidly connected using a BEAM connector element. The connecting rod between the crankshaft and the piston's head has HINGE connections at both ends. The gear transmission (2:1 gear ratio) between the crankshaft and the valve-cam is modeled using an equation constraint between two rotational degrees of freedom. An analytically defined rigid surface is used to model the contact surface of the cam. This surface is rigidly attached to the larger gear used to spin the cam. The contact between the cam followers and the cam is modeled using a single node on each cam follower's tip defined using a node-based surface. The connections between cam followers and push rods and between push rods and valve rockers are modeled using JOIN connectors. The valve rockers use HINGE connections about fixed points in space (the engine block is not modeled). The valve rockers are pushing against the spring loaded valves. The valve springs are precompressed and are defined using CARTESIAN connectors together with the reference lengths and angles for constitutive response and the connector elasticity behavior. The valve seats and the cylinders are not modeled. Steel material properties are used for all parts. The model is actuated by fixing the available components of relative motion to spin the right end of the crankshaft for a little more than two full revolutions. In some of the analyses the connector friction behavior is used to model frictional effects in the connector elements. Plasticity effects are specified in connectors in some of the analyses.

Models

The following three models are considered:

1. All parts are modeled using regular meshes (no substructures or rigid bodies). This is the largest model and could include the most modeling options, including nonlinear material modeling options (e.g., plasticity), if desired. A large computation time is needed to complete the analysis.
2. All parts are modeled as rigid bodies. While much less computational time is needed to complete this analysis, this model cannot compute stresses in the various parts. However, it reveals very useful information about the overall kinematics/dynamics of the model. Reaction forces at support points and connector output variables (such as connector reaction forces or elastic forces in the valve springs) are also predicted accurately.

3. Some parts (crankshaft, connection rod, piston head, cam, and cam followers) are modeled as substructures. The remainder of the parts (push rods, valve rockers, and valves) are modeled as rigid bodies. This model is very efficient in obtaining both overall kinematic/dynamic information as well as the stresses and strains in the selected parts.

The input files are organized so that one can very easily switch (just by changing a few lines) between the different models.

Before the global analysis can be run, the substructure for each part must be generated. To minimize the computational effort in the global analysis, the minimum necessary number of retained nodal degrees of freedom is chosen. The reference nodes of all coupling constraints must be retained since the parts are connected through these nodes. In addition, since the parts will undergo large rotations during the analysis, at least three nonaligned nodes must be retained at the substructure generation level to allow for the precise calculation of the rigid body motion of each part (see *Generating Substructures*). Since for most parts only two nodes in the coupling constraint are defined (most parts are connected at only two points), an additional node not aligned with the nodes in the coupling constraint is also retained for most substructures. Because of the small number of retained nodes the global analysis is very efficient. For these particular choices it runs faster than the model when all parts are modeled as rigid.

The reduced mass matrix for each substructure is generated by selecting the eigenmodes during the substructure generation. To improve the representation of the substructure's dynamic behavior in the global analysis, six dynamic modes, which are extracted using an eigenfrequency extraction preload step, are included during the substructure generation during eigenmode selection. The substructures are loaded by gravity in the vertical direction by requesting that the substructure's gravity load vectors be calculated during penetration and by specifying a distributed load at the substructure usage level.

Results and discussion

Both static and direct-integration implicit dynamic analyses are conducted. The vertical displacements of the valves and the contact forces on the cam followers are monitored.

In *Figure 3* the vertical displacement of the valve on the left is depicted for a static analysis. The valve descends (“opens”) in the beginning of the cycle and then “closes” for most of the analysis. Since the crankshaft rotates a little more than two revolutions, the valve starts opening again in the end. The normal contact force between the left cam follower and the cam is shown in *Figure 4*. A sharp peak is encountered when the valve “opens,” while the contact force stays constant when the valve is “closed.” Since the valve seats are not modeled, the “closed” positions for the rigid body model and for the substructure model are slightly different due to elastic deformation of the cam followers in the latter case. The stresses in the left cam follower are depicted in *Figure 5* for an intermediate position of the crankshaft. The stresses are recovered when output is obtained from the substructure.

Since the inertia forces of the moving parts are relatively small when compared to the compression forces in the valve springs, the contact forces and the valve displacements in the direct-integration implicit dynamic analyses are similar to those in the static analysis. However, if the valve spring stiffness is chosen to be very small (such as would happen if the spring failed), the inertial effects become very important and contact between the cam and the cam follower could be lost. In *Figure 6* the valve displacement is depicted for such a case. Contact is established only intermittently between the two parts (corresponding to zero valve displacement). For most of the time the two parts are not contacting each other, and the valve displacement is completely different from the intended behavior.

The **abaqus substructurecombine** execution procedure combines model and results data from two substructure output databases into a single output database. For more information, see *Combining Output from Substructures*.

Input files

crank_substr_sta.inp	**STATIC* substructure global analysis.

crank_substr_dyn.inp	**DYNAMIC* substructure global analysis.
crank_substr_sta_fric.inp	**STATIC* substructure global analysis with frictional effects.
crank_substr_dyn_fric.inp	**DYNAMIC* substructure global analysis with frictional effects.
crank_substr_dynimp.inp	**DYNAMIC* substructure impact analysis (softer valve springs).
crank_rb_sta.inp	**STATIC* analysis with rigid bodies only (no substructures).
crank_all_def_sta.inp	**STATIC* regular mesh analysis.
crank_all_def_dyn.inp	**DYNAMIC* regular mesh analysis.
crank_shaft_gen.inp	Substructure generation for the right half of the shaft.
crank_shaft2_gen.inp	Substructure generation for the left half of the shaft.
crank_conrod_gen.inp	Substructure generation for the connection rod.
crank_head_gen.inp	Substructure generation for the piston's head.
crank_cam_gen.inp	Substructure generation for the cam.
crank_camfollow1_gen.inp	Substructure generation for the first cam follower.
crank_camfollow2_gen.inp	Substructure generation for the second cam follower.
crank_conn_gr_ref_nodes.inp	Connector, ground, and reference node definitions.
crank_small_mass.inp	Small masses used to avoid numerical singularities in **DYNAMIC* analyses.
crank_materials.inp	Material properties.
crank_shaft_nodes.inp	Node definitions for the right half of the shaft.
crank_shaft_elts.inp	Element definitions for the right half of the shaft.
crank_shaft_coupling.inp	**COUPLING* definitions for the right half of the shaft.
crank_shaft2_nodes.inp	Node definitions for the left half of the shaft.
crank_shaft2_elts.inp	Element definitions for the left half of the shaft.
crank_shaft2_coupling.inp	**COUPLING* definitions for the left half of the shaft.
crank_conrod_nodes.inp	Connecting rod node definitions.
crank_conrod_elts.inp	Connecting rod element definitions.
crank_conrod_coupling.inp	Connecting rod **COUPLING* definitions.
crank_head_nodes.inp	Piston head node definitions.
crank_head_elts.inp	Piston head element definitions.
crank_head_coupling.inp	Piston head **COUPLING* definitions.
crank_cam_nodes.inp	Cam node definitions.
crank_cam_elts.inp	Cam element definitions.
crank_cam_coupling.inp	Cam **COUPLING* definitions.
crank_camfollow1_nodes.inp	Node definitions for the first cam follower.
crank_camfollow1_elts.inp	Element definitions for the first cam follower.
crank_camfollow1_coupling.inp	**COUPLING* definitions for the first cam follower.
crank_camfollow2_nodes.inp	Node definitions for the second cam follower.
crank_camfollow2_elts.inp	Element definitions for the second cam follower.
crank_camfollow2_coupling.inp	**COUPLING* definitions for the second cam follower.
crank_valves_nodes.inp	Node definitions for the crank valves.
crank_valves_elts.inp	Element definitions for the crank valves.
crank_valvelifters_nodes.inp	Node definitions for the crank valve lifters.
crank_valvelifters_elts.inp	Element definitions for the crank valve lifters.

crank_tierods_nodes.inp	Node definitions for the tie rods.
crank_tierods_elts.inp	Element definitions for the tie rods.

Figures

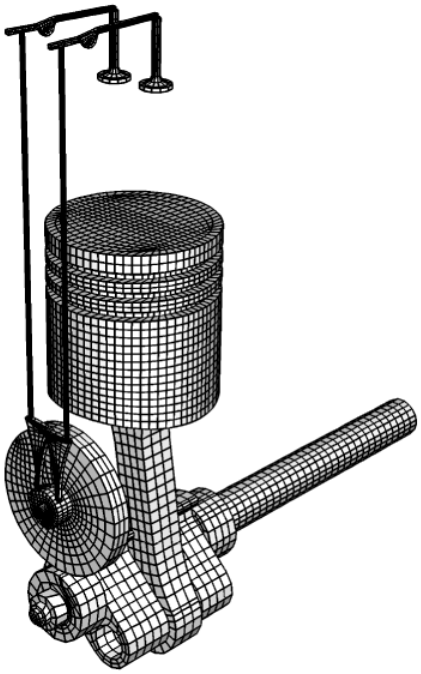


Figure 1: Mesh used for the complete one-piston model.

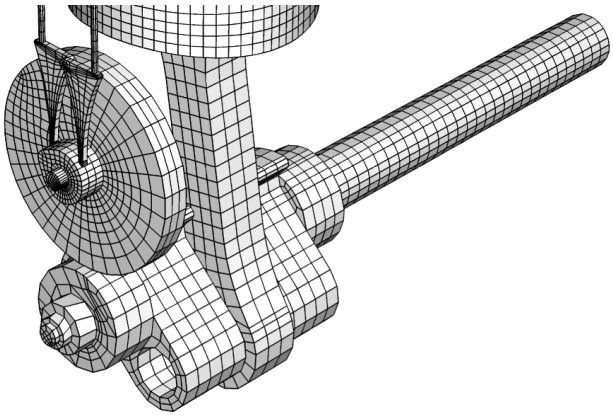


Figure 2: Detail of the mesh used for the one-piston model.

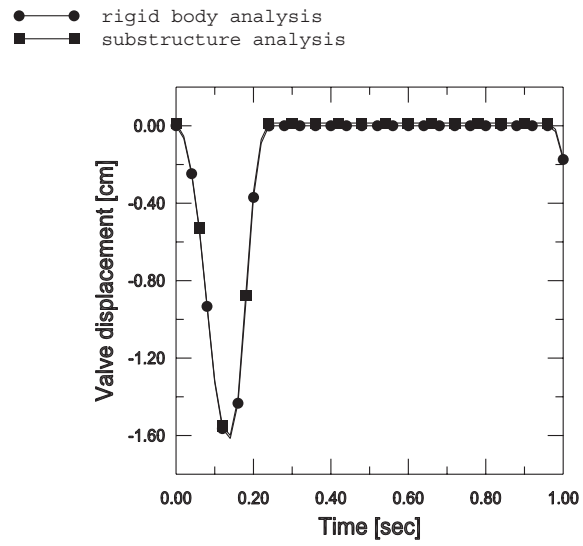


Figure 3: Valve displacement in static analyses.

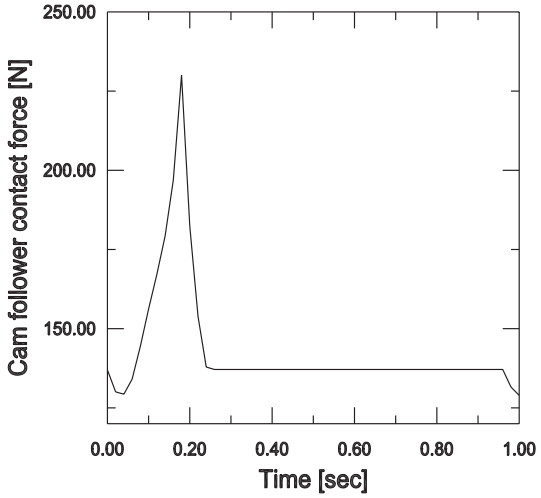


Figure 4: Cam follower contact force in a substructure static analysis.

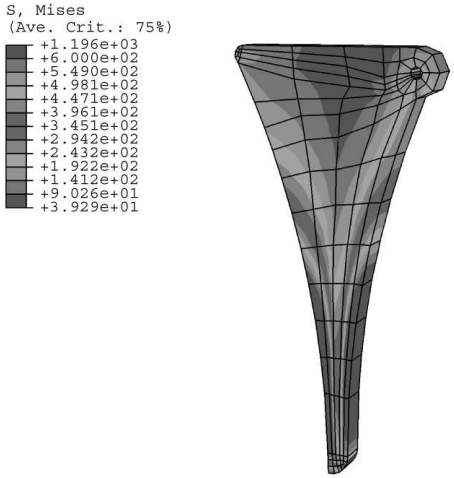


Figure 5: Cam follower stresses in a substructure static analysis.

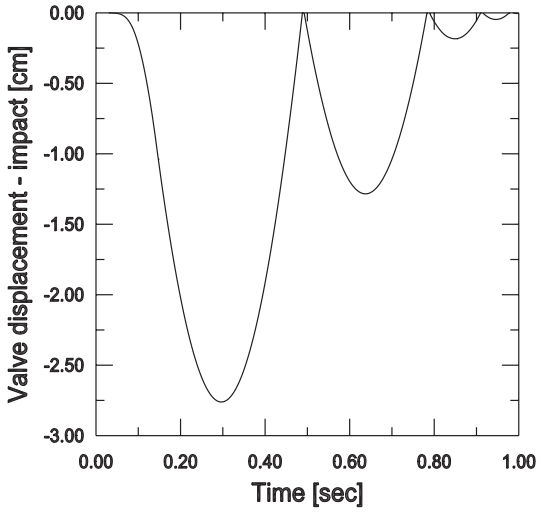


Figure 6: Valve displacement in a substructure direct-integration implicit dynamic analysis with very soft valve spring stiffness.

Application of bushing connectors in the analysis of a three-point linkage

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example illustrates the use of detailed finite element bushing models to calibrate the constitutive behavior of bushings (when experimental data for a particular bushing design are not available) so that a very inexpensive 2-node connector element representation of the bushing can be used in subsequent analyses involving the bushing.
The approach shown in this example is effective in reducing computational costs in assembly models while accurately capturing the macroscopic response of the bushing. To demonstrate this approach, a three-point linkage is simulated with connector elements used to model bushings.

Geometry and materials

The geometry of the three-point linkage (TPL) is shown in *Figure 1*. Since the main focus of this example is to illustrate modeling of the bushings, a simplified representation of the TPL is used. The TPL is connected to the subframe (assumed fixed in space) via the front and rear bushings. A ball joint connects the TPL to a wheel assembly (not shown). The TPL is a steel shell structure and is modeled as a linear elastic material with Young's modulus of 2.1×10^5 MPa, Poisson's ratio of 0.3, and a density of 7.82×10^{-9} tonnes/mm^3.

As shown in *Figure 2*, the rear bushing is a hollow cylinder with a rubber portion enclosed between two (inner and outer) thin steel tubes. The rubber portion has two symmetrically placed cutouts of different sizes, and it is rigidly bonded to the steel tubes. The steel tubes are 2 mm thick and use the same material properties as the TPL. The inner diameter of this bushing is 28 mm, the outer diameter is 72 mm, and the axial length is 20 mm. The outer steel tube is connected to the TPL using a kinematic coupling. The inner tube is connected to a fixed node that represents a subframe using a distributed coupling. The rubber is modeled as a second-order Ogden hyperelastic material (*Hyperelastic Behavior of Rubberlike Materials*), which may undergo nonlinear, finite deformation. The material parameters are $\mu_1 = 1.671$, $\alpha_1 = 9.0067$, $\mu_2 = 2.154\text{E}{-4}$, $\alpha_2 = -4.86970$, $D_1 = 1.0$, and $D_2 = 1.0$. The material density is 1.5×10^{-9} tonnes/mm^3.

The geometry of the front bushing is shown in *Figure 3*; it consists of three noncoaxial steel tubes with two rubber parts between them. The axial length of the bushing is 40 mm. The outer diameters of the three tubes are 40 mm, 28 mm, and 16 mm. All three tubes have a thickness of 1 mm. The outer steel tube is connected to the TPL using a kinematic coupling. The inner tube is connected to a fixed node that represents a U-shaped bracket. The front bushing uses the same rubber and steel materials as those used for the rear bushing.

Models

To characterize the nonlinear constitutive behavior of the bushings, a series of static analyses of the front bushing are performed in Abaqus/Standard and a series of quasi-static analyses of the rear bushing are performed in Abaqus/Explicit. Self-contact occurs in the rear bushing analyses due to severe deformation. In all cases the reference node of the kinematic coupling connected to the outer steel tube is held fixed in all degrees of freedom, while the second reference node connected to the inner tube moves. For the calibration analyses, a BUSHING connector for which no constitutive behavior is defined is placed between the two reference nodes. The use of a BUSHING connector to drive the deformation in these models is desirable since this choice ensures appropriate kinematic and kinetic modeling when the connector is used in other models (such as a full-system analysis). Since the main interest is capturing the hyperelastic response of the rubber bushings, both unidirectional and

coupled multidirectional tests (simultaneous deformations in up to three directions) are performed as summarized below.

To analyze several loading conditions, some connector components of relative motion can be fixed and others can be prescribed nonzero motion. The reaction forces and relative motions in the connector are saved as history output. The data from these analyses are then used in a subsequent analysis of the three-point linkage assembly to define nonlinear connector elasticity data.

Front bushing calibration tests

For the front bushing, static analyses are performed in Abaqus/Standard as follows:

Test 1_FB:

A translational motion is applied in the local 1-direction with all other relative motions fixed. The motion is applied in the positive and negative directions separately since the front bushing is not symmetric with respect to the local 1-axis. The magnitude of this motion is 5 mm. The CRF1 vs. CU1 data generated from this analysis are used to define nonlinear elasticity data for component 1.

Test 2_FB:

A translational motion is applied in the local 2-direction with all other relative motions fixed. The magnitude of this motion is 5 mm. The CRF2 vs. CU2 data generated from this analysis are used to define nonlinear elasticity data for component 2.

Test 3_FB:

A translational motion is applied in the local 3-direction with all other relative motions fixed. The magnitude of this motion is 5 mm. The CRF3 vs. CU3 data generated from this analysis are used to define nonlinear elasticity data for component 3.

Test 4_FB:

A 5° “bending” rotational motion about the local 1-direction is applied with all other relative motions fixed. The CRM1 vs. CUR1 data generated from this analysis are used to define nonlinear elasticity data for component 4.

Test 5_FB:

A 5° “bending” rotational motion about the local 2-direction is applied with all other relative motions fixed. The CRM2 vs. CUR2 data generated from this analysis are used to define nonlinear elasticity data for component 5.

Test 6_FB:

A 15° “twisting” rotational motion about the local 3-direction is applied with all other relative motions fixed. The CRM3 vs. CUR3 data generated from this analysis are used to define nonlinear elasticity data for component 6.

Test 7_FB:

A 5 mm displacement along the local 3-direction is applied about three preloaded configurations. The test attempts to capture coupling effects in the bushing after complex deformation is achieved. The preloaded configurations are:

1. A twisting of 0º about the local 3-direction.
2. A twisting of 7.5º about the local 3-direction.
3. A twisting of 15º about the local 3-direction.

This test is a collection of three two-step analyses. In each of the analyses a preload motion is applied in the first step, and the bending motion is applied in the second step. Assuming a hyperelastic-like quasi-static response in the bushing, for a given final coupled deformation, the deformation path is not relevant. Hence, the prestress and the actual loading steps can be run sequentially. The sets of CRF3 vs. CU3 data generated from the second steps of these analyses are used to define elasticity data for component 3 using independent component 6.

Test 8_FB:

A 5° bending motion about the local 2-direction is applied about three preloaded configurations. The test attempts to capture coupling effects in the bushing after complex deformation is achieved. The preloaded configurations are:

1. A twisting of 0º about the local 3-direction.
2. A twisting of 7.5º about the local 3-direction.
3. A twisting of 15º about the local 3-direction.

This test is a collection of three two-step analyses. In each of the analyses a preload motion is applied in the first step, and the bending motion is applied in the second step. Assuming a hyperelastic-like quasi-static response in the bushing, the deformation path is not relevant for a given final coupled deformation. Hence, the prestress and the actual loading steps can be run sequentially. The sets of CRM2 vs. CUR2 data generated from the second steps of these analyses are used to define elasticity data for component 5 using independent component 6.

Force vs. displacement and moment vs. rotation curves are shown in *Figure 4* for the cases without preload (Tests 1_FB to 6_FB) and in *Figure 5* for the cases with preload (Tests 7_FB and 8_FB).

Script files were generated to automatically create the input files and run the analyses for the uncoupled tests and each of the coupled tests. These scripts essentially build up a CROSS design parametric study for the uncoupled tests and a MESH design for the coupled tests (see *Scripting Parametric Studies*).

Rear bushing calibration tests

For the rear bushing, quasi-static analyses are performed in Abaqus/Explicit as follows:

Test 1_RB:

A translational motion is applied in the local 1-direction with all other relative motions fixed. The magnitude of this motion is 10 mm. The CRF1 vs. CU1 data generated from this analysis are used to define nonlinear elasticity data for component 1.

Test 2_RB:

A translational motion is applied in the local 2-direction with all other relative motions fixed. The magnitude of this motion is 5 mm. The CRF2 vs. CU2 data generated from this analysis are used to define nonlinear elasticity data for component 2.

Test 3_RB:

A translational motion is applied in the local 3-direction with all other relative motions fixed. The magnitude of this motion is 5 mm. The CRF3 vs. CU3 data generated from this analysis are used to define nonlinear elasticity data for component 3.

Test 4_RB:

A 5° "bending" rotational motion about the local 1-direction is applied with all other relative motions fixed. The CRM1 vs. CUR1 data generated from this analysis are used to define nonlinear elasticity data for component 4.

Test 5_RB:

A 15° "bending" rotational motion about the local 2-direction is applied with all other relative motions fixed. The CRM2 vs. CUR2 data generated from this analysis are used to define nonlinear elasticity data for component 5.

Test 6_RB:

A 5° "twisting" rotational motion about the local 3-direction is applied with all other relative motions fixed. The CRM3 vs. CUR3 data generated from this analysis are used to define nonlinear elasticity data for component 6.

Test 7_RB:

A 10 mm displacement in the local 1-direction is applied about several preloaded configurations. The displacement is applied in the positive and negative directions separately since the rear bushing is not symmetric with respect to the local 1-axis. The test attempts to capture coupling effects in the bushing after complex deformation is achieved. The preloaded configurations are a combination of a displacement in the local 2-direction and a rotation about the local 2-direction for which the following design points were selected:

1. A displacement of 0.0 mm in the local 2-direction.
2. A displacement of 2.5 mm in the local 2-direction.
3. A displacement of 5.0 mm in the local 2-direction.
4. A 0° "bending" rotational motion about the local 2-direction.
5. A 7.5° "bending" rotational motion about the local 2-direction.
6. A 15° "bending" rotational motion about the local 2-direction.

This test is a collection of 18 two-step analyses. In each of the analyses two preload motions are applied in the first step, and the translational motion in the local 1-direction is applied in the second step. Assuming a hyperelastic-like quasi-static response in the bushing, the deformation path is not relevant for a given final coupled deformation. Hence, the prestress and the actual loading steps can be run sequentially. The

sets of CRF1 vs. CU1 data generated from the second steps of these analyses are used to define elasticity data for component 1 using independent components 2 and 5.

Test 8_RB:

A 5 mm displacement in the local 2-direction is applied about several preloaded configurations. The test attempts to capture coupling effects in the bushing after complex deformation is achieved. The preloaded configurations are a combination of a displacement in the local 1-direction (positive and negative loadings are considered separately due to asymmetry) and a rotation about the local 2-direction, for which the following design points were selected:

1. A displacement of –10.0 mm in the local 1-direction.
2. A displacement of –5.0 mm in the local 1-direction.
3. A displacement of 0.0 mm in the local 1-direction.
4. A displacement of 5.0 mm in the local 1-direction.
5. A displacement of 10.0 mm in the local 1-direction.
6. A 0° “bending” rotational motion about the local 2-direction.
7. A 7.5° “bending” rotational motion about the local 2-direction.
8. A 15° “bending” rotational motion about the local 2-direction.

This test is a collection of 15 two-step analyses. In each of the analyses two preload motions are applied in the first step, and the translational motion in the local 2-direction is applied in the second step. Assuming a hyperelastic-like quasi-static response in the bushing, the deformation path is not relevant for a given final coupled deformation. Hence, the prestress and the actual loading steps can be run sequentially. The sets of CRF2 vs.CU2 data generated from the second steps of these analyses are used to define elasticity data for component 2 using independent components 1 and 5.

Force vs. displacement and moment vs. rotation curves are shown in *Figure 6* for the cases without preload (Tests 1_RB to 6_RB). *Figure 7* shows a comparison of the force vs. displacement with no preload and with the most extreme preload conditions for Tests 7_RB and 8_RB.

Script files were generated to automatically create the input files and run the analyses for the uncoupled tests and each of the coupled tests. As in the case of the front bushing, these scripts essentially build up a CROSS design parametric study for the uncoupled tests and a MESH design for the coupled tests. Furthermore, Python files were created to automatically gather the force vs. displacement data points and to create corresponding report files.

Additional comments on calibration tests

Since each bushing has several symmetry planes, the tests above (unless specified otherwise) are conducted only for positive relative motions when the responses in the opposite directions are symmetric. Therefore, nonlinear elasticity data are generated only for positive relative motions for those cases. Nonlinear elasticity is defined for negative relative motions by symmetrizing the elasticity data with respect to the origin.

The analyses chosen above are deemed appropriate to generate the necessary and complete nonlinear elasticity data to enable BUSHING connectors to represent the front and rear bushings in subsequent analyses of the TPL. The quasi-static analyses were run over sufficiently large motion ranges to cover the motion range expected in the analysis where the BUSHING connector is used. Nonlinear elasticity data are generated in all six relative uncoupled directions in the BUSHING connector. For the front bushing, elasticity data are generated (Tests 7_FB and 8_FB) for two coupled deformation modes (involving two components of local motion each) that were thought to be the dominant coupling modes in the subsequent analysis where the BUSHING connectors are used. For the rear bushing, elasticity data are generated (Tests 7_RB and 8_RB) for two coupled deformation modes (involving three components of local motion each) that were thought to be the dominant coupling modes

in the subsequent analysis where the BUSHING connectors are used. In general, the number and complexity of coupled deformation tests can be increased to match any particular modeling needs.

TPL models

Three different models are created for the analysis of the three-point linkage system: a fully meshed model, a bushing connector model, and a substructure model. They all model the TPL using the front and rear bushings to connect to the subframe (not modeled) that is assumed to be fixed in space. Loads are applied to a reference point where the ball joint is attached to model loads that the wheel assembly would exert on the TPL. The constraint imposed by the fixed subframe is modeled in all three cases by constraining the motion of the inner cylinder distributed coupling reference node in all six degrees of freedom.

In the fully meshed model the linkage itself is modeled using shell elements, while the front and rear bushings are modeled using continuum elements. The front bushing is connected to the reference node of a U-bracket that is modeled as a rigid body (see *Figure 1*). The rigid body reference node of the U-bracket is held fixed in all six degrees of freedom.

In the bushing connector model the bushings in the fully meshed model are replaced with BUSHING connectors. The constitutive data for the connectors are obtained from the series of tests described above. The front bushing is connected to the reference node of a U-bracket that is modeled as a rigid body (see *Figure 1*). The rigid body reference node of the U-bracket is held fixed in all six degrees of freedom.

In the substructure model the three-point linkage is modeled with a substructure, and the bushings are modeled with connector elements as defined in the bushing connector model. For clarity, the U-bracket is not represented in the substructure model since it is considered to be rigid and fixed. The rear bushing inner cylinder is connected to the subframe via a large bolt.

Two load cases are applied in all three models:

Load case 1:

A geometrically nonlinear, single-step, static analysis is performed where a concentrated force of 250 N is applied at the ball joint reference node in the negative global 1-direction. The analysis models a horizontal load at the ball joint.

Load case 2:

A geometrically nonlinear, two-step, static analysis is performed. In the first step the TPL is lifted by 10º about the global 1-direction by applying a displacement boundary condition of 20 mm to the ball joint reference node in the negative global 3-direction. In the second step a concentrated force of 250 N is applied at the ball joint reference node in the negative global 1-direction. The analysis models a horizontal load on the ball joint as the wheel and, hence, the ball joint go over a bump.

Results and discussion

It can be seen from the results of Tests 7_FB and 8_FB (*Figure 5*) that the front bushing behavior in both the local 3-direction and about the local 2-direction is affected significantly by the amount of preload about the local 3-direction. For the rear bushing it can be seen from the results of Tests 7_RB and 8_RB (*Figure 7*) that the behavior in the local 1-direction and especially the behavior in the local 2-direction are affected significantly by the amount of the combined preload (in the local 2-direction and about the local 2-direction for Test 7_RB, and in the local 1-direction and about the local 2-direction for Test 8_RB).

The Mises stresses in the TPL calculated for each of the three modeling approaches are very similar, as shown in *Figure 8*, *Figure 9*, and *Figure 10*. In addition, displacement histories at the ball joint (where the loading was applied) show very good agreement between the three models (*Figure 11*). As expected, the bushing connector

model analysis and the substructure model analysis produced identical displacements at the ball joint. When compared with the fully meshed model, the differences for the displacement histories in both directions are 4% toward the end of the curves.

The main motivation to use the BUSHING connector in this and similar applications is to reduce the complexity of the models and the computation time. In this TPL analysis the fully meshed model analysis takes approximately 50 times longer than the analysis that uses connectors to model the bushings. Furthermore, the substructure model analysis completes in about one-eighth of the time required for the analysis that uses connectors to model the bushings, approximately 400 times faster than the original fully meshed model.

In summary, this example demonstrates significant improvement in analysis efficiency when bushings are modeled with pre-calibrated BUSHING connectors or substructures without sacrificing accuracy.

Files

tpl_fb_uncoupled.inp	Front bushing model template used for generating constitutive data curves for all tests without preload.
tpl_fb_uncoupled.psf	Python script file to automatically generate input files and run analyses for all tests without preload.
tpl_fb_CRF3vsCU3.inp	Front bushing model template used for generating constitutive data curves for Test 7_FB.
tpl_fb_CRF3vsCU3.psf	Python script file to automatically generate input files and run analyses for Test 7_FB.
tpl_fb_CRM2vsCUR2.inp	Front bushing model template used for generating constitutive data curves for Test 8_FB.
tpl_fb_CRM2vsCUR2.psf	Python script file to automatically generate input files and run analyses for Test 8_FB.
tpl_rb_uncoupled.inp	Rear bushing model template used for generating constitutive data curves for all tests without preload.
tpl_rb_uncoupled.psf	Python script file to automatically generate input files and run analyses for tests without preload.
tpl_rb_CRF1vsCU1.inp	Rear bushing model template used for generating constitutive data curves for Test 7_RB.
tpl_rb_CRF1vsCU1.psf	Python script file to automatically generate input files and run analyses for Test 7_RB.
tpl_rb_CRF2vsCU2.inp	Rear bushing model template used for generating constitutive data curves for Test 8_RB.
tpl_rb_CRF2vsCU2.psf	Python script file to automatically generate input files and run analyses for Test 8_RB.
tpl_const_data_ext-rb_CRF1vsCU1.py	Python file to automatically gather force vs. displacement results for Test 7_RB.
tpl_const_data_ext-rb_CRF2vsCU2.py	Python file to automatically gather force vs. displacement results for Test 8_RB.
tpl_full_loadcase1.inp	Fully meshed model analysis for Load case 1.
tpl_full_loadcase2.inp	Fully meshed model analysis for Load case 2.
tpl_connectors_loadcase1.inp	Bushing connector model analysis for Load case 1.
tpl_connectors_loadcase2.inp	Bushing connector model analysis for Load case 2.
tpl_substructure_gen.inp	Substructure generation analysis for the three-point linkage.
tpl_substructure_loadcase1.inp	Substructure model analysis for Load case 1.

tpl_substructure_loadcase2.inp Substructure model analysis for Load case 2.

Figures

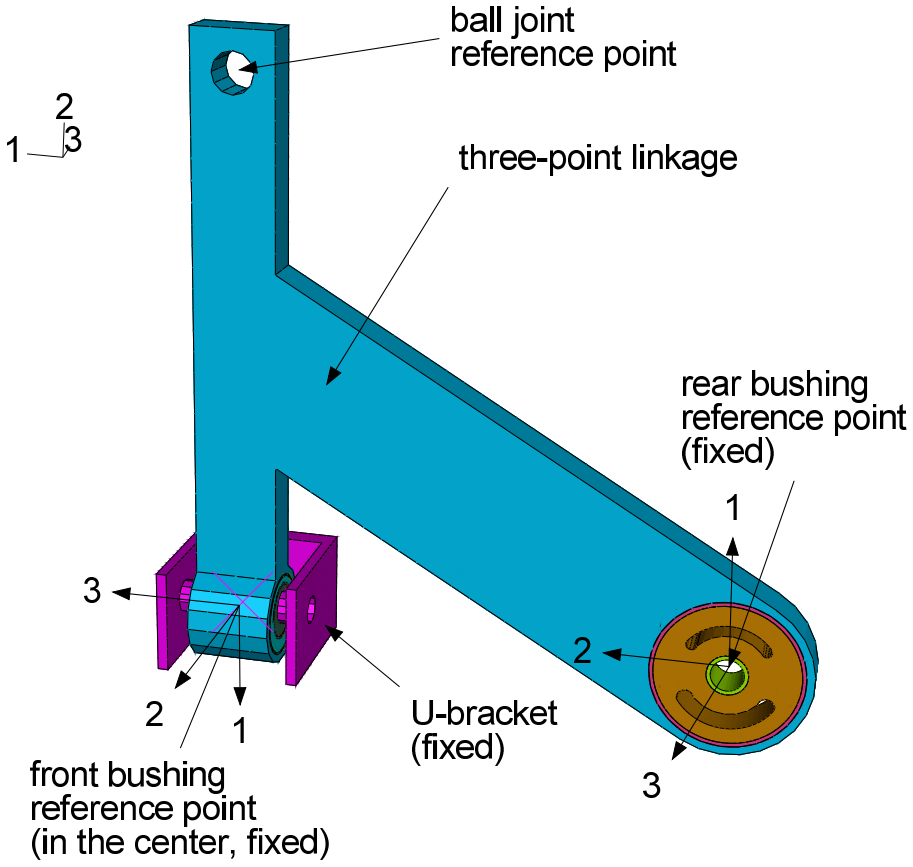


Figure 1: Three-point linkage (TPL) assembly.

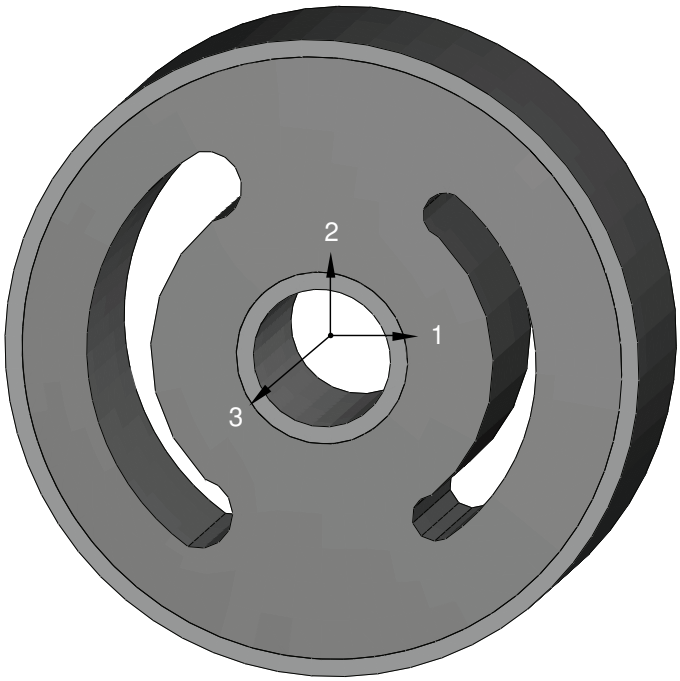


Figure 2: Rear bushing geometry.

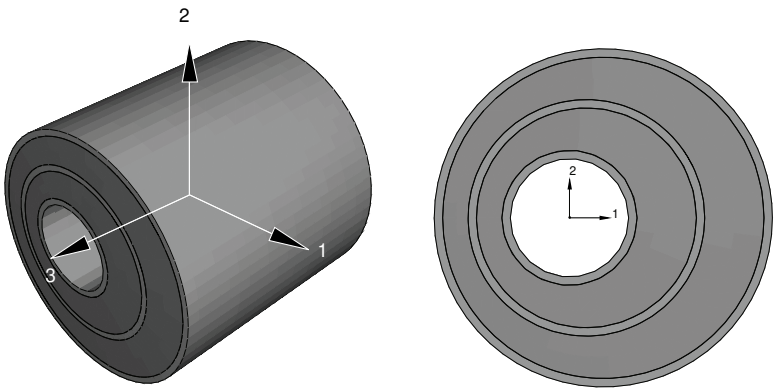


Figure 3: Front bushing geometry.

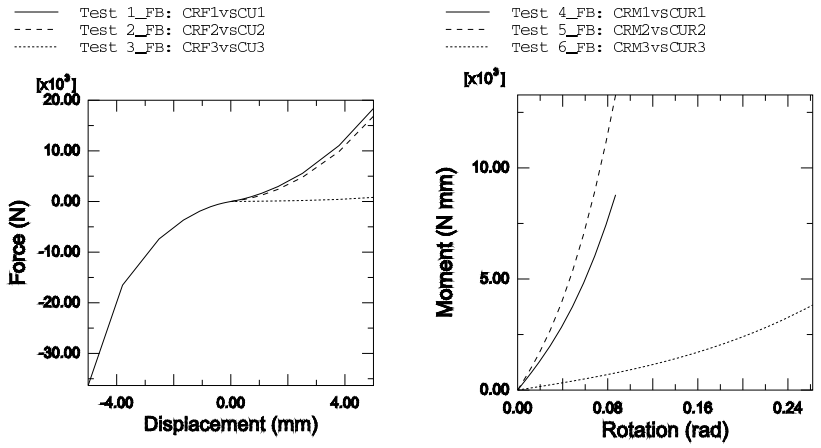


Figure 4: Front bushing coupled force vs. displacement and moment vs. rotation curves without preload.

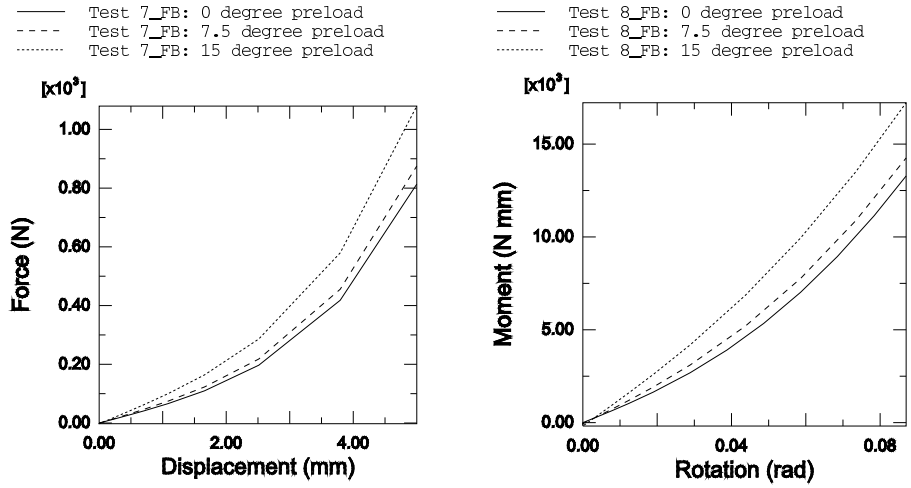


Figure 5: Front bushing coupled force vs. displacement and moment vs. rotation curves with preload.

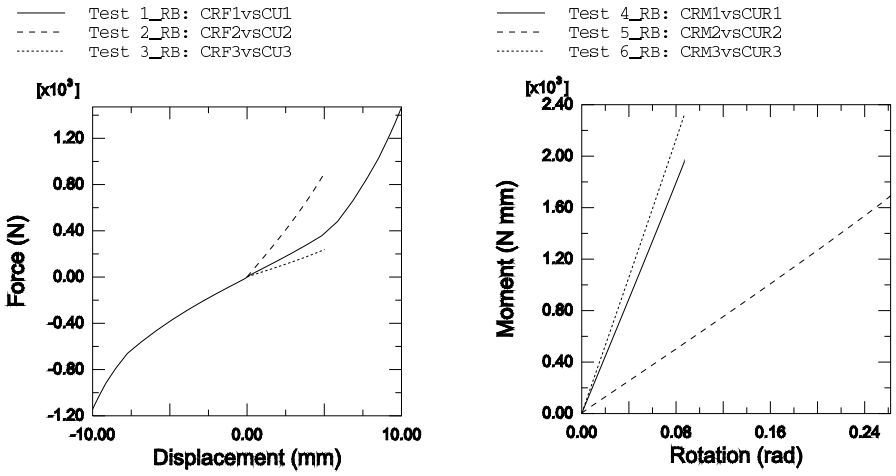


Figure 6: Rear bushing force vs. displacement and moment vs. rotation curves without preload.

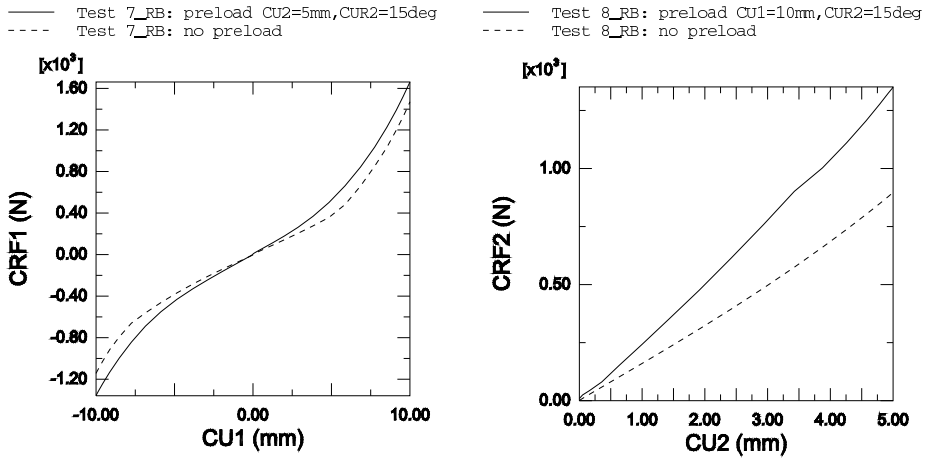


Figure 7: Comparison of rear bushing coupled force vs. displacement for Tests 7_RB and 8_RB.

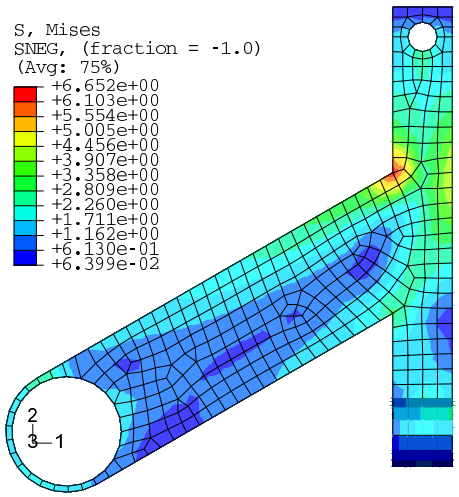


Figure 8: Mises stresses in the TPL for the fully meshed model (Load case 2).

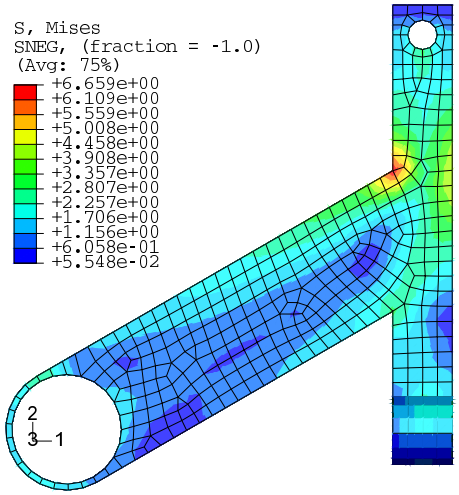


Figure 9: Mises stresses in the TPL for the bushing connector model (Load case 2).

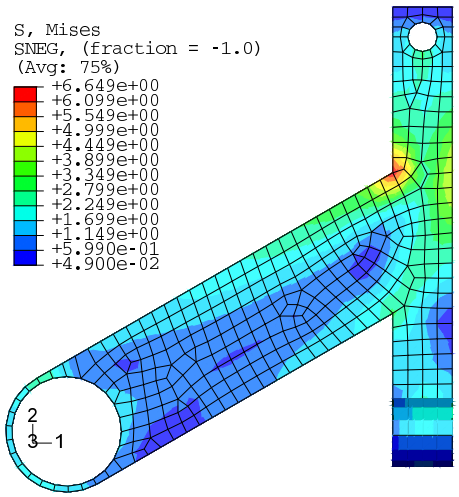


Figure 10: Mises stresses in the TPL for the substructure model (Load case 2).

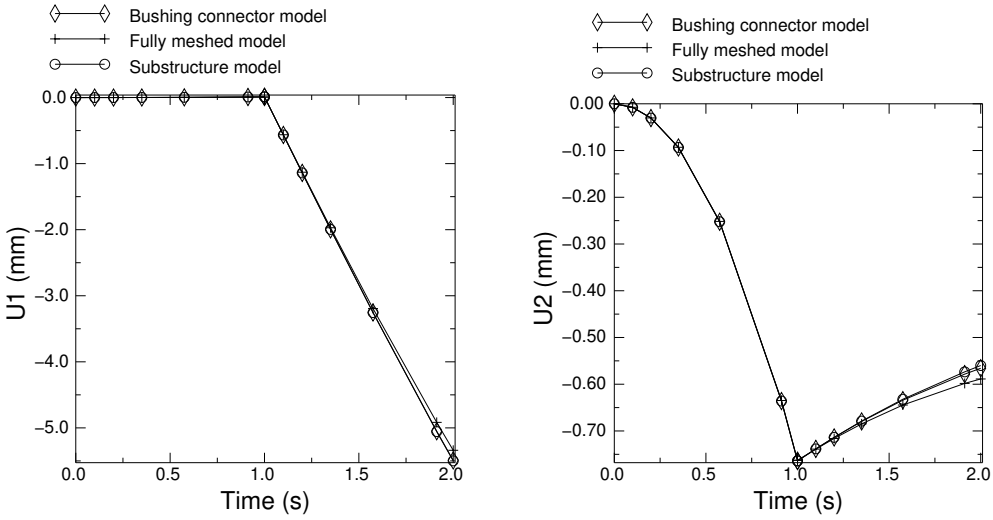


Figure 11: Comparison of displacements at the ball joint reference node (Load case 2).

Gear assemblies

Products: Abaqus/Explicit

Overview

This example illustrates the use of connectors in modeling the kinematics of gear assemblies.

Application description

Gears are widely used in a variety of machinery to transfer torque (rotational motion) from one shaft to another shaft and to act as a means to change the torque (rotational speed) acting on a shaft. Three types of gear assemblies (spur gear assembly, rack and pinion mechanism, and screw gear assembly) are studied in this example with a focus on the proper transmission of torque and rotational motion between the gears. The units used are millimeters-tonnes-seconds.

Geometry

Spur gears

A spur gear assembly consists of a driving gear (labeled `Pinion`) with input torque acting on it and a driven gear (labeled `Gear`) that receives the torque from the pinion, as shown in *Figure 1*. In this example the gear ratio is chosen as 2.5; i.e., the gear has 2.5 times more teeth than the pinion and, hence, it rotates 2.5 times slower than the pinion. To measure the power transmission in the gear system, a linear torsional spring is placed on each shaft such that the ratio of their elastic constants is the square of the ratio of the gear radius on the respective shaft. The whole gear assembly is mounted on a single rigid body whose reference node is constrained to prevent rigid body motion. The pitch circle radius of the pinion and the gear is 1.04573 units and 2.61433 units, respectively, as marked by the position of node 2 in *Figure 1*. The elastic constant of the torsional spring on the pinion shaft is chosen as 1 unit while that on the gear shaft, as calculated from the gear ratio, is 6.25 units.

Rack and pinion

A rack and pinion mechanism is a special case of the spur gear assembly with one of the gears having an infinite radius. The mechanism consists of a spinning pinion pulling the rack toward itself, as shown in *Figure 2*. A groove is provided to guide the movement of the rack. The whole rack and pinion assembly is mounted on a rigid body whose reference node is constrained to prevent any rigid body motion. The pitch circle radius of the pinion is 2.5 units.

Screw gear and housing

A screw gear mechanism is similar to a rack and pinion mechanism except that the translational motion takes place along the axis of rotation of the gear, as shown in *Figure 3*. As in the case of the rack and pinion arrangement, a slot guides the direction of the motion of the screw. The screw gear and housing assembly is mounted on a rigid body whose reference node is constrained to prevent any rigid body motion. The motion involves turning the screw one complete revolution as it runs along the grooves of the screw thread of a fixed housing. The screw has a radius of 1.5 units and a pitch of 2 units.

Abaqus modeling approaches and simulation techniques

This example illustrates the use of connectors for modeling gear assemblies in Abaqus/Explicit. The focus in this example is purely on the kinematics of gears without the consideration of gear-teeth forces.

Summary of analysis cases

Case 1: Spur gears	A rotating pinion driving a gear.
Case 2: Rack and pinion	A rotating pinion pulling a rack guided in a slot.
Case 3: Screw gear	A rotating screw moving through a groove in a housing.

Mesh design

All three cases of gear assemblies use CONN3D2 elements with the section behavior defined in detail in the following sections.

Case 1: Spur gears

This case illustrates using connectors to model the kinematics of a spur gear assembly.

Mesh design

Each spur gear is modeled using one CONN3D2 element with connection type HINGE and one CONN3D2 element with connection type FLOW-CONVERTER.

Materials

The HINGE of the pinion has a torsional stiffness of 1 unit, and the HINGE of the gear has a torsional stiffness of 6.25 units.

Boundary conditions

The first nodes of both HINGEs are fixed. The second node of the HINGE of the pinion is driven by 18.85 radians.

Interactions

As shown in *Figure 1*, the pinion and gear are each modeled with a HINGE and a FLOW-CONVERTER sharing a node (nodes 1 and 3, respectively). Nodes 1 and 3 of HINGEs `HG1` and `HG2`, respectively, are free to rotate, whereas their second nodes (node 4 and 6, respectively) form a rigid body along with node 5 (reference node). This rigid body has all of its degrees of freedom fixed to prevent overall rigid body motion of the gear system.

The second node of FLOW-CONVERTERs `FC1` and `FC2` is common (node 2) with all of its rotations fixed. Any relative rotation in HINGE`HG1` causes material flow at node 2 of FLOW-CONVERTER`FC1`. The gear mechanism formed by HINGE`HG2` and FLOW-CONVERTER`FC2` mirrors the pinion mechanism formed by FLOW-CONVERTER`FC1` and HINGE`HG1`. Thus, any material flow at common node 2 of FLOW-CONVERTER`FC2` results in a relative rotational motion between the two nodes of HINGE`HG2`.

The gear ratio of 2.5 is maintained by defining the scaling factor (with proper sign) in FLOW-CONVERTERs `FC1` and `FC2`; for this case, the scaling factors are 1.0 and −0.4, respectively. The display bodies of the gear and

pinion have reference nodes 10 and 11, respectively, that form a rigid body with the respective unconstrained nodes of HINGEs HG1 and HG2.

Case 2: Rack and pinion

This case illustrates using connectors to model the kinematics of a rack and pinion mechanism.

Mesh design

The pinion is modeled using one CONN3D2 element with connection type HINGE and one CONN3D2 element with connection type FLOW-CONVERTER. The rack is modeled using one CONN3D2 element with connection type SLIPRING. The groove in which the rack moves is modeled using one CONN3D2 element with connection type TRANSLATOR.

Materials

Both the HINGE and the SLIPRING are defined as rigid elastic.

Boundary conditions

The first nodes of the HINGE and TRANSLATOR are fixed. Degree of freedom 10 of the common node of the SLIPRING and TRANSLATOR is also fixed. The second node of the HINGE of the pinion is driven by 6.28 radians.

Interactions

The rack and pinion mechanism, as shown in *Figure 2*, is modeled similarly to the spur gears described in Case 1. The pinion is modeled with HINGEHG (nodes 1 and 6) and FLOW-CONVERTERFC (nodes 1 and 2) that share a common node (node 1). The rack is modeled with a SLIPRINGSR (nodes 2 and 3) and TRANSLATORTR (nodes 4 and 3) that is used to guide the motion of SLIPRINGSR. SLIPRINGSR and FLOW-CONVERTERFC share a common node (node 2).

Nodes 2, 4, and 6 are part of a rigid body with reference node 5 that is constrained to prevent any rigid body motion of the rack and pinion system. A fixed boundary condition on degree of freedom 10 is imposed at node 3 of SLIPRINGSR to prevent any material flow past this node. Any relative rotation between nodes 1 and 6 of HINGEHG results in material flow at node 2 of FLOW-CONVERTERFC. This material flow at node 2 results in tension in the adjoining SLIPRINGSR that drives the rack.

The relation between the translational motion of the rack and the rotational motion of the pinion is maintained by defining the scaling factor (with proper sign) of FLOW-CONVERTERFC; for this case, the scaling factor is 0.4. As mentioned before, the motion of the rack occurs parallel to the axis of the TRANSLATORTR, and it can be measured via the connector displacement of TRANSLATORTR. In this example available component 4 in HINGEHG and available component 1 in SLIPRINGSR are both defined as rigid elastic. The display bodies of the rack and pinion have reference nodes 10 and 11, respectively, that form a rigid body with the respective unconstrained nodes of SLIPRINGSR and HINGEHG.

Case 3: Screw gear

This case illustrates using connectors to model the kinematics of a screw gear and housing assembly.

Mesh design

The screw is modeled using one CONN3D2 element with connection type HINGE, one CONN3D2 element with connection type FLOW-CONVERTER, and one CONN3D2 element with connection type SLIPRING. The groove in which the screw moves is modeled using a CONN3D2 element with connection type TRANSLATOR.

Materials

Both the HINGE and the SLIPRING are defined as rigid elastic.

Boundary conditions

The first nodes of the TRANSLATOR are fixed. Degree of freedom 10 of the common node of the SLIPRING and TRANSLATOR is also fixed. The second node of the HINGE of the screw is driven by 20 radians.

Interactions

The screw gear system, as shown in *Figure 3*, is modeled similarly to the rack and pinion system described in Case 2. The screw is modeled with HINGE`HG` (nodes 6 and 1), FLOW-CONVERTER`FC` (nodes 1 and 2), and SLIPRING`SR` (nodes 2 and 3). The housing is modeled with TRANSLATOR`TR` (nodes 4 and 3). HINGE`HG` and FLOW-CONVERTER`FC` share a common node (node 1). SLIPRING`SR` and FLOW-CONVERTER`FC` share node 2.

Nodes 2, 4, and 11 are part of a rigid body with reference node 5 that is constrained to prevent any rigid body motion of the housing. Material flow at node 3 of the SLIPRING is prevented via a fixed boundary condition on degree of freedom 10. Similar to the rack and pinion mechanism, the relative rotation between nodes 1 and 6 of HINGE`HG` results in material flow at node 2 of FLOW-CONVERTER`FC`. This material flow at node 2 causes tension in the belt of SLIPRING`SR`, resulting in a translational displacement of node 3. Nodes 3 and 6 are tied together as a rigid body and, hence, the translational motion of node 3 results in the linear motion of the screw. The rotational motion of the screw is identical to the rotational motion of node 1.

The scaling factor of FLOW-CONVERTER`FC` is defined to scale the rotational motion of node 1 in relation to its translational motion; for this case, the scaling factor is −3.14. As in the case of the rack and pinion mechanism, TRANSLATOR`TR` guides the direction of the motion of the screw. Available component 1 in SLIPRING`SR` is defined as rigid elastic. The display bodies of the screw and the housing have reference nodes 10 and 11, respectively, that form a rigid body with the respective unconstrained nodes of HINGE`HG` and SLIPRING`SR`.

Discussion of results and comparison of cases

Case 1: Spur gears

The pinion shaft is given an angular displacement of 18.85 radians by defining connector motion in HINGE`HG1`. The final angular displacement of the gear shaft is measured to be 7.54 radians, with the gear ratio holding at 2.5 throughout the analysis duration, as shown in *Figure 4*. The ratio of the input to output power, which is calculated by taking the product of connector moment CTM1 and connector motion CUR1 of the respective HINGEs, is also found to be constant.

Case 2: Rack and pinion

The pinion is rotated one complete revolution by defining connector motion in HINGEHG. The rack moves 15.7 units, measured by CU1 of TRANSLATORTR in *Figure 5*. This distance is the same as the circumference of the pitch circle of the pinion.

Case 3: Screw gear

The screw is given an angular rotation of 20 radians by defining connector motion in HINGEHG. The screw moves 6.37 units, measured by CU1 of TRANSLATORTR as shown in *Figure 6*.

Input files

spur_gear.inp	Spur gear assembly.
rack_pinion.inp	Rack and pinion assembly.
screw_gear.inp	Screw gear assembly.

Figures

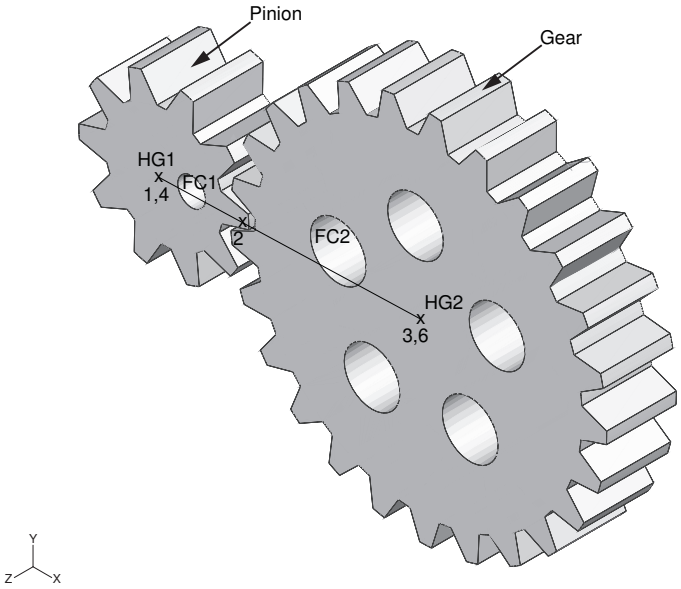


Figure 1: Spur gear assembly.

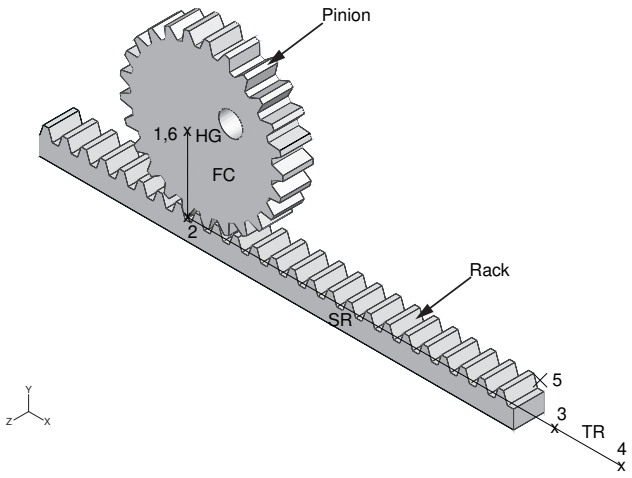


Figure 2: Rack and pinion assembly.

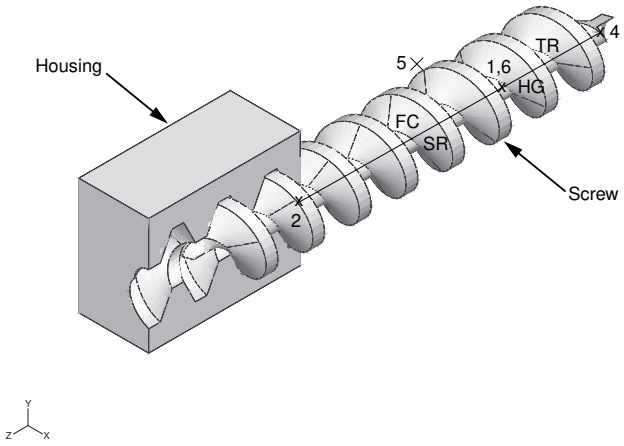


Figure 3: Screw gear assembly.

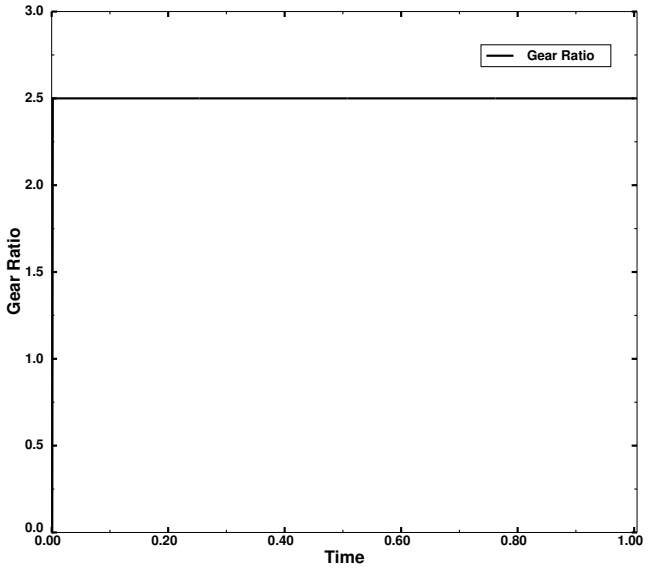


Figure 4: Ratio of angular displacement of the pinion (CUR1 of HINGE`HG1`) to that of the gear (CUR1 of HINGE`HG2`).

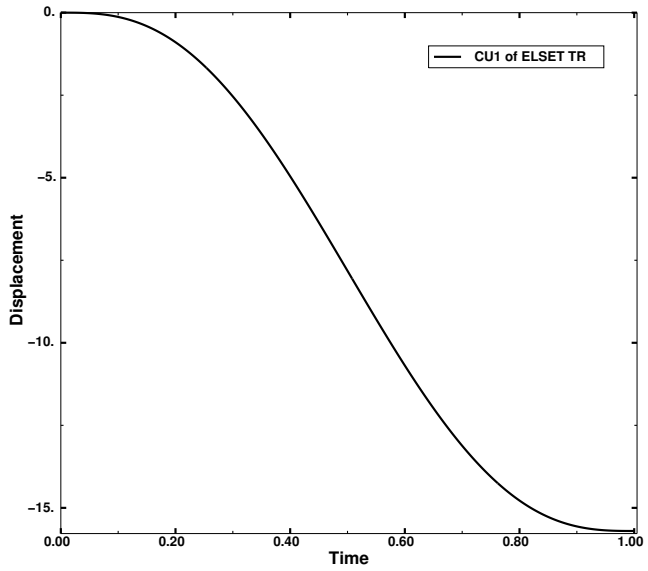


Figure 5: Displacement of the rack (CU1 of TRANSLATOR`TR`).

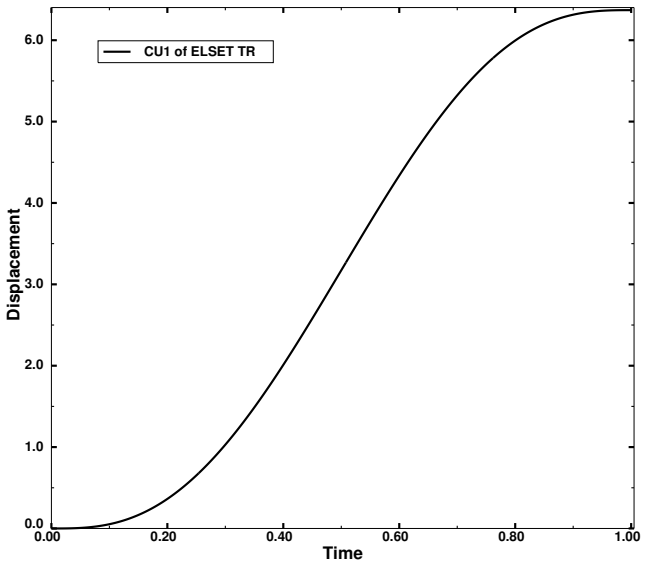


Figure 6: Displacement of the screw (CU1 of TRANSLATOR_TR).

Electrochemical and Electromagnetic Analyses

In this section:

- *Fully coupled thermal-electrochemical-structural analysis in batteries*
- *Electromagnetic analyses*

Fully coupled thermal-electrochemical-structural analysis in batteries

Products: Abaqus/Standard

References:

- *Fully Coupled Thermal-Electrochemical-Structural Analysis*

Overview

This example illustrates the capability of a fully coupled thermal-electrochemical-structural analysis for modeling coupled multiphysics-multiscale behaviors in lithium-ion batteries under mechanical constraints.

Application description

A fully coupled thermal-electrochemical-structural analysis utilizes the principles of both thermal-displacement analysis (see *Fully Coupled Thermal-Stress Analysis*) and thermal-electrochemical analysis (see *Coupled Thermal-Electrochemical Analysis*). This procedure is suited for the analysis of battery electrochemistry applications that require solving simultaneously for displacements, temperature, electric potentials in the porous solid electrodes and electrolyte, concentration of ions in the electrolyte, and concentration of lithium in the solid particles used in the electrodes. The primary application of battery electrochemistry simulation is the charging and discharging of lithium-ion batteries (LIBs). During charging, lithium ions are removed from the cathode (deintercalation) causing the active particles to shrink. The ions then travel through the electrolyte to the anode where they are absorbed into the active particles (intercalation), leading to an expansion in the active particle volume. These volumetric changes alter the structure of both electrodes and significantly impact their electrochemical behavior. This process is reversed during discharging. Heat is generated during the flow of electrons, ion migration, and the intercalation process.

Porous electrode theory (*Newman et al., 2004*) is commonly accepted as the leading method for modeling the charge-discharge behavior of lithium-ion cells. The method is based on a homogenized Newman-type approach that does not consider the details of the pore geometry. For more details on the fully coupled physics being modeled during a battery simulation, see *Coupled Thermal-Electrochemical Analysis*.

In this example, charge and discharge tests of LIBs under external mechanical constraints are carried out to investigate the effect of electrochemical-mechanical-coupled loading on battery performance. In some applications a detailed understanding of the effects of the thermal-electrochemical fields on deformation of the lithium-ion cell is important. Battery performance characteristics (such as energy storage capacity and discharge voltage) can degrade with deformation caused by particle swelling and thermal strain. Large deformation is often associated with failure of the separator and thermal runaway.

Geometry

In this example, a lithium-ion battery is precompressed and then charged at a 0.1C rate. The lithium-ion battery used in this model has a 4.1 ampere-hour capacity, and the maximum output voltage is 4.4 volts. The charge curves predicted by this model for the voltage and reaction force are compared with experimental results from *Yuan et al. (2022)*. The details of the experimental test are described in the paper.

The 3D geometry of the cell (*Figure 1*) follows a stacking structure, including the cathode (active material and current collector, 18 layers and 2 half-layers), separator (38 layers), and anode (active material and current collector, 19 layers), meshed with 22 elements through thickness for the anode and cathode layers and 10 elements through thickness for the separator layer. The cathode collector layers are electrically connected by the positive

tab and the anode current collector layers are connected by the negative tab. The total length, width, and thickness of the electrode stack are 76 mm, 57.21 mm, and 4.343 mm, respectively.

Abaqus modeling approaches and simulation techniques

Analysis types

A fully coupled thermal-electrochemical-structural analysis models the voltage and reaction force behaviors of the battery under mechanical constraints during charging.

Mesh design

QEC3D8 elements are used to model the whole assembly; 112 elements are used in plane, and 1218 elements are used through the thickness.

The QEC3D8 elements have displacement (degrees of freedom 1, 2, 3), electric potential in the solid (degree of freedom 9), temperature (degree of freedom 11), electric potential in the electrolyte (degree of freedom 32), and ion concentration in the electrolyte (degree of freedom 33) as nodal variables.

Materials

The anode material used in this simulation is graphite. The cathode material is LiCoO2 (LCO). Thermal conductivity and specific heat are defined for the battery components as thermal properties. The electrochemistry framework requires that the material definition contain the complete specification of properties required for the porous electrode theory, as described in *Material Options*. The electrochemical properties are provided in *Yuan et al. (2022)*. The mechanical properties of the separator (elastic-plastic), electrodes, current collectors and tabs (elastic) used in this model can be found in *Yuan et al. (2022)*. In this example, we assume a hyperelastic (Neo-Hooke) representation of the mechanical properties. The hyperelastic material properties are obtained based on the given elastic properties. The active mechanical properties of the electrodes are calibrated as a hyperfoam material usingconstant, universal gas constant, Avogadro number, Boltzmann constant, absolute zero temperature, and the elementary charge number that are required for this simulation are defined as model data.

Initial conditions

In a battery analysis, the initial potentials in the electrode and electrolyte are usually chosen such that the overpotentials are zero at the anode and cathode. Initial solid electric potential for cathode collector and active material is set to 3.0838 volt. Initial fluid electric potential for all nodes on the anode, cathode active layers, and separator is set as -0.8258 volt. Initial ion concentration on the nodes in the active material of the electrode is set to $600\frac{mole}{meter^3}$. Initial lithium concentration in the anode and cathode particles are set to $500\frac{mole}{meter^3}$ and $37500\frac{mole}{meter^3}$ respectively. The initial temperature for all nodes in the model is set to 298.15 K.

Boundary conditions

The anode collectors are grounded (that is, the solid electric potential is set to zero). All displacement components are constrained to be zero at all nodes. There are no boundary conditions on ion concentration since they evolve during the analysis based on the intercalation currents.

You can specify boundary conditions as functions of time by referring to amplitude curves.

Loads

Following *Yuan et al. (2022)*, the pouch cell is compressed to a predefined mechanical load of 1000N using prescribed displacement boundary conditions. A charging step is carried out at a constant rate of 0.1C with the compression platens held fixed. The battery is charged with a constant current from 3.0838 volts to 4.4 volts. The step control functionality in Abaqus is utilized to terminate the charging step when the voltage at the cathode tab reaches 4.4 volts.

Results and discussion

The voltage and reaction force of the battery under mechanical constraints during charging is validated against experimental results from *Yuan et al. (2022)*. The reaction force is measured as a surface integrated output on the topmost surface of the model in the Z direction. We focus on the first charge of the fresh cell with good contact between components. *Figure 2* displays the charge curve predicted by Abaqus showing the evolution of the voltage during charge for 0.1C and compared with experimental results from *Yuan et al. (2022)*. A reasonably good correlation is observed. *Figure 3* shows the simulation reaction force result for the same C rate, which also in good agreement with the experimental values from *Yuan et al. (2022)*. We observe that the mechanical reaction force rises as the lithium-ion battery is continuously charged and reaches the peak value at the end of charging.

Input files

battery_tecm_pt1C_charge.inp	Input file of the pouch cell for the fully coupled temperature-displacement, electrochemistry analysis at 0.1C charge rate.

References

- Newman, J., and K. E. Thomas-Alyea, “Electrochemical Systems,” Wiley-Interscience, Third Edition, 2004.
- Yuan, C., Y. Hahn, W. Lu, V. Oancea, and J. Xu, "Quantification of Electrochemical-Mechanical Coupling in Lithium-Ion Batteries," Cell Reports Physical Science, 3.12, 2022.

Figures

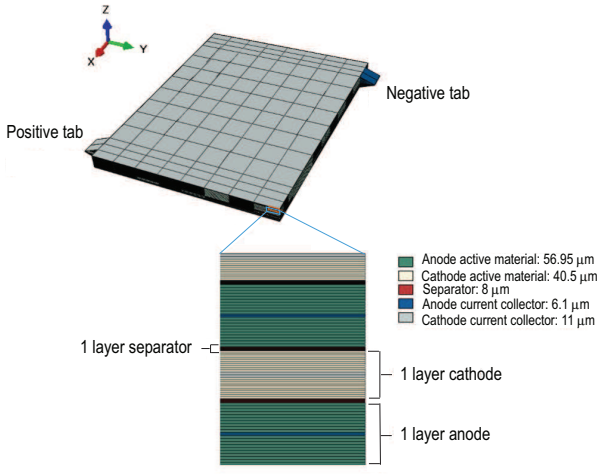


Figure 1: Pouch cell with a detailed layer of anode, cathode, separator, and electrode current collector.

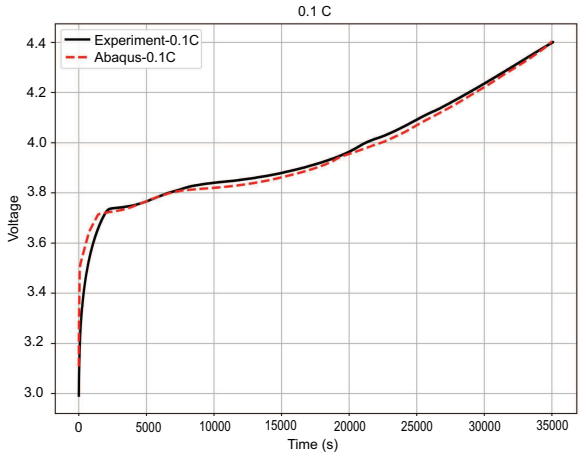


Figure 2: Voltage charge curve of the cell under mechanical constraints at 0.1C rate.

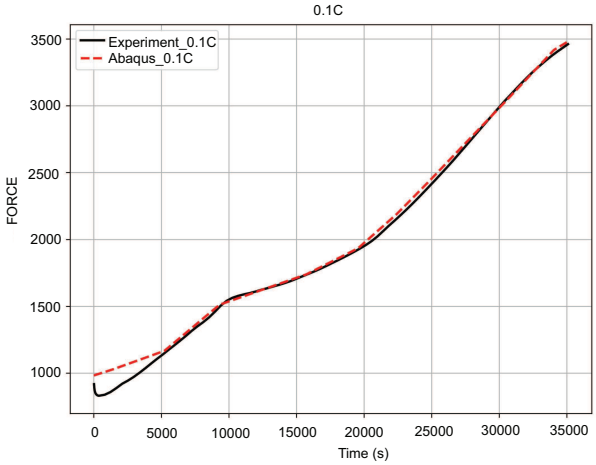


Figure 3: Mechanical force charge curve of the cell under mechanical constraints at 0.1C rate.

Electromagnetic analyses

In this section:

- *Eigenvalue analysis of a piezoelectric transducer*
- *Transient dynamic nonlinear response of a piezoelectric transducer*
- *Thermal-electrical modeling of an automotive fuse*
- *Modeling of a silicon piezoresistive pressure sensor*

Eigenvalue analysis of a piezoelectric transducer

Products: Abaqus/Standard

Overview

This problem performs an eigenspectrum analysis of a cylindrical transducer consisting of a piezoelectric material with brass end caps.
Various elements are used in the eigenspectrum analysis. The elements range from axisymmetric elements to three-dimensional elements, using both lower- and higher-order elements. The basis of the piezoelectric capability in Abaqus is described in *Piezoelectric analysis*.

Geometry and material

This problem is identical to the one discussed in a report by Mercer et al. (1987). The structure is shown in *Figure 1* and consists of a piezoelectric material PZT4 with brass end caps. The piezoelectric material is electroded on both the inner and outer surfaces.

The properties for PZT4 in a cylindrical system are:

Elasticity Matrix:

$$\begin{bmatrix} 115.4 & 74.28 & 74.28 & 0 & 0 & 0 \\ 74.28 & 139.0 & 77.84 & 0 & 0 & 0 \\ 74.28 & 77.84 & 139.0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 25.64 & 0 & 0 \\ 0 & 0 & 0 & 0 & 25.64 & 0 \\ 0 & 0 & 0 & 0 & 0 & 25.64 \end{bmatrix} \text{GPa}$$

Piezoelectric Coupling Matrix (Stress Coefficients):

$$\begin{bmatrix} 15.08 & 0 & 0 \\ -5.207 & 0 & 0 \\ -5.207 & 0 & 0 \\ 0 & 12.710 & 0 \\ 0 & 0 & 12.710 \\ 0 & 0 & 0 \end{bmatrix} \text{coulomb/m}^2$$

Dielectric Matrix:

$$\begin{bmatrix} 5.872 & 0 & 0 \\ 0 & 6.752 & 0 \\ 0 & 0 & 6.752 \end{bmatrix} 10^{-9} \quad \text{farad/m}$$

The 1-direction is radial, the 2-direction is axial, and the 3-direction is tangential. From these matrices it is seen that the poling direction is radially outwards from the axis of symmetry. (The order of the stresses in Abaqus may differ from those typically used in electrical applications. Abaqus uses the standard mechanical convention, where the stress components are ordered as $\{\sigma_{11}\sigma_{22}\sigma_{33}\tau_{12}\tau_{13}\tau_{23}\}$. See *Piezoelectric Behavior*.)

The brass is elastic and isotropic with a Young's modulus of 104 GPa and a Poisson's ratio of 0.37.

Models

The transducer is modeled with a variety of elements. It is modeled as an axisymmetric structure utilizing both the planar, axisymmetric elements and the three-dimensional elements. For the axisymmetric elements, five meshes employing 4-node, 6-node, and 8-node elements are used in the finite element discretization. The first two meshes use 4-node elements with two levels of refinement, the third mesh uses 6-node elements, and the last two meshes use the 8-node elements with two levels of refinement. Lumped mass matrices are used for the lower-order elements. Consistent mass matrices are used in the higher-order elements. The meshes used for the 4-node and 6-node axisymmetric elements are shown in *Figure 2*. The meshes used for the 8-node axisymmetric elements are shown in *Figure 3*.

The three-dimensional model uses a slice of the structure and applies axisymmetric boundary conditions; 8-node and 20-node brick elements are used. The discretization used for each model is shown in *Figure 3*. These models use a local coordinate system to maintain the proper definitions of the material properties. Also, in order to prescribe the axisymmetric boundary conditions, the nodal degrees of freedom are transformed into a cylindrical coordinate system.

All the models are considered to be open-circuited. The potentials on the inside surface are restrained to zero. The frequencies correspond to those for antiresonance.

Results and discussion

The solutions obtained with the various Abaqus models are shown in *Table 1*. Even for these coarse models, the results are quite close to the experimental results. In addition, the results from Abaqus for the lower-order axisymmetric elements with lumped mass and the higher-order axisymmetric elements with consistent mass matrices in the computation of both the resonant and antiresonant frequencies match well with the numerical results reported in Mercer et al. The first four mode shapes for the more refined model with CAX8RE elements are shown in *Figure 4*.

Similar analyses have been performed considering the problem to be closed-circuited to obtain the resonant frequencies. For this situation, the potentials on both the inner and outer surfaces are set to zero. The results also compare well with those given in Mercer et al.

Input files

eigenpiezotrans_cax4e_coarse.inp	Coarse mesh with 4-node axisymmetric elements.
eigenpiezotrans_cax4e_fine.inp	Refined mesh with 4-node axisymmetric elements.
eigenpiezotrans_cax6e.inp	Mesh with 6-node axisymmetric elements.
eigenpiezotrans_cax8re_coarse.inp	Coarse mesh with 8-node axisymmetric elements.
eigenpiezotrans_cax8re_fine.inp	Refined mesh with 8-node axisymmetric elements.
eigenpiezotrans_c3d8e.inp	Mesh with 8-node three-dimensional elements.
eigenpiezotrans_c3d8e.f	User subroutine *ORIENT* used in eigenpiezotrans_c3d8e.inp.
eigenpiezotrans_c3d20e.inp	Mesh with 20-node three-dimensional elements.
eigenpiezotrans_c3d20e.f	User subroutine *ORIENT* used in eigenpiezotrans_c3d20e.f.
eigenpiezotrans_elmatrixout.inp	Data that test the use of **ELEMENT MATRIX OUTPUT* with piezoelectric elements.
eigenpiezotrans_usr_element.inp	Data for a job that reads in the matrices output in eigenpiezotrans_elmatrixout.inp and performs an eigenvalue analysis.

References

- Mercer, C. D., B. D. Reddy, and R. A. Eve, “Finite Element Method for Piezoelectric Media,” *University of Cape Town/CSIR Applied Mechanics Research Unit Technical Report*, no. 92, April 1987.

Tables

Table 1: Piezoelectric transducer eigenvalue estimates.

Model		Frequencies (kHz) for mode number				
Type	**# of Elements**	**1**	**2**	**3**	**4**	**5**
CAX4E	13	14.1	39.1	56.2	66.1	79.3
CAX4E	320	18.6	40.3	57.8	64.2	88.1
CAX6E	10	20.0	43.2	63.2	70.4	98.8
CAX8RE	5	19.6	42.8	61.0	66.9	96.3
CAX8RE	80	18.6	40.3	57.6	64.2	87.6
C3D8E	16	19.8	41.8	62.0	68.7	95.2
C3D20E	16	19.7	42.9	60.4	66.5	91.7
Experimental[(1)]		18.6	35.4	54.2	63.3	88.8
(1): Experimental results obtained from Mercer et al. (1987).						

Figures

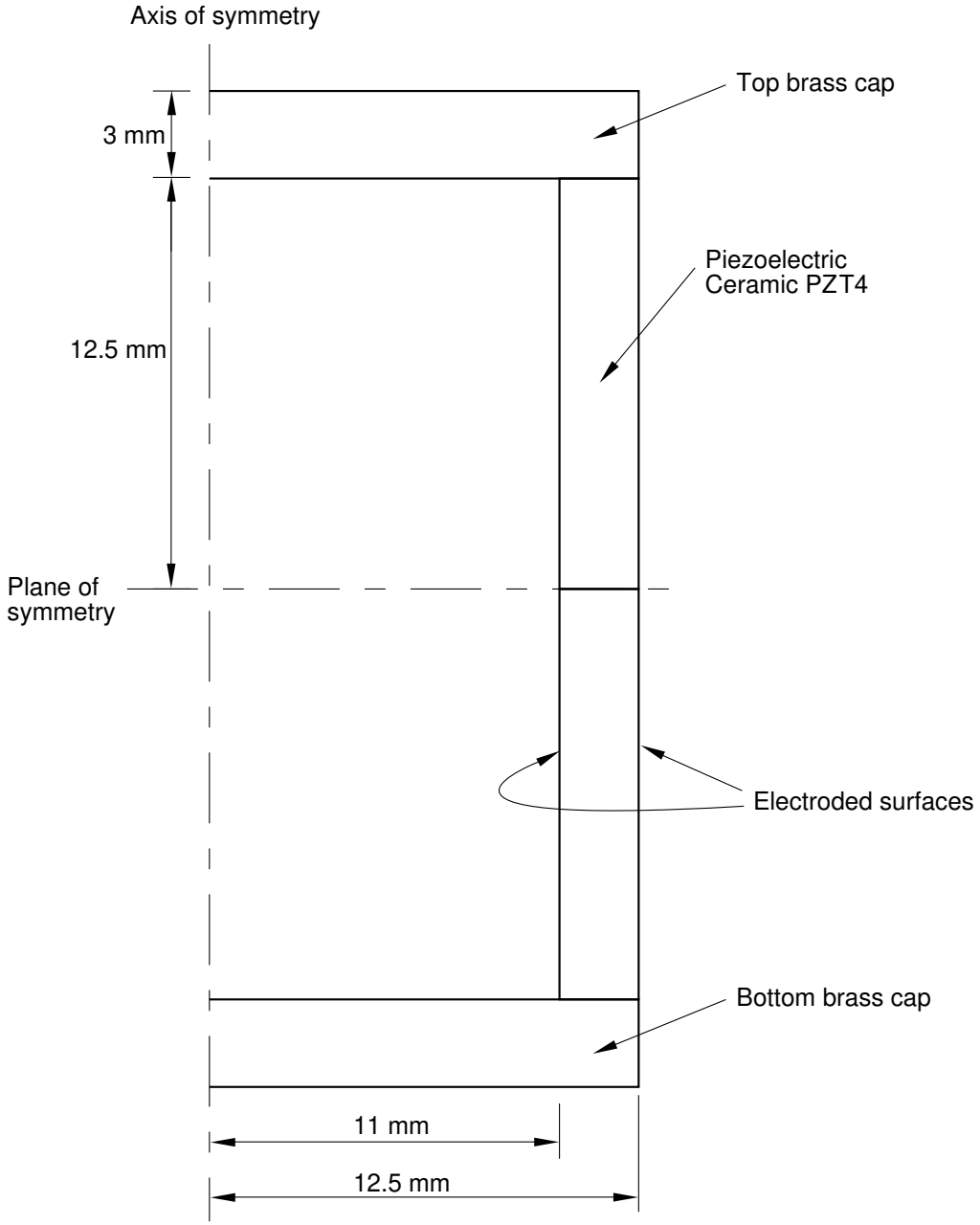


Figure 1: Piezoelectric transducer.

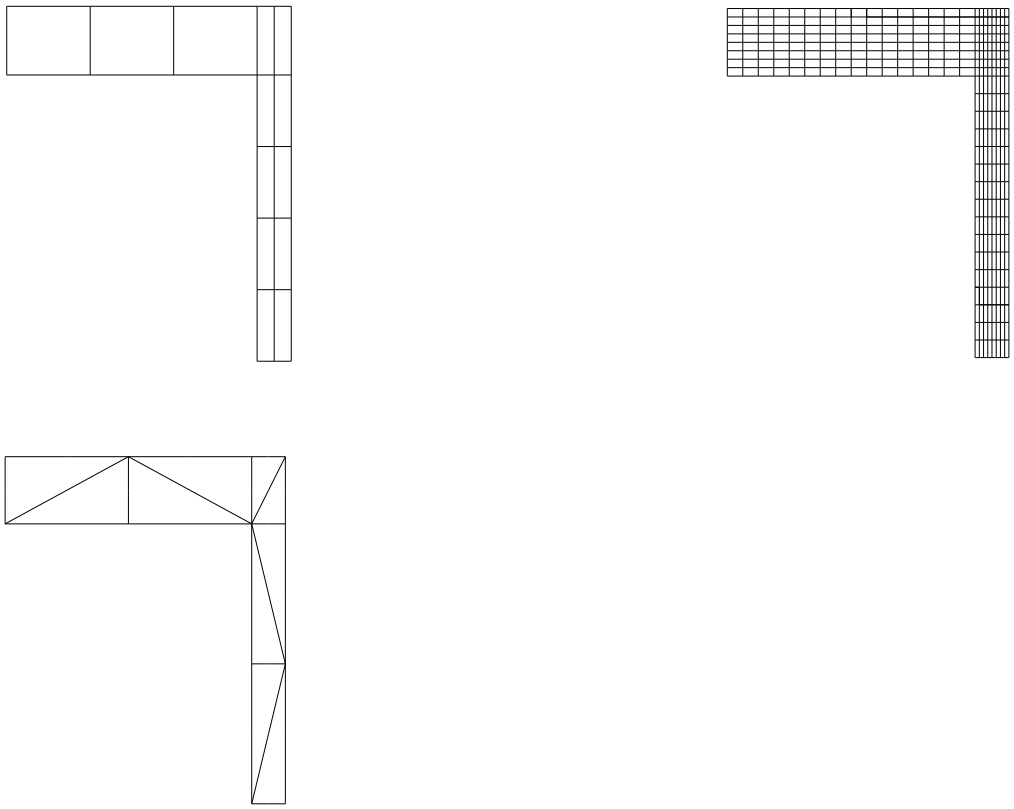


Figure 2: Meshes used with 4-node and 6-node axisymmetric elements.

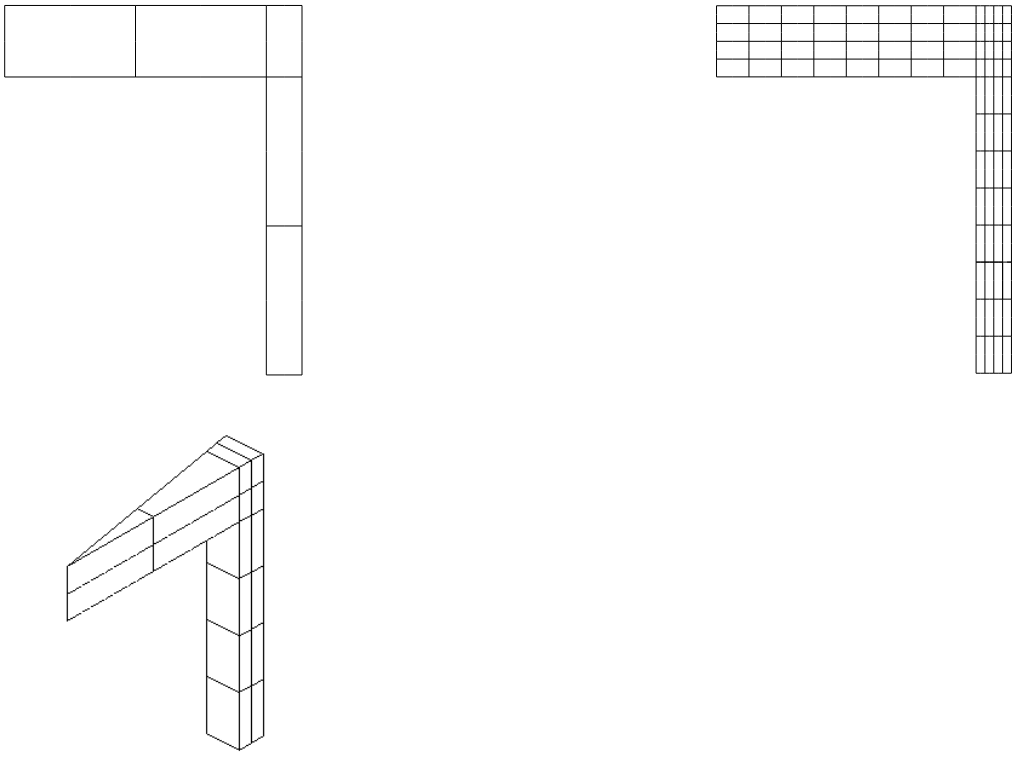


Figure 3: Meshes used with 8-node axisymmetric and 8-node and 20-node three-dimensional elements.

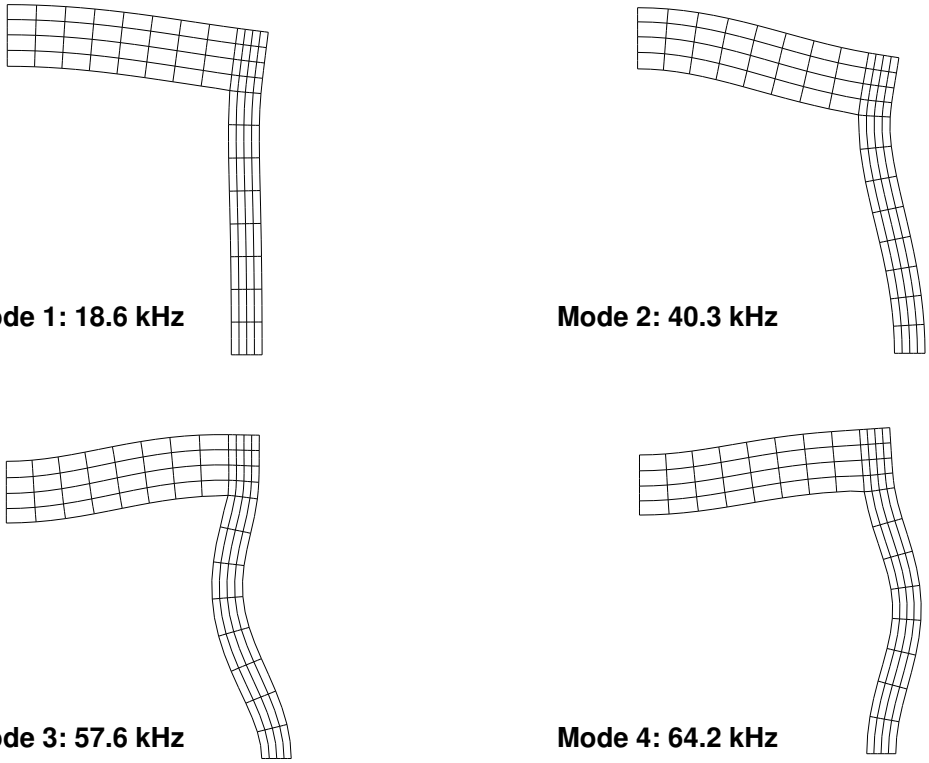


Figure 4: Mode shapes for 8-node axisymmetric elements.

Transient dynamic nonlinear response of a piezoelectric transducer

Products: Abaqus/Standard

Overview

This example demonstrates the Abaqus capability for predicting the nonlinear transient dynamic behavior of structural systems that incorporate piezoelectric components.
The example utilizes a piezoelectric bending-type transducer that is idealized as a simple layered beam structure. The fabrication of the transducer is modeled, and the eigenfrequencies of the preloaded structure are extracted. Finally, the dynamic response due to a transient electrical potential pulse is monitored. Piezoelectric transducers are commonly used in the following application areas:

- Ultrasonic imaging systems
- Ultrasonic cleaning systems
- Ultrasonic welding/bonding systems
- Audio systems
- Acoustic transducers
- Active vibration control systems

Geometry and material

The transducer is shown in *Figure 1*. It has a composite beam that is 0.0025 m wide and 0.18 m long. The initially straight beam has a 0.0005 m thick insulating core; 0.000125 m thick PZT–5H piezoelectric strips are bonded to the top and the bottom faces of the core material. The piezoelectric strips are only 0.060 m long and are centered at the beam mid-span. The piezoelectric strips are polarized along the beams in the through-thickness direction. The properties for the PZT–5H material are as follows:

Elastic properties:

Engineering constants	
E_1	60.61 GPa
E_2	48.31 GPa
E_3	60.61 GPa
ν_{12}	0.512
ν_{13}	0.289
ν_{23}	0.408
G_{12}	23.0 GPa
G_{13}	23.5 GPa
G_{23}	23.0 GPa

Piezoelectric coupling matrix (strain coefficients):

$$\begin{bmatrix} 0 & 0 & 0 & 741 & 0 & 0 \\ -274 & 593 & -274 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 741 \end{bmatrix} 10^{-12} \quad \text{m/volt}$$

Dielectric matrix:

$$\begin{bmatrix} 1.505 & 0 & 0 \\ 0 & 1.301 & 0 \\ 0 & 0 & 1.505 \end{bmatrix} 10^{-8} \quad \text{farad/meter}$$

The local 1-direction is in the beam longitudinal direction, and the local 2- and 3-directions are in the beam cross-section. From these matrices it can be seen that the poling direction is in the local 2-direction of the piezoelectric strips.

The core material is elastic and isotropic with a Young's modulus of 6 GPa, a density of 1500 kg/m^3, and a Poisson's ratio of 0.35.

Model

The beam core material is modeled with 46 C3D20 elements, and the piezoelectric strips are modeled using 16 C3D20E elements each. A perfect bond between the core and the piezoelectric materials is defined with a surface-based tie constraint, for which the piezoelectric surfaces are retained as the main surfaces. The electrical potentials for the top and the bottom surfaces of each piezoelectric strip are coupled to the electrical potentials of the main nodes assigned to each surface using linear constraint equations. The electrical potentials and the reaction charges can be monitored at these main nodes. The piezoelectric surfaces bonded to the core material are assigned a zero electrical potential throughout the analysis.

The first 8 steps are used to represent the fabrication procedure and to investigate the behavior of the fabricated transducer. These include linear perturbation steps to investigate the eigenmodes of the fabricated transducer at various stages during its fabrication. Steps 9 through 11 represent a general nonlinear transient analysis of the transducer. A square wave electrical potential pulse of 200 volts is applied using a boundary condition with a step function amplitude. Immediately after the pulse, a closed circuit condition (potential gradients are prescribed) is maintained and the reaction charges are monitored. Subsequently, the circuit is opened (the potential gradient is not prescribed; the potential is an active degree of freedom that is determined as part of the solution). In the case of an open circuit condition the resulting voltage can be used to measure the transducer's open circuit free vibration. The analysis steps are as follows:

1. Static shape fabrication: deformation induced via an applied potential of 1000 volts.
2. Static shape fabrication: fix support ends, and reduce the applied potential to 0 volts.
3. Closed circuit modal analysis about the base state obtained at the end of Step 2.
4. Static shape fabrication: apply open circuit conditions.
5. Open circuit modal analysis about the base state obtained at the end of Step 4.
6. Static test at an operational load of 200 volts.
7. Closed circuit modal analysis about the base state obtained at the end of Step 6.
8. Reset to zero voltage condition.
9. Transient dynamic response: apply voltage pulse of 200 volts for 0.00265 seconds.
10. Transient dynamic response: free vibration under closed circuit condition with 0 volts.
11. Transient dynamic response: free vibration under open circuit condition.

Results and discussion

In *Figure 2* the deformed and superimposed undeformed shapes are shown after applying 1000 volts at both piezoelectric strips. In *Figure 3* the deformed shape is shown at the end of Step 2, where both ends are fixed and an applied potential of 0 volts is prescribed. Subsequently, the eigenfrequencies are extracted about this preloaded state. In *Figure 4* the third eigenmode with a frequency of 150.9 cycles/sec is shown for this closed circuit condition. *Figure 5* shows the third mode shape with a frequency of 154.4 cycles/sec under open circuit conditions. The third mode shape for the closed circuit condition with a prescribed voltage of 200 volts is again almost identical to the third mode shape shown in *Figure 4*. However, the eigenfrequency has changed to 181.7 cycles/sec.

In Steps 9 and 10 a closed circuit condition is prescribed for the piezoelectric strips, and subsequently in Step 11 the voltage boundary conditions are removed resulting in open circuit conditions. *Figure 6* shows the time history of the potential under closed and open circuit conditions. Typically, the transducer acts as a "driver" in a closed circuit condition as the potential gradient is prescribed, thereby driving the structure. On the other hand, the transducer acts as a "receiver" in an open circuit condition as the voltage output can be used to measure the mechanical response. In *Figure 7* the reaction charge at the top piezoelectric strip is shown under the above conditions. In a closed circuit condition the reaction charge changes in time, while in an open circuit condition the reaction charge is equal to zero. The displacement of the transducer's center is demonstrated in *Figure 8*.

The results are presented to illustrate the general capabilities in Abaqus for predicting the transient response of piezoelectric structures. If this system represented part of an actual ultrasound system, additional output related to the design/analysis objective would be created and analyzed.

Input files

dynamictransducer.inp	Transient dynamic nonlinear response of a piezoelectric transducer.
dynamictransducer_mesh.inp	Assembly definition.

Figures

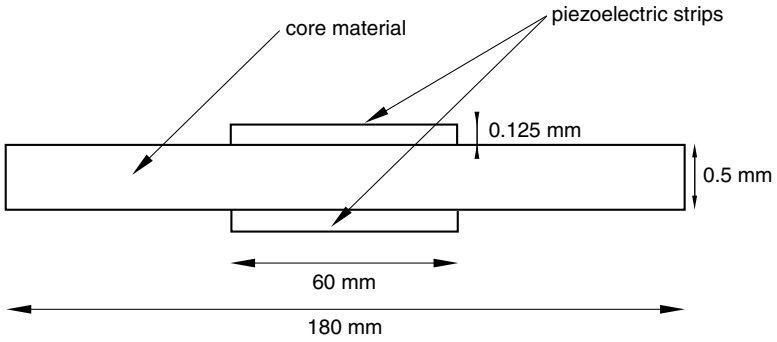


Figure 1: Geometry of the piezoelectric transducer.

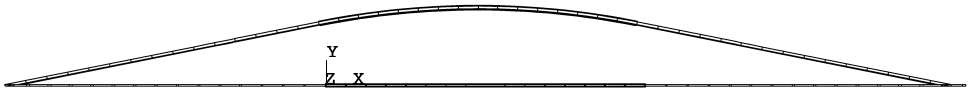


Figure 2: Deformed shape with superimposed undeformed shape at the end of Step 1.



Figure 3: Deformed shape with superimposed undeformed shape at the end of Step 2.

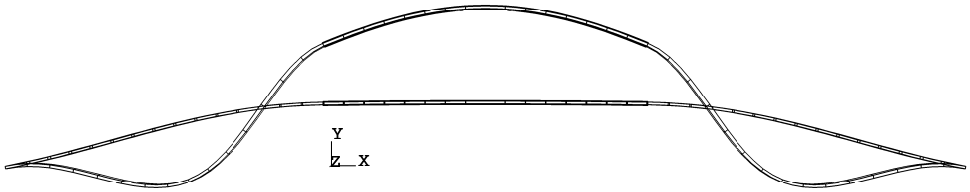


Figure 4: Mode shape 3 with superimposed shape of the base state after Step 2.

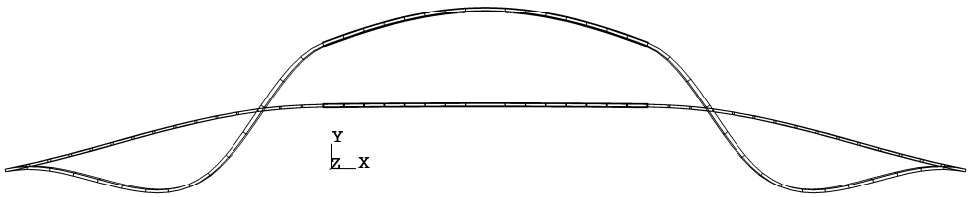


Figure 5: Mode shape 3 with superimposed shape of the base state after Step 4.

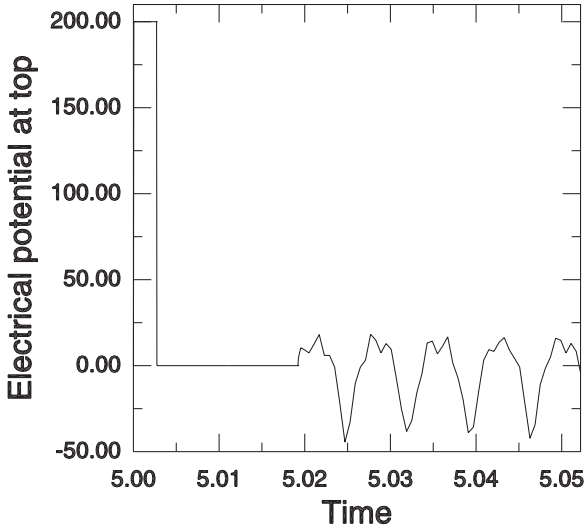


Figure 6: Transient response of potential at top piezoelectric strip.

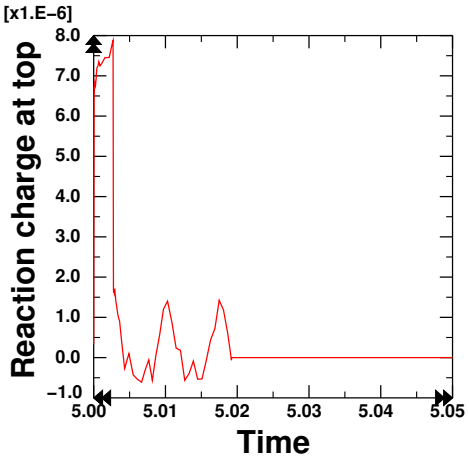


Figure 7: Transient dynamic response of the reaction charge at top piezoelectric strip.

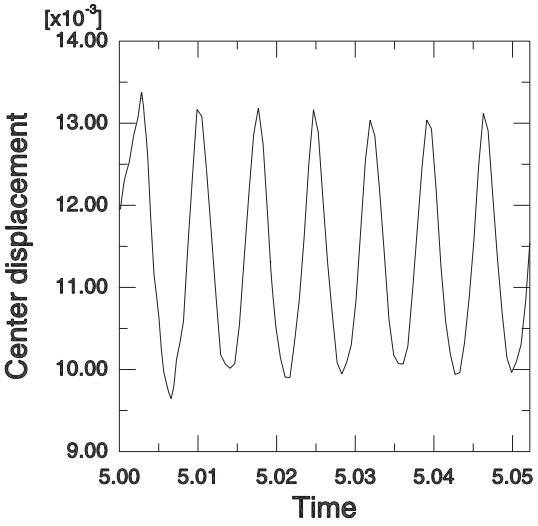


Figure 8: Transient response of center displacement.

Thermal-electrical modeling of an automotive fuse

Products: Abaqus/Standard

Overview

This example shows the use of the fully coupled thermal-electrical procedure to analyze an automotive fuse.

Joule heating arises when the energy dissipated by electrical current flowing through a conductor is converted into thermal energy. Abaqus provides a fully coupled thermal-electrical procedure for analyzing this type of problem.

An overview of the capability is provided in *Coupled Thermal-Electrical Analysis*. This example illustrates the use of the capability to model the heating of an automotive electrical fuse due to a steady 30 A electrical current. Fuses are the primary circuit protection devices in automobiles. They are available in a range of different current ratings and are designed so that when the operating current exceeds the design current for a period of time, heating due to electrical conduction causes the metal conductor to melt and—hence—the circuit to disconnect.

A description of the original problem, as well as experimental measurements, can be found in Wang and Hilali (1995). The experimental data and some of the material properties were refined subsequent to this publication. These properties are used here, and the finite element results are compared with the refined measurements (Hilali, July 1995).

Problem description

An automotive electrical fuse consists of a metal conductor, such as zinc, embedded within a transparent plastic housing. The plastic housing, which only protects and supports the thin conductor, is not represented in the finite element model. *Figure 1* shows front and top sections of the geometry of the conductor. It consists of two 0.76 mm thick blades, with an S-shaped fuse element supported between the blades. The blades fit tightly into standard electrical terminals that are built into the circuit and provide the connection between the electrical circuit and the fuse element. The fuse element is usually much thinner than the fuse blades (in this case 0.28 mm thick) and is designed to melt when the operating current exceeds the design current for a period of time. The fuse blades are 8 mm wide and 30.4 mm long. The fuse element is approximately 3.6 mm wide.

The model is discretized (see *Figure 1*) with 8-node first-order brick elements (element type DC3D8E), using one element through the thickness. Two 6-node triangular prism elements (element type DC3D6E) are used to fill regions where the geometry precludes the use of brick elements. For comparison refined mesh input files are also included.

The electrical conductivity of zinc varies linearly between $16.75 \times 10^3 \mathbf{1/\Omega}$ mm at 20°C and $12.92 \times 10^3 \mathbf{1/\Omega}$ mm at 100°C. The thermal conductivity varies linearly between 0.1120 W/mm°C at 20°C and 0.1103 W/mm°C at 100°C. The density is 7.14×10^{-6} kg/mm^3, and the specific heat is 388.9 J/kg°C. The fraction of dissipated electrical energy released as heat is specified by the Joule heat fraction. We assume that all electrical energy is converted into thermal energy.

The analysis is done in two steps. In the first step heating of the conductor due to current flow is considered. Once steady-state conditions are reached, the current is switched off and the fuse is allowed to cooldown to the ambient temperature in a second step. During the first part of the analysis, the coupled thermal-electrical equations are solved for both temperature and electrical potential at the nodes using the coupled thermal-electrical procedure. In the subsequent cooldown period, since there is no longer any electric current in the fuse, an uncoupled heat transfer analysis (*Uncoupled Heat Transfer Analysis*) is performed. The coupled thermal-electrical procedure definition is general enough to request either a steady state or a transient solution; input files illustrating both analysis types are provided. The transient analysis uses an automatic time incrementation scheme that is controlled by specifying the maximum temperature change allowed in an increment to 20°C. The transient analysis is set to terminate when steady-state conditions are reached. Steady state is defined here as the point at which the

temperature rate change is less than 0.1°C/s. This condition is defined as part of the definition of the coupled thermal-electrical procedure. We specify a total analysis time of 100 s, with an initial time step size of 0.1 s.

The electrical loading is a steady 30 A current. This is applied as a concentrated current on each of the nodes on the bottom edge of the left-hand-side terminal. The total current and the total heat flux across a section defined through the fuse element are output. The electrical potential (degree of freedom 9) is constrained at the bottom edge of the right-hand-side blade by using a boundary condition (*Boundary Conditions*). This option is also used to keep the bottom edges of the fuse blades at sink temperatures (degree of freedom 11) of 29.4°C and 30.2°C, respectively.

It is assumed that the exposed metal surfaces lose heat through convection to an ambient temperature of $\theta^0 =$ 23.3°C. Heat loss from the thin edges is ignored. The film coefficient varies with temperature according to the empirical relation

$$h = h^0 \left(\theta - \theta^0\right)^{1/4},$$

where θ is surface temperature (°C); h is the film coefficient (W/mm^2°C); and h^0 is a constant that depends on the surface geometry—$h^0 = 4.747 \times 10^{-6}$ W/mm^2 for the blade surfaces, and $h^0 = 5.756 \times 10^{-6}$ W/mm^2 for the fuse element surfaces. This dependence is entered as a table of film property values in the film coefficient definition (see *Thermal Loads*).

Results and discussion

Figure 2 shows a contour plot of the magnitude of the electrical current density vector at steady-state conditions. Since the dissipated electrical energy—and, hence, the thermal energy—is a function of current density, this figure represents contours of the heat generated. The figure indicates that most of the heat is generated near the inside curves of the S-shaped fuse element and near the center hole. The dissipated energy in the fuse blades is negligible compared to that in the fuse element.

Figure 3 shows a contour plot of the temperature distribution at the end of the first analysis step. The maximum temperature is reached near the center of the S-shaped fuse element. This area is expected to fail first when the operating current exceeds the design current. *Figure 4* compares the temperatures at the measuring positions (defined in *Figure 1*) with the experimental measurements (Hilali, July 1995). While the results show some discrepancies between the experiment and analysis, it is clear that the analysis is sufficiently representative to provide a useful basis for studying such systems. *Figure 5* shows the variation of temperature at measuring position 6 during the heating and subsequent cooldown periods.

The results discussed above are for the coarse mesh model. The refined mesh models yield slightly different results from the coarse ones. The maximum difference in the magnitude of the electrical current density vector for the steady-state analysis is approximately 11.4%.

Acknowledgments

Mr. Hilali and Dr. Wang of Delphi Packard Electric Systems supplied the geometry of the fuse, the material properties, and the experimental results. Delphi Packard assumes no responsibility for the accuracy of the analysis method or data contained in the analysis.

Input files

thermelectautofuse_steadystate.inp	Steady-state analysis.
thermelectautofuse_transient.inp	Transient analysis.
thermelectautofuse_transient_po.inp	**POST OUTPUT* analysis.
thermelectautofuse_node.inp	Nodal coordinates for the model.

thermelectautofuse_element.inp	Element definitions.
thermelectautofuse_controls.inp	Identical to thermelectautofuse_steadystate.inp, except that it uses the **CONTROLS* option for control of convergence criteria.
teaf-steadystate-refined.inp	Refined mesh model for the steady-state analysis.
teaf-transient-refined.inp	Refined mesh model for the transient analysis.

References

- Hilali, S. Y., Private communication, July 1995.
- Hilali, S. Y., and B. -J. Wang, "Abaqus Thermal Modeling for Electrical Assemblies," *1995 Abaqus Users' Conference, Paris, May 1995*, pp. 441–457.
- Wang, B. -J., and S. Y. Hilali, "Electrical-Thermal Modeling Using Abaqus," *1995 Abaqus Users' Conference, Paris, May 1995*, pp. 771–785.

Figures

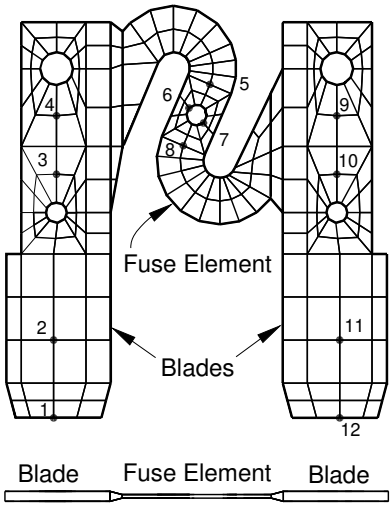


Figure 1: Geometry and finite element discretization.

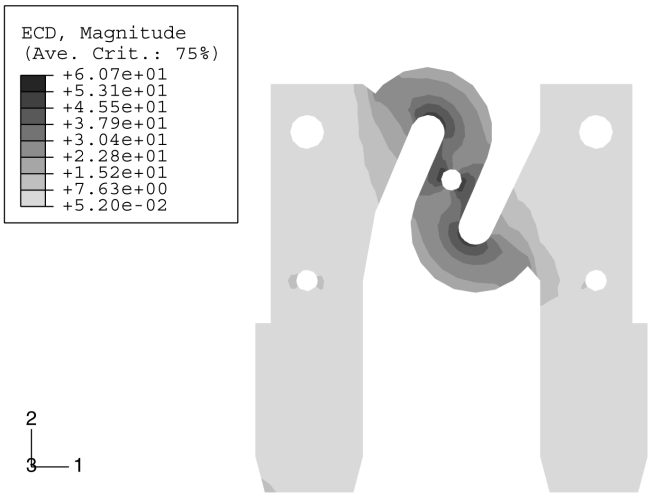


Figure 2: Contours of the magnitude of the current density vector (A/mm^2).

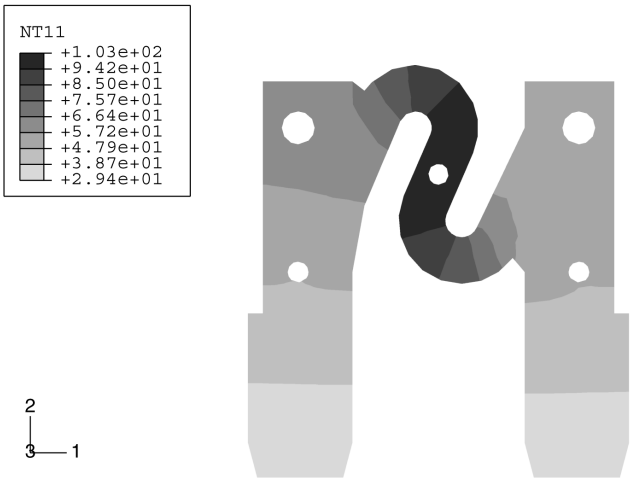


Figure 3: Contours of temperature field (°C).

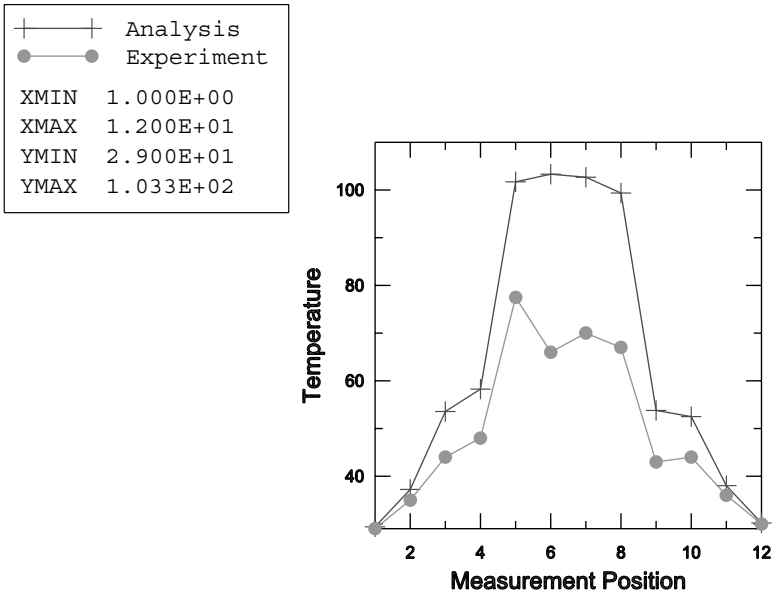


Figure 4: Temperature (°C) at measuring positions.

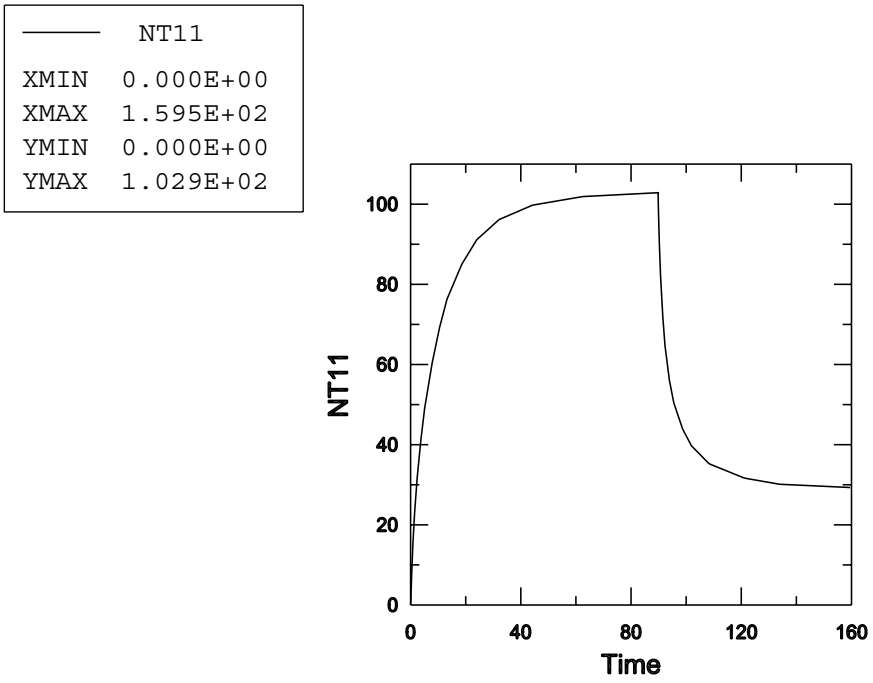


Figure 5: Variation of temperature (°C) at measuring position 6 with time (s).

Modeling of a silicon piezoresistive pressure sensor

Products: Abaqus/Standard

References:

- *Fully Coupled Thermal-Electrical-Structural Analysis*
- *Piezoresistive Effect*

Overview

This example illustrates the use of a piezoresistive material model to simulate a silicon piezoresistive pressure sensor.

Application description

The model shown in *Figure 1* represents a simplified piezoresistive membrane-chip pressure sensor. It consists of a silicon piezoresistive membrane chip bonded with glass and then glued to a base. Four p-doped piezoresistors (labeled R1, R2, R3, and R4 in *Figure 2*) are placed on top of the membrane and arranged to form a Wheatstone bridge circuit as shown in *Figure 2*.

A constant current is applied to the circuit. Without any applied pressure, the resistances of the four resistors are the same, and the output voltage between nodes 1 and 2 is zero. When a uniform pressure is applied on the top surface, the resistances of the four resistors change with the applied pressure. The stress state in resistors R1 and R4 is different from the stress state in resistors R2 and R3. Therefore, the resistances of R1/R4 and R2/R3 change differently with the applied pressure. The output voltage between nodes 1 and 2 can be measured for different values of the applied pressure.

Geometry

The four piezoresistors are placed 0.396 mm from the center of the membrane, as shown in *Figure 3*. They are aligned with the [1 1 0] direction, which represents the direction cosines of the longitudinal resistor direction and the crystal axis. The size of the piezoresistors $l_0 \times w_0$ is 90 μm × 25 μm, and the thickness h_0 is 0.3086 μm, as shown in *Figure 3* and *Figure 4*. The height of the base is 2 mm, and the diameter is 4 mm. *Table 1* provides a summary of the other model dimensions.

Abaqus modeling approaches and simulation techniques

The four piezoresistors are meshed using coupled thermal-electrical-structural elements Q3D8R, while all the other region of the model are meshed using C3D8R elements. To model the Wheatstone bridge circuit, each pair of adjacent resistors is connected to a reference node virtually by uniformly coupling the electrical potential of the nodes on the end surfaces of the resistors to that of the reference node. A total of four reference nodes are introduced, corresponding to nodes 1, 2, 3, and 4 in *Figure 2*

Analysis types

A fully coupled thermal-electrical-structural analysis is performed.

Mesh design

The silicon is partitioned into two parts: the silicon membrane top and the silicon support. These two parts are meshed separately. The glass and the base are also meshed separately. All the parts are connected by tie constraints.

Materials

The orthotropic elasticity properties (see *Table 2*) for the silicon material were obtained from *Hopcroft, Nix, and Kenny (2010)*. *Table 3* lists the piezoresistive coefficients for the p-doped silicon (*Kumar and Pant, 2014*). The initial resistivity of the four piezoresistors is 0.3 mm. The thermal properties of silicon are summarized in *Table 4*.

The Young's modulus and Poisson's ratio of the glass is assumed to be 6.28×10^7 KPa and 0.2, respectively. The Young's modulus and Poisson's ratio of the base material is assumed to 1.38×10^8 KPa and 0.317, respectively.

Boundary conditions

All the bottom nodes of the base are fixed. The temperature of all the resistors is set to be 25°C. The electrical potential of node 4 is constrained to be 0 mV.

Loads

A uniform pressure 1000 KPa is applied on the top of the silicon membrane. The electrical loading is a steady 1mA current over the duration of the step. This loading is applied as a distributed current density on the surfaces of R2 and R4 that are coupled to reference node 3.

Acknowledgements

SIMULIA gratefully acknowledges Kistler Instrumente AG for providing some of the input data used in this example.

Results and discussion

Figure 5 shows the output voltage across nodes 1 and 2 versus the applied pressure for cases with and without piezoresistivity, demonstrating the importance of the piezoresistive effect for simulating the transducer's sensitivity to the applied pressure. The figure shows an essentially linear relationship between the applied pressure and the output voltage for both cases. The maximum deflection at the center of membrane is very small compared to the membrane thickness yielding small deformations that change linearly with the applied pressure. Without piezoresistivity, these small deformations are the only source of resistance change, yielding an output voltage sensitivity of about -1.02×10^{-3} mV/kPa for the transducer. The piezoresistive effect increases the magnitude of the transducer's sensitivity to the applied pressure over 100 times, yielding an output voltage sensitivity of 0.113 mV/kPa. *Figure 6* and *Figure 7* show the contours of the electrical potential field and the von Mises stress field, respectively, in the four resistors. The stresses in resistors R1 and R3 are different, and this difference leads to the differences in the electrical potential in these two resistors. Similar observations can be made regarding the fields in resistors R2 and R4.

Input files

Silicon_PR_Pressure_Sensor.inp	Input file for the example.

References

- Kumar, S.S., and B. D. Pant, "Design Principles and Considerations for the 'Ideal' Silicon Piezoresistive Pressure Sensor: a Focused Review," Microsystem Technologies, vol. 20, no. 7, pp. 1213–1247, July 2014.
- Hopcroft, M. A. , W. D. Nix, and T. W. Kenny, "What is the Young's Modulus of Silicon?," Journal of Microelectromechanical Systems, vol. 19, no. 2, pp. 229-238, April 2010.

Tables

Table 1: Geometry of the model.

Piezoresistor	l_0	90 μm
	w_0	25 μm
	h_0	0.3086 μm
Membrane	L1	0.912 mm
	W1	0.912 mm
	t	50.4 μm
Silicon	L	2 mm
	W	2 mm
	H1	0.36 mm
		54.7°
Glass	L	2 mm
	W	2 mm
	H2	0.96 mm
Position	p	0.396 mm

Table 2: Orthotropic elasticity properties of silicon (modulus unit: 10^7 KPa).

E_1	E_2	E_3	$_{12}$	$_{13}$	$_{23}$	G_{12}	G_{13}	G_{23}
13.0	13.0	13.0	0.28	0.28	0.28	7.96	7.96	7.96

Table 3: Piezoresistive coefficients for p-doped silicon (unit: 10^{-12} Pa^{-1}).

$_{1111}$	$_{1122}$	$_{2222}$	$_{1133}$	$_{2233}$	$_{3333}$	$_{1212}$	$_{1313}$	$_{2323}$
66.0	-11.0	66.0	-11.0	-11.0	66.0	690.5	690.5	690.5

Table 4: Thermal properties of silicon.

Specific heat	Thermal conductivity	Thermal expansion	Density
7.94×10^8 μJ/Kg/°C	1.24×10^5 μW/mm/°C	2.46×10^{-6} $°C^{-1}$	2.33×10^{-6} Kg/mm^3

Figures

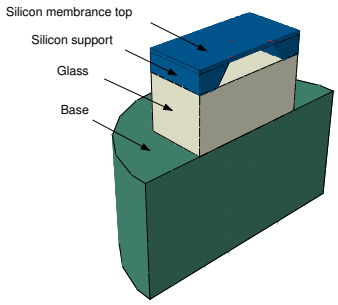


Figure 1: Half cut view of the model.

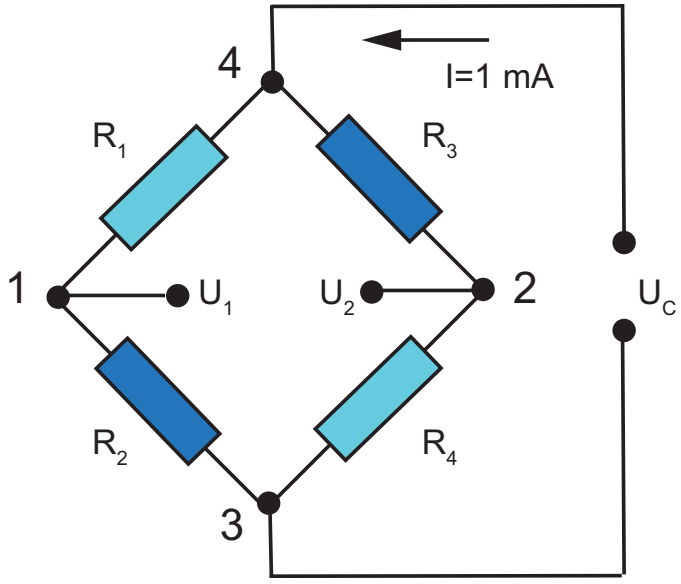


Figure 2: Wheatstone bridge circuit.

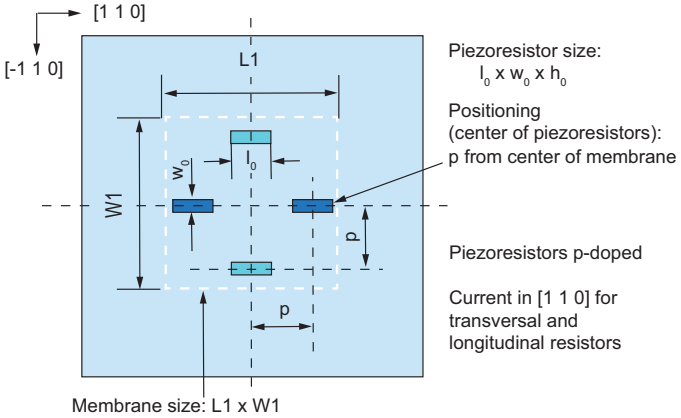


Figure 3: Top view of the model.

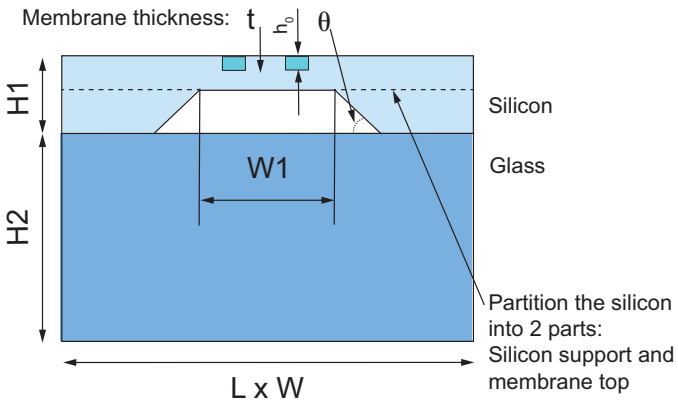


Figure 4: Cut view of the silicon and glass part.

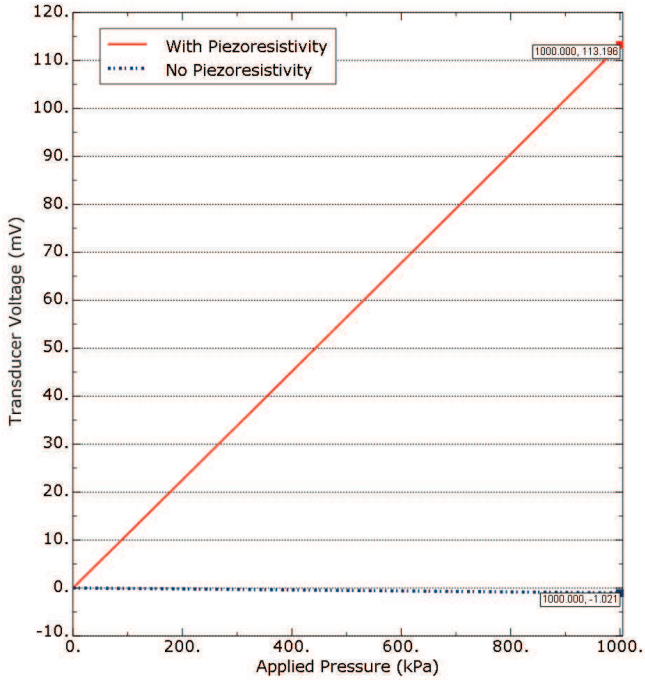


Figure 5: Output voltage vs the applied pressure.

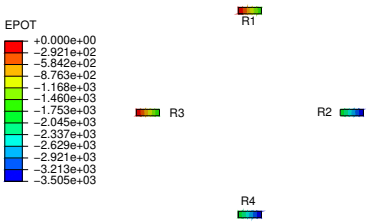


Figure 6: Electrical potential contour of the four piezoresistors.

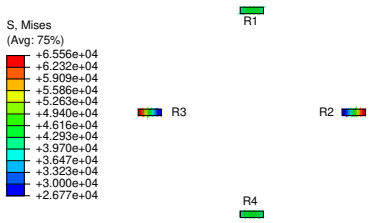


Figure 7: Von Mises stress contour of the four piezoresistors.

Heat Transfer and Thermal-Stress Analyses and Particle Methods

In this section:

- *Heat transfer and thermal-stress analyses*
- *Particle methods*

Heat transfer and thermal-stress analyses

In this section:

- *Thermal-stress analysis of a disc brake*
- *A sequentially coupled thermomechanical analysis of a disc brake with an Eulerian approach*
- *Exhaust manifold assemblage*
- *Coolant manifold cover gasketed joint*
- *Conductive, convective, and radiative heat transfer in an exhaust manifold*
- *Thermal-stress analysis of a reactor pressure vessel bolted closure*
- *Coupled thermomechanical analysis of viscoelastic dampers*

Thermal-stress analysis of a disc brake

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example shows the fully coupled thermal-stress analysis of a disc brake.

Disc brakes operate by pressing a set of composite material brake pads against a rotating steel disc: the frictional forces cause deceleration. The dissipation of the frictional heat generated is critical for effective braking performance. Temperature changes of the brake cause axial and radial deformation; and this change in shape, in turn, affects the contact between the pads and the disc. Thus, the system should be analyzed as a fully coupled thermomechanical system.

In this section two thermally coupled disc brake analysis examples are discussed. The first example is an axisymmetric model in which the brake pads and the frictional heat generated by braking are "smeared" out over all 360° of the model. This problem is solved using only Abaqus/Standard. The heat generation is supplied by user subroutine *FRIC*, and the analysis models a linear decrease in velocity as a result of braking.

The second example is a three-dimensional model of the entire disc with pads touching only part of the circumference. The disc is rotated so that the heat is generated by friction. This problem is solved using both Abaqus/Standard and Abaqus/Explicit.

It is also possible to perform uncoupled analysis of a brake system. The heat fluxes can be calculated and applied to a thermal model; then the resulting temperatures can be applied to a stress analysis. However, since the thermal and stress analyses are uncoupled, this approach does not account for the effect of the thermal deformation on the contact which, in turn, affects the heat generation.

Another type of geometrical model for a disc brake is used by Gonska and Kolbinger (1993). They model a "vented" disc brake (*Figure 1*) and take advantage of radial repetition by modeling a pie-slice segment (*Figure 2*). Like the axisymmetric model, this requires the effect of the pads to be smeared, but it allows the modeling of radial cooling ducts while still reducing the model size relative to a full model.

Geometry and model

Both models analyzed in this example have solid discs, which allows the models to use coarser meshes than would be required to model the detail of a typical disc brake that has complicated geometrical features such as cooling ducts and bolt holes. The first example further simplifies the model by considering the pads to be "smeared" around the entire 360° so that the system is axisymmetric. The second example is a full three-dimensional model of the entire annular disc with pads touching only part of the circumference. However, the geometry of the disc has been simplified by making it symmetrical about a plane normal to the axis. Therefore, only half of the disc and one brake pad is modeled, and symmetry boundary conditions are applied.

The dimensions of the axisymmetric model are taken from a typical car disc brake. The disc has a thicker friction ring connected to a conical section that, in turn, connects to an inner hub. The inner radius of the friction ring is 100.0 mm, the outer radius is 135.0 mm, and it is 10.0 mm thick. The conical section is 32.5 mm deep and 5.0 mm thick. The hub has an inner radius of 60.0 mm, an outer radius of 80.0 mm, and is 5.0 mm thick. The pads are 20.0 mm thick and initially cover the entire friction ring surface.

Two analyses of the axisymmetric model are performed in which the pads and disc are modeled using fully integrated and reduced-integration linear axisymmetric elements. Reduced integration is attractive because it decreases the analysis cost and, at the same time, provides more accurate stress predictions. Frictional contact between the pads and the disc is modeled by contact pairs between surfaces defined on the element faces in the contact region. Small sliding is assumed. The mesh is shown in *Figure 3*, with the pads drawn in a darker gray

than the disc. There are six elements through the thickness of the friction ring and four elements through the thickness of each of the pads. The mesh is somewhat coarse but is optimized by using thinner elements near the surfaces of the disc and pads where contact occurs for better resolution of the thermal gradients in these areas.

The disc for the three-dimensional model has an outer radius of 135.0 mm, an inner radius of 90.0 mm, and a thickness of 10.0 mm (the half-model has a thickness of 5.0 mm). The ring has a thinner section out to a radius of 100.0 mm, which has a thickness of 6.0 mm (the half-model has a thickness of 3.0 mm). The pad is 10.0 mm thick and covers a little less than one-tenth the circumference. The pad does not quite reach to the edge of the thicker part of the friction ring.

The pad and disc of the three-dimensional model are modeled with C3D8T elements in Abaqus/Standard and with C3D8RT elements in Abaqus/Explicit; the contact and friction between the pad and the disc are modeled by contact pairs between surfaces defined on the element faces in the contact region. The same mesh is used in both Abaqus/Standard and Abaqus/Explicit. It is shown in *Figure 4*, with the pad drawn in a darker gray than the disc. The disc is a simple annulus with a thinner inner ring. This mesh is also rather coarse with only three elements through the thickness of the disc and three elements through the pad. The elements on the contact sides are thinner since they will be in the areas of higher thermal gradients. There are 36 elements in the circumferential direction of the disc.

Material properties

The thermomechanical properties for the axisymmetric model were taken from a paper by Day and Newcomb (1984) describing the analysis of an annular disc brake. The pad is made of a resin-bonded composite friction material, and the disc is made of steel. Although Day and Newcomb note that material changes occur in the pad material because of thermal degradation, the pad in the axisymmetric model has the properties of the unused pad material. For the axisymmetric model the modulus, density, conductivity, and coefficient of friction are divided by 18 since the pads actually cover only a 20° section of the disc, even though they are modeled as being smeared around the entire circumference.

The pad for the three-dimensional model is also a resin-bonded composite friction material whose thermomechanical properties are listed in *Table 1* and coefficient of friction is listed in *Table 2*. The properties were taken from a paper by Day (1984). It is noted that above certain temperatures, approximately 400°C, the pad material becomes thermally degraded and μ is assumed constant from this point on.

It is assumed that all the frictional energy is dissipated as heat and distributed equally between the disc and the pad; therefore, the fraction of dissipated energy caused by friction that is converted to heat is set to 1.0, and the default distribution is used. This fractional value allows the user to specify an unequal distribution, which is particularly important if the heat conduction across the interface is poor. In this example the conductivity value specified with the gap conductance is quite high; hence, the results are not very sensitive to changes in distribution. In Abaqus/Explicit arbitrarily high gap conductivity values may cause the stable time increment associated with the thermal part of the problem to control the time incrementation, possibly resulting in a very inefficient analysis. In this problem the gap conductivity value used in the Abaqus/Explicit simulation is 20 times smaller than the one used in the Abaqus/Standard simulation. This allows the stable time increment associated with the mechanical part of the problem to control the time incrementation, thus permitting a more efficient solution while hardly affecting the results.

Loading

The pads of the axisymmetric model are first pressed against the disc. The magnitude of the load is divided by 18 since the pads are not actually axisymmetric. The frictional forces are then applied through user subroutine *FRIC* to simulate a linear decrease in velocity of the disc relative to the pads. The braking is done over three steps; then, when the velocity is zero, a final step shows the continued heat conduction through the model.

The pad of the three-dimensional model is fixed in the nonaxial degrees of freedom and is pressed against the disc with a distributed load applied to the back of the pad. In Abaqus/Standard the disc is then rotated by 60°

using an applied boundary condition to the center ring. In Abaqus/Explicit this boundary condition is prescribed using smooth step data to minimize the effects of centrifugal forces at the beginning and end of the step. Frictional forces between the surfaces generate heat in the brake.

The initial temperature of both models is 20°C.

Solution controls (Abaqus/Standard only)

Since the three-dimensional model has a small loaded area and, thus, rather localized forces and heat fluxes, the default averaged flux values for the convergence criteria produce very tight tolerances and cause more iteration than is necessary for an accurate solution. To decrease the computational time required for the analysis, the solution controls override the automatic calculation of the average forces and heat fluxes. Solution controls are first used to set parameters for the displacement field and warping degrees of freedom equilibrium equations. The convergence criterion ratio is set to 1%, and the time-average and average fluxes are set to a typical nodal force (displacement flux):

$$f_d = pA = (1.7\,\mathrm{E6})\ (1.77\mathrm{E\text{-}4}) \approx 300,$$

where p is the pressure and A is the area of a typical pad element. Solution controls are next used to set parameters for the temperature field equilibrium equations. The convergence criterion ratio is set to 1%, and the time-average and average fluxes are set to the nodal heat flux (temperature flux) for a typical pad element. The heat flux density generated by an interface element due to frictional heat generation is $q_g = \eta \tau v$, where η is the gap heat generation factor, τ is the frictional stress, and v is the velocity. Therefore, the nodal heat flux is

$$f_t = q_g A = \eta\,(\mu p)\ (\omega r)\,A,$$

where A is the contact area of a typical pad element, μ is the friction coefficient, and p is the contact pressure.

The angular velocity, ω, is obtained as the total rotation, $\pi/3$, divided by the total time, 0.015 sec. The radius, r, is set to 0.120 m, which is the distance from the axis to a point approximately in the middle of the pad surface. This yields

$$f_t = \frac{(1.0)(.37)(1.7\,\mathrm{E6})\left(\frac{\pi}{3}\right)(0.12)(1.77\mathrm{E\text{-}4})}{0.015} \approx 900.$$

Additional solution controls can reduce the solver cost for an increment by improving the initial solution guess, solving thermal and mechanical equations separately, and reducing the wavefront of three-dimensional finite-sliding contact analysis. These features are discussed below. The impact of combining these features is also discussed.

When the default convergence controls are used, it is possible to obtain faster convergence with a parabolic extrapolation step. For the three-dimensional model the use of this feature yields a 14% enhancement in computational speed per increment.

The coupling between the thermal and mechanical fields in this problem is relatively weak. It is, therefore, possible to obtain a more efficient solution by specifying separate solutions for the thermal and mechanical equations each increment. This technique results in faster per-iteration solution times at the expense of poorer convergence when a strong interfield coupling is present. Use of this technique also permits the use of the symmetric solver and storage scheme. The resulting symmetric approximation of the mechanical equations was also found to be cost effective for this problem, when combined with a quality initial solution guess obtained by specifying parabolic extrapolation in the step. Neither of these approximations impacts solution accuracy. For the three-dimensional model the use of the separated solution scheme, parabolic extrapolation, and symmetric matrix storage yields a 50% decrease in the total solution time.

In the three-dimensional model the deformable main surface is defined from a large number of connecting elements resulting in a large wavefront. By default, Abaqus/Standard employs an automated contact patch algorithm to reduce the wavefront and solution time. For instance, in the coupled thermomechanical analysis a substantial savings in solution time (a 30% to 50% decrease) is obtained when the automatic contact patch

algorithm is employed compared to an analysis that uses a fixed contact patch encompassing the entire main surface. The reduction in solution time is system dependent and depends on several factors, such as CPU type, system memory, and IO speed. This solution time savings is in addition to any of the other savings discussed in this section. The additional savings is, therefore, realized when the separated solution scheme and parabolic extrapolation are also specified.

Results and discussion

The temperature distribution of the axisymmetric model at an early time increment is shown in *Figure 5*. The temperature is greatest at the interfaces between the disc and pads, and the heat has just started to conduct into the disc. *Figure 6* shows the temperature distribution at the end of the analysis when the velocity is zero. The heat has conducted through the friction ring of the disc. *Figure 7* is a displaced plot of the model at the end of the analysis and shows the characteristic conical deformation due to thermal expansion. The displacement has been magnified by a factor of 128 to show the deformation more clearly.

The temperature distribution of the disc surface of the three-dimensional model after a rotation of 60° is shown in *Figure 8* (Abaqus/Standard) and *Figure 9* (Abaqus/Explicit). The agreement between the two results is excellent. The hottest region is the area under the pad, while the heat in the regions that the pad has passed over has dissipated somewhat. *Figure 10* shows the temperature distribution of the inside of the brake pad predicted by Abaqus/Standard, while *Figure 11* shows the same result obtained with Abaqus/Explicit. Again excellent agreement between the two results is noted. *Figure 12* shows the temperature distribution in the disc predicted by Abaqus/Standard with the thickness magnified by a factor of 20. The heat has conducted into the disc in the regions that the pad has passed over.

The stresses predicted by Abaqus/Standard do not account for the effects of centrifugal loads (fully coupled thermal-stress is a quasi-static procedure), while the stresses predicted by Abaqus/Explicit do. These effects can be significant, especially during the early transient portion of the simulation when the initially stationary disc is brought up to speed. To compare the stress results between Abaqus/Standard and Abaqus/Explicit, we gradually initiated and ended the disc rotation in the Abaqus/Explicit simulation; thus, in Abaqus/Explicit, the centrifugal stresses at the beginning and end of the step are small compared with the thermal stresses. At points in between, however, the effects of centrifugal loading are more pronounced and differences between the stress states predicted by Abaqus/Standard and Abaqus/Explicit are observed. The overall effect on the thermal response, however, is negligible.

The Abaqus/Explicit analysis did not include mass scaling because its presence would artificially scale the stresses due to the centrifugal loads. It is possible to include mass scaling to make the analysis more economical, but any results obtained with mass scaling must be interpreted carefully in this problem.

Input files

Abaqus/Standard input files

discbrake_std_cax3t.inp	Axisymmetric model with CAX3T elements.
discbrake_std_cax3t.f	User subroutine *FRIC* used in discbrake_std_cax3t.inp.
discbrake_std_cax4t.inp	Axisymmetric model with CAX4T elements.
discbrake_std_cax4t.f	User subroutine *FRIC* used in discbrake_std_cax4t.inp.
discbrake_std_cax4rt.inp	Axisymmetric model with CAX4RT elements.
discbrake_std_cax4rt_surf.inp	Axisymmetric model with CAX4RT elements using the surface-to-surface approach.
discbrake_std_cax4rt.f	User subroutine *FRIC* used in discbrake_std_cax4rt.inp.
discbrake_3d.inp	Three-dimensional model.
discbrake_postoutput.inp	**POST OUTPUT* analysis of the three-dimensional model.

discbrake_3d_extrapara.inp	Three-dimensional model with the second step run with **STEP*, EXTRAPOLATION=PARABOLIC and with the default **CONTROLS* option.
discbrake_3d_extrapara_300c.inp	Three-dimensional model with the second step run with **STEP*, EXTRAPOLATION=PARABOLIC. It is assumed that several revolutions occurred and the initial temperature for the disc brake and pad is 300°C.
discbrake_3d_separated.inp	Three-dimensional model run using the **SOLUTION TECHNIQUE*, TYPE=SEPARATED option.

Abaqus/Explicit input file

discbrake_3d_xpl.inp	Three-dimensional model.

References

- Day, A. J., "An Analysis of Speed, Temperature, and Performance Characteristics of Automotive Drum Brakes," *Journal of Tribology*, vol. 110, pp. 295–305, 1988.
- Day, A. J., and T. J. Newcomb, "The Dissipation of Frictional Energy from the Interface of an Annular Disc Brake," *Proc. Instn. Mech. Engrs*, vol. 198D, no. 11, pp. 201–209, 1984.
- Gonska, H. W., and H. J. Kolbinger, "Abaqus Application Example: Temperature and Deformation Calculation of Passenger Car Brake Disks," *Abaqus Users' Conference Proceedings*, 1993.

Tables

Table 1: Thermomechanical properties.

Temperature of property measurement (°C)	20	100	200	300
Young's modulus, E (N/mm^2)	2200	1300	530	320
Poisson's ratio, ν	0.25	0.25	0.25	0.25
Density, ρ (kg/m^3)	1550	1550	1550	1550
Thermal expansion coefficient (K^{-1})	10×10^{-6}	–	30×10^{-6}	–
Thermal conductivity, κ (w/mK)	0.5	0.5	0.5	0.5
Specific heat, C_p (J/kgK)	1200	1200	1200	1200

Table 2: Brake lining temperature characteristic.

Temperature of property measurement (°C)	100	200	300	400
Friction coefficient, μ	0.38	0.41	0.42	0.24

Figures

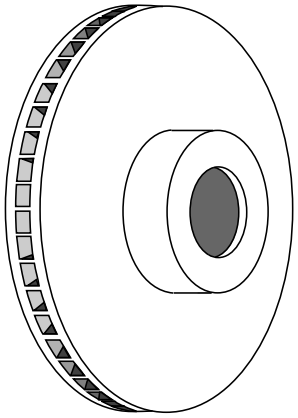


Figure 1: A vented brake disc design.

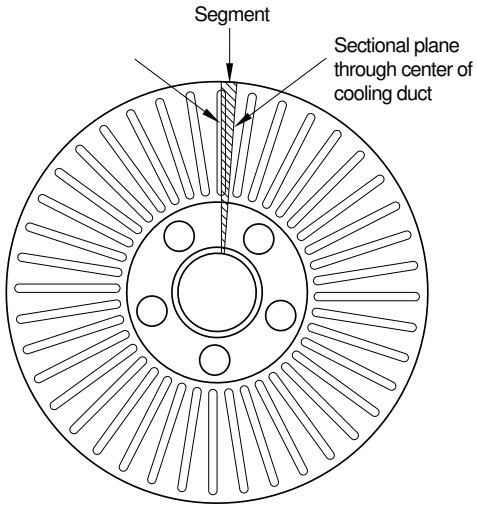


Figure 2: Modeling a segment of a brake disc.

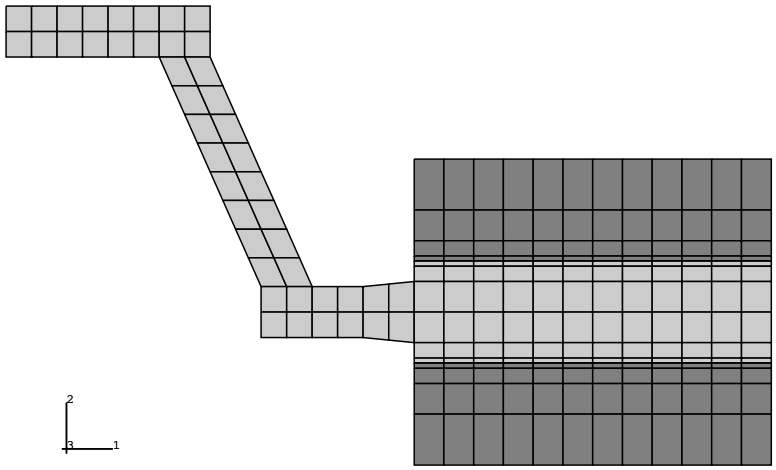


Figure 3: Mesh for the axisymmetric model, Abaqus/Standard.

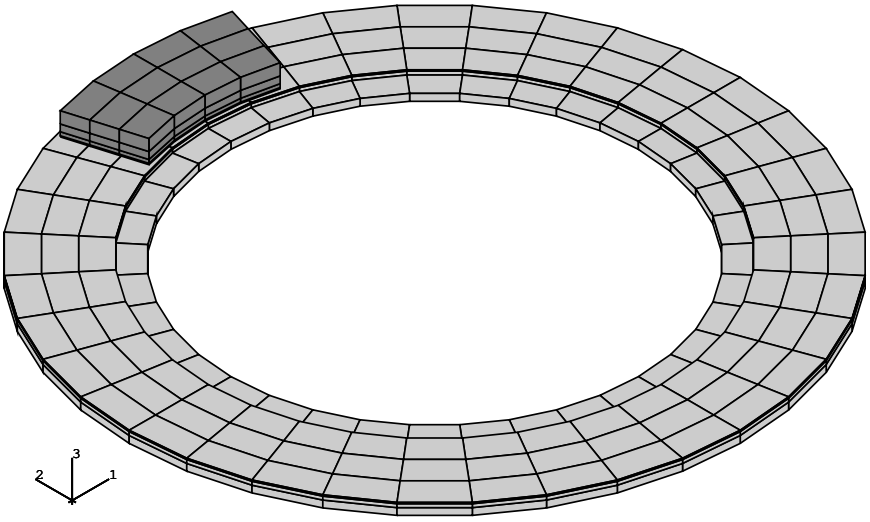


Figure 4: Mesh for the three-dimensional model.

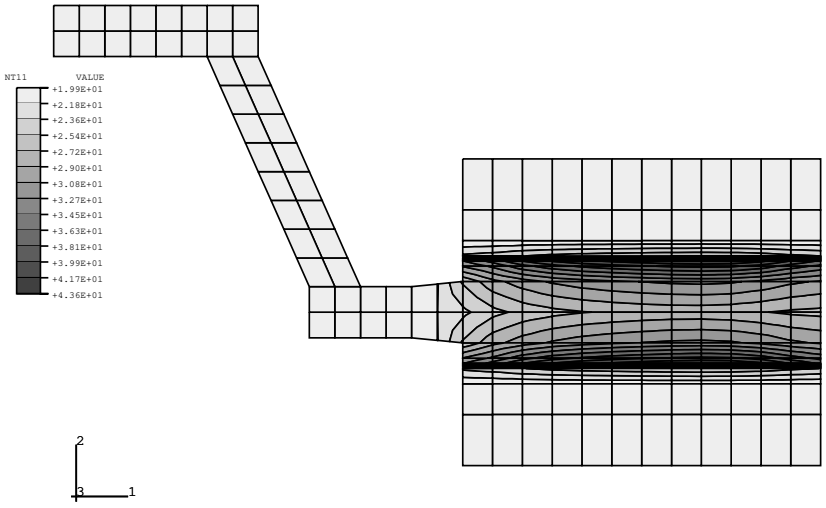


Figure 5: Isotherms of the axisymmetric model at $t = 0.675$, Abaqus/Standard.

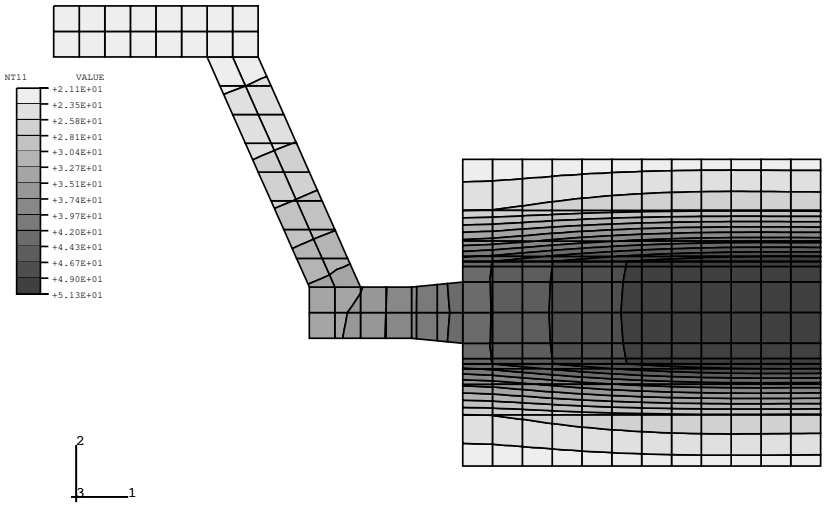


Figure 6: Isotherms of the axisymmetric model when braking has ended, Abaqus/Standard.

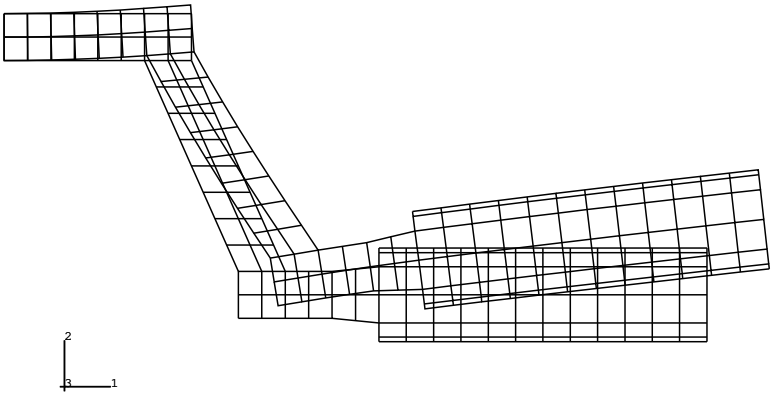


Figure 7: Deformation of the axisymmetric disc, displacement magnified by 128, Abaqus/Standard.

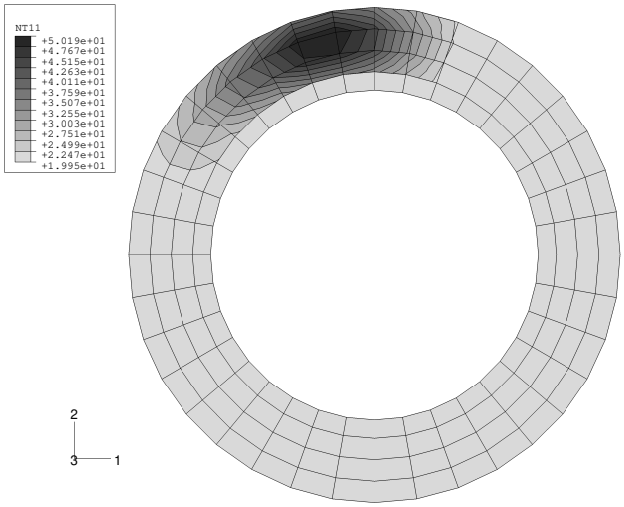


Figure 8: Isotherms of the disc surface, Abaqus/Standard.

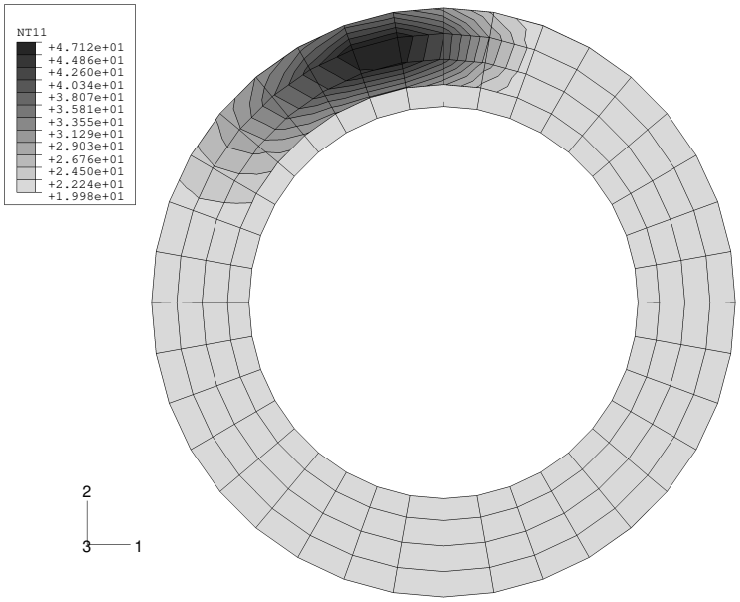


Figure 9: Isotherms of the disc surface, Abaqus/Explicit.

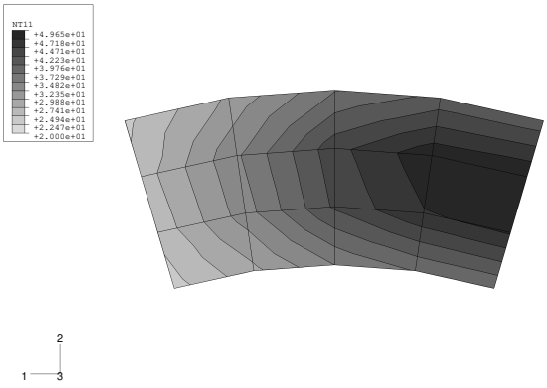


Figure 10: Isotherms of the inside of the brake pad, Abaqus/Standard.

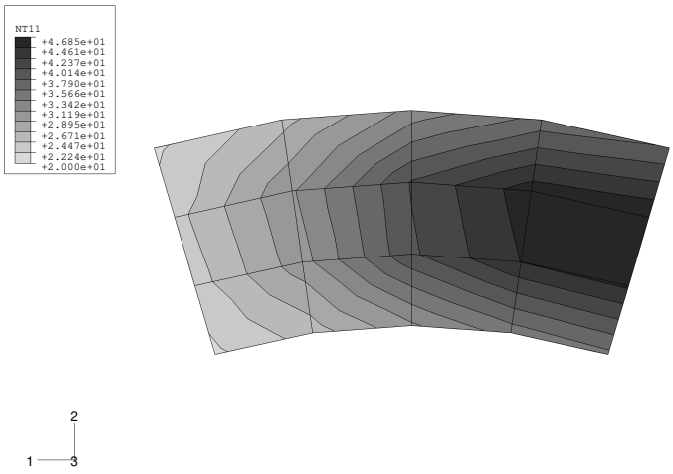


Figure 11: Isotherms of the inside of the brake pad, Abaqus/Explicit.

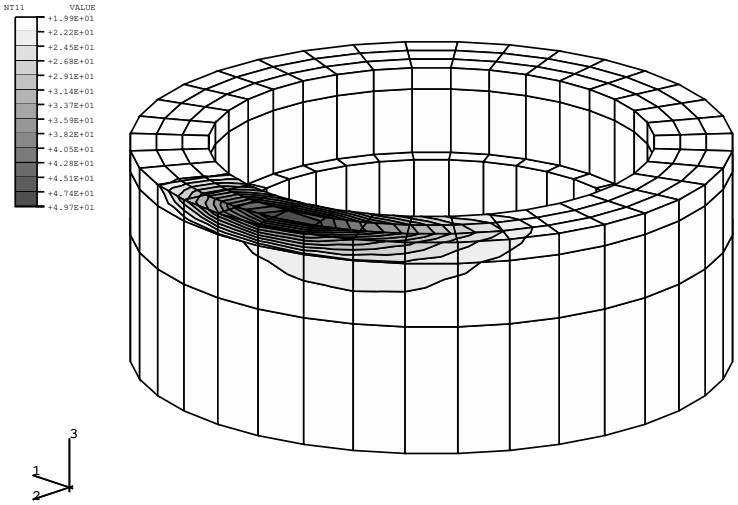


Figure 12: Isotherms of the disc with the thickness magnified 20 times, Abaqus/Standard.

A sequentially coupled thermomechanical analysis of a disc brake with an Eulerian approach

Products: Abaqus/Standard

Overview

This example shows the sequentially coupled thermomechanical analysis of a disc brake using an Eulerian approach.

The prediction of fatigue and failure of a disc brake system is fundamental in assessing product performance. Disc brakes operate by pressing a set of brake pads against a rotating disc. The friction between the pads and the disc causes deceleration. The brake system then converts the kinetic energy of vehicle motion into heat. Severe temperature changes as well as mechanical loadings cause inelastic deformation and circumferential tensile stress in the disc, which may eventually lead to the failure of the disc.

The traditional way of analyzing this kind of problem is to use a Lagrangian approach in which the mesh used to discretized the disc rotates relative to the brake assembly. Since many revolutions are typically required to reach the state of interest to the analyst, this approach is prohibitively expensive and cumbersome. The steady-state transport analysis capability in Abaqus/Standard (*Steady-State Transport Analysis*), which uses the Eulerian method in which the finite element mesh of the disc does not rotate relative to the brake assembly but the material "flows" through the mesh, provides a cost-effective alternative approach. The paths that the material points follow through the mesh are referred to as streamlines. This kinematic description converts the moving disc brake problem into a pure spatially dependent simulation. Thus, the mesh has to be refined only in a fixed region where the brake pads are in contact with the disc initially.

Geometry and model

The model analyzed in this example is a solid disc. An axisymmetric model is created to define the cross-sectional geometry of the disc, as shown in *Figure 1*. The disc has a thicker friction ring connected to a conical section that, in turn, connects to an inner hub. The inner radius of the friction ring is 86.5 mm, the outer radius is 133.0 mm, and the ring is 13.0 mm thick. The conical section is 27.2 mm deep. The inner radius of the conical section is 64.75 mm, the outer radius is 86.5 mm, and the section is 6.4 mm thick. The conical section has a thinner section out to a radius of 71.25 mm, which has a thickness of 4.5 mm. The hub has an inner radius of 33.0 mm, an outer radius of 71.25 mm, and is 6.2 mm thick. Symmetric model generation (*Symmetric Model Generation*) is used to create a three-dimensional disc, as shown in *Figure 2*, by revolving the two-dimensional cross-section about the symmetry axis and to create the streamlines needed for the steady-state transport analyses in this example. There are eight elements through the thickness of the friction ring, four elements through the thickness of the hub, and four elements through the conical section. There are 40 element sectors in the circumferential direction of the disc, with a more refined mesh used in the region with higher thermal and stress gradients. The model consists of 9440 first-order forced convection/diffusion bricks (DCC3D8) in the heat transfer analysis, giving a total of about 11520 degrees of freedom; and it consists of 9440 first-order bricks (C3D8) in the subsequent steady-state transport analysis, giving a total of about 34560 degrees of freedom.

The disc pads are not modeled in the example. Instead, the thermal and mechanical interactions between the disc and the pads are represented by the application of appropriate distributed heat fluxes in the heat transfer analysis and by the application of appropriate concentrated loads in the steady-state transport analysis, respectively.

Material properties

The disc is made of metallic material, with a Young's modulus of 93.5 GPa, a yield stress of 153 MPa, a Poisson's ratio of 0.27, and a coefficient of thermal expansion of 11.7×10^{-6} per °C at room temperature. In this example

the dissipation of the frictional heat-generated temperature fluctuates, ranging from a minimum value of 40°C to a maximum value of 560°C over the entire braking cycle. The temperature distribution when the disc is heated to its peak value is shown in *Figure 3*. Under such operating conditions plastic deformation, as well as creep deformation, is observed. The two-layer viscoelastic-elastoplastic model, which is best suited for modeling the response of materials with significant time-dependent behavior as well as plasticity at elevated temperatures, is used to model the disc (see *Two-Layer Viscoplasticity*). This material model consists of an elastic-plastic network that is in parallel with an elastic-viscous network. The Mises metal plasticity model with kinematic hardening is used in the elastic-plastic network, and the power-law creep model with strain hardening is used in the elastic-viscous network. Because the elastic-viscoplastic response of the material varies greatly over this temperature range, temperature-dependent material properties are specified.

The thermal properties for the disc are temperature dependent with a conductivity of 51×10^{-3}W/mm per °C, a specific heat of 501 J/kg per °C, and a density of 7.15×10^{-6} kg/mm at room temperature.

Problem description and loading

A simulation of braking a solid disc rotating initially at an angular velocity of 155.7 rad/sec is performed. The braking time is approximately 5 seconds, followed by a cooling period of 600 seconds. A sequentially coupled thermomechanical analysis is performed on the solid disc using the Eulerian approach: a forced convection/diffusion heat transfer analysis is followed by a steady-state transport analysis. Heat fluxes with film condition and prescribed mass flow velocity through user subroutine *UMASFL* are applied to the thermal model, which consists of three steps. The first step, which lasts 0.2 seconds, simulates the response of the disc under constant distributed fluxes and a constant angular velocity. The second step involves 4.8 seconds during which the distributed fluxes and the angular velocity are decreased linearly to small values near zero at the end of the step. The final step, which lasts 600 seconds, simulates the continued cooling in the model. The resulting temperatures obtained during the heat transfer analysis are applied to the subsequent mechanical analysis, which involves five steady-state transport analysis steps.

The purpose of the first step in the mechanical analysis is to obtain a steady-state solution for a disc under constant concentrated loads due to the application of the brake pads to the disc. There is only one increment in this step. A constant temperature of 40°C is used, and a constant angular transport velocity of 155.7 rad/sec is specified.

The second step obtains a series of quasi-steady-state transport solutions under different temperature loading passes through the disc. The pass-by-pass steady-state transport analysis technique is used for this purpose. This step lasts 0.2 seconds with a constant angular velocity of 155.7 rad/sec throughout the entire step. Several increments are involved, with each increment corresponding to a complete temperature loading pass through the disc. The temperature values obtained during the first step of the heat transfer analysis are read into this step.

The third step also obtains a series of quasi-steady-state transport solutions under different temperature loading passes through the disc. However, this step involves 4.8 seconds over which the angular velocity is decreased linearly from 155.7 rad/sec at the beginning of the step to a small value close to zero at the end of this step. There are several increments in this step, with each increment corresponding to a complete temperature loading pass through the disc. The temperature values obtained during the second step of the heat transfer analysis are read into this step.

The fourth step obtains a steady-state solution for the disc due to the removal of the concentrated loads. There is only one increment in this step.

The last step obtains a series of quasi-steady-state transport solutions when the disc cools down. There are several increments over a step period of 600 seconds. The temperatures obtained during the last step of the heat transfer analysis are read into this step. Since the angular velocity is very small, this step essentially simulates a long-term elastic-plastic response for the disc.

Solution controls

Since the modified Newton's method is used in a steady-state transport analysis, more numerical iterations are necessarily required to obtain a converged solution. To decrease the computational time required for the analysis due to the unnecessary cutback of the increment size, the time incrementation control parameters are used to override the default values (see *Time Integration Accuracy in Transient Problems*).

Results and discussion

One of the considerations in the design of a disc brake system is the stress distribution and deformation in the region where the brake pads are applied. Circumferential tensile stress, which may cause the fracture of the disc, will develop, making this region critical in the design. The results shown in *Figure 6* through *Figure 11* are measured in this region (element 7817, integration point 5; see point A in *Figure 2*). The temperature in this region (node 7820) is shown in *Figure 4* as a function of the time over the entire braking process. *Figure 5* shows the von Mises stress distribution just before the distributed loads are removed and the cooling period starts (Step 3, increment 60).

Figure 6, *Figure 7*, and *Figure 8* show the evolution of the circumferential stress, circumferential plastic strain, and circumferential viscous strain, respectively, as a function of the time throughout a complete braking cycle. A tensile stress of 54 MPa is developed after the disc is cooled down completely. Both plastic strain and viscous strain reach their saturation levels during the cooling period. The time evolution of the circumferential stress versus the circumferential plastic strain, shown in *Figure 9*, is obtained by combining *Figure 6* with *Figure 7*. Similarly, the time evolution of the circumferential stress versus the circumferential viscous strain, shown in *Figure 10*, is obtained by combining *Figure 6* with *Figure 8*. The shapes of the stress-strain curves represent the plastic and viscous energies dissipated, respectively, over an entire braking cycle. These dissipated energies, which could be used to predict the fatigue life for the disc, are shown in *Figure 11* as a function of the time.

Acknowledgements

SIMULIA gratefully acknowledges PSA Peugeot Citroën and the Laboratory of Solid Mechanics of the Ecole Polytechnique (France) for their cooperation in developing the Eulerian algorithm for steady-state transport analysis and for supplying the geometry and material properties used in this example.

Input files

discbrake_sst_heat_axi.inp	Axisymmetric model for the heat transfer analysis.
discbrake_sst_heat_symm.inp	Three-dimensional model for the heat transfer analysis.
exa_discbrake_sst_heat_symm.f	User subroutine *UMASFL* used in discbrake_sst_heat_symm.inp.
discbrake_sst_axi.inp	Axisymmetric model for the mechanical analysis.
discbrake_sst_symm_pbp.inp	Three-dimensional model for the mechanical analysis.

References

- Maitournam, M. H., “Formulation et Résolution Numérique des Problèmes Thermoviscoplastiques en Régime Permanent,” *Thèse de l’Ecole des Ponts et Chaussées*, 1989.
- Nguyen-Tajan, T. M.L., “Modélisation Thermomécanique des Disques de Frein par une Approche Eulérienne,” *Thèse de l’Ecole Polytechnique*, 2002.

Figures

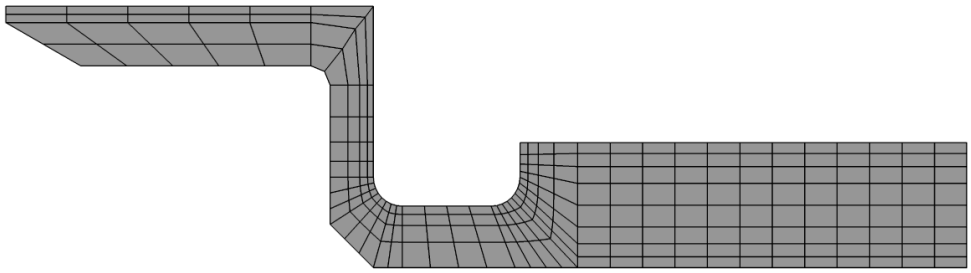


Figure 1: Mesh for the axisymmetric model.

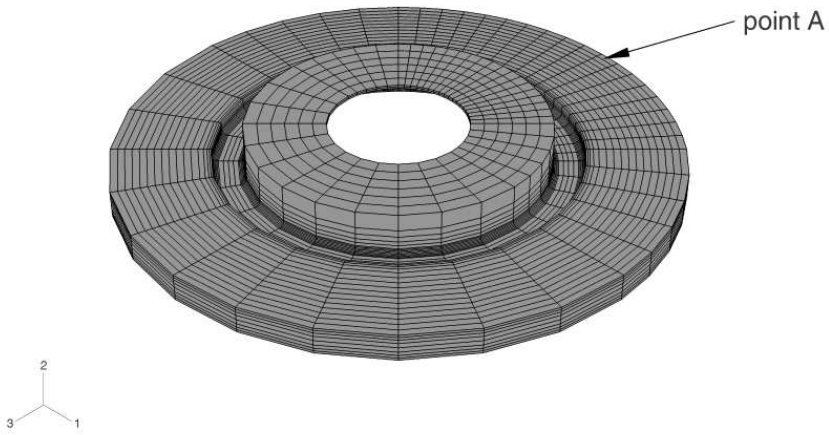


Figure 2: Mesh for the three-dimensional model.

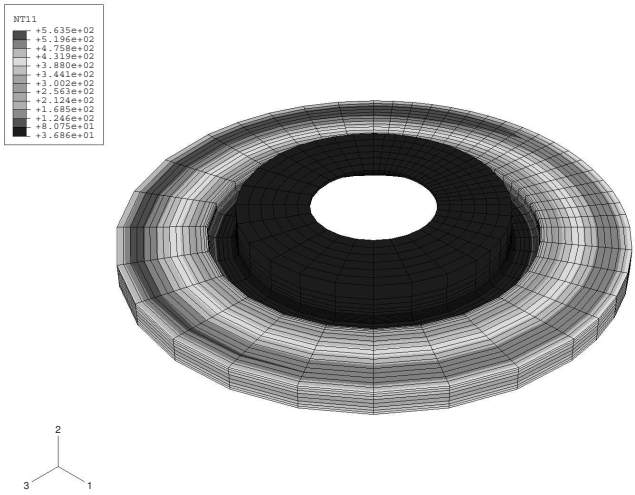


Figure 3: Temperature distribution when the disc is heated to its peak value.

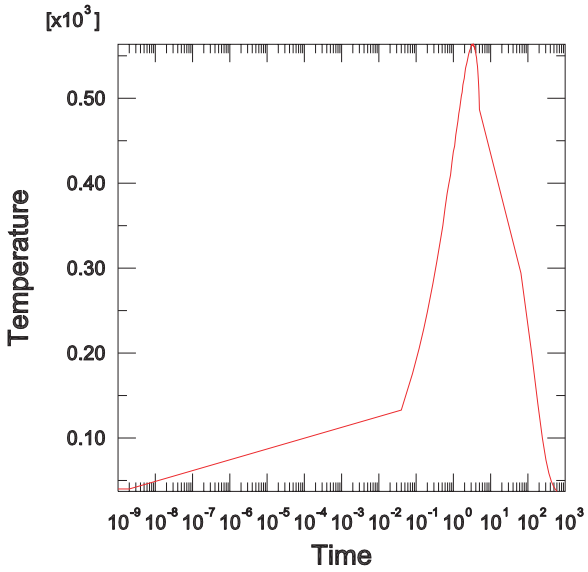


Figure 4: Temperature at node 7820 as a function of time during the entire braking period.

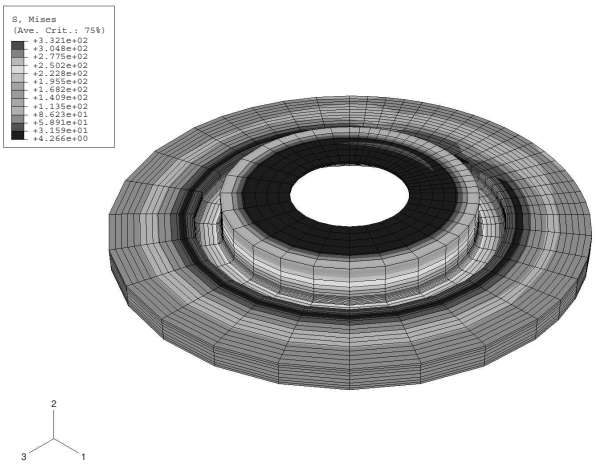


Figure 5: Von Mises stress distribution (Step 3, increment 60) just before the cooling period starts.

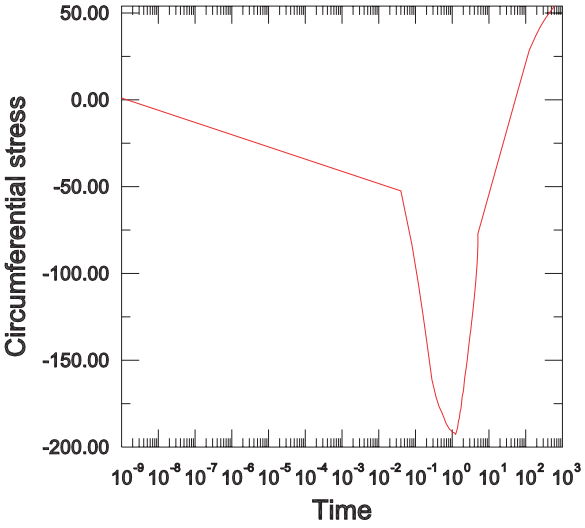


Figure 6: Evolution of the circumferential stress as a function of time.

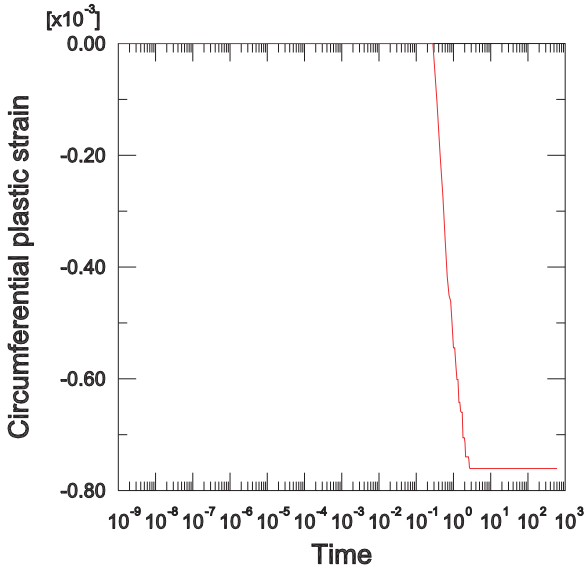


Figure 7: Evolution of the circumferential plastic strain as a function of time.

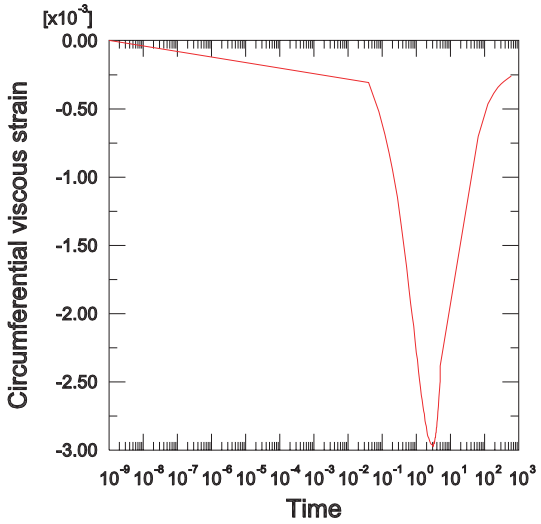


Figure 8: Evolution of the circumferential viscous strain as a function of time.

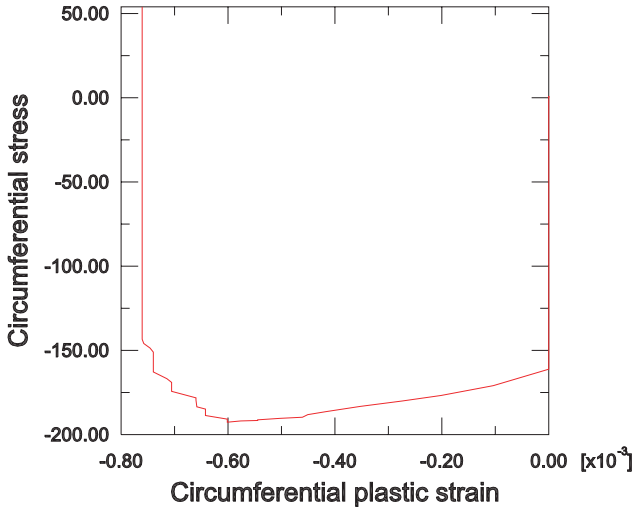


Figure 9: Evolution of the circumferential stress versus the circumferential plastic strain.

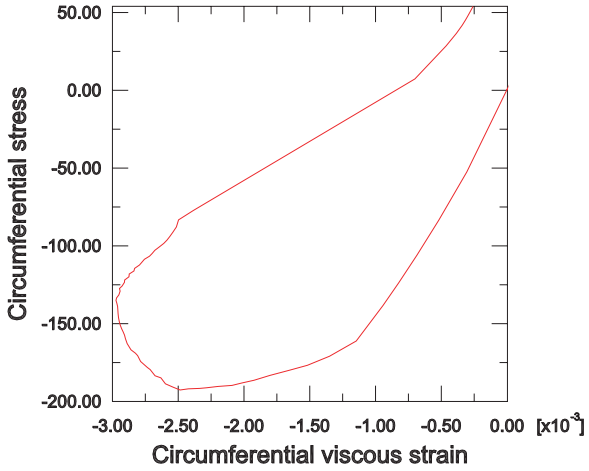


Figure 10: Evolution of the circumferential stress versus the circumferential viscous strain.

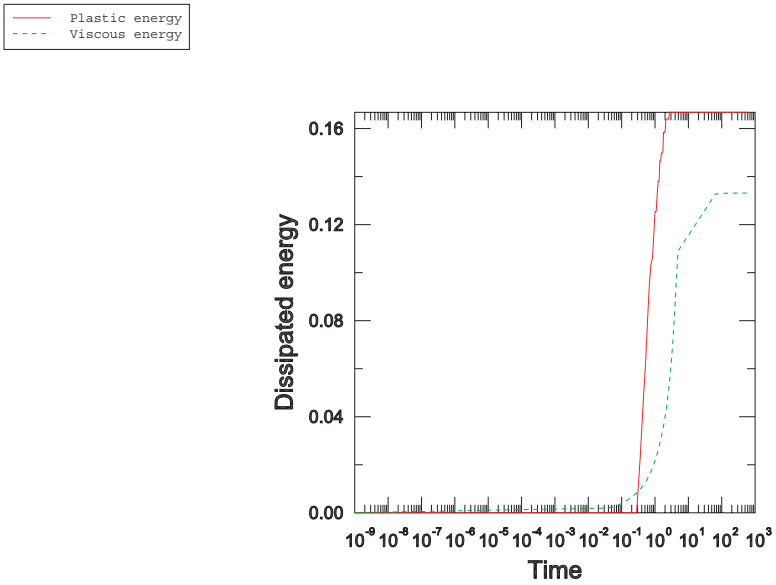


Figure 11: Evolution of the plastic dissipated energy and the viscous dissipated energy as a function of time.

Exhaust manifold assemblage

Products: Abaqus/Standard

Overview

This example shows how to model varying boundary constraints with the prescribed assembly load capability of Abaqus.

Engine exhaust manifolds are commonly subject to severe thermal cycles during operation and upon shutdown. Thermal expansion and contraction of the manifold is constrained by its interaction with the engine head to which it is bolted. These constraints govern the thermomechanical fatigue life of the manifold.

The initial assembly procedure consists of bolting the flanges of the manifold to the engine head with prescribed bolt forces that produce uniform axial bolt stresses. Under subsequent operating conditions such as thermal cycling and creep, these bolt forces may increase or relax, possibly changing normal pressures and resulting in lateral slippage between the engine head and the manifold flanges. Thus, the boundary constraints on the manifold flanges are a function of the response of the entire assembly to its operating conditions. As such, these boundary constraints cannot be prescribed *a priori*. This example shows how to simulate these varying boundary constraints with the prescribed assembly load capability of Abaqus.

The problem scenario consists of three steps:

1. Apply prescribed bolt loads to fasten the exhaust manifold to the engine head.
2. Subject the assembly to the steady-state operating temperature distribution.
3. Return the assembly to ambient temperature conditions.

Geometry and model

The exhaust manifold assemblage being analyzed is depicted in *Figure 1*. It consists of a four tube exhaust manifold with three flanges, bolted with seven bolts to a small section of the engine head.

The manifold is cast from gray iron with a Young's modulus of 138 GPa, a Poisson's ratio of 0.283, and a coefficient of thermal expansion of 13.8×10^{-6} per °C. In this example the region of the manifold where the hot exhaust gases converge is subject to temperatures ranging from an initial value of 300 K to an extreme of 980 K. The elastic-plastic response of gray cast iron varies greatly over this range of temperatures, so the temperature-dependent plasticity curves shown in *Figure 2* are used for the manifold material. Gray cast iron exhibits different behavior in tension and compression; therefore, these curves represent the average response. The Mises metal plasticity model with isotropic hardening is used. The three manifold flanges contain a total of seven bolt holes. The 9.0 mm diameter of these bolt holes is slightly greater than the 8.0 mm diameter of the bolt shanks to allow for some unobstructed lateral motion of the manifold.

For simplicity, only a portion of engine head directly beneath the manifold flanges is modeled. The head is made from aluminum, with a Young's modulus of 69 GPa, a Poisson's ratio of 0.33, and a coefficient of thermal expansion of 22.9×10^{-6} per °C. The head has four exhaust ports leading into the manifold tubes. It has seven bolt holes used to secure the manifold.

Seven bolts fasten the manifold to the head. The bolts are made from steel, with a Young's modulus of 207 GPa, a Poisson's ratio of 0.3, and a coefficient of thermal expansion of 13.8×10^{-6} per °C. The bolt shanks have a diameter of 8 mm. The bolt head diameters are 16 mm.

Three-dimensional, deformable-to-deformable, small-sliding contact conditions apply to the model. The bottoms of the bolt heads form contact bearing surfaces, with the top surfaces of the manifold flanges lying directly

beneath them. In addition, the bottoms of the manifold flanges form contact bearing surfaces with the top of the engine head. Each of these surfaces is defined in Abaqus with a surface definition. Respective mating surfaces are paired together with contact pairs. Normal pressures will be transmitted through these contact pairs as a result of the bolt tightening forces in Step 1. The forces carried by the bolts will vary as they respond to the thermal cycling of the assembly in subsequent steps. These fluctuations in bolt loads will result in varying normal pressures transmitted across the contact pairs. Lateral slip of the mating components will occur if the critical frictional shear stress limit is surpassed by lateral forces developed in the system. A friction coefficient of 0.2 is used between all contacting surfaces. Contact conditions are not necessary between the bolt shanks and the holes in the manifold flanges because of the design clearance between them. Contact between the bolt shanks and the holes in the engine head is not modeled.

All three structural components (manifold, head, and bolts) are modeled with three-dimensional continuum elements. The model consists of 7450 first-order brick elements with incompatible deformation modes, C3D8I, and 282 first-order prism elements, C3D6. The C3D6 elements are used only where the complex geometry precludes the use of C3D8I elements. The C3D8I elements are selected to represent the bending of the manifold walls with only one element through the thickness of the tube walls.

Loading and boundary constraints

It is assumed that the engine head is securely fixed to a stiff and bulky engine block, so the nodes along the base of the head are secured in the direction normal to the base (the global x-direction) but are free to move in the two lateral directions to account for thermal expansion. It is also assumed that the bolts are threaded tightly into the engine head, with the bolt threads beginning directly beneath the section of engine head modeled. Therefore, the nodes at the bottom of the bolt shanks are shared with the nodes of the surrounding engine head elements and are also secured in the global x-direction. The manifold flanges are sandwiched between the top of the engine head and the base of the bolt heads using contact pairs. The line of action of the bolt forces (bolt shank axes) is along the global x degree of freedom. Soft springs acting in the global y- and z-directions are attached to the outlet end of the manifold and to the two ends of the head to suppress rigid body motions of the manifold and head, respectively. These springs have no influence on the solution.

In the first step of the analysis each of the seven bolts is tightened to a uniform bolt force of 20 kN. In subsequent steps the variation of the bolt loads is monitored as the bolts respond to the thermal loading on the assembly as a whole. The “prescribed assembly load” capability of Abaqus is used. For each bolt we define a “cut,” or pre-tension section, and subject the section to a specified tensile load. As a result, the length of the bolt at the pre-tension section will change by the amount necessary to carry the prescribed load, while accounting for the compliance of the rest of the system. In the next step the prescribed bolt loads are replaced by the condition that the length changes calculated in the previous step remain fixed. The remainder of the bolt is free to deform.

The same procedure is used for all seven bolts. First, pre-tension sections are defined as “cuts” that are perpendicular to the bolt shank axes by defining surfaces on the faces of a group of elements within each bolt shank, as shown in *Figure 3*. The line of action of the bolt force is in the direction that is normal to this surface. Next, each bolt is assigned an arbitrary, independent node that possesses one degree of freedom (DOF 1), to which the bolt force will be applied. These nodes are called the “pre-tension nodes” (all seven bolt pre-tension nodes are placed into a node set named `BOLTS`). The spatial position of a pre-tension node is irrelevant. Finally, each surface is associated with the appropriate pre-tension node using a pre-tension section.

A portion of the Abaqus model definition section defining the pre-tension section is shown below:

```
*ELSET, ELSET=BCUT1, GENERATE
19288,19307
*SURFACE, NAME=BOLT1
BCUT1,S2
*NODE, NSET=BOLTS
99991,   21.964    ,  -139.80    ,  -12.425
…
99997,   21.964    ,   137.38    ,  -12.226
*PRE-TENSION SECTION, SURFACE=BOLT1, NODE=99991
```

In Step 1 of the analysis a concentrated clamping load of 20 kN is applied to each of the pre-tension nodes in node set BOLTS. In Step 2 the concentrated load from Step 1 is removed and replaced by a "fixed" boundary condition that will hold the pre-tension section length changes from Step 1 fixed. Over the course of a step in which a load is replaced by a boundary condition, CF1 is ramped down, while RF1 is ramped up to replace it. Therefore, the total force across the bolt is the sum of the concentrated force (CF1) and the reaction force (RF1) on the pre-tension node. This total force is available as TF1. Additionally in this step of the analysis nodal temperatures depicting the steady-state temperature distribution in the manifold are read from an external file. The temperature distribution is shown in *Figure 4*. These nodal temperatures can be generated by an Abaqus heat transfer analysis. Each of the nodes in the model has its temperature ramped up from the initial ambient temperature of 300 K to its final steady-state temperature. These nodal temperatures are interpolated to the element integration points so that the correct temperature-dependent plasticity data can be used in the constitutive calculations. Finally, in Step 3 the nodal temperatures are ramped back down to the initial ambient temperature of 300 K.

Results and discussion

The analysis is performed as a small-displacement analysis. The nonlinearities in the problem are the result of changing contact conditions, frictional slip and stick, and temperature-dependent plasticity.

Figure 5 shows the lateral displacement of the bottom surface of the flange at the end of the heat-up step. As a result of frictional sticking, the ends of the two outer manifold flanges have expanded outward relative to one another by only about 0.75 mm. Plastic yielding conditions result since thermal expansion of the remainder of the manifold is constrained by this limited lateral flange motion. A separate thermal-stress analysis of the manifold only, with no bolt constraints included, produced relative lateral expansions of about 1.1 mm and very little plasticity.

Figure 6 is a plot of the forces carried by each of the seven bolts throughout the load history. This plot can be obtained with the *X*–*Y* plotting capabilities in Abaqus/CAE. The curves contain the values of the total forces (TF1) for the pre-tension nodes in node set BOLTS. The loads carried by the bolts increase significantly during the heat-up step. The loads do not return precisely to the original bolt load specification upon cool down because of the residual stresses, plastic deformation, and frictional dissipation that developed in the manifold.

Input files

manifold.inp	Input data for the analysis.
manifold_node_elem.inp	Node and element definitions.
manifold_nodaltemp.inp	Nodal temperature data.

Figures

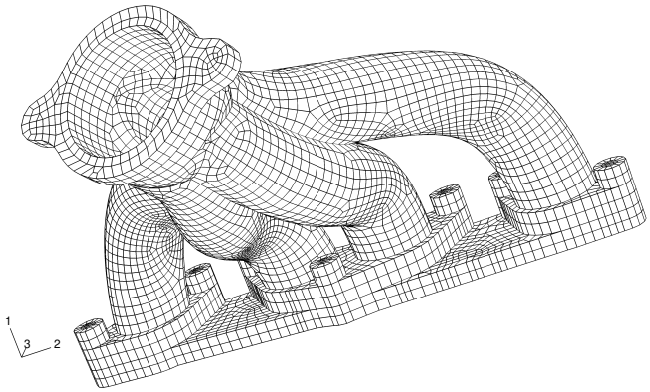


Figure 1: Manifold assemblage.

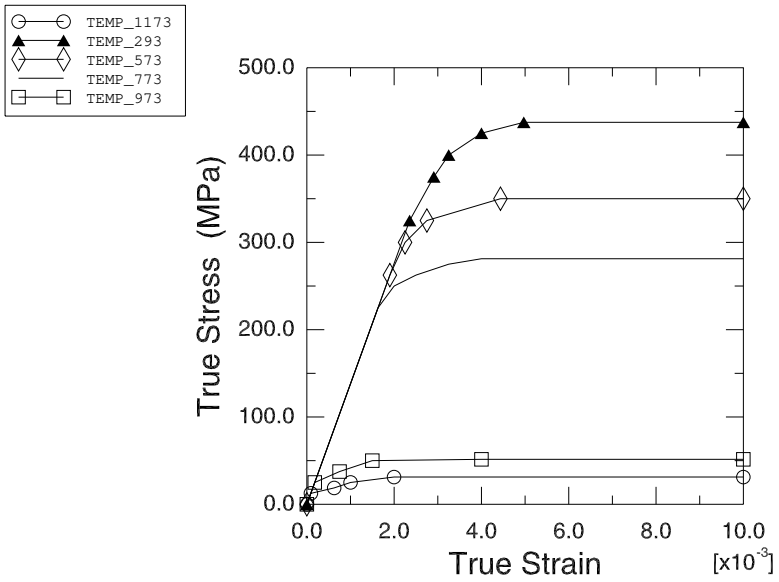


Figure 2: Gray cast iron temperature-dependent plasticity curves.

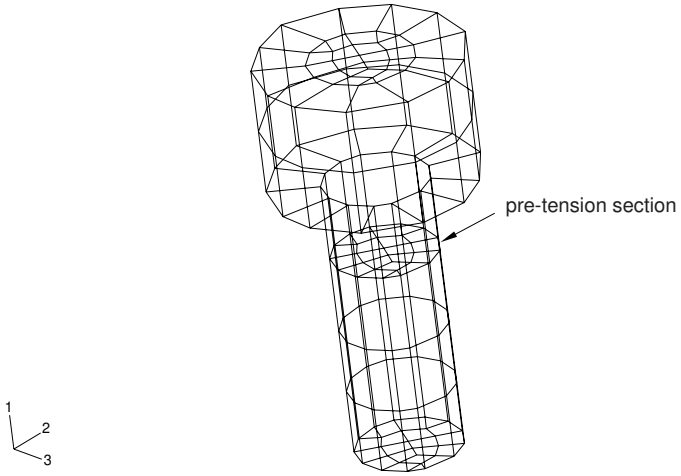


Figure 3: Pre-tension section.

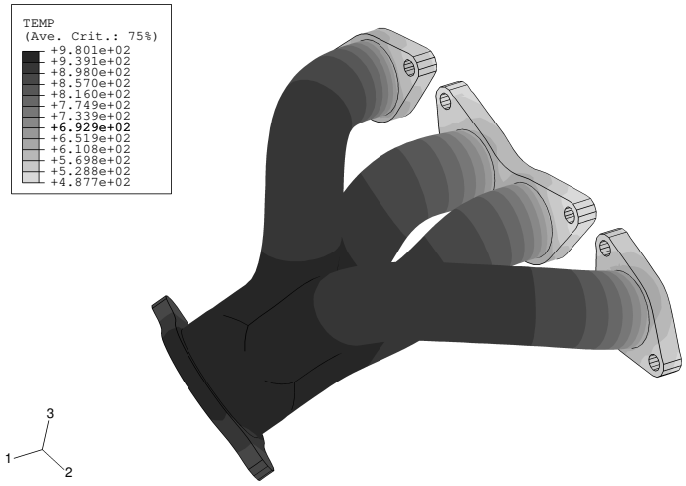


Figure 4: Steady-state temperature distribution.

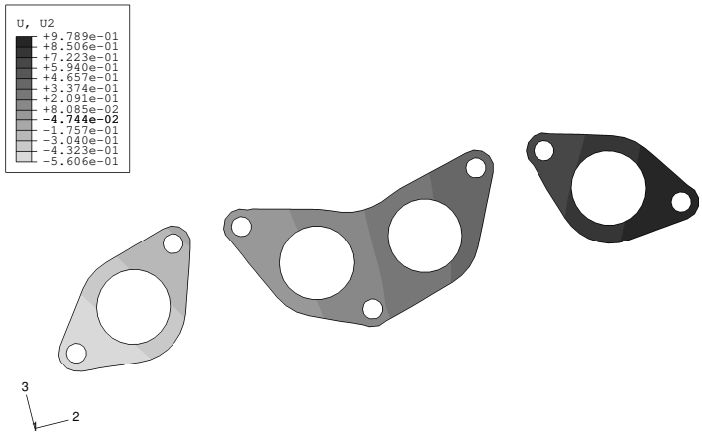


Figure 5: Lateral expansion of manifold footprint.

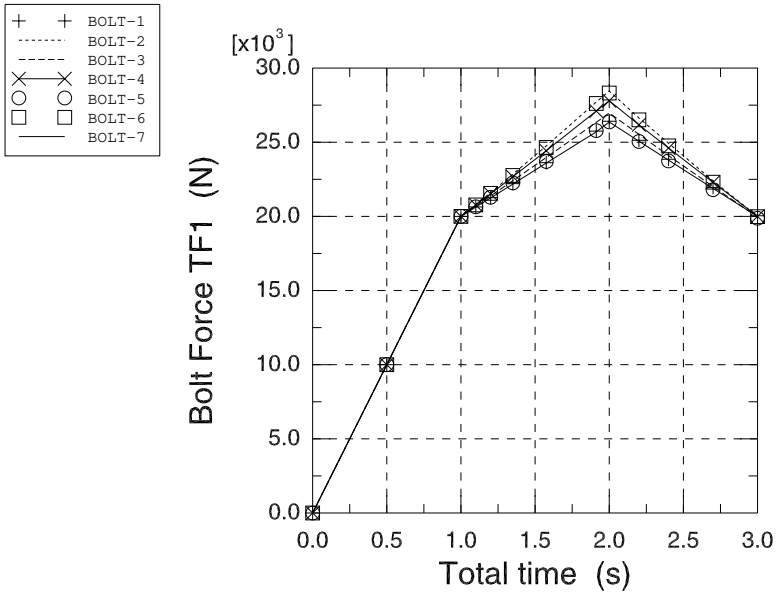


Figure 6: History of bolt forces.

Coolant manifold cover gasketed joint

Products: Abaqus/Standard

Overview

This example shows how to use gasket elements to model a coolant manifold cover gasketed joint.

Engine gaskets are used to seal the mating surfaces of engine components to maintain the integrity of the closed system throughout a wide range of operating loads and environmental conditions. Inadequate gasket performance leads to diminished engine pressure and fluid leakage, resulting in degradation of engine performance and potential engine damage. The gasket, the engine component flanges, and the fasteners—collectively referred to as a gasketed joint—must be considered as a unit when determining the system sealing performance because most gasketed joints do not obtain a uniform contact stress distribution due to nonuniform bolt spacing and flange distortion during assembly and subsequent operational loading.

Engine gaskets are often complicated geometric constructs of various engineering materials and are subject to large compressive strains. The compressive response of the gasket is highly nonlinear. Such complexities make detailed modeling of gaskets with continuum elements difficult and impractical when analyzing complete assemblies.

Abaqus has a dedicated class of elements, referred to as gasket elements, that simplify the modeling of such components while maintaining the essential ingredients of the nonlinear response. Typical use of these gasket elements involves a tabular representation of the pressure versus closure relationship in the thickness direction of the gasket. The pressure versus closure models available in Abaqus allow the modeling of very complex gasket behaviors, including nonlinear elasticity, permanent plastic deformation, and loading/unloading along different paths. These behaviors are usually calibrated directly from test data. In this manner a complex gasket can be modeled effectively using a single gasket element in the thickness direction.

In this example a paper foam gasket with a silkscreened silicone bead is compressed between the lower engine intake manifold and the coolant manifold cover. The coolant manifold cover seals the lower intake manifold coolant passages so that the coolant can be distributed to the cylinder heads. An exploded view of the gasketed joint model is shown in *Figure 1*. It consists of two steel bolts, an aluminum coolant manifold cover, a paper foam gasket with a silicone bead, and—for simplicity—only a portion of the lower intake manifold, which is composed of steel. Symmetry conditions reduce the structure to a half model. The gasketed joint is subjected to the following mechanical and environmental load conditions:

1. Simulate the bolt loading sequence to fasten the joint.
2. Heat the assembly to the maximum operating temperature and apply interior cavity pressure.
3. Cool the assembly to the minimum operating temperature while maintaining interior cavity pressure.
4. Return the assembly to ambient conditions with the interior pressure removed.
5. Disassemble the gasketed joint.

Geometry and material

The portion of the lower intake manifold that is modeled has two passages. Coolant flows from one passage into the manifold cover and back out through the other passage. Two steel bolts secure the cover to the manifold. The bolt shanks have a diameter of 6.0 mm, and the bolt heads have a diameter of 11.8 mm. The bolts and the lower intake manifold are assigned a Young's modulus of 2.0×10^5 MPa, a Poisson's ratio of 0.28, and a coefficient of thermal expansion of 1.6×10^{-5} per °C. The aluminum coolant manifold cover has a Young's modulus of 7.1×10^4 MPa, a Poisson's ratio of 0.33, and a coefficient of thermal expansion of 2.3×10^{-5} per °C.

The metal components (bolts, cover, and intake manifold) are modeled with three-dimensional continuum elements: 1304 first-order brick elements with incompatible deformation modes (C3D8I) and 208 first-order prism elements (C3D6). The C3D8I elements are chosen to capture the bending of the cover, using only one element through its thickness. The C3D6 elements are used only where geometric constraints preclude the use of C3D8I elements.

The gasket schematic shown in *Figure 2* has two distinct regions. The majority of the gasket is composed of a 0.79 mm thick, flat, crushable paper foam material. To ensure proper sealing pressures for this joint, a 0.076 mm thick silicone bead has been silkscreened along the top surface of the gasket encircling the interior cavity. Placing silicone beads on gaskets results in a change in the load transmitting characteristics of the gasket, which often improves both the recovery properties of the gasket and its potential to remain sealed for the long term.

The entire gasket, including the bead, is modeled as a flat sheet with one gasket element through the thickness (see *Figure 3*). A relatively fine mesh is used for the gasket to capture the in-plane variation of the gasket sealing pressure. This creates a mismatched mesh across the contacting surfaces, but Abaqus contact definitions do not require one-to-one matching meshes across contact pairs. The gasket components (silicone bead region and paper foam region) are modeled with 973 first-order 8-node area elements (GK3D8) and 29 first-order 6-node area elements (GK3D6). The physical thickness of the entire sheet of gasket elements corresponds to the initial combined height of the paper foam and the silicone bead, 0.866 mm. The elements in the region of the gasket beneath the silicone bead are assigned different gasket properties from the rest of the elements in the gasket model. The paper foam region is initially not in contact with the cover. The initial gap is 0.076 mm. No pressure is generated in this portion of the gasket until the gap has closed. Gasket region property distinctions, such as initial gaps and different pressure versus closure relationships, are assigned to corresponding element sets by referring to different gasket behavior definitions.

Experimentally determined pressure versus closure curves for the two distinct gasket regions without the initial gap taken into account are shown in *Figure 4*. Tabular representations of these curves are specified using a gasket thickness-direction behavior that is associated with the respective gasket behavior definitions. Creep/relaxation properties of the gasket and temperature-dependent pressure versus closure properties, capturing such effects as the glassy transition temperature of the silicone bead, are not accounted for in this example. Initially, Abaqus considers the gasket behavior to be nonlinear elastic, such that loading and unloading occur along the same user-defined nonlinear path. Abaqus considers yielding to occur once the slope of the pressure versus closure curve decreases by at least 10%. In addition to the single loading curve, whose closure increases monotonically, the user can define any number of unloading curves at different levels of plastic closure. Yielding occurs at a closure of 0.1118 mm for both regions of the gasket in this example, after which the gasket stiffness decreases slightly up to a closure of 0.15 mm, the final point on the loading curve. Beyond the data of the loading curve defined by the user, Abaqus considers the gasket to behave with a fully crushed elastic response by linearly extrapolating the last segment of the last specified unloading curve (alternatively, the user could have specified a piecewise linear form).

A single unloading curve is defined for each of the two gasket regions: the unloading curve for the silicone bead region is defined at 0.11 mm of plastic closure, and the unloading curve for the paper foam region is defined at 0.09 mm of plastic closure. Any unloading of the gasket beyond the yield point occurs along a curve interpolated between the two bounding unloading curves, which—for this example—are the initial, nonlinear elastic curve and the single unloading curve.

Gasket materials often have higher coefficients of thermal expansion than most of the metals from which the bolts and flanges are made. For situations involving wide and rapid temperature fluctuations resultant differences in relative expansion and contraction can have a significant effect on the sealing properties of the gasket. The coefficient of thermal expansion for the silicone bead region is 1.2×10^{-4} per °C, and for the paper foam region it is 3.0×10^{-5} per °C.

In this case, because of the differences in thermal expansion between the aluminum cover and the steel intake manifold, it is important to account for the membrane and transverse shear properties of the gasket and to model frictional effects between mating surfaces. For this analysis the silicone bead region of the gasket is defined to have a membrane stiffness of 75 MPa and a transverse shear stiffness of 40 MPa. The base foam material is

defined with a value of 105 MPa for the membrane stiffness and a value of 55 MPa for the transverse shear stiffness. A friction coefficient of 0.2 is used between all mating surfaces.

A separate analysis is included in this example problem using the "thickness-direction only" version of the gasket elements (GK3D8N and GK3D6N). These elements respond only in the thickness direction and have no membrane or transverse shear stiffness properties. They possess only one degree of freedom per node. As a result, frictional effects cannot be included at the surfaces of these elements. They are more economical than more general gasket elements that include membrane and transverse shear responses and may, thus, be preferable in models where lateral response can be considered negligible.

Loading and boundary constraints

Symmetry boundary constraints are placed along the nodes on the symmetry plane. Furthermore, it is assumed that the intake manifold is a stiff and bulky component, so nodes along the base of the portion of the manifold modeled are secured in the normal direction (the global z-direction). Except for a soft spring constraint to eliminate rigid body motion, these manifold base nodes are free to displace laterally to allow for thermal expansion. Soft springs are also attached to the cover to eliminate rigid body motion in the x- and z-directions.

The bottoms of the bolt heads form contact bearing surfaces with the top surface of the cover flange. In addition, the top of the gasket interacts with the bottom of the cover, while the bottom of the gasket contacts the top of the manifold. Each of these surfaces is defined with a separate surface definition. Mating surfaces are paired together with contact pairs. Three-dimensional, deformable-to-deformable, small-sliding contact conditions apply to each of these contact pairs. The gasket is attached to the manifold base using the no-separation contact behavior, thus constraining it against rigid body motion in the global z-direction. The gasket membrane is allowed to stretch, contract, or shear as a result of frictional effects on both sides of the gasket. The bolts are assumed to be threaded tightly into the base. Therefore, the nodes at the bottom of the bolt shanks are shared with the intake manifold. Contact between the bolt shanks and the bolt holes is not modeled.

The "prescribed assembly load" capability is used to define pre-tension loads in each of the bolts. For each of the two bolts we define a "cut" or pre-tension section and subject the section to a specified load. As a result, the length of the bolt at the pre-tension section changes by the amount necessary to carry the prescribed load, while accounting for the compliance of the rest of the joint. Once a bolt has been pre-tensioned, the applied concentrated bolt load is replaced with a "fixed" boundary condition, which specifies that the length change of the bolt at the "cut" remains fixed, while the remainder of the bolt is free to deform.

The sequence in which the bolts are tightened can have an impact on the distribution of the resultant contact area stress. A poorly specified bolt sequence can cause excessive distortion of the gasket and the flanges, which may lead to poor sealing performance. In the first step of the analysis the left bolt is pre-tensioned to a load of 6000 N using a pre-tension section. In the second step the right bolt is pre-tensioned to 6000 N and the prescribed load on the left bolt is replaced with a fixed boundary condition as described above. Since only half of each bolt is modeled, a total load of 12000 N is carried by each bolt.

Step 3 is the beginning of the three-step thermomechanical operational cycle. In Step 3 the entire assembly is heated uniformly to its maximum operating temperature of 150°C, while simultaneously the interior cavity is pressurized to 0.689 MPa and the prescribed load on the pre-tension section of the right bolt is replaced with a fixed boundary condition. In Step 4 the system temperature is decreased to the minimum operating temperature of −40°C while maintaining the interior pressure load of 0.689 MPa. In Step 5 the gasketed joint is returned to the ambient temperature conditions and the internal cavity pressure is removed.

The sixth and final step in the analysis simulates disassembly of the gasketed joint by removing the bolt loads. This process demonstrates the interpolated unloading response for the different regions of the permanently deformed gasket.

Results and discussion

The prime interest in this problem is the variation of bolt forces during the initial assembly and thermomechanical cycle and the resultant distribution and variation of the gasket sealing pressure.

The function of the fasteners in a gasketed joint is to apply and maintain the load required to seal the joint. The bolt pattern and tension are directly related to the sealing pressure in the clamped gasket. At the maximum service temperature the bolt loads can be expected to be at their peak as a result of thermal expansion effects. It is important to ensure that the stress values of the metal engine components remain below yield and that there is no significant bending of the flanges, which may cause improper sealing of the gasket. At the minimum operating temperature the bolt loads are expected to reach a minimum as a result of thermal contraction effects. Hence, it is necessary to assess that adequate sealing pressure is retained throughout the gasket.

Figure 5 shows the bolt load variation over the course of the six analysis steps. During the first step the pre-tension section node on the right bolt was prescribed a zero change of length constraint, which implies that the right bolt has just been placed in position but not torqued tightly. Hence, as the left bolt is tightened during Step 1, a small reaction load is generated in the right bolt. At the end of the second step during which the right bolt is tightened to carry a force of 6000 N, the force in the left bolt increases to 6200 N. In Step 3 the deformation of the assembly causes the bolt forces to increase to maximum values of 6800 N in the left bolt and 6600 N in the right bolt because of thermal expansion and interior pressurization. When the assembly is cooled to the minimum operating temperature, the bolt loads reach their minimum values. Due to thermal cycling and interior cavity pressure inducing inelastic response in the gasket, the bolt forces at the end of the operational cycle reduce to 6050 N in the left bolt and 5950 N in the right bolt.

The gasket sealing pressure pattern depends on the rigidity of the flanges. Hence, it is useful to predict how the structure will deform due to the applied loading. *Figure 6* shows the deformed shape of the coolant manifold cover at a displacement magnification factor of 50. Bowing of the cover from initial assembly and subsequent operational loads will lead to a nonuniform sealing pressure distribution in the gasket. *Figure 7* illustrates the distribution of the contact pressure along with the contact pressure error indicator on the gasket surface after initial fastening of the joint. The error indicator field is used to assess the contact pressure accuracy and validate the existing mesh refinement. High values of the error indicator are observed in the regions near the boundary between the two materials used for modeling the gasket response. Capturing the contact pressure gradients more accurately would require increasing the mesh refinement in these regions.

Figure 8 shows the sealing pressure as a function of position along the perimeter of the silicone bead at the end of each of the analysis steps. The sealing pressure reaches a minimum at the point equidistant from the bolts, making this the critical point in the gasketed joint design. This figure also reflects the reduction in the sealing pressure near the bolt holes as a result of plastic deformation of the gasket body during the operational cycle. *Figure 9* is a contour plot of the permanent deformation in the gasket after completion of the thermomechanical cycle.

Figure 10 follows the pressure/closure history of one point in the gasket during this analysis in relation to the user-specified loading/unloading test data. The “mechanical closure” (total closure, E11, minus thermal closure, THE11) is plotted along the abscissa of this figure. The material point traced (element 18451, integration point 1) is located along the inside periphery of the silicone bead at the symmetry plane of the assembly nearest the left bolt. Step 1 shows that this point follows the initial elastic loading curve up to the closure of 0.1118 mm. After this amount of closure, further loading causes plastic deformation. In the second step the tightening of the bolt results in a very small amount of unloading for this material point. For purposes of clarity, this deformation is not shown in the figure. Step 3 involves heating the system to the maximum operating temperature and pressurizing the interior cavity so that further yielding of the material point occurs. Step 4 results in the partial unloading of the point due to the thermal contraction associated with cooling the assembly to the minimum operating temperature. For this case the unloading path is based on a curve interpolated between the initial, nonlinear elastic curve and the single unloading curve. The return of the assembly to ambient conditions partially reloads this point along the same path as the previous unloading; however, no further yielding of this material point occurs during this step. In the final step the gasket is unloaded completely.

The analysis using the “thickness-direction only” gasket elements runs in nearly half the CPU time of the full three-dimensional gasket element model. Minimum gasket sealing pressures in Step 4 of this analysis are predicted to be about 20% lower because frictional effects are neglected.

Input files

manifoldgasket.inp	Input data for the analysis.
manifoldgasket_mesh.inp	Node, element, and surface definitions.
manifoldgasket_thick.inp	“Thickness-direction only” gasket element analysis.
manifoldgasket_thick_mesh.inp	Node, element, and surface definitions for the “thickness-direction only” gasket element analysis.

References

- Czernik, D. E., Gasket Handbook, McGraw-Hill, New York, 1996.

Figures

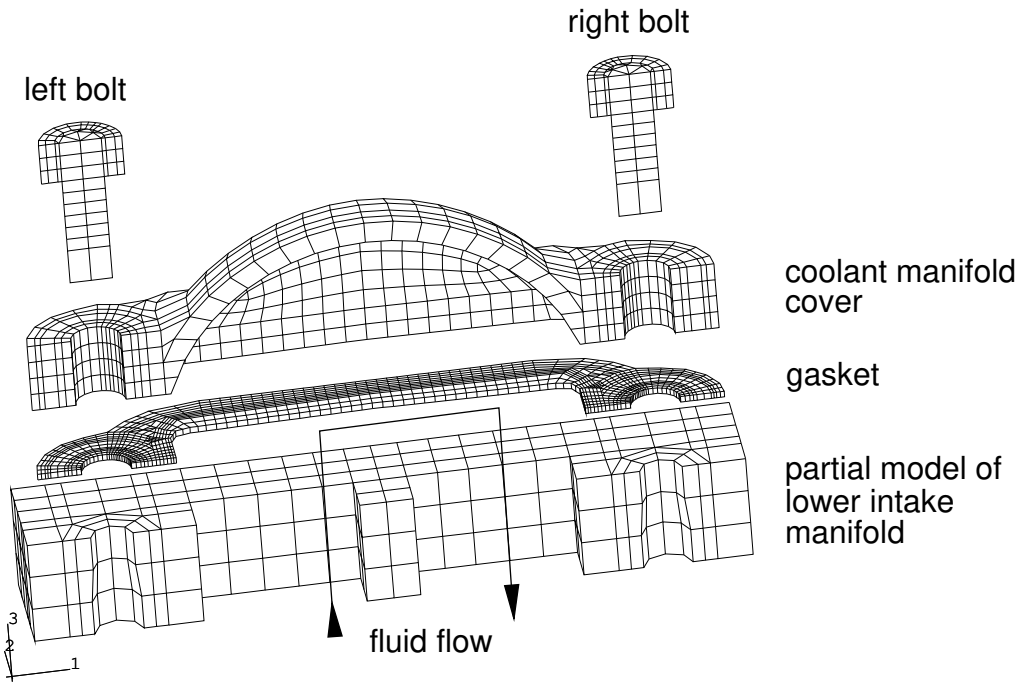


Figure 1: Coolant manifold assemblage.

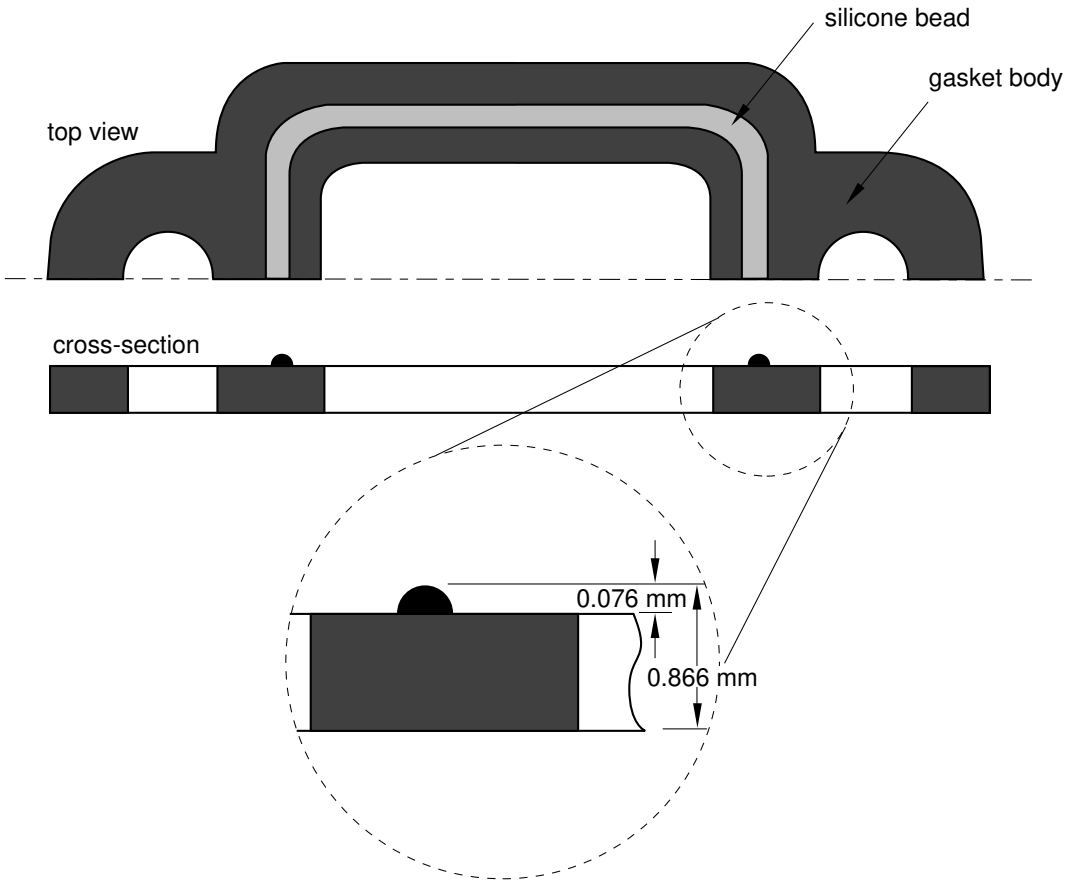


Figure 2: Schematic representation of a silicone bead printed on the gasket body.

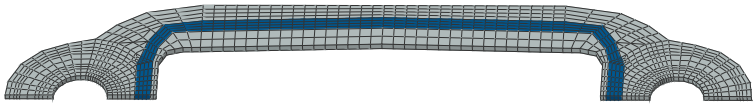


Figure 3: Mesh of gasket with silicone bead highlighted.

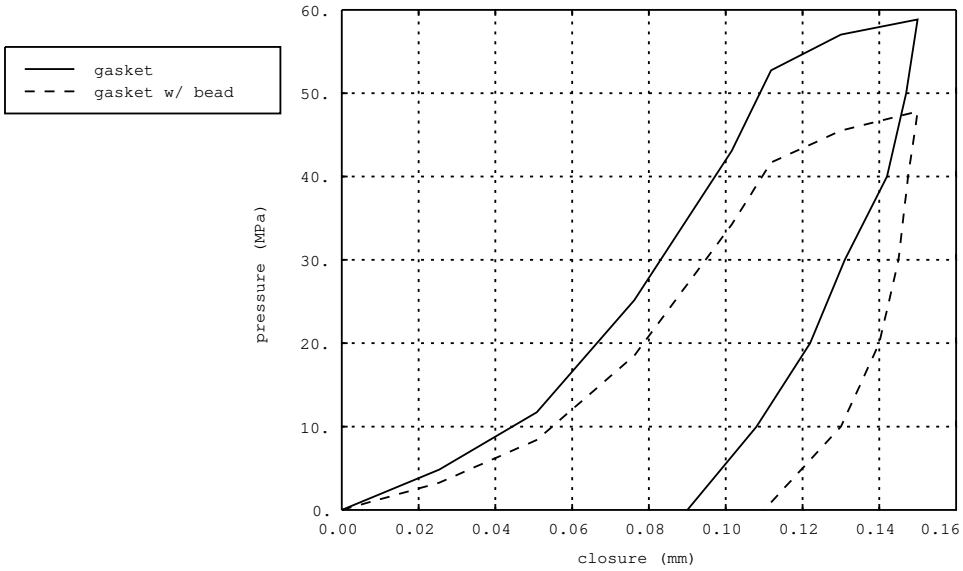


Figure 4: Pressure versus closure behavior for the gasket and the gasket with silicone bead.

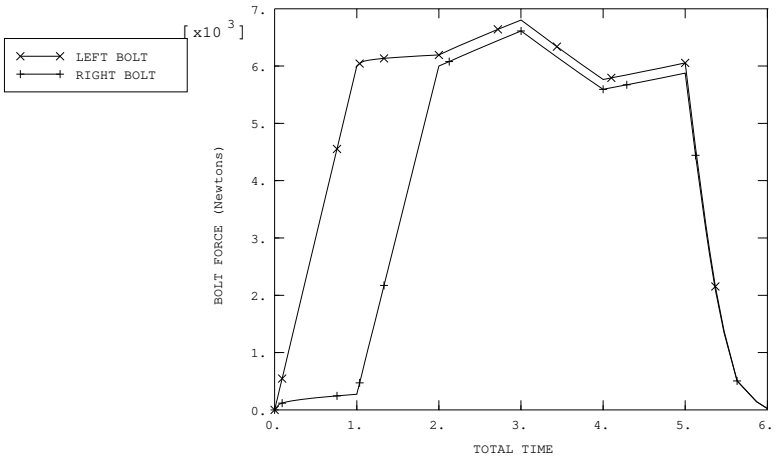


Figure 5: History of bolt force.

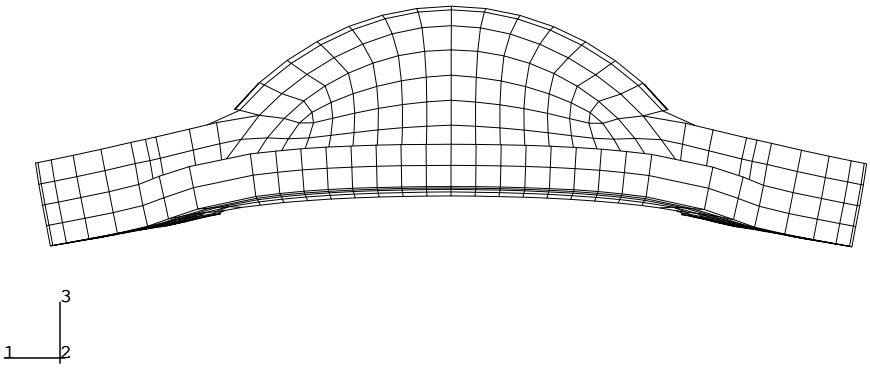


Figure 6: Deformed shape of coolant manifold cover at a displacement magnification factor of 50.

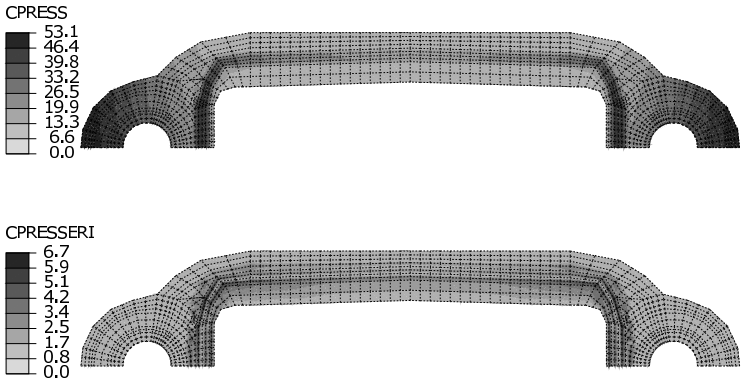


Figure 7: Contact pressure and contact pressure error indicator on the gasket surface after the initial fastening sequence.

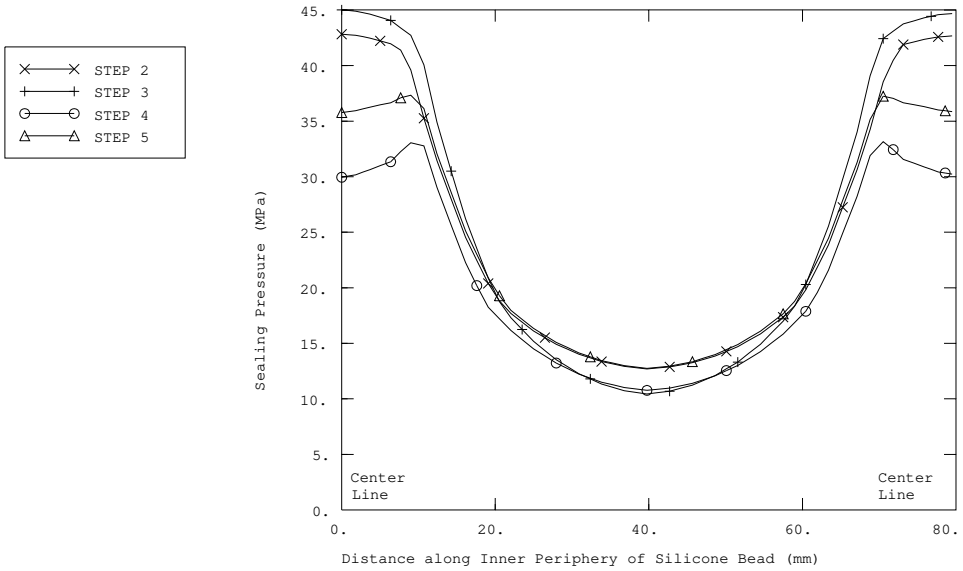


Figure 8: Sealing pressure along inside periphery of silicone bead region of gasket.

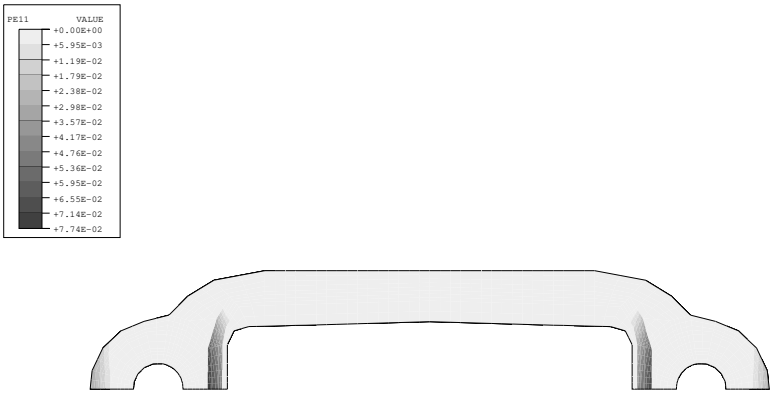


Figure 9: Plastic closure in gasket after operational cycle.

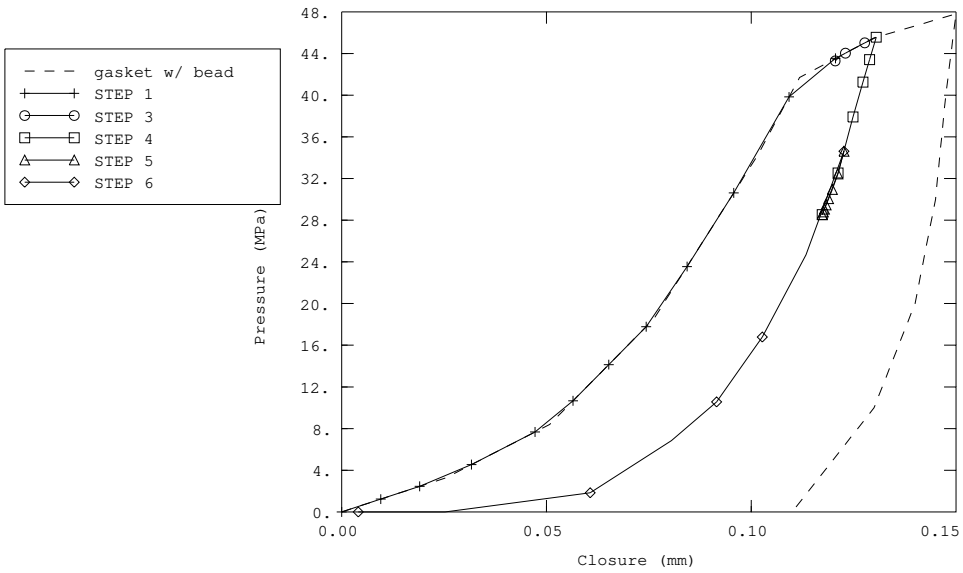


Figure 10: Typical pressure-closure diagram for material point in silicone bead region of gasket.

Conductive, convective, and radiative heat transfer in an exhaust manifold

Products: Abaqus/Standard

References:

- *Steady-State Analysis*
- *Cavity Radiation in Abaqus/Standard*
- **CAVITY DEFINITION*
- **HEAT TRANSFER*
- **RADIATION VIEW FACTOR*
- **SFILM*
- **SRADIATE*
- *Uncoupled heat transfer analysis*
- *Cavity radiation*
- *View factor calculation*

Overview

This example shows conductive, convective, and radiative heat transfer of an exhaust manifold.

The following Abaqus features are demonstrated:

- computing steady-state heat transfer in an exhaust manifold,
- comparing results for radiation heat transfer formulations using cavity radiation and average-temperature radiation conditions, and
- using film conditions to simulate the convective heat transfer from the exhaust gases.

Application description

Heat transfer in engine exhaust manifolds is governed by three effects: conduction through the metal, convection from the hot exhaust gases, and radiative exchange between different parts of the metal surface. This example illustrates the computation of the equilibrium thermal state of a manifold subject to these effects. The units of length in this example are millimeters; otherwise, standard metric units are used.

The procedure consists of a single heat transfer step in which the thermal loading conditions are ramped up from zero. The boundary constraints on the manifold flanges are a simplification of those experienced under operating conditions: the temperatures at the cylinder head and the outlet are fixed. Convection due to heat transfer from the hot exhaust is applied at the internal surfaces of the manifold tubes. Radiation is modeled between the internal surfaces of the tubes using several methods: the cavity radiation method, with and without cavity parallel decomposition enabled, and using average-temperature radiation conditions.

Geometry

The exhaust manifold part being analyzed is depicted in *Figure 1*. It consists of a four tube exhaust manifold with three flanges, as in *Exhaust manifold assemblage*.

Materials

The manifold is cast from gray iron with a thermal conductivity of 4.5×10^{-2} W/mm/°C, a density of 7800×10^{-9} kg/mm^3, and a specific heat of 460 J/kg/°C. The manifold begins the analysis with an initial temperature of 20°C. The part is dimensioned in millimeters, and the temperature is measured in °C, so the Stefan Boltzmann constant is taken as 5.669×10^{-14} W/mm^2/K^4 and absolute zero is set at 273.15°C below zero. The surface emissivity of gray iron is taken as a constant value of 0.77.

Initial conditions

The initial temperature of the manifold is set to 20°C.

Boundary conditions and loading

The hot exhaust gases create a heat flux applied to the interior tube surfaces. In this example this effect is modeled using a surface-based film condition, with a constant temperature of 816°C and a film condition of 500×10^{-6} W/mm^2/°C. A temperature boundary condition of 355°C is applied at the flange surfaces attached to the cylinder head, and a temperature boundary condition of 122°C is applied at the flange surfaces attached to the exhaust.

Abaqus modeling approaches and simulation techniques

The radiative transfer between the interior surfaces of the manifold tubes is modeled using several methods for comparison: through the cavity radiation method, with and without cavity parallel decomposition enabled, and through average-temperature radiation conditions (see *Cavity Radiation in Abaqus/Standard*). Using the cavity radiation methods, geometric view factors are computed in Abaqus between each facet of the mesh on the exposed interior tube surface. These view factors quantify the effect of radiative transfer between each facet and each of the other facets in the user-defined cavity. The view factors, in turn, are used to compute a fully populated interaction matrix to compute the radiation flux between each pair of facets in the model. When modeling radiation through average-temperature radiation conditions, the flux at each facet is equal to that resulting from a black enclosure, held at the average temperature in the cavity, enclosing the facet. For the cavity radiation methods, some of the facets on the interior of the manifold have a view of the exterior, which is not modeled in this example. The exterior ambient temperature is taken to be the average of the temperatures used for the cylinder head and exhaust boundary conditions. Only the temperatures on the surface are considered when using average-temperature radiation conditions, so an ambient temperature does not need to be defined. For simplicity, all methods are defined using a single surface that includes all of the interior facets of the manifold tubes.

Summary of analysis cases

Case 1	Steady-state heat transfer with film and radiation effects; radiation modeled using the cavity radiation method without parallel decomposition enabled.
Case 2	Steady-state heat transfer with film and radiation effects; radiation modeled using the cavity radiation method with parallel decomposition enabled.
Case 3	Steady-state heat transfer with film and radiation effects; radiation modeled using average-temperature radiation conditions.

The following sections discuss analysis considerations that are applicable to all cases.

Analysis types

Due to the fourth-order dependence of the radiation flux on the surface temperatures, this example problem is intrinsically nonlinear. For all cases the steady-state heat transfer procedure is used. This is a general analysis step in Abaqus, chosen because iteration is required for convergence. An initial increment is chosen as one-tenth of the final value.

Mesh design

The manifold is meshed using linear hexahedral and wedge heat transfer elements with linear interpolation.

Discussion of results and comparison of cases

Figure 2 shows the nodal temperature field for the manifold. On the left, results obtained using the cavity radiation method are shown (results with and without cavity parallel decomposition enabled were identical); on the right, results using the average-temperature radiation conditions are shown. In this problem we observe good agreement between all methods, although some differences can be discerned between the cavity radiation and average-temperature radiation condition results in the plots.

The peak temperature in the field is higher when using the cavity radiation method. The effect of radiation heat transfer is to smooth out the temperature field in the equilibrium solution: high-temperature zones radiate more heat, which is absorbed by the cooler areas. In the cavity radiation method, this smoothing effect is limited and affected by the geometric view factors: the distance and orientation of the surface facets affects the degree to which radiation exchange can occur. This is not the case when using average-temperature radiation conditions. Each facet absorbs or emits radiative heat flux based on its temperature and the averaged cavity temperature only; the localizing effects of view factors are ignored. Therefore, the average method results reflect the greater smoothing effect of the radiation model used, resulting in lower peak values.

Figure 3 shows the flux magnitude results. The flux field shows even greater agreement than the temperature field.

Solving the cavity radiation equations without parallel decomposition enabled involves the inversion of a fully populated matrix operator. Therefore, this method is significantly more computationally expensive than either the cavity radiation method with parallel decomposition enabled (even when only 1 CPU is used) or the average-temperature radiation condition method. *Table 1* illustrates the differences and compares the computational costs between the methods. In the case of the cavity radiation method with parallel decomposition enabled, we include the performance results for 1, 2, and 4 CPUs. No significant performance enhancement was observed when using multiple CPUs with the other two methods. In this problem the cavity surface contained 4505 facets—it consists of the entire interior of the manifold. The timing results were obtained on a desktop computer using Xeon processors, but the relative comparisons between run times are more pertinent than the specific run times.

Input files

heattransfermanifold.inp	Input data for the analysis using average-temperature radiation conditions.
heattransfermanifold_cavity.inp	Input data for the analysis using the cavity radiation method without parallel decomposition enabled.
heattransfermanifold_cavity_parallel.inp	Input data for the analysis using the cavity radiation method with parallel decomposition enabled.

Tables

Table 1: Relative computational costs of the cavity radiation methods and average-temperature radiation conditions.

	Cavity Radiation				Average-Temperature Radiation Conditions
	Parallel Decomposition			No Parallel Decomposition	
	1 CPU	2 CPUs	4 CPUs		
Increments	6	6	6	6	6
Total iterations	14	14	14	7	10
Wall clock time (sec)	147	81	51	375	12

Figures

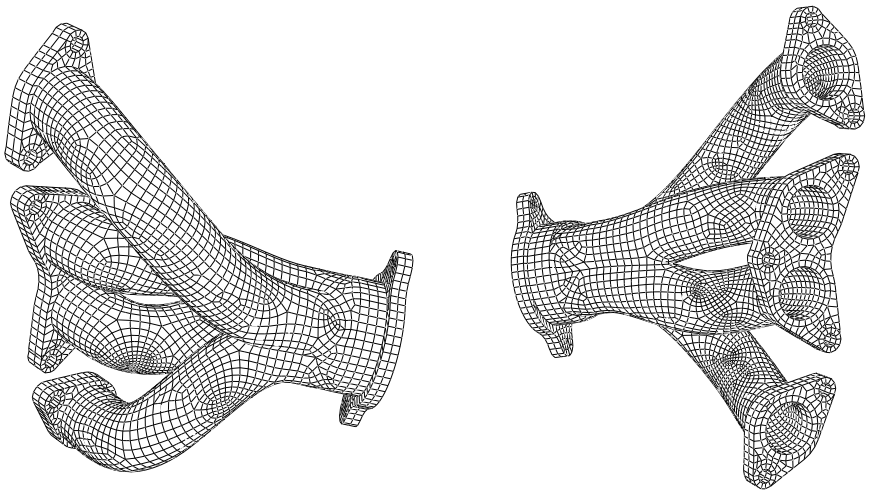


Figure 1: Manifold mesh.

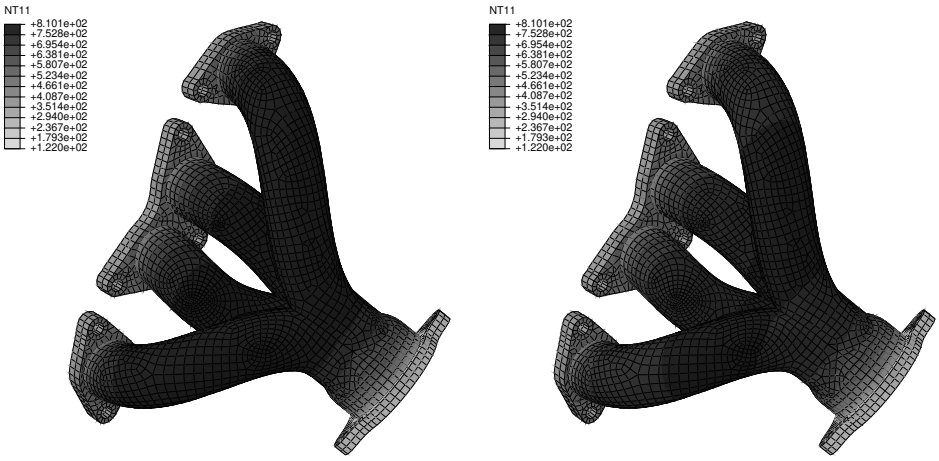


Figure 2: Equilibrium temperature field in the manifold using cavity radiation (left) and average-temperature radiation conditions (right).

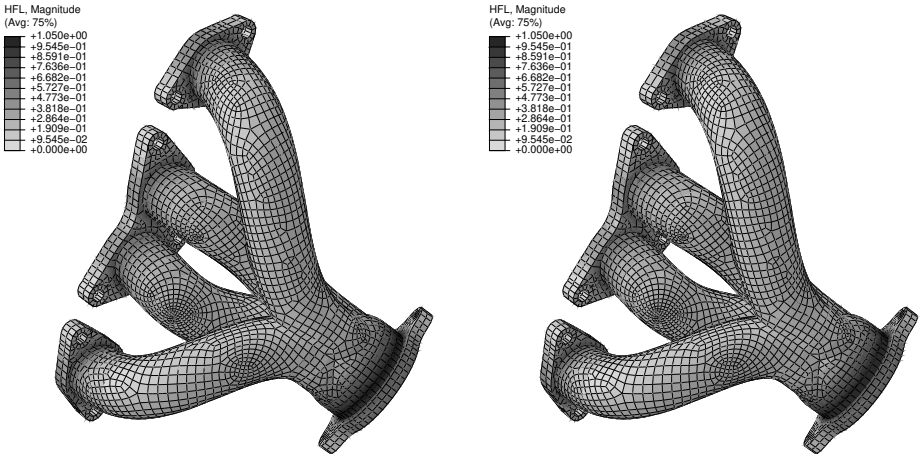


Figure 3: Equilibrium heat flux magnitude field in the manifold using cavity radiation (left) and average-temperature radiation conditions (right).

Thermal-stress analysis of a reactor pressure vessel bolted closure

Products: Abaqus/Standard Abaqus/CAE

References:

- *About Adaptive Remeshing*
- *Understanding adaptive remeshing*

Overview

This example examines the thermal and stress behavior of the bolted closure region of a nuclear reactor vessel assembly.

The following Abaqus features for heat transfer and static stress analyses are demonstrated:

- specifying adaptive remeshing rules in different regions of a model in a particular analysis step;
- using an automated process to remesh the model adaptively based on the remeshing rules specified; and
- viewing error indicator results as a means of assessing mesh quality.

Application description

The vessel assembly forms the pressure boundary surrounding the fuel core. This example considers the strength of sustaining the following loading conditions:

- pre-tension load in the stud bolts,
- constant internal pressure, and
- specified heat-up/cool-down rate.

These loading conditions cover the most basic design requirements of a reactor vessel. A short and rapid temperature change is one of the most severe loading cases and will be considered in this example. The International System of units (SI) will be used in the following sections to describe the model. The analysis itself is performed in English units. The model and analysis are derived from details of the Shippingport pressurized water reactor (1958).

Geometry

The problem domain comprises a cylindrical vessel shell, a hemispherical bottom head, a dome-shaped closure head, and the closure and seal assembly, as shown in *Figure 1*. The overall height of the vessel shell including the bottom head is 7650 mm (301 in). The bottom head has an inner radius of 1410 mm (55.5 in) and a thickness of 157 mm (6.18 in). The inner radius of the vessel shell is 1380 mm (54.5 in), and the thickness is 213 mm (8.40 in). The closure head has a height of 2330 mm (91.8 in), an inner radius of 1310 mm (51.5 in), and a thickness of 210 mm (8.25 in). The closure head is attached to the vessel shell by a seal and closure assembly. The assembly includes 40 stud bolts passing through the bolting flanges of the closure head and the vessel shell, each of which is restrained by two cap nuts. To complete the closure assembly, an omega seal is welded to the under surface of the closure head and top surface of the vessel shell. The stud bolt is 2290 mm (90 in) in length and has a diameter of 146 mm (5.75 in). The closure nuts are 304 mm (12 in) long with a thickness of 28.6 mm (1.13 in).

Boundary conditions and loading

The outside of the vessel is exposed to air that has a constant temperature of 21°C (70°F). The inside of the vessel is filled with hot water with an operating temperature of 320°C (600°F). During a cool-down process, the internal temperature is reduced by 38°C (100°F) in two hours. The water inside imposes a constant pressure of 1.38×10^7 Pa (2000 psi) on the internal surface of the vessel.

Abaqus modeling approaches and simulation techniques

The objective of this analysis is an understanding of stresses near the vessel-to-head interfaces. Although the assembly contains many features, such as inlet and outlet nozzles, the example ignores these details since they are far away from the vessel-to-head interface. The rest of the geometry is cyclically symmetric, which allows the example to model the entire 360° structure at a reduced computational expense by analyzing only a single repetitive sector of the model. Since there are 40 stud bolts along the circumference of the reactor vessel, the following analysis is performed on a 9° model with one sector as shown in *Figure 2*.

The example also takes advantage of the fact that the thermal and mechanical responses of the vessel are only weakly coupled. Based on this fact, a sequentially coupled thermal-stress analysis is performed on the reactor vessel. The distribution of the temperature field is obtained first through a heat transfer analysis, then the mechanical response of the vessel is obtained by performing a static stress analysis with the temperature field specified using the results of the thermal analysis.

Summary of analysis cases

Case 1	Steady-state and transient heat transfer analyses, with adaptive remeshing in Abaqus/CAE.
Case 2	Static stress analysis, with adaptive remeshing in Abaqus/CAE.

The following sections discuss analysis considerations that are applicable to both analyses. More detailed descriptions are provided later including discussions of results and listings of the files provided. The models for the two analyses were generated using Abaqus/CAE and imported ACIS-format files.

Analysis types

The thermal analysis includes a steady-state and a transient heat transfer step. The structural analysis is performed using multiple linear general static steps.

Mesh design

The omega seal is meshed with first-order brick elements, while the rest of the model is meshed with second-order tetrahedral elements. The geometry is partitioned to create a fine initial mesh in the area near the closure assembly, where the geometry is most complex and high stress and heat flux are expected. The purpose of this mesh design is to obtain an accurate estimate of the error indicators specified in the remeshing rules, which will result in faster convergence in the adaptivity procedure.

Adaptive remeshing rules

The patch recovery techniques that Abaqus/CAE uses to calculate the error indicator variables can have a significant impact on the analysis solution time. In particular, the element energy density is calculated after each increment and is more costly than the other error indicator variables. To reduce the computational expense, the omega seal is excluded from the region where remeshing rules are specified. The closure head and the vessel shell are also partitioned in such a way that regions relatively farther away from the stud bolt are not included in the remeshing regions. Four separate remeshing rules are defined in the closure head, the vessel shell, the cap nuts, and the stud bolt. In both the thermal and structural analyses, the remeshing rules are specified in only the last step.

Additional details of the remeshing rules used in the two analyses are discussed with each example.

Constraints

To simulate the welding constraints, the bottom surfaces of the omega seal are tied to the surfaces of the bolting flanges in both analyses.

Adaptivity process

Each adaptivity process specifies a maximum number of three remesh iterations.

Heat transfer analysis

The example starts by performing a heat transfer analysis to obtain the temperature distribution in the pressure vessel under the thermal loading.

Analysis types

The analysis consists of a steady-state heat transfer step, representing the steady operation of the reactor. This step is followed by a transient heat transfer step, representing a rapid cool-down event. The resulting temperatures obtained are applied to the subsequent mechanical analysis.

Mesh design

When performing the heat transfer analysis, first-order hexahedral diffusive heat transfer elements (DC3D8) are used in the omega seal, and the rest of the geometry is meshed with second-order tetrahedral diffusive heat transfer elements (DC3D10).

Adaptive remeshing rules

The heat flux error indicator is chosen in all remeshing rules. For each remeshing rule the sizing method is set to uniform error distribution and the error indicator target is set to automatic target reduction.

Materials

The heat transfer analysis requires specification of thermal conductivity, which is 46.7 W/m/°C (2.25 Btu/h/in/°F), and specific heat, which is 460 J/kg/°C (0.11 Btu/lb/°F). The density of the material is also specified, which is 7850 kg/m^3 (0.284 lb/in^3). One solid, homogenous section is used to assign material properties to the elements.

Initial conditions

The initial temperature is set to 21°C (70°F) in the entire model.

Boundary conditions

No temperature boundary conditions are applied. The thermal response of the model is driven entirely by thermal loading through film coefficients.

Interactions

Conductive heat transfer is defined between adjacent/contacting surfaces, and a gap conductance coefficient is specified, which is 1220 W/m^2/°C (1.5 Btu/h/in^2/°F). Heat flux on the surfaces is applied by film conditions. The outer surfaces are exposed to air, which has a film coefficient of 28 W/m^2/°C (0.035 Btu/h/in^2/°F). The inner surfaces are in contact with water with a film coefficient of 580 W/m^2/°C (0.70 Btu/h/in^2/°F). The outer surfaces are initially associated with a sink temperature of 21°C (70°F), and the inner surfaces, 320°C (600°F).

During a subsequent two-hour cooling process, the sink temperature associated with the inner surfaces is reduced by 38°C (100°F).

Analysis steps

The heat transfer analysis is performed using a steady-state step followed by a transient step. The purpose of the first step is to obtain a steady-state solution of temperature distribution in the whole model. The second step lasts for 7200 seconds (2 hours), and it simulates the response of the thermal model during a rapid cooling process.

Convergence

The convergence of the error indicator HFLERI during a three-iteration adaptivity process is presented in *Table 1*. Convergence is observed in each of the remeshing regions.

Run procedure

The model for the heat transfer analysis is generated using Abaqus/CAE to import the geometry, create the thermal loading, mesh the assembly, create the remeshing rules, and run the adaptivity process. Python scripts are provided to build the model and to submit the adaptivity process. The scripts can be run interactively or from the command line.

To create the heat transfer model, select **File->Run Script** from the Abaqus/CAE main menu and select `adaptReactorVesselHT_model.py`.

When you are ready to run the adaptivity process, select **File->Run Script** and select `adaptReactorVesselHT_job.py`. After the Abaqus Scripting Interface scripts have created the model and run the adaptivity process, you can use Abaqus/CAE to view the model and to explore variations of the example.

Results and discussion

The magnitude of the heat flux (HFL) along a particular path near the head to vessel interface is shown in *Figure 3*. This figure shows the spatial variation of the heat flux as the mesh is refined through the three remesh iterations. *Figure 4* shows the original mesh and the final mesh resulting from the adaptive remeshing process. The refined mesh shows how Abaqus/CAE reacted to the higher temperature gradients in the bolted flange region.

Structural analysis

The temperature distribution calculated in the heat transfer case will now augment bolting and pressure loads to define the structural loading of the vessel assembly.

Analysis types

A series of static steps is performed to simulate the mechanical response of the model under both thermal and force loading.

Mesh design

The omega seal is meshed with first-order reduced-integration continuum elements (C3D8R), and the rest of the geometry is meshed with modified second-order tetrahedral elements (C3D10M).

Adaptive remeshing rules

The response of the model varies from step to step during the analysis; therefore, the time history-dependent error indicator ENDENERI is chosen to capture the extreme of the model's response to the load history. For each

remeshing rule the sizing method is set to uniform error distribution and the error indicator target is set to automatic target reduction.

Materials

The linear static structural analysis requires specification of Young's modulus, which is 2.07×10^7 N/m^2 (3.0×10^7 lbf/in^2), and Poisson's ratio, which is 0.29. A thermal expansion coefficient is also defined, which is 6.3×10^{-6}. One solid, homogenous section is used to assign material properties to the elements.

Boundary conditions

Symmetry boundary constraints are placed on the two side surfaces of the sector. Since the two symmetry constraints overlap at the centerline and such definitions are not allowed by the analysis input file processor, partitions are made on the side surfaces so that the centerline and a small part of its surrounding region are excluded from the symmetry boundary constraints. The nodes on the centerline are constrained separately and are free to move only in the axial direction. The center node on the outer surface of the bottom head is fixed to prevent rigid body motion.

Loads

A pre-tension load of 2200 kN (5×10^6 lbf) is applied to the stud bolt. The inner surfaces of the head and the vessel shell are subject to a constant pressure of 1.38×10^7 Pa (2000 psi) from the water.

Predefined fields

When the bolt loading is applied in the pre-assembly step, a constant temperature of 21°C (70°F) is applied. The temperature field is specified using the thermal results from the previous steady-state heat transfer analysis when the inner pressure is applied. The results of the temperature after each increment during the transient heat transfer analysis are imported in the last step when no additional loading is applied.

Interactions

The outer surface of the stud bolt is tied to the inner surfaces of the cap nuts. Small-sliding surface-to-surface contact interactions are defined between the contact surfaces of the cap nuts and the bolting flange on the head and vessel shell. A friction coefficient of 0.2 is specified in the contact between the cap nuts and the bolting flanges. The analysis assumes that the contact between the closure head and the vessel shell is frictionless. The augmented Lagrange method is chosen to enforce the contact constraints.

Analysis steps

The structural analysis is performed by using two static steps. The bolt force and internal pressure are both applied in the first step, a predefined temperature field is also specified using the results obtained in the steady-state heat transfer step. The temperature obtained from the transient heat transfer step is specified in the last loading step, and the structure expands with the change of temperature.

Output requests

Default field output requests are specified in the first step. In the second step, field output requests are made at specified time points to match the results of the structural analysis to those of the thermal analysis at the exact step times.

Convergence

The convergence of error indicators ENDENERI and MISESERI during a three-iteration adaptivity process is presented in *Table 2*.

Run procedure

The model for the static analysis is generated using Abaqus/CAE to import the geometry, create the structural loading, mesh the assembly, create the remeshing rules, and run the adaptivity process. Python scripts are provided to build the model and to submit the adaptivity process. The scripts can be run interactively or from the command line.

To create the structural model, select **File->Run Script** from the Abaqus/CAE main menu and select `adaptReactorVesselSTR_model.py`.

When you are ready to run the adaptivity process, select **File->Run Script** and select `adaptReactorVesselSTR_job.py`. After the Abaqus Scripting Interface scripts have created the model and run the adaptivity process, you can use Abaqus/CAE to view the model and to explore variations of the example.

Results and discussion

The magnitude of the von Mises stress (MISES) along a particular path near the head to vessel interface is shown in *Figure 5*. This figure shows the spatial variation of the von Mises stress as the mesh is refined through the three remesh iterations. *Figure 6* shows the original mesh and the final mesh resulting from the adaptive remeshing process. The refined mesh shows how Abaqus/CAE reacted to the higher stress gradients near the nut-to-bolted flange interfaces and the vessel-to-head interface.

Discussion of results and comparison of cases

The thermal and structural cases presented in this example are complementary; the structural case depends on the thermal case. You can compare how adaptive remeshing refines the mesh in each case. As seen, in *Figure 4* and *Table 1* for the heat transfer case and *Figure 6* and *Table 2* for the structural case, the mesh refinement is significantly different.

Files

To create the models and to run the adaptivity processes, you can use the Python scripts listed below.

Heat transfer analysis

adaptReactorVesselHT_model.py	Script to create the model.
adaptReactorVesselHT_job.py	Script to analyze the model.

Structural analysis

adaptReactorVesselSTR_model.py	Script to create the model.
adaptReactorVesselSTR_job.py	Script to analyze the model.

References

- Naval Reactors Branch, Division of Reactor Development, United States Atomic Energy Commission,*The Shippingport Pressurized Water Reactor,* Reading, Massachusetts: Addison Wesley Publishing Company, 1958.

Tables

Table 1: The convergence of the thermal error indicator HFLERI during an adaptivity process with three iterations.

Remeshing Region	Error Indicator Result (%)			Element Count		
	1	2	3	1	2	3
Stud Bolt	11.7	7.0	4.1	1080	1386	2112
Closure Head	4.9	2.5	1.7	2210	4846	9226
Cap Nuts	2.8	1.5	1.4	1500	4210	5317
Vessel Shell	5.4	2.3	1.5	2443	5124	11036

Table 2: The convergence of the stress-based error indicator ENDENERI during an adaptivity process with three iterations.

Remeshing Region	Error Indicator Result (%)			Element Count		
	1	2	3	1	2	3
Stud Bolt	1.8	1.7	1.5	1334	1737	2211
Closure Head	8.7	4.7	3.2	2210	5120	13320
Cap Nuts	6.0	5.7	4.8	1500	3315	7534
Vessel Shell	6.7	3.6	2.7	2438	5813	15290

Figures

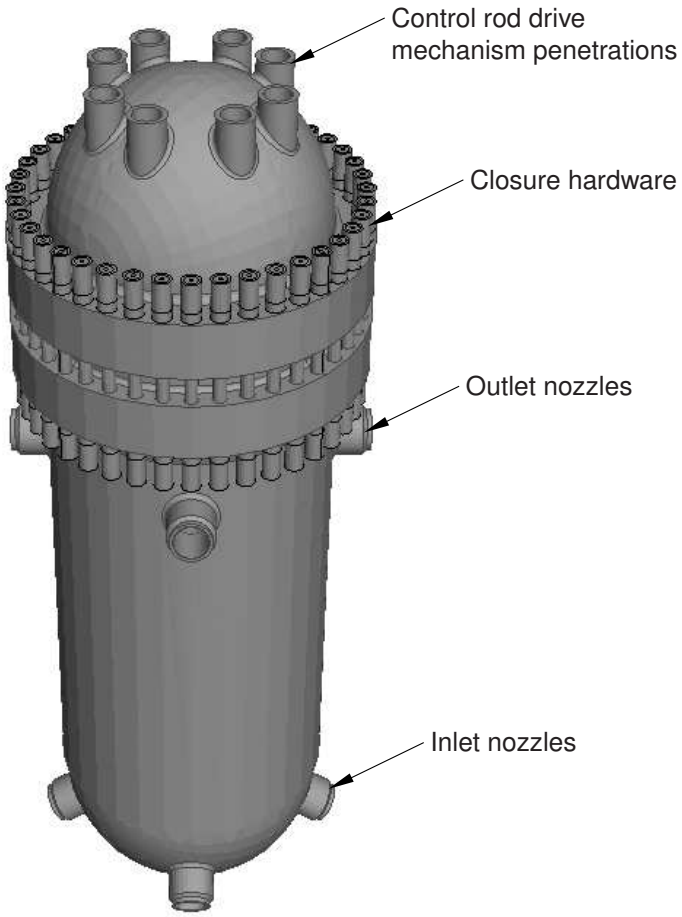


Figure 1: Reactor vessel assembly.

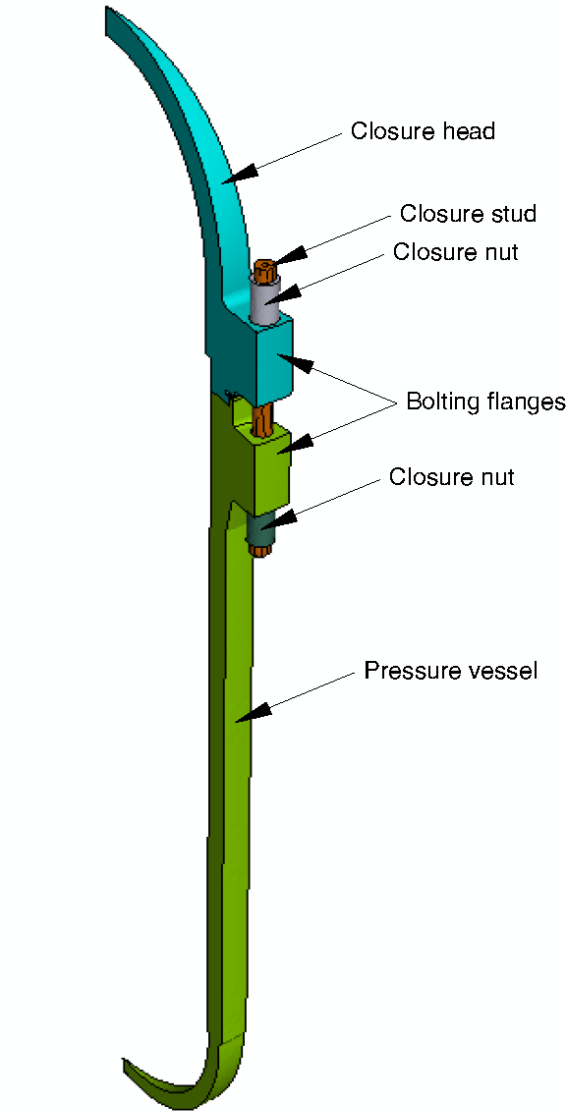


Figure 2: 9° sector model.

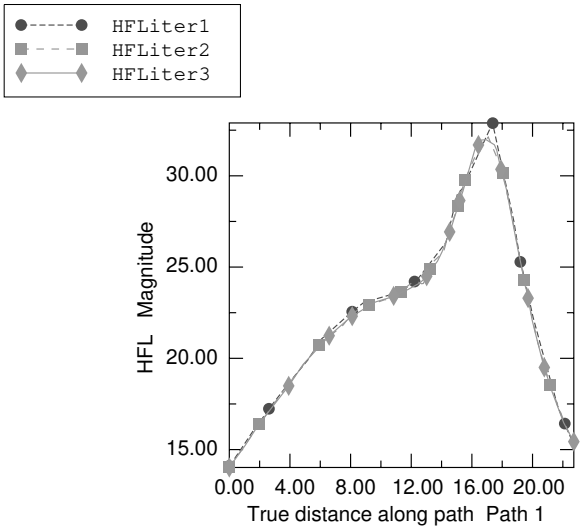


Figure 3: Heat flux along a path near the vessel-to-head interface.

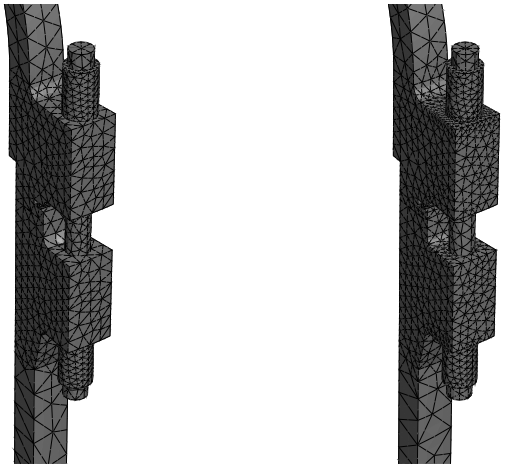


Figure 4: Original and refined mesh for the heat transfer analysis.

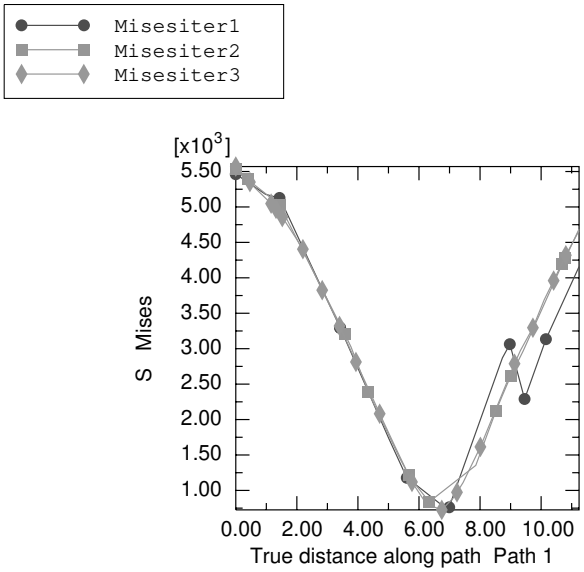


Figure 5: Von Mises stress along a path near the vessel-to-head interface.

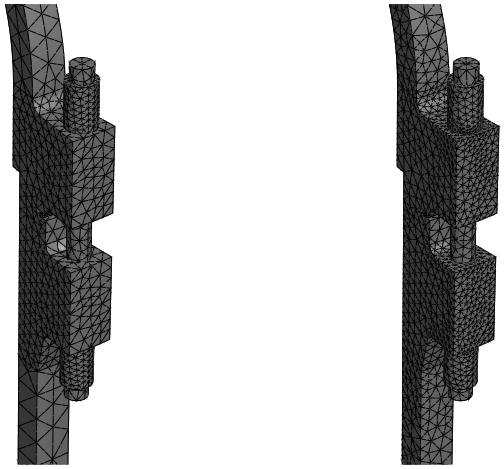


Figure 6: Original and refined mesh for the structural analysis.

Coupled thermomechanical analysis of viscoelastic dampers

Products: Abaqus/Standard

References:

- *Fully Coupled Thermal-Stress Analysis*
- *Time Domain Viscoelasticity*
- **COUPLED TEMPERATURE-DISPLACEMENT*
- **INELASTIC HEAT FRACTION*
- **TRS*
- **VISCO*
- **VISCOELASTIC*

Overview

This example examines the response of a viscoelastic damper under cyclic loading with the damper material modeled as a linear viscoelastic material using a Prony series calibrated to capture the hysteresis response accurately.

The following Abaqus features and techniques are demonstrated:

- using Prony-series viscoelasticity to account for material hysteresis,
- using the thermorheologically simple (TRS) material model to account for temperature dependence in a viscoelastic material,
- accounting for heat generated by mechanical energy dissipation through coupled temperature-displacement analysis, and
- comparing the response of a structure experiencing large deformations using elastic or hyperelastic material models.

Application description

One tool commonly used in Abaqus to model stress relaxation in viscoelastic materials is the Prony series. However, using the Prony series to model stress relaxation in dissipating materials often leads to underprediction of the size of the hysteresis loop for the material when subjected to cyclic loading (Dalrymple et al., 2007). If both the stress relaxation and hysteretic behavior of a given material must be modeled accurately, a more complicated material model, such as the parallel rheological framework model, is required. However, in some structures the stress relaxation response may be of secondary importance to the hysteretic energy dissipation under cyclic loading. In such cases it can be reasonable to calibrate the Prony series coefficients in a manner that accurately captures the hysteresis response of the material at the expense of accurately modeling the stress relaxation response. In this example the response of a viscoelastic damper under cyclic loading is examined. The damper material is modeled as a linear viscoelastic material with a Prony series calibrated to capture the hysteresis response accurately.

One of the salient characteristics of damping materials is their ability to convert mechanical energy into other forms of energy, typically heat. However, the mechanical response of these materials is often highly sensitive to temperature. This analysis examines the importance of accounting for the interdependence of heat generation and transport with mechanical response through the use of a fully coupled thermomechanical analysis.

Geometry

The model consists of a layer of viscoelastic damping material between two steel plates, as shown in *Figure 1*. The damper dimension is 3.5 inches in the x-direction, and the viscoelastic material and steel plates are each 0.5 inches thick, for a total dimension of 1.5 inches in the y-direction.

Materials

The model contains two materials, steel and a viscoelastic damping material with a response described by Shen and Soong (1995). Details of the material response are provided in *Materials* below.

Initial conditions

The initial temperature in the structure is uniformly 21.7 °C.

Boundary conditions and loading

The damper is subjected to sinusoidal shear loading at a rate of 1 Hz for a period of 10 seconds such that the nominal engineering shear strain in the viscoelastic damping material has an amplitude of approximately 50%. This loading is imposed by specifying the displacement of the top surface.

For heat conduction, the damper's boundaries are assumed to be adiabatic (i.e., no heat is transferred to or from the environment).

Abaqus modeling approaches and simulation techniques

Four analyses are performed comparing the predicted response if heat generation from mechanical dissipation is or is not accounted for and comparing the effect of modeling the viscoelastic damping material as a linear elastic or hyperelastic material.

Summary of analysis cases

Case 1	Quasi-static stress analysis with linear elasticity for the viscoelastic damper material
Case 2	Quasi-static stress analysis with hyperelasticity for the viscoelastic damper material
Case 3	Coupled temperature-displacement analysis with linear elasticity for the viscoelastic damper material
Case 4	Coupled temperature-displacement analysis with hyperelasticity for the viscoelastic damper material

Analysis types

Two types of analyses are performed: a transient, static, stress/displacement analysis that neglects the generated heat and a coupled transient temperature-displacement analysis. Both analysis types neglect inertial effects (i.e., the model is quasi-static) and use general steps with geometric nonlinearity since the model undergoes large displacements. In the coupled analysis (see *Fully Coupled Thermal-Stress Analysis*), transient heat transfer resulting from heat generated by dissipation of mechanical energy is modeled along with the force-displacement. The resulting temperature changes will affect the structure's response due to material temperature dependence in the viscoelastic damper and thermal expansion.

Mesh design

A single mesh refinement is examined, with 8 elements through the thickness of each layer of the damper (resulting in a total of 24 elements in the y-direction) and 32 elements in the x-direction. Plane strain is assumed. CPE4 and CPE4H elements are used for the stress analysis for the steel and viscoelastic damper materials, respectively. The coupled analysis uses CPE4T and CPE4HT elements.

Materials

This section provides the properties used for the various materials in the model.

Steel

The steel's mechanical response is modeled with linear elasticity. The Young's modulus of the steel is $E = 29.0 \times 10^6$ psi, and the Poisson's ratio is $\nu = 0.3$.

Viscoelastic damper

The mechanical response of the viscoelastic damper material is determined from experimental data provided by Shen and Soong. For temperature dependence, the material is assumed to be thermorheologically simple. Shen and Soong provide the following TRS shift function based on temperature:

$$\log A = -0.0561\,(\theta - \theta_0)\,,$$

which is a simplified form of the Williams-Landell-Ferry (WLF) relationship implemented within Abaqus (see *Time Domain Viscoelasticity*):

$$\log A = -\frac{C_1\,(\theta - \theta_0)}{C_2 + (\theta - \theta_0)}.$$

The form of the TRS shift function given by Shen and Soong can be replicated using the WLF relationship by selecting values of C_1 and C_2 such that $C_1/C_2 = 0.0561$ and $C_2 \gg \theta - \theta_0$. Therefore, the parameters of the WLFTRS definition of the material implemented within Abaqus are defined as $\theta_0 = 21.7$ °C, $C_1 = 56.1$, and $C_2 = 1000$ °C.

The storage and loss moduli along with the corresponding frequencies from Shen and Soong were shifted using the TRS shift function to a reference temperature of 21.7 °C. This choice of reference temperature, which in practice is arbitrary, was made based on the temperature at which the regression performed by Shen and Soong yields no shift. The original moduli, along with the shifted values, are given in *Table 1*. As a general practice, it is important that the time scales or frequencies of the loading in the analysis be within the scope of the time scales and frequencies of the experimental data used to characterize the material response. The loading in this analysis is 1 Hz, which is within the range of frequencies given in *Table 1*.

The Levenburg-Marquardt algorithm (Press et al., 1992), a least-squares approach, was used to obtain the Prony series parameters (see *Table 2*) for the Maxwell model using the temperature-shifted frequencies and storage and loss moduli. The initial shear modulus is 2.0845 ksi.

For the instantaneous elastic response of the material, it is assumed that the bulk modulus is constant with respect to time or frequency. The material is assumed to be nearly incompressible with an initial Poisson's ratio of 0.495. This results in a bulk modulus of $K = 2.078 \times 10^5$ ksi.

This example compares two different approaches for modeling the instantaneous elastic response of the damping material. The first is to model the response as linear. The initial Young's modulus of $E = 6.232$ ksi is determined from the initial shear modulus and Poisson's ratio.

The second approach utilizes a neo-Hookean hyperelastic material definition (see *Hyperelastic Behavior of Rubberlike Materials*). Hyperelasticity is based on finite deformation theory, which may be more appropriate for this model considering the large strains that are experienced. Another consequence of this change is that the stress-strain relationship for the material will be nonlinear. The neo-Hookean model is chosen due to the lack of comprehensive data for the instantaneous stress-strain response of the viscoelastic damping material. The strain energy potential constants ($C_{10} = 1.0423 \times 10^3$ and $D_1 = 9.6267 \times 10^{-6}$) are determined from the initial shear and bulk moduli.

Heat transfer

The properties of the materials related to heat transfer are given in *Table 3*. The inelastic heat fraction (see *Fully Coupled Thermal-Stress Analysis*) for the viscoelastic damper material is specified as $\mathbf{1.071 \times 10^{-4}}$. This assumes that all energy dissipated by viscoelasticity is converted into heat and accounts for the change of energy units from in-lbf in the mechanical analysis to BTUs, which are used to define the material properties related to heat transfer given in *Table 3*.

Initial conditions

The initial temperature of all nodes is 21.7 °C.

Boundary conditions

The nodes on the top surface are constrained in the *y*-direction. The nodes on the bottom surface are constrained in both the *x*- and *y*-directions. No thermal boundary conditions are specified, meaning that there is no heat transfer with the environment.

Loads

A sinusoidal time-varying *x*-direction displacement with an amplitude of 0.25 inches and a frequency of 1 Hz is applied to the nodes on the top surface.

Convergence

Adaptive time stepping is used with a creep strain error tolerance (CETOL) of 5×10^{-3}. Increments that are sufficiently small to satisfy this tolerance typically converge within 2–3 iterations.

Discussion of results and comparison of cases

Figure 2 shows a contour of the logarithmic shear strain in the damper at the maximum stroke of the final cycle in the coupled model using hyperelasticity, showing the large deformations present in the viscoelastic damper material. The hysteresis responses predicted using the two different material models are shown in *Figure 3* for the coupled thermomechanical analysis and *Figure 4* for the uncoupled stress analysis. The importance of accounting for the heat generated by dissipation of mechanical energy is immediately apparent in the way that the hysteresis loops flatten and decrease in area over time in the coupled analyses. These changes occur because the material loses stiffness as the temperature increases. No such decrease is noted in the pure stress analysis—the cyclic hysteresis response reaches a steady state after a single cycle has been completed.

Figure 3 and *Figure 4* show that the forces are slightly higher when using the hyperelastic model for the damping material than when using the linear elastic model. This difference is expected due to the nonlinear nature of the stress-strain relationship obtained using hyperelasticity and the use of finite deformation theory.

Figure 5 shows the contour of the temperature in the damper, indicating that some regions of the damper experience a temperature increase of approximately 8 degrees over the 10-second duration of the analysis. This increase is due to the heat generated by mechanical energy dissipation. The generated heat accumulates in the viscoelastic material faster than it is conducted to the cooler steel plates, leading to a steady temperature increase. *Figure 6* shows the time history of the temperature at the central node (node 1217) of the damper in the coupled analysis. The temperature rise is more pronounced in the model using hyperelasticity due to the higher stresses (and subsequently greater energy dissipation) experienced when using the hyperelastic material definition.

One additional aspect of interest for this material definition is how the predicted stress relaxation compares to the material's actual response. Recall that for this viscous damper, the hysteresis response is of primary interest. Therefore, the viscoelastic Prony series was calibrated to capture that aspect of the material response accurately. Shen and Soong performed relaxation experiments on a test fixture using the material in this example. The force response of that test fixture can be approximated using the model from this example by multiplying the sum of

the x-direction reaction forces in the upper nodes of the model by 5, permitting a direct comparison between experiment and simulation.

Figure 7 compares the time history of the experimentally measured and simulated reaction forces resulting from loading the damping material to a nominal engineering shear strain of 20% over one second and then holding the displacements of the test fixture constant. As seen in the figure, the force in the simulation drops more quickly than it does in the experiment, and to a much lower value. This result highlights the intrinsic limitation of linear viscoelasticity described in the introduction of this example—it is generally not possible to represent both the hysteresis response and the stress relaxation response accurately using a linear viscoelastic Prony series. Therefore, while the approaches described in this example are appropriate for modeling a viscoelastic material under cyclic loading, the approach described here is not recommended for analyses in which stress relaxation is of primary importance.

Acknowledgements

SIMULIA would like to thank Professor Gary Dargush at SUNY Buffalo for providing the model used in this example, which is based on work by Rajesh Radhakrishnan (2000).

Input files

Case 1

visco_damper_uncoupled_linelastic.inp Uncoupled analysis with linear elasticity.

Case 2

visco_damper_uncoupled_hyperelastic.inp Uncoupled analysis with hyperelasticity.

Case 3

visco_damper_coupled_linelastic.inp Coupled analysis with linear elasticity.

Case 4

visco_damper_coupled_hyperelastic.inp Coupled analysis with hyperelasticity.

References

- Dalrymple, T., J. Choi, and K. Miller, “Elastomer Rate-Dependence: A Testing and Material Modeling Methodology,” 172nd Technical Meeting of the Rubber Division of the American Chemical Society, Cleveland, OH, pp. 1–16, 2007.
- Press, W. H., S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in Fortran*, 172nd Technical Cambridge University Press, Cambridge, U.K, 1992.
- Radhakrishnan, R., *Coupled Thermomechanical Analysis of Viscoelastic Dampers*, Master's Thesis, State University New York, Buffalo, NY, 2000.
- Shen, K. L., and T. T. Soong, “Modeling of Viscoelastic Dampers for Structural Applications,” Journal of Engineering Mechanics, vol. 121, issue 6, pp. 694–701, 1995.

Tables

Table 1: Original and shifted storage and loss moduli from Shen and Soong (1995).

θ (°C)	f (Hz)	G′ (ksi)	G″ (ksi)	fr (Hz)	Gr′ (ksi)	Gr″ (ksi)
21	1	0.199	0.259	1.096	0.199	0.258
	1.5	0.265	0.326	1.644	0.264	0.325
	2	0.3	0.395	2.192	0.299	0.394
	2.5	0.365	0.463	2.74	0.364	0.462
	3	0.386	0.487	3.288	0.385	0.486
32	1	0.093	0.128	0.265	0.09	0.124
	1.5	0.1	0.158	0.398	0.097	0.153
	2	0.131	0.189	0.53	0.127	0.183
	2.5	0.147	0.213	0.663	0.142	0.206
	3	0.182	0.242	0.795	0.176	0.234
38	1	0.074	0.09	0.122	0.07	0.085
	1.5	0.068	0.102	0.183	0.064	0.097
	2	0.1	0.114	0.244	0.095	0.108
	2.5	0.106	0.125	0.305	0.1	0.119
	3	0.114	0.14	0.366	0.108	0.133

Table 2: Prony series parameters.

$g_1 = 0.0396$	$t_1 = 1.766$ s
$g_2 = 0.1018$	$t_2 = 0.1536$ s
$g_3 = 0.8586$	$t_3 = 0.0127$ s

Table 3: Thermal properties.

Quantity	Steel	Viscoelastic Damper
Density (lbf-s^2/in^4)	7.30×10^{-4}	9.93×10^{-5}
Thermal expansion (in/in-°C)	10.8×10^{-6}	9.0×10^{-5}
Thermal conductivity (BTU/s-in-°C)	7.0×10^{-4}	6.0×10^{-6}
Specific heat (BTU-in/lbf-s^2-°C)	76.3	300

Figures

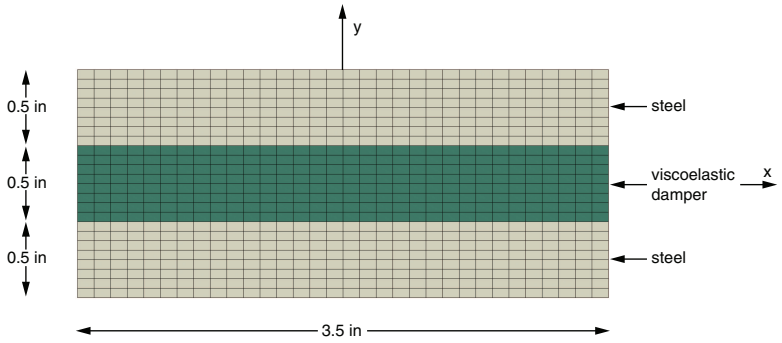


Figure 1: Viscoelastic damper geometry.

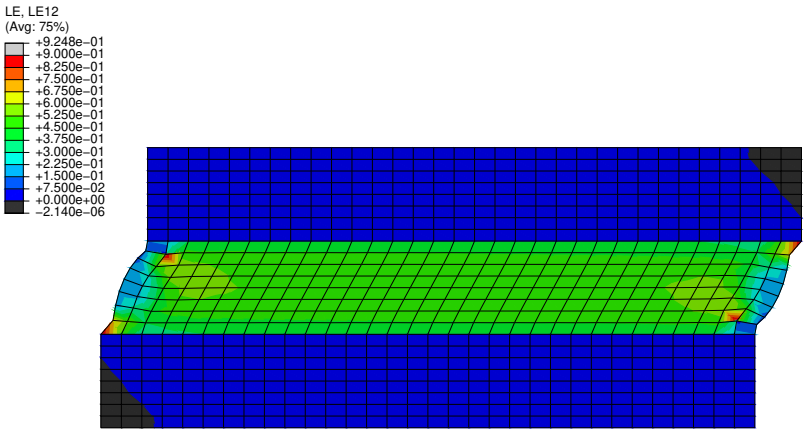


Figure 2: Shear component of the log strain at t=9.25 s for the coupled temperature-displacement model using hyperelastic materials.

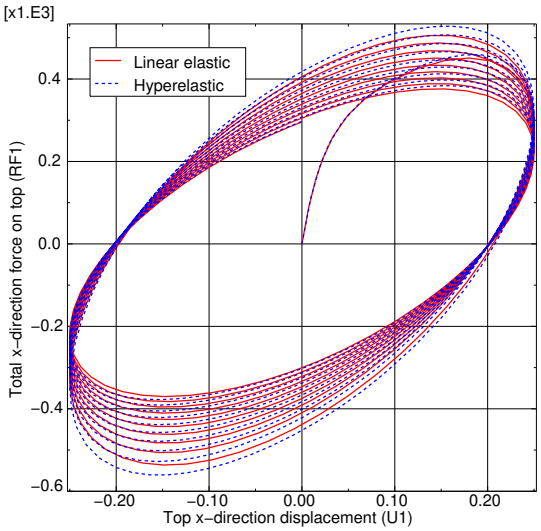


Figure 3: Hysteresis response for the coupled thermomechanical analysis.

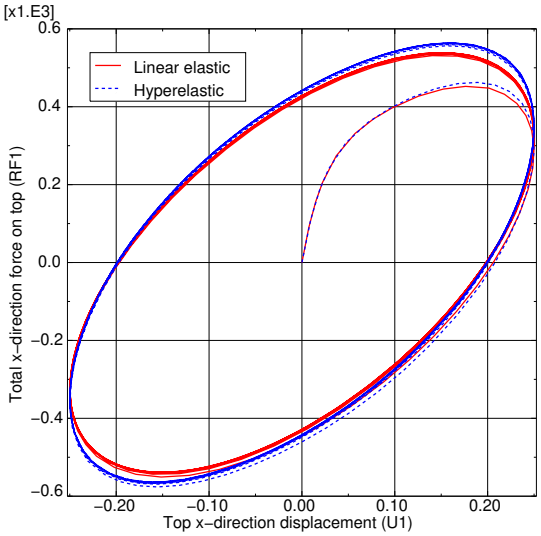


Figure 4: Hysteresis response for the pure stress analysis (no thermal coupling).

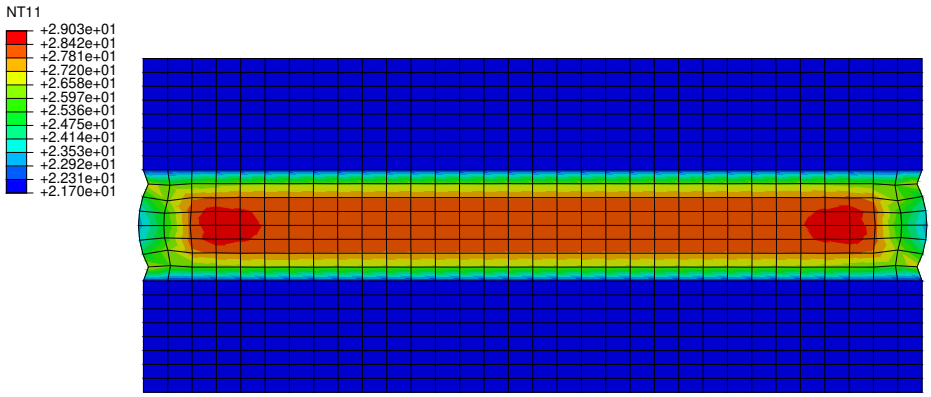


Figure 5: Temperature contour in fully coupled model with hyperelasticity at t=10 s.

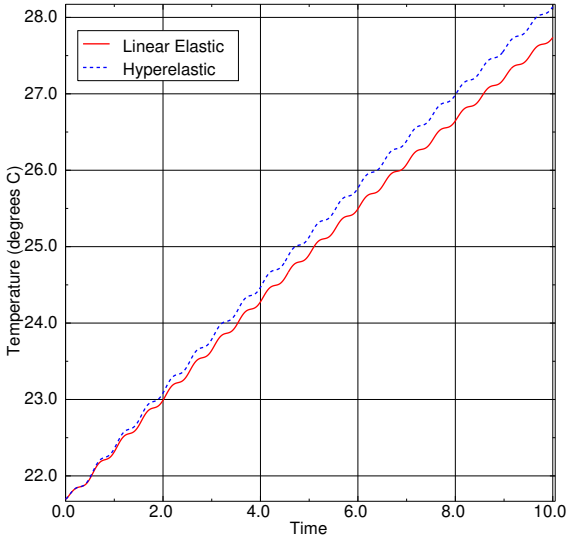


Figure 6: Temperature history at center of viscoelastic region for coupled analysis.

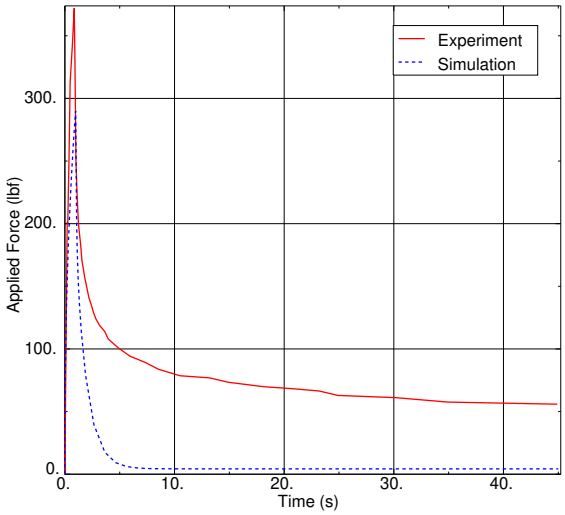


Figure 7: Comparison of relaxation response from experiments by Shen and Soong with prediction using material calibrated to capture hysteresis response.

Particle methods

In this section:

- *Mixing of granular media in a drum mixer*
- *Particle cluster drop*

Mixing of granular media in a drum mixer

Products: Abaqus/Explicit

References:

- *Discrete Element Method*
- *Discrete Particle Elements*

Overview

This example demonstrates using the discrete element method (DEM) in Abaqus to analyze mixing of different particulate media in a tumbling mill.

Application description

Rotating drum mixers and tumbling mills are used for grinding, mixing, and drying of ore and granular materials. Such applications can be found in a wide range of industrial sectors such as mining. Several factors (including shape, size, density, and contact stiffness of particles; friction; adhesion between particles, speed of rotation; and tilt of the drum axis) influence the level of mixing that will be achieved in a given amount of time. These factors also influence the amount of energy required to operate the mixer. The discrete element method (DEM) is a useful tool in understanding the effects of these factors on the mixing process. This example demonstrates the use of DEM to analyze mixing of granular media with nonadhesive as well as adhesive contact behavior.

Geometry

Figure 1 shows the drum geometry. The drum length, L, is 760 mm; the drum outer diameter, D_o, is 620 mm; and the drum mouth diameter, D_m, is 315 mm. The interior of the drum has five equally spaced baffles to aid the mixing process. The baffles taper from the rear to the front of the drum. The wall of the drum is hollow; and the drum inner radius, R, is 300 mm. The axis of the drum is inclined at an angle of 30°. Although this laboratory size drum mixer is not of the scale of an industrial mixer, it is sufficiently large to demonstrate the mixing process.

To analyze nonadhesive contact between particles, the granular media consist of two batches of spherical limestone pellets. The first batch has a mass of 16.3 kg, and each pellet has a radius of 5 mm. The second batch has a mass of 19.3 kg, and each pellet has a radius of 6 mm. To analyze adhesive contact between particles, the granular media consist of two batches of spherical polyethylene pellets. The first batch has a mass of 9.8 kg, and each pellet has a radius of 5 mm. The second batch has a mass of 11.2 kg, and each pellet has a radius of 6 mm.

Materials

The drum is made of steel with a Young's modulus of 2.08×10^5 N/mm^2, density of 7850×10^{-9} kg/mm^3, and Poisson's ratio of 0.3.

Boundary conditions and loading

Mixing of particles in a rotating drum is influenced by the drum radius, the speed of rotation, and the degree of filling of the drum. At slow rotating speeds, particles tend to slip and slump along the walls of the drum; while at very high speeds, centrifuging occurs, pushing the particles up along the drum wall. Rolling and cascading of particles in a rotating drum results in good mixing. The Froude number specifies the tendency of the particles to roll and cascade during mixing in a rotating drum. The Froude number is defined as $\omega^2 R/g$, where ω is the angular speed of the drum, R is the drum radius, and g is the acceleration due to gravity. A Froude number in

the range of 0.001–0.1 is recommended for mixing operations. In this example the reference node of the drum is given a rotating speed of slightly less than 0.25 revolutions per second, which results in a Froude number of 0.068. The two batches of pellets together occupy less than half of the interior volume of the drum (i.e., the degree of filling is less than 0.5). The entire model in this example is subjected to gravity loading.

Interactions

Two different contact conditions between particles are considered: nonadhesive contact between limestone pellets, and adhesive contact between polyethylene pellets. Contact between particles and the drum is nonadhesive. The friction coefficient for contact between particles is 0.35. For contact between particles and the drum wall, the friction coefficient is 0.3.

Abaqus modeling approaches and simulation techniques

For this analysis the drum is assumed to be a rigid body. It is meshed with shell elements and made rigid by assigning it to a rigid body. A CARDAN connection type connector element that is aligned with the drum axis is attached to the reference node of the drum. The connector element is used to apply the torque to rotate the drum. The limestone and polyethylene pellets are modeled using PD3D elements. The particles are spherical in shape. The model used in this example has 8556 PD3D elements with a 6 mm radius and 12478 PD3D elements with a 5 mm radius.

Summary of analysis cases

Case 1	Nonadhesive contact between limestone pellets.
Case 2	Adhesive contact between polyethylene pellets.

Mesh design

It is very difficult to start such a simulation with the particles positioned precisely in an equilibrated configuration. A common DEM modeling technique is used for this analysis in which arrays of particles are initially positioned in the model and allowed to settle under gravity during the first analysis step with no other loading. The desired loading response is studied in subsequent steps.

In this case, layers of nonoverlapping particles of both sizes are introduced inside the drum. The two batches of particles are initially positioned next to each other and at a certain initial height from the interior wall of the drum. Next, these two batches of particles are dropped inside the drum and allowed to settle under gravity. This is done via a dummy step of 0.5 seconds duration in which only the gravity load is active. The drum is held fixed in its initial position during this step. At the end of the gravity settling step, we have the two batches of particles in a compacted stable configuration at the lower part of the drum. The cost of gravity settling is an additional overhead incurred in most DEM analyses.

Boundary conditions

An encastre boundary condition is applied to the free end of the connector, and all translational degrees of freedom of the reference point of the rigid body are held fixed for the duration of the analysis.

Loads

A gravity load is applied to the model. An acceleration of -9800 mm/s^2 is applied in the z-direction. Velocity type connector motion with an amplitude is applied about the connector component aligned with the drum axis. The other two connector components are held fixed. Mass proportional damping is used in the analysis to reduce the analysis noise. The total time period for the analysis, which includes gravity settling and mixing, is 5.5 seconds.

Case 1 Nonadhesive contact between limestone pellets

Two batches of spherical limestone pellets are used to analyze nonadhesive contact between particles.

Materials

Limestone has a Young's modulus of 2.0×10^4 N/mm^2, density of 2500×10^{-9} kg/mm^3, and Poisson's ratio of 0.25.

Interactions

As discussed in *Discrete Element Method*, PD3D elements are rigid and the contact stiffness for interactions between DEM particles can be tuned to reflect the Hertz contact solution for contact between two elastic spheres (see Timoshenko and Goodier, 1951; and *The Hertz contact problem*). The Hertz solution relating the contact force, F, to the approach distance, δ, between remote points on two contacting spheres is

$$\mathrm{F} = \frac{4}{3}\mathrm{E}^*\sqrt{\mathrm{R}}\sqrt{\delta^3},$$

where

$$\mathrm{R} = \frac{\mathrm{R}_1\mathrm{R}_2}{\mathrm{R}_1+\mathrm{R}_2},$$

and

$$\frac{1}{\mathrm{E}^*} = \frac{1-\nu_1{}^2}{\mathrm{E}_1} + \frac{1-\nu_2{}^2}{\mathrm{E}_2}.$$

E_1, ν_1 and E_2, ν_2 are the Young's modulus and Poisson's ratio of the two particles, respectively. R_1 and R_2 are the radii of the two particles, respectively.

The equation relating F and δ was used to generate tabular force versus overclosure relationships for contact between DEM particles. Different relationships are used for different combinations of particle radii. In Abaqus force-overclosure tables are specified under tabular type pressure-overclosure contact surface behavior since the particle contact area is unity.

A simpler alternative is to specify an approximate linear contact stiffness over the range of δ of interest. The Hertz contact stiffness is not linear because F is not linearly dependent on δ. The normal contact stiffness (or slope of the F vs. δ curve) for a given value of indentation between particles is

$$\mathrm{K} = \frac{d\mathrm{F}}{d\delta} = 2\mathrm{E}^*\sqrt{\mathrm{R}}\sqrt{\delta}.$$

In Abaqus the contact stiffness can be specified using a linear type pressure-overclosure contact surface behavior. The maximum indentation δ_{max} can be assumed to be a certain percentage of the particle radius (between 0.05 and 1.0 percent). Substituting a value such as $\delta = 0.002$ into the expression for K may provide a linear contact stiffness that is adequately representative. For a given particle mass, a larger contact stiffness would require a smaller time increment. For a slow mixing process (i.e., the Froude number is at the low end of the mixing operation range), it may be possible to get reasonably accurate results with a lower contact stiffness and using larger time increments.

Case 2 Adhesive contact between polyethylene pellets

Two batches of spherical polyethylene pellets are used to analyze adhesive contact between particles.

Materials

Polyethylene has a Young's modulus of 1.0×10^3 N/mm^2, density of 1400×10^{-9} kg/mm^3, and Poisson's ratio of 0.3.

Interactions

The Johnson-Kendall-Roberts (JKR) adhesive contact model discussed in *Discrete Element Method* is used for modeling adhesive contact between PD3D elements. The surface energy per unit area, $\mathbf{\Gamma}$, is used to specify adhesion between contacting particles; in this case, $\mathbf{\Gamma}$ = 50 J/m^2. Increasing the value of $\mathbf{\Gamma}$ leads to stronger adhesive forces between particles. The JKR model reduces to nonadhesive Hertz contact for $\mathbf{\Gamma = 0}$. Nonadhesive contact between polyethylene pellets is modeled ($\mathbf{\Gamma}$ = 0 J/m^2) for comparison with adhesive contact results.

Discussion of results and comparison of cases

Figure 2 shows a series of deformed plots obtained at different times during the mixing analysis of limestone pellets. The initially generated mesh can be seen in the plot at 0.0 seconds. The configuration at the end of the gravity settling step can be seen in the plot at 0.5 seconds. The remaining four plots show the progress of the mixing process and the rolling and cascading of the particulate media. *Figure 3* shows the total energy input to the mixing process as a function of time.

Figure 4 shows a comparison of the mixing process between the nonadhesive and adhesive mixing of polyethylene pellets. A series of deformed plots obtained at different times during the mixing analysis for the two conditions are juxtaposed in the figure. The steel drum has been removed from the plots for clarity. The left column of plots in *Figure 4* corresponds to nonadhesive contact, and the right column of deformed plots corresponds to adhesive contact. The configuration at the end of the gravity settling step can be seen in the plot at 0.5 seconds. The remaining five plots show the progress of the mixing process. As can be seen from the figure, adhesion reduces the level of mixing. Also, the rolling and cascading of the particulate media that is observed in the modeling of nonadhesive contact for the polyethylene pellets is absent for adhesive contact.

Input files

Case 1

rotating_drum_mixer.inp	DEM nonadhesive drum mixing analysis using limestone pellets.
dem_drummixer_generator_2s_autodt.inp	DEM nonadhesive drum mixing analysis with particle generator and automatic time incrementation.

Case 2

rotating_drum_mixer_jkr.inp	DEM adhesive drum mixing analysis using polyethylene pellets.

References

- Timoshenko, S., and J. N. Goodier,Theory of Elasticity, Second edition, McGraw-Hill, New York, 1951.

Figures

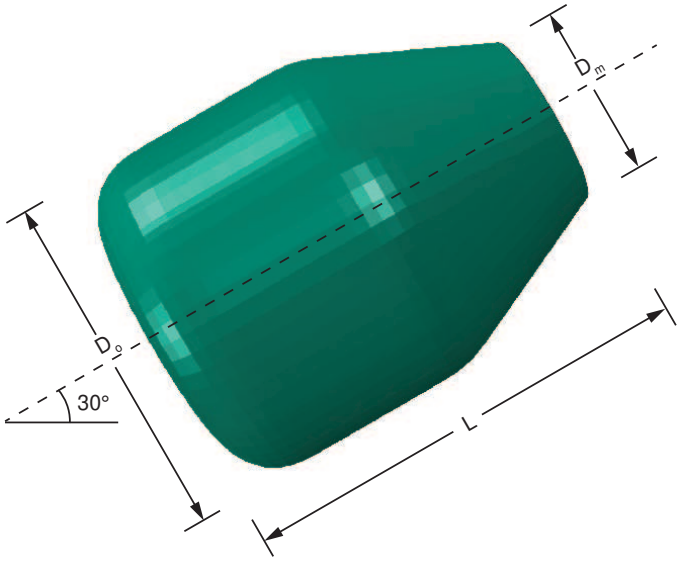


Figure 1: Mixer dimensions.

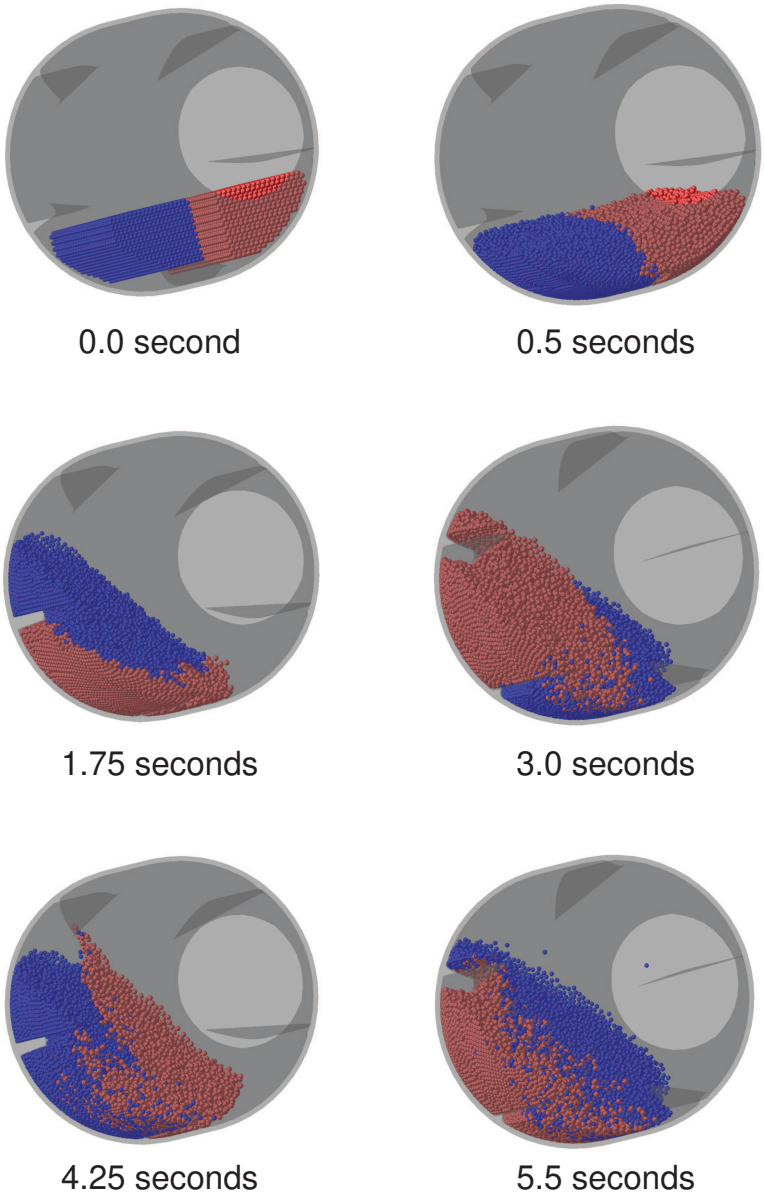


Figure 2: Case 1: Snapshots of the mixing process for limestone pellets.

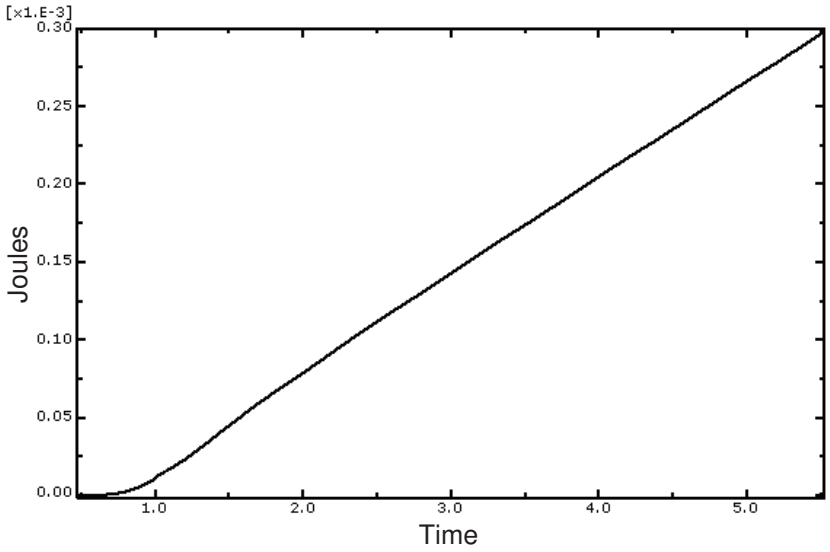


Figure 3: Case 1: Energy input to the drum mixer.

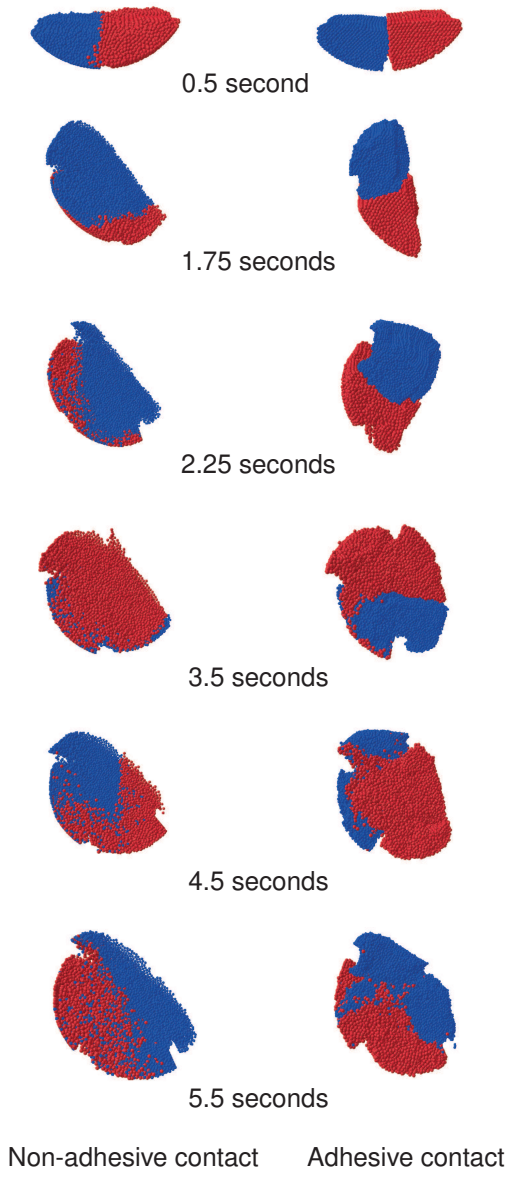


Figure 4: Case 2: Comparison of the mixing process for nonadhesive and adhesive contact between particles.

Particle cluster drop

Products: Abaqus/Explicit

References:

- *Discrete Element Method*

Overview

This example demonstrates the use of clustering in the discrete element method (DEM) in Abaqus.

Application description

In this example we simulate the deposition of a batch of gravel on a surface.

Commercial gravel is produced by crushing stones and grading the crushed aggregates by size into batches. In material handling plants, stone aggregates are transported by conveyor belts and deposited on a surface. Stone aggregates are rarely spherical in shape, and the shape and the size of aggregates influences the contact interaction between the aggregates. In general, the interlocking of irregularly shaped grains impacts the bulk behavior of granular materials.

To improve the fidelity of a discrete element analysis, it is important to describe the grain shape and the grain size as accurately as possible. In this simplified example a collection of limestone aggregates is dropped onto a rigid floor under gravity loading. We use particle clustering to describe the nonspherical shape of the aggregates.

Geometry

In general, commercially produced aggregates exhibit less variation in size and shape compared to naturally occurring gravel. For this simplified model we assume that all aggregates have an identical oblong geometry. The length of each aggregate is 40 mm, and the width is 20 mm. *Figure 1* shows the shape and size of the aggregate. Since clustering might involve overlapped particles, computing the mass and inertia of a cluster directly from individual spherical particles results in incorrect mass and inertia values for the original gravel geometry. Instead, you should directly specify the mass and inertia values of the original oblong shaped body. The oblong shape here consists of a cylinder and two hemispheres. Referring to the radius, R, and length, h, of the oblong aggregate (as shown in *Figure 1*) and using the density, , of limestone, we can compute the mass and inertia of the aggregate using the following equations:

$$M_{hemisphere} = \frac{2}{3}\pi R^3 \rho$$
$$M_{cylinder} = \pi R^2 h \rho$$
$$Mass = M_{cylinder} + 2M_{hemisphere}$$

The oblong body is aligned along the y-axis. The inertia about the y-axis can be computed using the following equations:

$$I_y^{cylinder} = \frac{1}{2} M_{cylinder} R^2$$
$$I_y^{hemisphere} = \frac{2}{5} M_{hemisphere} R^2$$
$$I_y = I_y^{cylinder} + 2I_y^{hemisphere}$$

The inertias about the x- and z-axes can be computed using the following equations:

$$I_x^{hemisphere} = I_z^{hemisphere} = \frac{83}{320} M_{hemisphere} R^2$$
$$I_x^{cylinder} = I_z^{cylinder} = \frac{1}{12} M_{cylinder} \left(3R^2 + h^2\right)$$
$$I_x = I_z = I_x^{cylinder} + 2I_x^{hemisphere} + 2M_{hemisphere} \left(\frac{3}{8}R + \frac{h}{2}\right)^2$$

Plugging in the values of R and h shown in *Figure 1* and using density $= 2500 \times 10^{-9}$ kg/mm^3, the mass of each cluster computes to 2.61797×10^{-05} kg and the mass moments of the inertia components are $I_{xx} = 0.003168$ kg mm^2, $I_{yy} = 0.001204$ kg mm^2, and $I_{zz} = 0.003168$ kg mm^2. All other components of the mass moment of inertia have a value of zero.

Abaqus modeling approaches and simulation techniques

The discrete element method (DEM) is used to simulate the piling of the aggregates being dropped onto a rigid floor under gravity.

Mesh design

We model the oblong aggregate as shown in the left image of *Figure 1* with a DEM cluster. Each cluster consists of one parent particle and two child particles. The parent particle has a radius of 10mm. The parent particles are defined as PD3D elements. The child particles have the same size as the parent particles, and they overlap with the parent particle by one particle radius. The child and parent particles are aligned collinearly with the child particles positioned on the opposite ends of the parent particle.

The right image in *Figure 1* shows the different parameters that are used to define the geometry of the cluster in a local axes system that is centered on the parent particle. In this model the local axes is aligned with the global axes. The parameters α_1 and α_2 define the offset of the two child particles, respectively, in terms of the radius R_p of the parent particle. Similarly, parameters β_1 and β_2 define the size of the two child particles, respectively, in terms of R_p. The orientation of the two child particles in the local axes is defined by the spherical angles θ_1,θ_2, β_1, and β_2, respectively. *Figure 1* also shows the values of all the parameters used in the model.

For simplicity, we use only three particles to approximate the oblong shape of the cluster. While the ends of the cluster are smooth, the sides of the cluster have a dimpled shape as can be see in *Figure 1*. The shape of the aggregate can be described more accurately using additional child particles. Even with many child particles, the sides have some degree of nonsmoothness. This is a limitation of clustering spherical particles to represent any smooth geometry.

There are a total of 4839 clusters in the model. The clusters are initially positioned in layers above the plate as shown in *Figure 2*. The closest layer is 100 mm above the plate, and the farthest layer is 800 mm above the plate. The floor is meshed using S4R elements.

Materials

The particle clusters in this model have properties of limestone, and the plate is modeled with steel.

	Limestone	Steel
Young's modulus	2.0×10^4 N/mm^2	2.08×10^5 N/mm^2
Density	2500×10^{-9} kg/mm^3	7850×10^{-9} kg/mm^3
Poisson's ratio	0.25	0.3

The particle clusters in the model behave like rigid bodies. The Young's modulus and Poisson's ratio values for the limestone are used for Hertzian contact interaction.

Initial conditions

The particle clusters fall freely under gravity with an initial velocity of 0.0 mm/s.

Boundary conditions

The displacement degrees of freedom at all nodes of the plate are held fixed.

Loads

The aggregates drop freely under acceleration due to gravity of 9810 mm/s^2 that acts in the vertical direction normal to the plate.

Interactions

Contact between the clusters as well as contact between the clusters and the plate uses Hertz-type contact interactions, with a coefficient of friction of 0.5. The clusters have mass proportional damping of 10.0 s^{-1}.

Analysis steps

The total duration of the analysis is 1 second. The kinetic energy is monitored to ensure that it is small at the end of the analysis, indicating that the aggregates have reached an equilibrium state.

Run procedure

The job is executed in domain parallel mode. Only the parent particles are included in the original output database (`.odb`) file; therefore, visualizing the original output database displays only the parent particles. To visualize the clusters, you run the python script `createDEMcluster_model.py` (in the directory where the output database from the job execution is located) using the following command:

```
abq python createDEMcluster_model.py demclusterdrop.odb
```

The script creates a new output database with information for all the particles in the cluster. The script reads the output database file from the original job and generates a new output database by appending `_clst` to the original name. In this example, the original output database from the job execution is named `demclusterdrop.odb`; therefore, the name of the new output database create by the script is `demclusterdrop_clst.odb`. You can open the new output database in Abaqus/Viewer to visualize all the particles in the model.

Results and discussion

Figure 3 shows the final configuration of the particle clusters after being dropped onto the floor. The peak of the pile is about 200 mm above the floor. The final configuration depends on several factors such as the geometry and initial position of the clusters, the drop height, and the friction coefficients of the particle-particle and particle-plate contact interactions. You can explore the interlocking of particle clusters further by altering these parameters.

Files

demclusterdrop.inp	DEM nonspherical limestone aggregate drop.
createDEMcluster_model.py	Python script for creating a DEM cluster model.

Figures

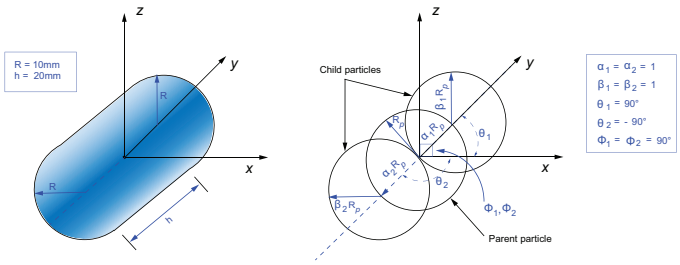


Figure 1: Aggregate geometry and cluster geometry.

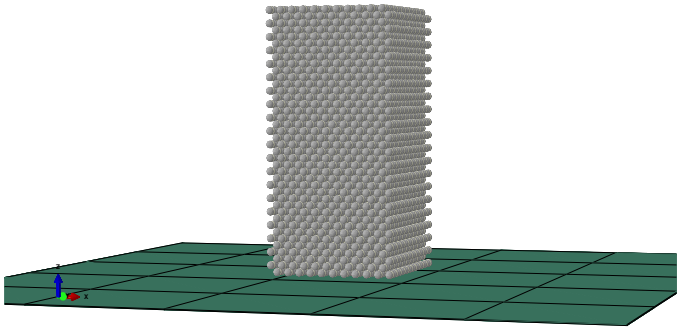


Figure 2: Initial configuration of the clusters.

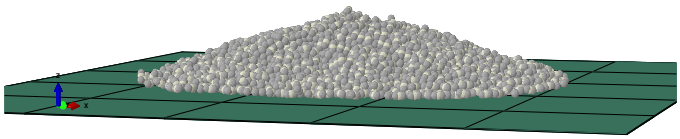


Figure 3: Final configuration of the clusters.

Additive Manufacturing Process Simulation

Additive manufacturing (AM) offers the potential to produce complex parts that are not possible to produce using traditional manufacturing methods.

In this section:

- *Sequential thermomechanical analysis of a laser powder bed fusion build*
- *Sequential thermomechanical analysis of a directed energy deposition build*

Sequential thermomechanical analysis of a laser powder bed fusion build

Products: Abaqus/Standard

References:

- *Thermomechanical Simulation of Additive Manufacturing Processes*
- *Special-Purpose Techniques for Additive Manufacturing*
- *Progressive Element Activation*
- *Thermal Loads*
- *About Table Collections, Parameter Tables, and Property Tables*
- *About Event Series*

Overview

This example illustrates the sequential thermomechanical analysis of a laser powder bed fusion three-dimensional build of a bridge structure.
The model in this problem is created based on the AMB2018-01 Additive Manufacturing benchmark problem published by the National Institute of Standards and Technology (NIST). A transient heat transfer analysis is performed first taking into account the material deposition sequence and the scanning path of the laser beam. The temperature field of the transient heat transfer analysis is then used to drive a subsequent static structural analysis. The predicted results of distortions and residual elastic strains show good correlation with the benchmark test data.

This example demonstrates the following Abaqus features and techniques:

- using temperature-dependent thermal and mechanical properties;
- performing thermomechanical simulation of additive manufacturing processes, including techniques of progressive element activation, progressive heating by a moving nonuniform heat flux, and progressive cooling on evolving free surfaces; and
- using special-purpose techniques for additive manufacturing.

Application description

Additive manufacturing (AM) technology has revolutionized design and manufacturing. Laser powder bed fusion (LPBF) is one of the common additive manufacturing technologies. During laser powder bed fusion, the recoater first deposits a thin layer of material powder, and then the laser beam scans along a predefined path acting as a heat source to melt and bind the powdered material into a solid structure. This process is repeated until the desired three-dimensional part is printed layer by layer.

Geometry

As shown in *Figure 1*, the bridge structure is 75 mm long, 5 mm wide, and 12 mm tall, with twelve legs that are of three different sizes. The bridge is positioned on a build plate. The length, width, and height of the build plate are 90mm, 15mm, and 12.7 mm, respectively.

Laser scan strategy and parameters

For each layer, a contour scan is followed by an infill scan. The contour scan laser travels at a speed of 900 mm/s using a power of 100 W. The infill scan laser has a speed of 800 mm/s and a power of 195 W. The laser diameters

for both contour and infill are 50 μm. For odd-numbered layers, the infill scans are horizontal lines (parallel to the *X*-axis) that are separated by 0.1 mm (hatch spacing), as shown in *Figure 2*. For even-numbered layers, the infill scans are vertical lines (parallel to the *Y*-axis) that are also separated by 0.1 mm, as shown in *Figure 3*.

Recoater parameters

The recoating blade spreads powder across the powder bed surface at a speed of 80 mm/s. Each powder layer is 0.02 mm thick, and a total of 625 layers is needed to build the bridge structure. The average printing time for each layer in the legs (z=0.02 mm to z=5.00 mm) is 52 s.

Abaqus modeling approaches and simulation techniques

Abaqus/Standard provides a general framework for modeling common additive manufacturing processes, such as laser powder bed fusion. At the core of the Abaqus additive manufacturing technology is the toolpath-mesh intersection module—a powerful geometry-based engine that takes process toolpath data as input and intersects it with an arbitrary mesh. The time-location history of both material deposition sequence (recoater motion) and heating source (laser beam scanning path) can be represented by an event series in the form of a table of time, spatial coordinates, and process parameters. The toolpath-mesh intersection module takes the event series data and automatically computes all of the information required to activate elements and apply the proper thermal energy to the model. Table collections that encapsulate parameter tables or property tables can be used to define additional process parameters needed for the simulation.

The analyses in this example use the streamlined solution for powder bed–type additive manufacturing processes in Abaqus/Standard (see *Thermomechanical Analysis of Powder Bed–Type Additive Manufacturing Processes Using the Trajectory-Based Method*).

Progressive element activation

The toolpath-mesh intersection module uses the current position of the recoater and the finite element mesh of the bridge structure to compute which elements are active at any given time. These elements are included in a specific progressive element activation definition so that they can be activated in the required time increment. Full activation is used in this analysis. When an element is activated, its volume fraction is set to one. (See *Progressive Element Activation*.)

Progressive heating

The energy deposition into the system is computed by taking into account the actual path of the heat source. The toolpath-mesh intersection module provides the information pertaining to the energy deposition. The laser heat source can be modeled as a concentrated moving heat flux. (See *Specifying a Concentrated Moving Heat Source*F.)

Progressive cooling

In an additive manufacturing process, radiation and convection cooling take place on the continuously evolving free surfaces. Abaqus keeps track of the free surfaces that reflect the current shape of the build part. Film or radiation conditions can be applied to the facet area of an element during progressive element activation. (See *Specifying Element-Based Film Conditions on Evolving Faces of an Element in Abaqus/Standard* and *Specifying Element-Based Radiation Conditions on Evolving Faces of an Element in Abaqus/Standard*.)

Heat transfer analysis

A transient heat transfer analysis is first performed to obtain the temperature distribution during the build process. The total step time includes the addition of 600 seconds of cooling time after the last heating event defined by the laser event series.

Mesh design

The mesh of the bridge structure consists of 8-node linear diffusive heat transfer DC3D8 elements. The elements vary in size but have a common characteristic height of 0.2 mm so that there are approximately ten layers of material in each element. The build plate is more coarsely meshed with DC3D8 elements.

Materials

The material for both the build part and the plate is nickel-based superalloy IN625. The density is taken as 8.44×10^{-9} ton/mm^3. The liquidus temperature is 1350°C, the solidus temperature is 1290°C, and the latent heat of fusion is 272×10^9 mJ/tonne.

The temperature-dependent thermal conductivity and specific heat are shown in *Table 1* (source: *http://www.specialmetals.com/assets/smc/documents/alloys/inconel/inconel-alloy-625.pdf*).

The Stefan-Boltzmann constant is taken as 5.67×10^{-11} mW/(mm^2·K^4). The absolute zero is set at –273.15°C. The film coefficient is 0.018 mW/(mm^2·°C). For radiation heat transfer, the emissivity is 0.45.

Initial conditions

An initial temperature of 80°C is applied to the build plate. The initial temperature of the build part is 40°C, which corresponds to the temperature of the powder material as it is spread from the powder bed reservoir.

Loads

The bottom of the build tray is held at a constant 80°C. Film and radiation conditions are applied to the evolving free surfaces during the build process. The reference ambient temperature is taken to be 40°C. The table collection `ABQ_AM.Moving Heat Flux.1` referenced from the distributed flux loading defines the laser as a moving heat source.

Constraints

The connection between the bottom surface of the bridge and the top surface of the build plate is modeled as a tie constraint with the top surface of the plate as the main surface.

Output requests

Nodal temperature (NT) is requested at time points corresponding to each layer's build time.

Structural analysis

Mesh design

Full-integration solid C3D8 elements are used for both the bridge and the build plate, maintaining the same mesh topology used in the heat transfer analysis.

Materials

The temperature-dependent coefficient of thermal expansion, elastic modulus, and Poisson's ratio are shown in *Table 1*. Orthotropic plasticity with hardening is used. The material's yield stress is 725 MPa in the horizontal x- and y-directions and 615 MPa in the vertical z-direction. The stress ratio, R_{33}, is 0.8483. The stress corresponding

to a plastic strain of 0.35 is 990 MPa in the horizontal directions, and the stress in the vertical direction is scaled with the same stress ratio as that used for the yield stress (source: *http://www.eos.info/material_m/werkstoffe/download/NickelAlloy_IN625.pdf*).

Initial conditions

In the structural analysis the initial temperature of the build part represents a relaxation temperature (not room temperature) above which thermal straining induces negligible thermal stress (see *Controlling the Scale of the Simulation and the Solution Fidelity*). Upon material activation, it represents the temperature from which the initial thermal contraction occurs. In this analysis the initial temperature of the bridge is set to 750°C. The initial temperature for the build plate is 80°C.

Boundary conditions

The bottom of the build plate is fixed in all degrees of freedoms.

Predefined fields

The thermal results for each increment during the previous transient heat transfer analysis are applied to the structural analysis as predefined fields. Abaqus automatically maps the nodal values of temperature by interpolation (both in space and time) of the previous results.

Constraints

Similar to the heat transfer analysis, a tie constraint is used to model the connection between the bottom surface of the bridge and the top surface of the build plate, with the top surface of the plate as the main surface.

Output requests

Nodal displacement (U) and displacement measured from the time the node is activated (UACT) are requested at time points corresponding to the build time of every eighth layer to reduce the size of the output database. Stress (S) and elastic strain (EE) are requested at the end of printing and after cutting. You can also request logarithmic strain (LE), plastic strain (PE), equivalent plastic strain (PEEQ), and the volume fraction of the material in the current element (EACTIVE).

Discussion of results and comparison of cases

In the physical test the part is cut via wire electron discharge machining (EDM) such that only the end portion of the part remains attached to the plate. In the simulation the cut process is modeled in a separate step with the model change feature to remove a layer of elements near the bottom of the legs. The cut section of the part deflects upward from relaxation of the as-built residual stresses.

Distortion

The distortion values are measured by the difference of the vertical deflection after and before the cut for the top of 11 ridges along the center plane. Thus, $\delta_i = z_i^{after} - z_i^{before}$, where δ_i is the vertical deflection of edge i. The distortions along the path are shown in *Figure 4*. The solid blue curve is the simulation result, and the dashed red curve is the benchmark test data.

Residual elastic strain

Residual elastic strain contour plots within the as-built bridge structure are generated and compared with the x-ray diffraction measurements results. *Figure 5* and *Figure 6* (*https://www.nist.gov/ambench/results-chal-amb2018-01-rs-part-residual-strains*) show the simulation results and benchmark test results of the elastic strain in the x-direction. The simulation results and benchmark test results of the elastic strain in the z-direction are shown in *Figure 7* and *Figure 8* (*https://www.nist.gov/ambench/results-chal-amb2018-01-rs-part-residual-strains*). *Figure 9* and *Figure 10* show the comparison between x-ray diffraction measurements and simulation results of the strain in the z-direction along the z=10.75 mm and z=2.75 mm paths in the center plane. Good correlation is found between the simulation results and the benchmark test data.

Acknowledgements

SIMULIA gratefully acknowledges TWI Ltd for their collaboration in developing this model and the National Institute of Standards and Technology for creating the Additive Manufacturing benchmark test series.

Figure 1, *Figure 2*, and *Figure 3* are provided courtesy of the NIST website (*https://www.nist.gov/ambench/amb2018-01-description*).

Figure 6 and *Figure 8* are provided courtesy of the NIST website (*https://www.nist.gov/ambench/results-chal-amb2018-01-rs-part-residual-strains*).

Input files

am_bridge_thermal_analysis.inp	Heat transfer analysis.
am_bridge_structural_analysis.inp	Static structural analysis.

The following input files contain definitions or data included in the input files listed above:

ABQ_am_special_purpose_types.inp	Types of property tables, parameter tables, and event series used by the special-purpose techniques for the simulation of common additive manufacturing processes in Abaqus.
node_bridge_part.inp	Node definitions of the bridge structure.
elem_bridge_part.inp	Element definitions of the bridge structure.
node_bridge_substrate.inp	Node definitions of the build plate.
elem_bridge_substrate.inp	Element definitions of the build plate.
es_bridge_roller.inp	Event series data of the roller motion.
es_bridge_laser.inp	Event series data of the laser scanning path.

References

- *https://www.nist.gov/ambench/amb2018-01-description*
- *https://www.nist.gov/ambench/benchmark-test-data*
- *http://www.specialmetals.com/assets/smc/documents/alloys/inconel/inconel-alloy-625.pdf*
- *http://www.eos.info/material_m/werkstoffe/download/NickelAlloy_IN625.pdf*

Tables

Table 1: Temperature-dependent material properties of IN625.

Temperature	Thermal Conductivity (mW/(mm·°C))	Specific Heat (mJ/(tonne·°C))	Coefficient of Thermal Expansion (1/°C)	Elastic Modulus (MPa)	Poisson's Ratio
21 °C	9.8	4.1×10^8	–	2.075×10^5	0.278
93 °C	10.8	4.27×10^8	1.28×10^{-5}	2.041×10^5	0.28
204 °C	12.5	4.56×10^8	1.31×10^{-5}	1.979×10^5	0.286
316 °C	14.1	4.81×10^8	1.33×10^{-5}	1.917×10^5	0.29
427 °C	15.7	5.11×10^8	1.37×10^{-5}	1.855×10^5	0.295
538 °C	17.5	5.36×10^8	1.4×10^{-5}	1.786×10^5	0.305
649 °C	19	5.65×10^8	1.48×10^{-5}	1.703×10^5	0.321
760 °C	20.8	5.9×10^8	1.53×10^{-5}	1.606×10^5	0.34
871 °C	22.8	6.2×10^8	1.58×10^{-5}	1.475×10^5	0.336
927 °C	–	–	1.62×10^{-5}	–	–
982 °C	25.2	6.45×10^8	–	–	–
1093 °C	–	6.7×10^8	–	–	–

Figures

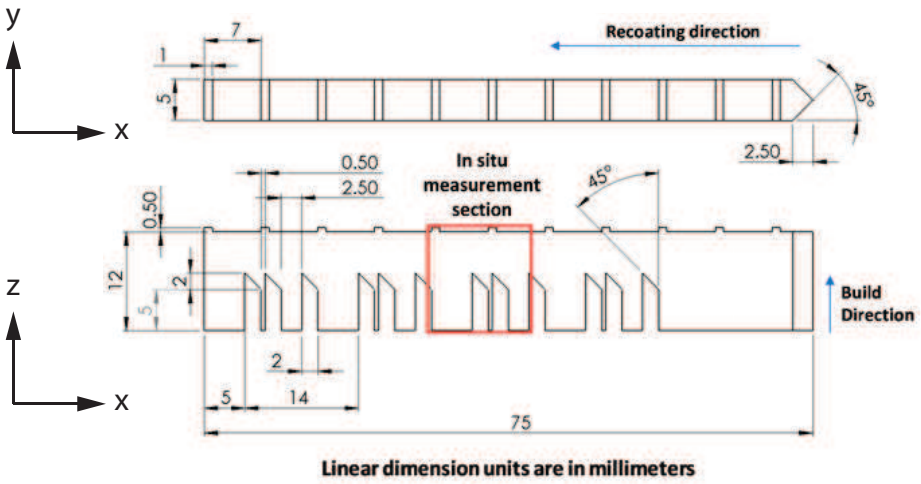


Figure 1: Schematic of the bridge structure as given by NIST.

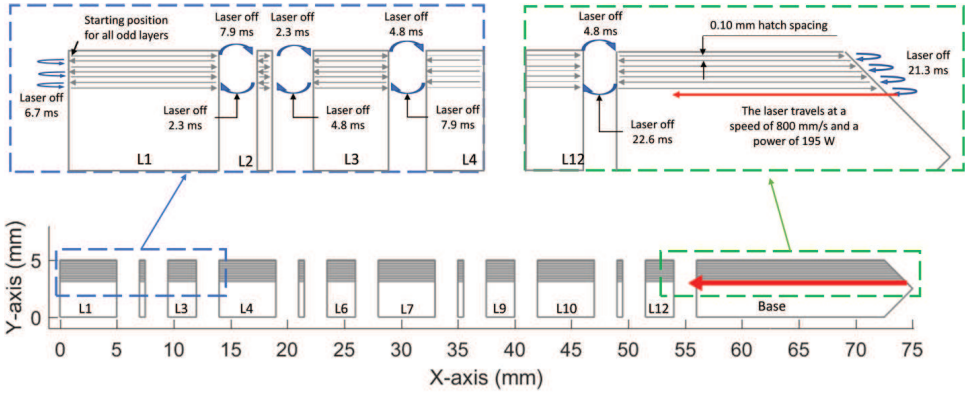


Figure 2: Odd-numbered layer scan strategy as given by NIST.

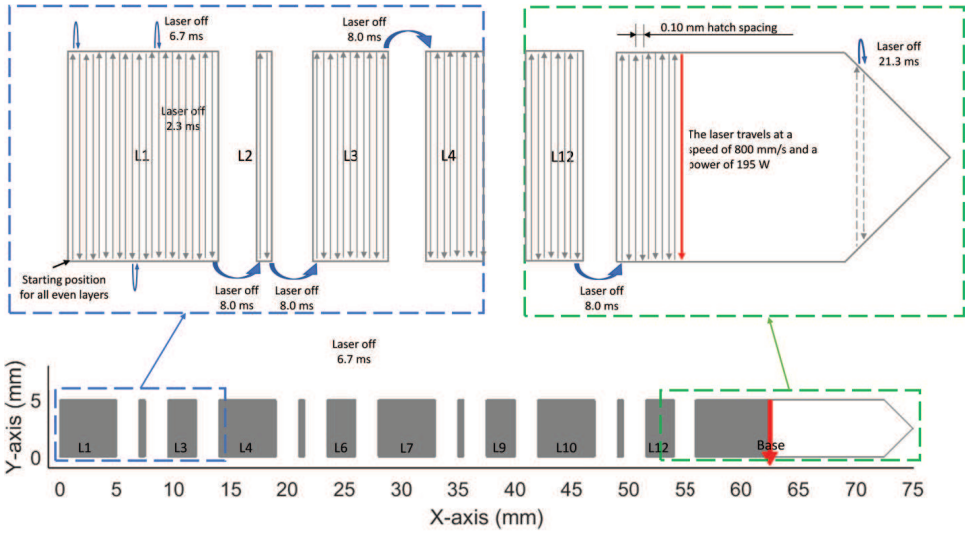


Figure 3: Even-numbered layer scan strategy as given by NIST.

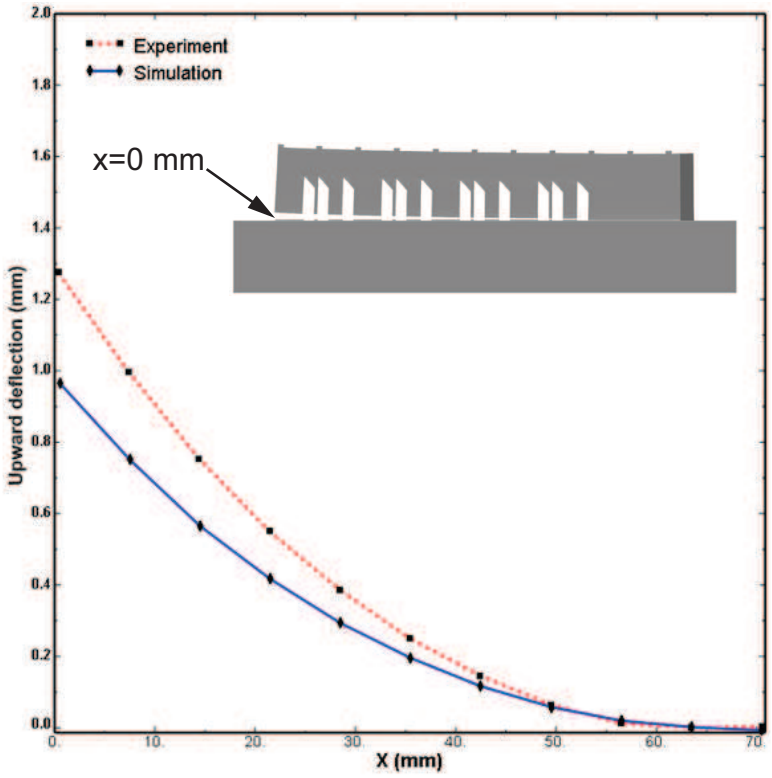


Figure 4: Upward deflection of the top of 11 ridges in the center plane.

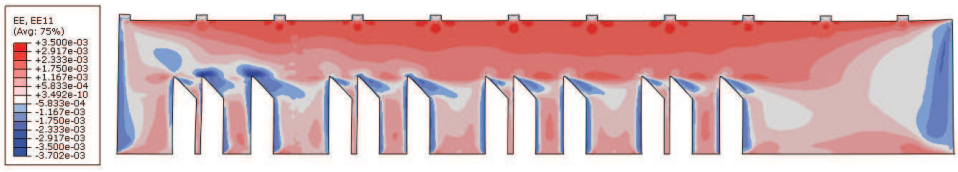


Figure 5: Simulation result of elastic strain in the x-direction.

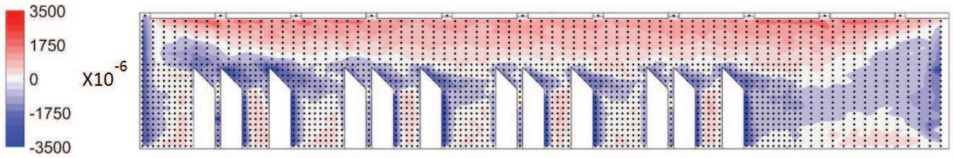


Figure 6: X-ray diffraction measurement of elastic strain in the x-direction as published by NIST.

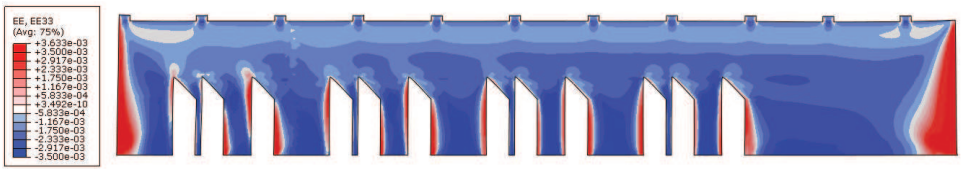


Figure 7: Simulation result of elastic strain in the z-direction.

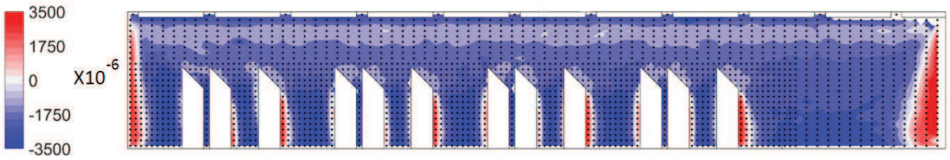


Figure 8: X-ray diffraction measurement of elastic strain in the z-direction as published by NIST.

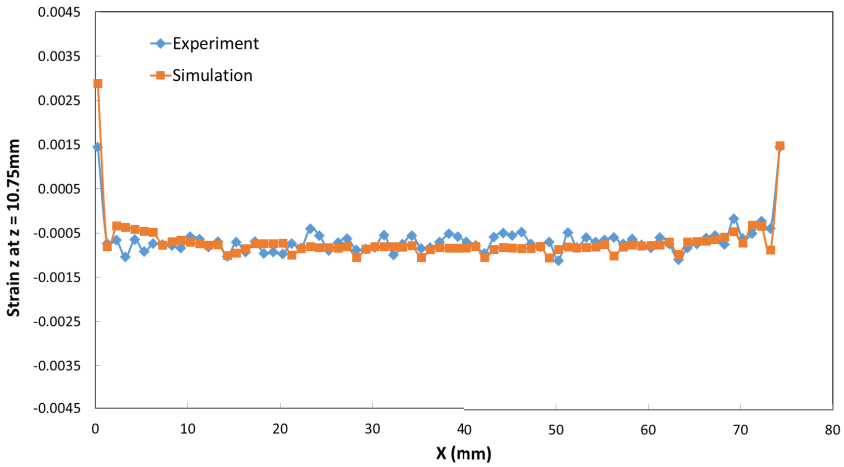


Figure 9: X-ray diffraction measurement and simulation result of elastic strain in the z-direction along the z=10.75 mm path.

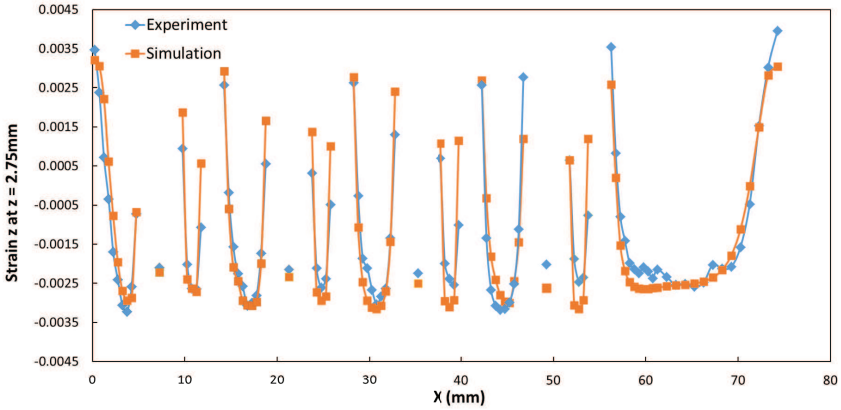


Figure 10: X-ray diffraction measurement and simulation result of elastic strain in the z-direction along the z=2.75 mm path.

Sequential thermomechanical analysis of a directed energy deposition build

Products: Abaqus/Standard

References:

- *Thermomechanical Simulation of Additive Manufacturing Processes*
- *Special-Purpose Techniques for Additive Manufacturing*
- *Progressive Element Activation*
- *Thermal Loads*
- *About Table Collections, Parameter Tables, and Property Tables*
- *About Event Series*

Overview

This example illustrates sequential thermomechanical analyses of directed energy deposition builds of a thin-wall structure on a cantilevered substrate.
The model in this problem is created based on published experiments (Denlinger et al., 2015). The predicted results of temperature and distortions histories during printing are in good agreement with experimental measurements.

This example demonstrates the following Abaqus features and techniques:

- using temperature-dependent thermal and mechanical properties;
- performing thermomechanical simulation of additive manufacturing processes, including techniques of progressive element activation, progressive heating by a moving nonuniform heat flux, and progressive cooling on evolving free surfaces; and
- using special-purpose techniques for additive manufacturing.

Application description

Additive manufacturing (AM) technology has revolutionized design and manufacturing. Directed energy deposition (DED) is one of the common additive manufacturing technologies. During directed energy deposition, the material is deposited by a nozzle mounted on a multi-axis arm and simultaneously melted by a heat source (such as a laser or an electron beam). New material is added and solidifies in a layer-by-layer fashion until the desired three-dimensional part is built.

This example problem simulates the fabrication of a thin-wall structure on a cantilevered substrate using the directed energy deposition process (Denlinger et al., 2015). The test setup consists of an aluminum clamp, a substrate, and a wall to be built on the center of the substrate. The substrate and the wall are made of Inconel nickel-chromium alloy 625.

Geometry

As shown in *Figure 1*, the dimensions of the thin-wall structure are 101.6 mm (L) × 6.7 mm (W) × 38.1 mm (H). The dimensions of the substrate are 152.4 mm (L) × 38.1 mm (W) × 12.7 mm (H). The clamped region of the substrate is 8.46 mm long. The dimensions of the clamp are 38.1 mm (L) × 38.1 mm (W) × 28.6 mm (H).

Material deposition

The wall is built using a three-bead deposition sequence per layer and a total of 42 layers. The in-plane material deposition motion is shown in *Figure 2*. For each layer, the center bead is deposited first, followed by the two side beads. All beads in a layer are deposited in the same direction. The deposition direction alternates between layers.

The travel speed of the nozzle is 10.6 mm/second. Thus, it takes 9.58 seconds to deposit one bead. After the deposition of each bead, there is a cooling period of 4.66 seconds. Three dwell times are considered for additional cooling after the deposition of each layer: 0 seconds, 20 seconds, and 40 seconds.

The raw material (powder) is melted upon deposition by a laser with a power of 2 kW. The laser beam spot size at the part surface is 4 mm in diameter. The penetration depth of the laser is 1.1 mm.

Experimental measurements

Temperature histories were measured during the printing process using three thermocouples placed on the bottom of the substrate, away from the action zone. A laser displacement sensor was used to measure the end deflection history of the substrate. *Figure 3* shows the location of the thermocouples and the measurement location of the displacement sensor.

Abaqus modeling approaches and simulation techniques

Three pairs of sequentially coupled thermomechanical analyses are performed in Abaqus/Standard to simulate three test cases of the Inconel builds of the thin-wall structure with different interlayer dwell times.

Summary of analysis cases

Case 1	Sequential thermomechanical analysis of the build with a 0 second interlayer dwell time
Case 2	Sequential thermomechanical analysis of the build with a 20 second interlayer dwell time
Case 3	Sequential thermomechanical analysis of the build with a 40 second interlayer dwell time

The following sections discuss analysis considerations that are applicable to all cases.

Analysis Types

A transient heat transfer analysis is performed first, considering thermal loads introduced by the deposition process on the thin-wall structure. This analysis is followed by a static structural analysis that is driven by the temperature field obtained by the thermal analysis.

Analysis Techniques

The analyses use the special-purpose techniques for directed energy deposition additive manufacturing processes available in Abaqus/Standard (see *Thermomechanical Analysis of FDM- and LDED-Type Additive Manufacturing Processes*).

The wall mesh is progressively activated using full element activation (see *Progressive Element Activation*). The cross-section of a bead of material being deposited is assumed to be rectangular with dimensions 3.35 mm (W) × 0.9071 (H), which is four elements wide and one element high. The material deposition sequence is defined through an event series.

Mesh design

Figure 4 shows the finite element mesh of the model. The thin-wall structure is modeled with a uniform mesh of 8-node linear brick elements. The element size is 1.016 mm (L) × 0.838 mm (W) × 0.907 mm (H). A coarser mesh is used for the substrate and the clamp. The heat transfer analysis and the structural analysis share the same mesh strategy. DC3D8 elements are used in the heat transfer analysis, and C3D8 elements are used in the structural analysis.

Materials

The substrate and the wall are made of Inconel 625. The temperature-dependent thermal conductivity, specific heat, the coefficient of thermal expansion, elastic modulus, and yield stress are shown in *Table 1* (Denlinger and Michaleris, 2016). The density is 8.44×10^{-9} tonne/mm^3. The solidus temperature is 1290°C, the liquidus temperature is 1350°C, and the latent heat of fusion is 2.72×10^{11} mJ/tonne. The Poisson's ratio is 0.366.

The clamp is made of aluminum. Constant material properties are used:

Density	2.70×10^{-9} tonne/mm^3
Conductivity	237 mW/(mm·°C)
Specific heat	9.1×10^{8} mJ/(tonne·°C)
Elastic modulus	70×10^{3} MPa
Poisson's ratio	0.366
Coefficient of thermal expansion	2.31×10^{-5} /°C

Analysis steps

Each simulation is performed using three analysis steps. The deposition process is modeled in the first step with a small time increment of 0.5 seconds. The second and the third steps simulate additional cooling periods after the built with larger time increments, 10 seconds and 100 seconds, respectively. The total time for cooling is 10,500 seconds.

Heat transfer analysis

Initial conditions

Newly deposited material comes in at room temperature, 26°C. The initial temperature of the clamp and the substrate are also at room temperature.

Loads

A moving heat flux with a Goldak distribution is used to model the heating by the laser upon deposition (see *Specifying a Moving Heat Source with a Goldak Distribution*). The laser beam spot at the intersection with the part surface is assumed to be circular. The laser scanning path is defined through the same event series that defines the material deposition sequence. The energy absorption efficiency is calibrated to be 40% for all cases.

With the deposition of new material during printing, previously exposed material surfaces are covered and new free surfaces are created. Surface convection and radiation are defined on the continuously evolving free surfaces (see *Specifying Element-Based Film Conditions on Evolving Faces of an Element in Abaqus/Standard* and *Specifying Element-Based Radiation Conditions on Evolving Faces of an Element in Abaqus/Standard*). The ambient temperature is 26°C. The emissivity is 0.28. The film coefficient is 0.018 mW/(mm^2·°C).

Output requests

Nodal temperature (NT) field output is requested for the whole model at every increment of the analysis for use in the subsequent structural analysis. In addition, nodal temperature (NT11) history output is requested for the three nodes at the locations where the three thermocouples were placed in the experiments.

Static structural analysis

Initial conditions

Based on the mesh size and the time incrementation used, the analyses presented in this example can be categorized as part-level simulations of additive manufacturing processes. To capture the melting effect in the structural analysis accurately, it is often necessary to assign an initial temperature representing a relaxation temperature above which thermal straining induces negligible thermal stress (see *Controlling the Scale of the Simulation and the Solution Fidelity*). In the structural analysis, the initial temperature of the wall is set to the melting temperature of the material, 1290°C. The substrate and the clamp are initially at the room temperature, 26°C.

Boundary conditions

All degrees of freedom of the nodes on the bottom and top surfaces of the clamp are fixed.

Predefined fields

Nodal temperatures stored in the output database (`.odb`) file of the previous transient heat transfer analysis are read as a predefined field. Abaqus automatically maps the nodal values of temperature by interpolation (both in space and time) of the previous results.

Output requests

Nodal displacement (U), stress (S), strain(E), and equivalent plastic strain (PEEQ) field output are requested for the whole model. In addition, nodal displacement (U3) history output is requested for the node at the location where the deflection of the substrate was measured in the experiments.

Discussion of results and comparison of cases

As shown in *Figure 5*, the simulations of the temperature histories of the three locations on the bottom of the substrate agree well with the in-situ experimental measurements for all cases. The agreement in temperature histories at locations that are away from the action zone indicates that the heat energy balance of the system, including heat energy input by the laser, thermal conduction, and cooling by convection and radiation, is well captured.

Figure 6 compares the simulated and measured deflections of the free end of the substrate for all cases. The oscillation due to the alternating deposition and cooling periods and the accumulated deflection of the substrate are well captured. The substrate bends downward during deposition due to a larger thermal expansion of the top surface relative to the bottom surface, while it bends upward during the cooling period because the substrate cools down and the deposited material also starts to contract (Denlinger et al., 2015). The final distortion and residual stresses of the substrate are caused primarily by the thermal contraction of the thin-wall structure.

Files

am_lded_thinwall_inconel625_dwell0_ht.inp	Heat transfer analysis of the Inconel 625 build with a 0 second interlayer dwell time.
am_lded_thinwall_inconel625_dwell0_st.inp	Static structural analysis of the Inconel 625 build with a 0 second interlayer dwell time.
am_lded_thinwall_inconel625_dwell20_ht.inp	Heat transfer analysis of the Inconel 625 build with a 20 second interlayer dwell time.
am_lded_thinwall_inconel625_dwell20_st.inp	Static structural analysis of the Inconel 625 build with a 20 second interlayer dwell time.
am_lded_thinwall_inconel625_dwell40_ht.inp	Heat transfer analysis of the Inconel 625 build with a 40 second interlayer dwell time.
am_lded_thinwall_inconel625_dwell40_st.inp	Static structural analysis of the Inconel 625 build with a 40 second interlayer dwell time.

The following input files contain definitions or data included in the input files listed above:

ABQ_am_special_purpose_types.inp	Types of property tables, parameter tables, and event series used by the special-purpose techniques for the simulation of common additive manufacturing processes in Abaqus.
mesh_thinwall_ht.inp	Node and element definitions of the model, used by the heat transfer analyses.
mesh_thinwall_st.inp	Node and element definitions of the model, used by the static structural analyses.
es_thinwall_dwell0.inp	Event series data of the material deposition (and laser scanning) motion, used by the analyses of the builds with a 0 second interlayer dwell time.
es_thinwall_dwell20.inp	Event series data of the material deposition (and laser scanning) motion, used by the analyses of the builds with a 20 second interlayer dwell time.
es_thinwall_dwell40.inp	Event series data of the material deposition (and laser scanning) motion, used by the analyses of the builds with a 40 second interlayer dwell time.

References

- Denlinger, E. R., J. C. Heigel, P. Michaleris, and T. A. Palmer, "Effect of Inter-layer Dwell Time on Distortion and Residual Stress in Additive Manufacturing of Titanium and Nickel Alloys," *Journal of Materials Processing Technology*, vol. 215, pp. 123–131, 2015.
- Denlinger, E. R., , and P. Michaleris, "Effect of Stress Relaxation on Distortion in Additive Manufacturing Process Modeling," *Additive Manufacturing*, vol. 12, pp. 51–59, 2016.

Tables

Table 1: Temperature-dependent material properties of Inconel 625 (Denlinger and Michaleris, 2016).

Temperature (°C)	Conductivity (mW/(mm·°C))	Specific Heat (mJ/(tonne·°C))	Coefficient of Thermal Expansion (1/°C)	Elastic Modulus (MPa)	Yield Stress (MPa)
20	9.9	4.10×10^{8}	1.28×10^{-5}	2.08×10^{5}	493
93	10.8	4.27×10^{8}	1.28×10^{-5}	2.04×10^{5}	479
205	12.5	4.56×10^{8}	1.31×10^{-5}	1.98×10^{5}	443
315	14.1	4.81×10^{8}	1.33×10^{-5}	1.92×10^{5}	430
425	15.7	5.11×10^{8}	1.37×10^{-5}	1.86×10^{5}	424

Temperature (°C)	Conductivity (mW/(mm·°C))	Specific Heat (mJ/(tonne·°C))	Coefficient of Thermal Expansion (1/°C)	Elastic Modulus (MPa)	Yield Stress (MPa)
540	17.5	5.36×10^8	1.40×10^{-5}	1.79×10^5	423
650	19.0	5.65×10^8	1.48×10^{-5}	1.70×10^5	422
760	20.8	5.90×10^8	1.53×10^{-5}	1.61×10^5	415
870	22.8	6.20×10^8	1.58×10^{-5}	1.48×10^5	386

Figures

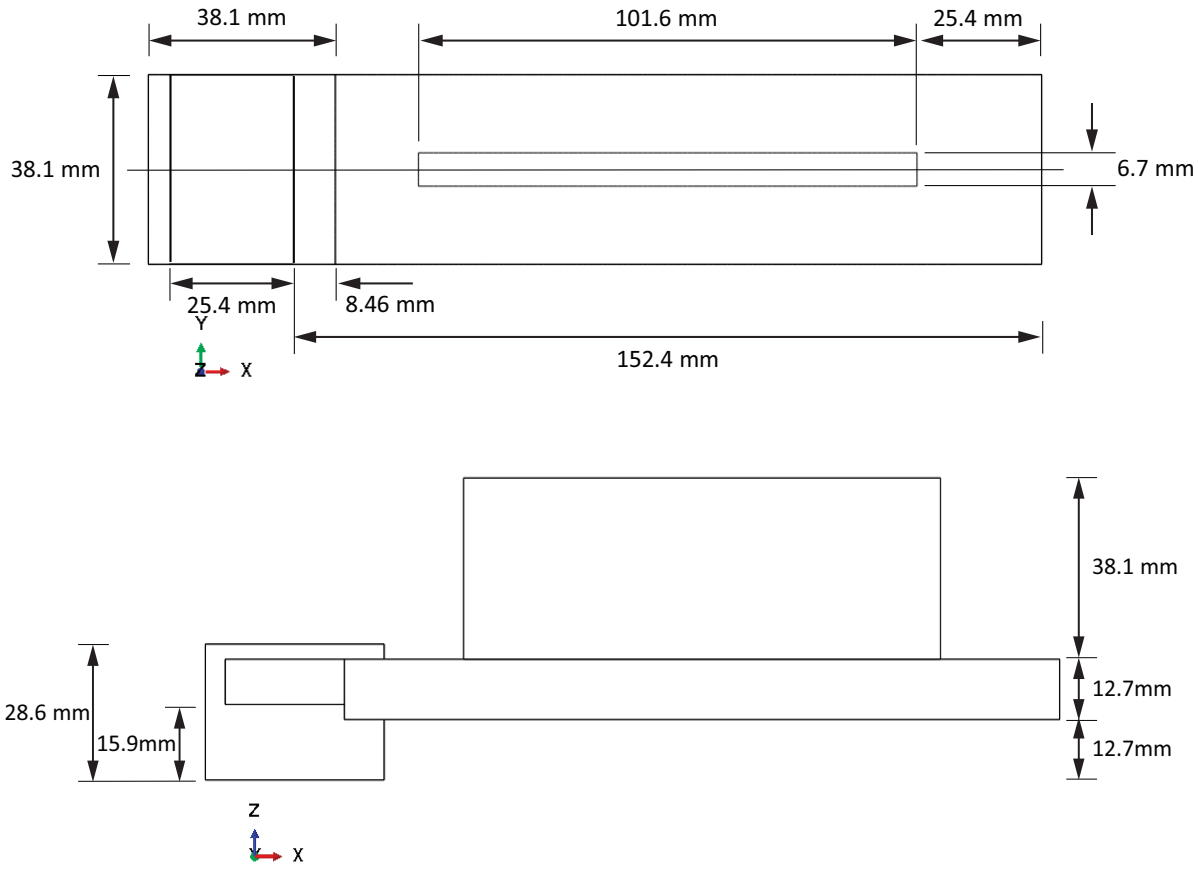


Figure 1: Dimensions (Denlinger et al., 2015).

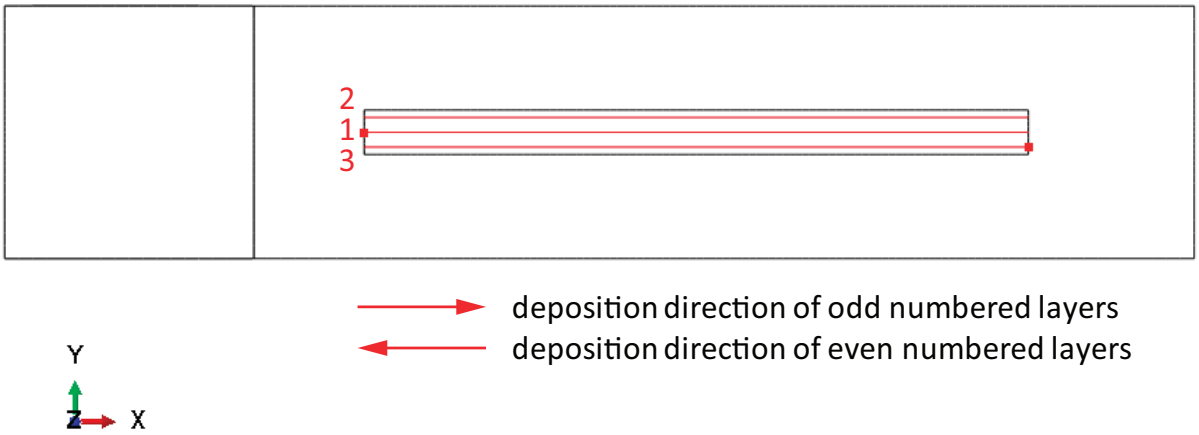


Figure 2: The material deposition (and laser scanning) path (Denlinger et al., 2015).

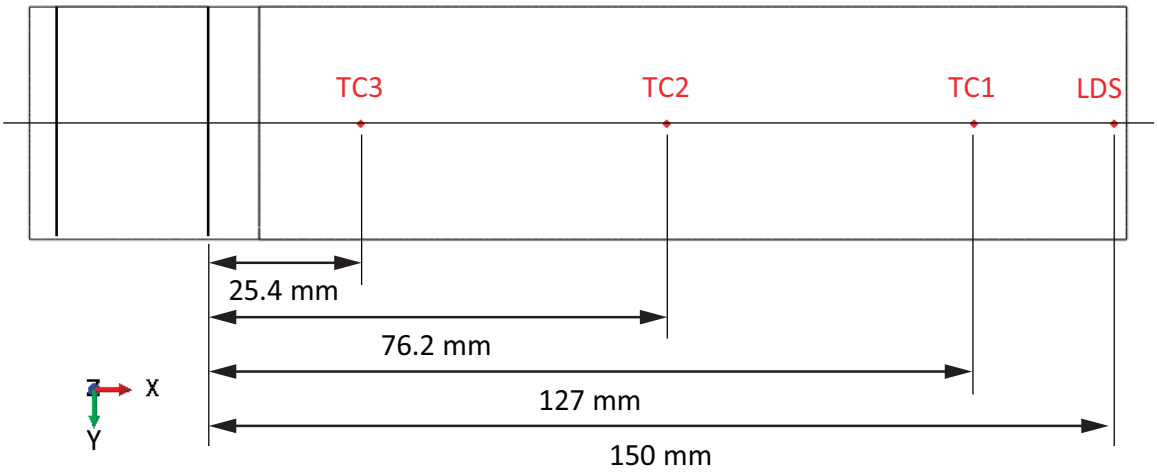


Figure 3: Locations of the thermocouples (TC) and the measurement location of the laser displacement sensor (LDS) on the bottom of the substrate (Denlinger et al., 2015).

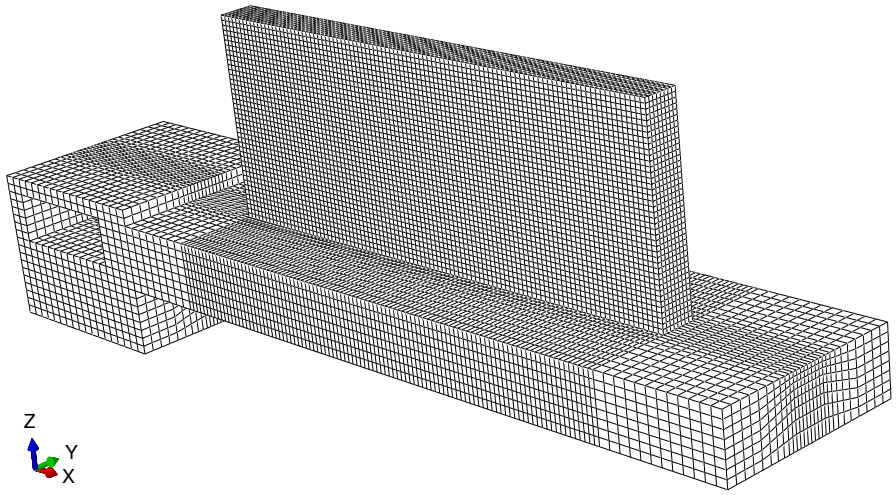
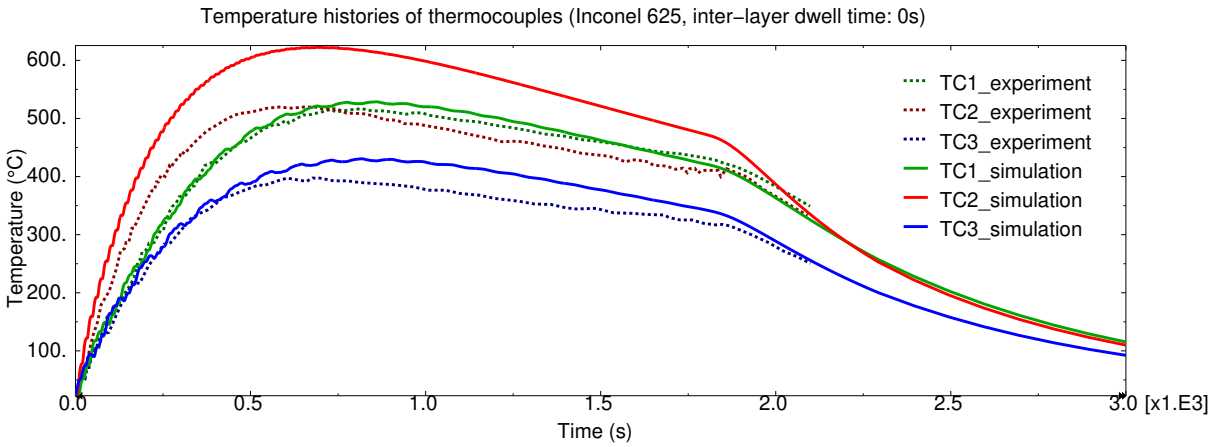
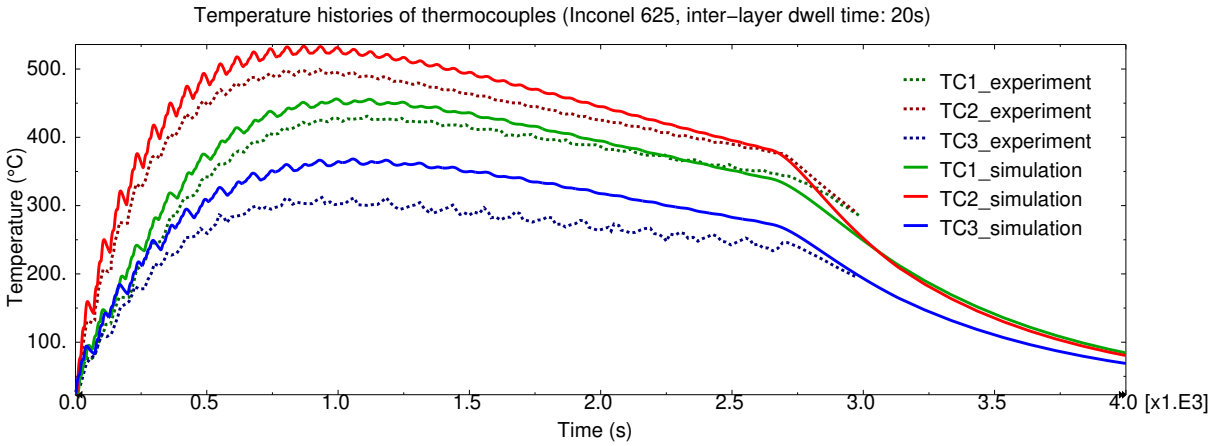


Figure 4: Finite element mesh.





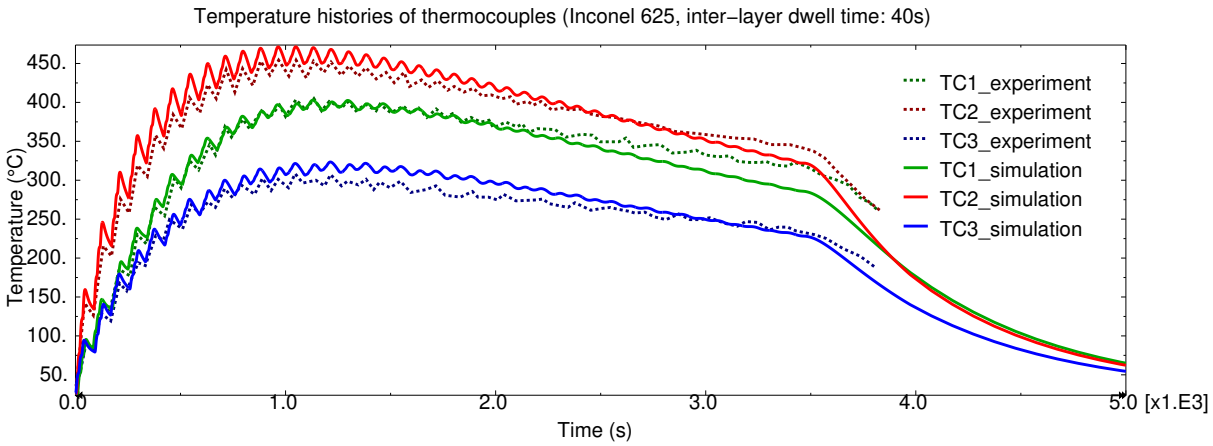
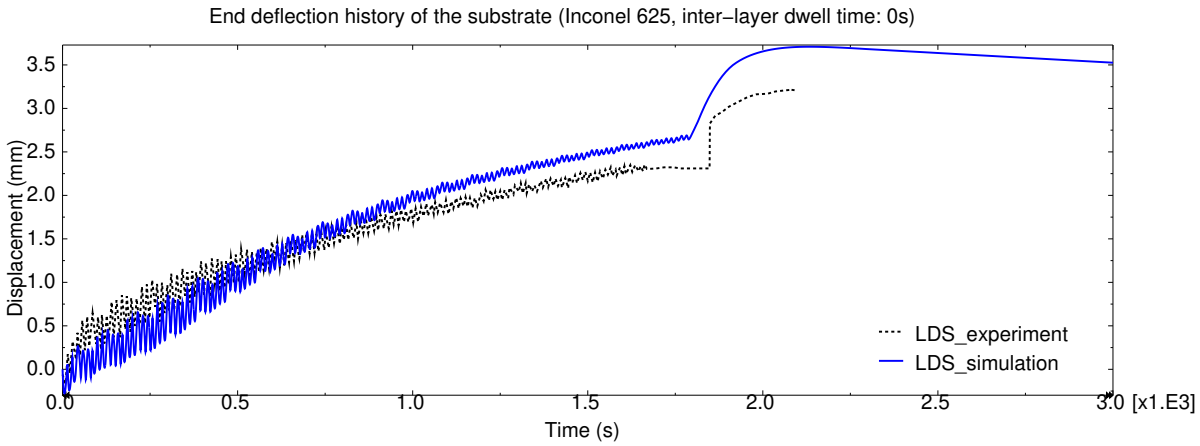
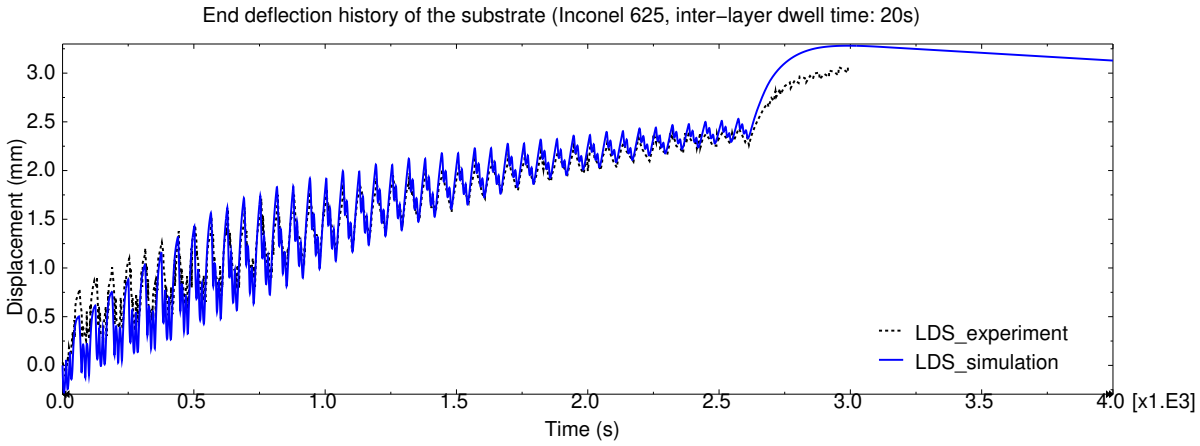


Figure 5: Temperature histories of thermocouples.





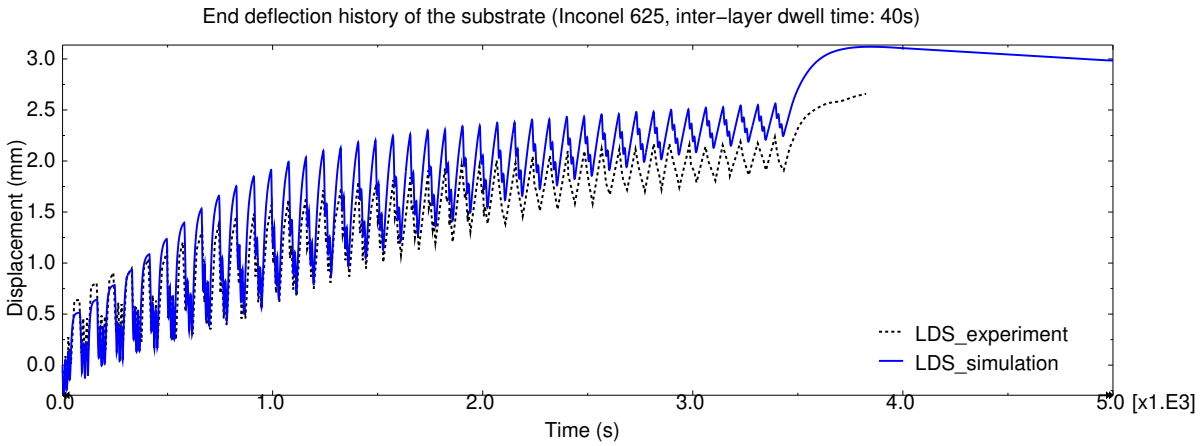


Figure 6: End deflection histories of the substrate.

Cure Process Simulation

Cure process modeling using Abaqus/Standard is demonstrated.

In this section:

- *Modeling the Watts test of adhesives*
- *Prediction of cure-induced spring-in of an angle bracket*

Modeling the Watts test of adhesives

Products: Abaqus/Standard

References:

- *Modeling the Cure Process in Thermosetting Polymers*
- *Thermal Expansion*
- *Time Domain Viscoelasticity*

Overview

This example demonstrates how to model the curing process of adhesives using the cure modeling capabilities in Abaqus/Standard.

This example illustrates the following Abaqus features and techniques:

- the cure modeling capabilities;
- the use of the thermorheologically simple (TRS) material model to account for temperature dependence and dependence on degree of cure in a viscoelastic material;
- the use of the tangent thermal expansion to define the thermal expansion coefficient; and
- accounting for the thermal and chemical dependence of mechanical properties in a fully coupled temperature-displacement analysis.

Application description

Adhesives are commonly used in the aerospace, automobile, and electronics industries. Structural adhesives are preferred over mechanical fastening methods such as welding due to their special characteristics (for example, corrosion resistance, high strength to weight ratio, and reduced thermal and mechanical damage to the substrates).

When the adhesives are applied, they typically undergo a curing process. Cross linking of the polymer chains form during an irreversible chemical reaction, and the material turns from a viscous liquid to a rubbery solid and eventually cools to a glassy solid. The heat generated during the curing process can cause thermal expansion of the material, and the cross-linking of polymers can cause the material to shrink. The thermal and chemical strains developed during the curing process can potentially weaken the bonded components.

It is important to simulate the curing process to better design the curing conditions to reduce the residual strains and stresses in the adhesives. The Watts test (*Watts and Cash, 1991*) is a simple curing test that is designed to validate the cure modeling process by comparing simulation predictions with experimental measurements.

Geometry

The Watts test measures the polymerization shrinkage of materials. In the Watts test, a disc-shaped specimen is sandwiched between two glass plates, as shown in *Figure 1*. The top plate is a thin glass diaphragm that rests on the adhesive and a brass ring. The bottom plate is a thick glass slide that is centered within the brass ring. The brass ring serves as a support for the diaphragm as it deflects due to the shrinkage of the specimen. The dimensions of the adhesive and other parts of the assembly are shown in *Figure 1*.

Abaqus modeling approaches and simulation techniques

The Watts test is modeled with a curing step followed by a cooling step and a relaxation step.

Analysis types

A fully coupled temperature-displacement analysis models the deformation of the adhesive during the curing process, taking into account the thermal and the chemical dependencies of the mechanical properties. A quasi-static analysis performed after curing models the stress relaxation in the adhesive.

Mesh design

Axisymmetric elements are used to model the whole assembly, as shown in *Figure 2*. CAX4RHT elements are used to model the adhesive due to its nearly incompressible mechanical properties in the liquid state. CAX4RT elements are used to model the glass plates, brass ring, and air gap.

Materials

The Kamal equation (*Reaction Kinetics*) is used to define the cure kinetics of the adhesive, which controls the rate of cure as a function of cure and temperature. *Table 1* lists the ccoefficients used in the Kamal equation. *Table 2* list the other curing and thermal properties of the adhesive.

The history-dependent mechanical behavior of the adhesive can be characterized with a viscoelastic material model. Viscoelastic properties of adhesives can be measured with DMA (Dynamic Mechanical Analysis) methods, in which the specimens are subjected to small oscillatory strains at various combinations of frequencies and temperature.The same tests can also be performed for adhesives with various degrees of cure (conversion). Assuming the adhesive materials are thermorheologically simple, the time-temperature-cure superposition principles can be applied (*Lindeman and Hedegaard, 2023*), which enables the construction of a single master curve by shifting the test data in both the temperature-frequency space and the cure-frequency space. A shifting factor can then be characterized that is dependent both on temperature and the degree of cure as shown in *Figure 3*. The temperature and cure dependency of the viscoelastic model is modeled with the thermorheologically simple (TRS) material model (*Temperature Effects*). The instantaneous modulus of the adhesive is measured directly with the DMA test at various temperatures and degrees of cure, as shown in *Figure 4* and *Figure 5*. The Young's modulus is small when the adhesive is in the liquid state, but increases rapidly after gelation when the adhesive turns from liquid to solid. The Young's modulus also increases as the material cools and transforms from a rubbery state to a glassy state. The Poisson's ratio is close to 0.5 when the adhesive is in the liquid state and decreases as the material solidifies. *Figure 5* shows that the Poisson's ratio is not sensitive to the temperature change for this adhesive. *Figure 6* shows the tangent thermal expansion coefficients at different temperatures and degrees of cure.

Table 3 lists the properties of the brass ring, glass, and air.

Initial conditions

The initial temperature of the whole assembly is set to 22°C. The initial degree of cure for the adhesive is set to 0.0018.

Boundary conditions

Symmetric boundary conditions are applied at the nodes on the center-line. The bottom nodes of the glass slides are constrained in the vertical direction. *Figure 2* shows the applied boundary conditions.

Loads

Thermal loads are applied to the exterior surfaces of the assembly (*Figure 2*) to simulate the curing conditions for the adhesive specimen. The film coefficient is set to 0.035 mW/mm^2K during the curing step and 0.05 mW/mm^2K during the cooling step. The sink temperature is set to 65°C during the curing step, which lasts 30 minutes. It is then set to 22°C during the cooling step, which also lasts 30 minutes. Thermal loads are removed after the cooling step when the temperature in the whole assembly reaches uniform room temperature.

Interactions

The interfaces between the adhesive and the glass (diaphragm and bottom slide) are assumed to be perfectly bonded (mechanically and thermally). Surface-to-surface contact is specified between the diaphragm and the top surfaces of the air and ring, and conductive heat transfer is specified between these surfaces. The bottom surfaces of the air and the brass ring are also assumed to be perfectly bonded to the top of the glass slide.

Analysis steps

The analysis consists of three steps. The first step is a coupled temperature-displacement step, in which the exterior of the assembly is heated to 65°C to activate the curing process. The second step is a cooling step, which is also modeled with the coupled temperature-displacement procedure. The sink temperature is dropped to 22°C, and a slightly higher film coefficient is used in this step. The final step is a quasi-static step in which the thermal load is removed and the residual stresses are relaxed.

Results and discussion

Figure 7 shows the cure history of the adhesive, and *Figure 8* shows the temperature and cure history of the adhesive at a location near the center-line. An abrupt increase in the temperature is observed due to the high amount of heat produced when the material is cured. As a result, we also observe a fast increase of thermal expansion as shown in *Figure 9*. This exothermic reaction results in both a fast temperature increase and a high cure rate, which might cause potential problems in a real assembly when components are made with temperature-sensitive materials. However, exothermic reaction is not uncommon in a Watts test since a large amount of adhesive is used to facilitate the test measurements. *Figure 10* shows the deflection of the glass diaphragm and the shrinkage of the adhesive, and *Figure 11* shows the deflection profile of the diaphragm. The predicted deflection profile compares well with the test data presented in *Lindeman and Hedegaard, 2023*.

Acknowledgment

SIMULIA gratefully acknowledges 3M for providing input data used in this example.

Input files

watts_test.inp	Input file for the Watts test.

References

1. Watts, D. C., and A. J. Cash, “Determination of Polymerization Shrinkage Kinetics in Visible-Light-Cured Materials: Methods Development,” *Dental Materials*, vol. 7, no. 4, pp. 281–287, 1991.
2. Lindeman, D., and A. Hedegaard, “31-Residual Stress Development in Curing Processes: Material Characterization and Modeling,” *Advances in Structural Adhesive Bonding (Second Edition)* 1011-1033, 2023.

Tables

Table 1: Kamal model coefficients.

Z_1 (1/sec)	3.91×10^{15}
E_1/R (K)	1.34×10^{4}
m	1.15
n	1.2
b_1	0.0

Table 2: Other adhesive properties.

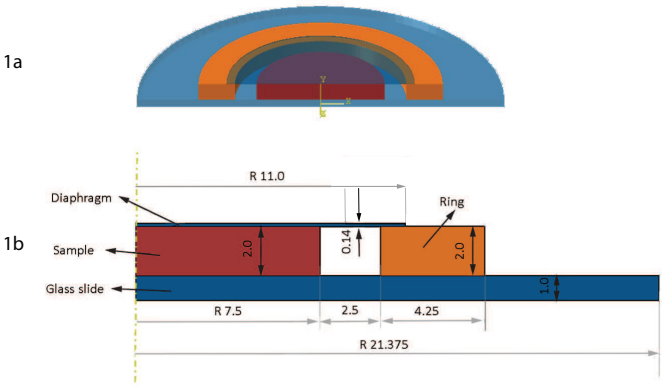
Density (ton/tmm^3)	1.26×10^{-9}
Heat of reaction (mJ/ton)	3.03×10^{11}
Cure shrinkage coefficient	0.043
Conductivity (mW/mm/K)	0.32
Specific heat (mJ/ton/K)	1.86×10^{9}

Table 3: Properties of brass, glass and air.

Properties	Brass	Glass	Air
Density (ton/mm^3)	8.49×10^{-9}	2.42×10^{-9}	1.2×10^{-12}
Young's modulus (MPa)	106×10^{3}	71.5×10^{3}	1×10^{-6}
Poisson's ratio	0.32	0.21	0
Conductivity (mW/mm/K)	124	1.1	2.4×10^{-2}
Specific heat (mJ/ton/K)	3.8×10^{8}	7.5×10^{8}	1.006×10^{9}
Thermal expansion coefficient (K^{-1})	2.01×10^{-5}	3.25×10^{-6}	0

Figures

Figure 1: Watts test geometry.



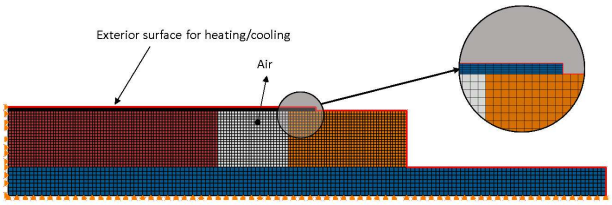


Figure 2: Axisymmetric mesh of the Watts test model and boundary conditions.

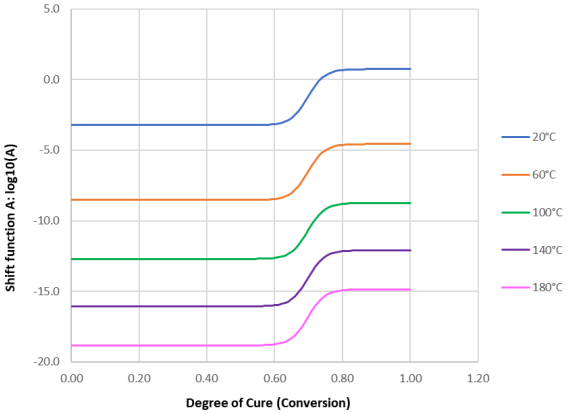


Figure 3: Shift function of the adhesive.

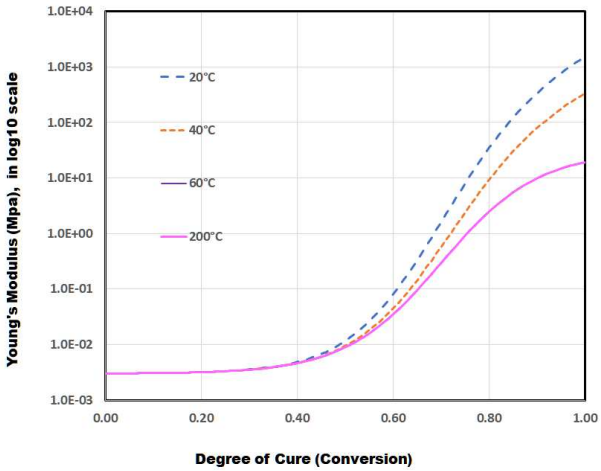


Figure 4: Young's modulus of the adhesive.

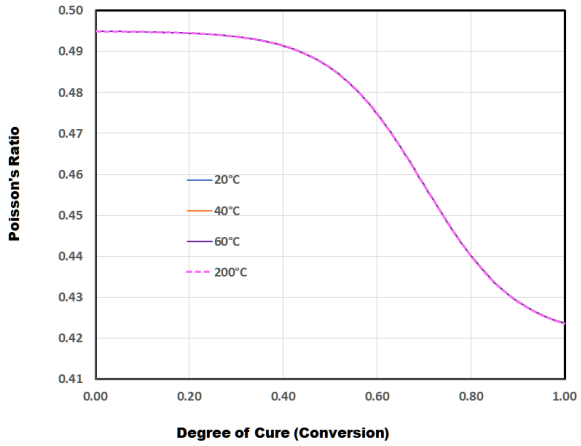


Figure 5: Poisson's ratio of the adhesive.

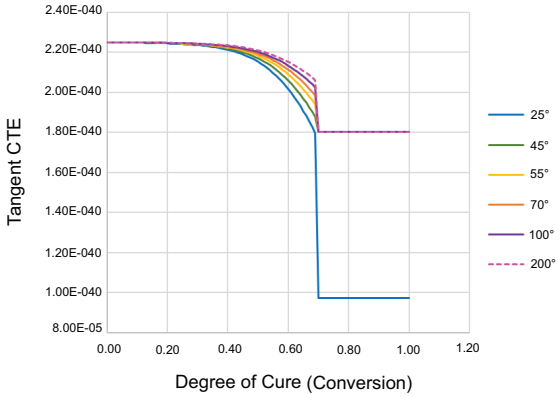


Figure 6: Tangent thermal expansion coefficient of the adhesive.

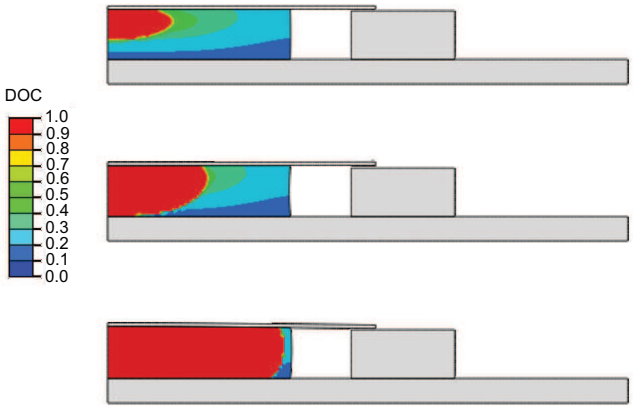


Figure 7: Conversion history of the adhesive during curing.

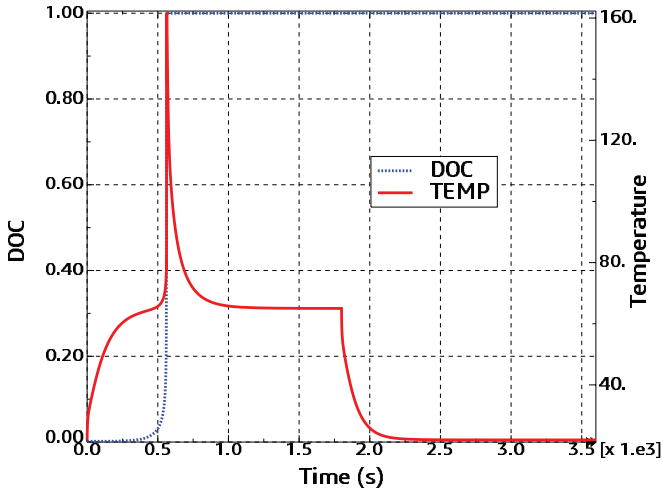


Figure 8: Temperature and cure history at the center of the specimen.

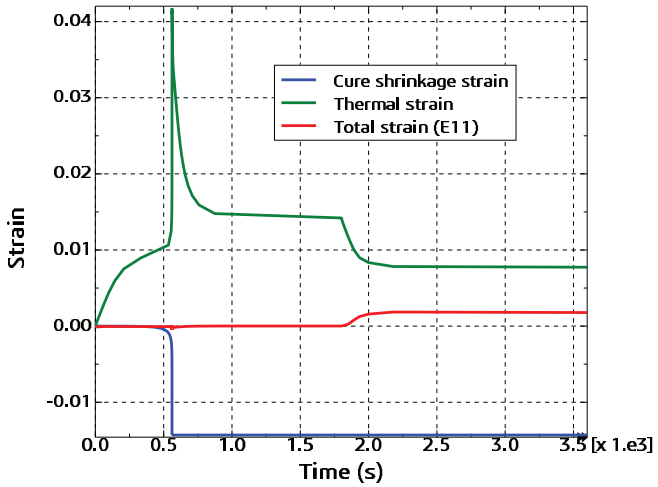


Figure 9: Time history of strains in element 681.

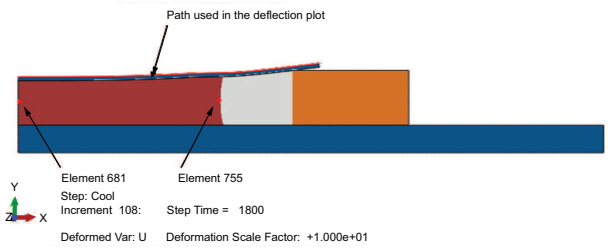


Figure 10: Deformation of the specimen at the end of the cooling step.

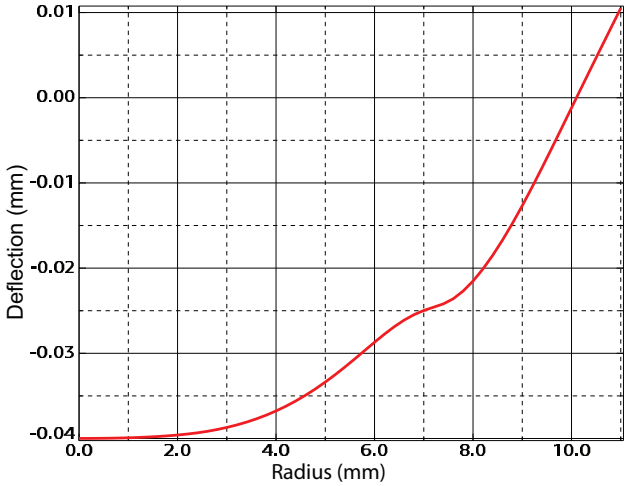


Figure 11: Predicted deflection profile.

Prediction of cure-induced spring-in of an angle bracket

Products: Abaqus/Standard

References:

- *Mean-Field Homogenization*
- *Curing Processes in Polymers*
- *Importing an External Field to Define History-Dependent Fields*

Overview

This example demonstrates the use of multiscale technology combined with cure modeling in Abaqus/Standard to model the cure-induced spring-in deformation of an angle bracket.

This example demonstrates the following Abaqus features and techniques:

- composite cure modeling in Abaqus/Standard;
- a viscoelasticity model with thermorheologically simple dependence on temperature and degree of cure;
- the use of a multiscale material model to homogenize thermal, chemical, and mechanical properties;
- a fully coupled thermal-displacement procedure to model the transient response of the composite during curing; and
- the model change technique to remove contact definitions for spring-in deformation prediction.

In addition, it compares the fully coupled approach with a sequentially coupled approach.

Application description

Composite cure process–induced shape distortion presents significant challenges in composite products manufacturing. Cure process simulation requires the modeling of thermal, chemical, and mechanical responses of the composite material. During the curing process, the manufacturer-recommended heating cycle is applied to the composite part. The degree of cure of the composite increases with time, and this evolution is characterized by the cure kinetics of the process. Residual stress/strain can develop with the change of thermal/chemical condition, leading to shape distortion once the composite part is separated from the mold/tool.

Geometry

The geometry of the angle bracket assembly has two main components: the composite angle bracket and the steel tool. *Figure 1* shows the dimensions of the cross-section of the L-shaped bracket. The width of the bracket is 60 cm. The bracket has a $(0/90)_n$ layup. The width of the tool is 80 cm, with a thickness of 50 cm. *Figure 2* shows the bracket-tool assembly.

Boundary conditions and loading

During the curing simulation, the bottom surfaces of the tool are fixed in all directions. Film conditions are applied to all exterior surfaces. The film coefficient is set to 60 W/m^2K for the top of the bracket and the tool. The film coefficient for the side and bottom of the tool is set to 30 W/m^2K. *Figure 3* shows the temperature applied during the curing cycle.

Abaqus modeling approaches and simulation techniques

Two modeling approaches are used in this example:

- Fully coupled temperature-displacement analysis.
- Sequentially coupled analysis. A heat-transfer analysis is performed first to predict the temperature change in the composite. A quasi-static analysis is then performed using the temperature history imported from the heat-transfer analysis.

Mesh design

C3D8T elements are used in the fully coupled analysis. In the sequentially coupled analysis, DC3D8 elements are used in the heat-transfer analysis and C3D8 elements are used in the subsequent mechanical analyses. There are 10 layers of elements in the thickness direction.

Materials

The tool is made of steel, with properties as listed in *Table 1*. The bracket material is modeled with LY5052 epoxy reinforced with 55% E-glass fiber. *Table 2* lists the properties of the E-glass, which are based on those in *Table 1*.

The cure characteristics of LY5052 are based on those in *Saseendran and Wysocki (2016)*. The Kamal definition is used for the cure kinetics, and the standard DiBenedetto equation is used to compute the glass transition temperature from the degree of cure. *Table 3* and *Table 4* list the Kamal coefficients and the parameters in the DiBenedetto equation, respectively. Another important aspect of the cure reaction is that a considerable amount of energy can be released during curing, which can result in exothermal temperature peaks and drastic changes in temperature and degree of cure. The cure heat generation is set to 4.82e5 J/Kg (*Saseendran and Wysocki (2016)*).

During the curing process, the cure temperature in the resin changes with time. With this change, the resin material undergoes two transformations. The first one is a transformation from a liquid to a solid, which is called gelation. At this point, the material is in a rubbery state. As the glass transition temperature increases and eventually exceeds the cure temperature, the material transforms from a rubbery state to a glassy state. This second transformation is called vitrification. *Table 5* and *Table 6* list the properties of the resin at the glassy state and the rubbery state.

The mechanical constitutive relationship of the resin material can be modeled as linear viscoelastic with thermorheologically simple effects. For simplicity, a simple shift function proposed by *Svanverg (2002)* is used:

$$A = \lim_{\omega \to 0} \begin{cases} \omega, T \geq T_g \\ \frac{1}{\omega}, T < T_g \end{cases}.$$

With this simplified shift function (see *Temperature Effects*), the resin material reaches full relaxation instantaneously in the rubbery state and has no relaxation once it reaches the glassy state. In the shift function definition, 10^{-20} is used for the rubbery state and 10^{35} is used for the glassy state. The relaxation of the modulus is adapted from Prony series coefficients in *Kim and White (1996)*.

During the curing process, the resin material undergoes permanent shrinkage due to the cross-linking reaction. Cure shrinkage, thermal strain, and residual stresses developed during the curing process are the main contributing factors to the shape distortion of the composite part after tool removal. The thermal strain in this example is modeled in the following rate:

$$\dot{\epsilon}^{th} = \alpha \dot{\theta},$$

in which α is the thermal expansion coefficient given by

$$\alpha = \begin{cases} \alpha^r, T \geq T_g \\ \alpha^g, T < T_g \end{cases},$$

in which α^r and α^g are, respectively, the thermal expansion coefficient of the rubbery state and the glassy state. The thermal expansion coefficient is assumed to be zero before gelation—the degree of cure is less than 0.5. Two approaches are used to model the thermal expansion: one is through user subroutine *UEXPAN*, and the other is through a tabular tangent definition (see *Thermal Expansion*). The cure shrinkage is modeled with a similar rate formulation (see *Cure Shrinkage Strain*).

Two approaches are demonstrated to model the local fiber directions. In the first approach, the composite is modeled as a unidirectional fiber-reinforced composite , and the local fiber direction is specified in the 1- and 2-direction in alternate layers of the mesh. In the second approach, the composite is modeled with a single woven material, which contains two inclusions that are orthogonal to each other.

Initial conditions

The initial temperature of both the bracket and the tool is 22°C. The initial degree of cure of the resin in the composite is 0.001.

Interactions

During the analysis the normal interaction between the bracket and tool is specified as "hard" contact with no separation. Rough friction is specified to describe the interaction in the tangential direction during the mechanical analysis. Node-to-surface contact is specified to ensure that all nodes on the bottom surface of the bracket stay attached to the tool. In the last step, contact between the bracket and the tool is removed.

Analysis steps

The fully coupled analysis consists of two steps. The first step is a coupled temperature-displacement step, in which the temperature shown in *Figure 3* is applied to the exterior of the bracket and tool to simulate the cure process. The second step is also a coupled temperature-displacement step, in which contact between the bracket and the tool is removed. The bracket continues to cool to room temperature during this step, and shape distortion of the bracket is evaluated under the free-standing condition.

The sequentially coupled analysis consists of three steps. The first step is a heat-transfer step, in which the temperature shown in *Figure 3* is applied to the exterior of the bracket and tool to simulate the cure process. The second step is a quasi-static step, in which the temperature history from the heat-transfer step is imported. Thermal expansion, cure shrinkage, and residual stresses are predicted in this step. The third step is a static step, in which contact between the bracket and the tool is removed and shape distortion of the bracket is evaluated under a "free-standing" condition. For efficiency, room temperature is prescribed, and the bracket reaches equilibrium in a single increment.

Results and discussion

The time history of the temperature, degree of cure, and glass transition temperature evaluated at one of the corner elements of the bracket is shown in *Figure 4*. The part temperature is close to the applied temperature with little exothermic overshoot. *Figure 5* and *Figure 6* show the temperature distribution in the whole assembly predicted by the two approaches (fully coupled and sequentially coupled). Little temperature gradient over the assembly is observed, which validates the proper choice of the tool material. The difference in results between the fully coupled and sequentially coupled approaches is very small, indicating insignificant effect of deformation on the change of temperature and degree of cure.

For the case when the bracket is modeled with alternate layers of UD composite, the final shape distortion results are shown in *Figure 7* and *Figure 8*. The deformed shape of the bracket is overlaid on the undeformed shape.

The amplitude of the deformation is amplified five times. The predicted spring-in angle predicted by both approaches is approximately 1°, which is close to the prediction by *Svanverg (2002)*. The displacements of the top corner node predicted by both approaches are shown in *Figure 10*, and the predictions are very close. The in-plane strains (E11, CURESE11, THE11) and out-of-plane strains (E22, CURESE22, THE22) predicted by the sequential analysis are shown in *Figure 11* and *Figure 12*. Due to the unconstrained condition at the top of the bracket, the strains in the thickness direction are significant larger than the in-plane strains. The strains predicted by the fully coupled analysis are very close to the prediction of the sequentially coupled analysis. *Figure 13* shows the in-plane (S11) and out-of-plane (S22) residual stresses predicted by both approaches. Most of the residual stresses developed after the composite transforms from the rubbery state to the glassy state.

For the case when the bracket is modeled with a woven composite, the final shape distortion result is shown in *Figure 9*. The analysis is fully coupled. The spring-in angle predicted is close to the prediction given by the UDcomposite.

Input files

anglebracket3d_composite.inp	Fully coupled temperature-displacement model (UD definition).
anglebracket3d_composite_heat.inp	Heat-transfer model in the sequentially coupled analysis (UD definition).
anglebracket3d_composite_mech.inp	Subsequent quasi-static model in the sequentially coupled analysis (UD definition).
anglebracket3d_woven_composite.inp	Fully coupled temperature-displacement model (woven definition).
anglebracket_material.inp	Input file containing material definition of the composite material (UD definition). Thermal expansion coefficient is specified with user subroutine *UEXPAN*.
anglebracket_woven_material.inp	Material definition for the composite material (woven definition). The thermal expansion coefficient is specified with user subroutine *UEXPAN*.
anglebracket_material_tangent.inp	Material definition for the composite material (UD definition). The thermal expansion coefficient is specified with the tangent definition.
uexpan_cureconst.f	User subroutine *UEXPAN*.

References

1. Svanverg, J., “Shape Distortion of Non-Isothermally Cured Composite Angle Bracket,” *Plastics, Rubber and Composites*, vol. 31, pp. 398–404, 2002.
2. Saseendran, S., M. Wysocki, and J. Varna, “Evolution of Viscoelastic Behavior of a Curing LY5052 Epoxy Resin in the Glassy State,” *Advanced Manufacturing: Polymer & Composites Science*, vol. 2, pp. 74–82, 2016.
3. Saseendran, S., , and M. Wysocki, “Characterisation of Viscoelastic Material Properties during Curing Process,” *Challenges in Mechanics of Time-Dependent Materials*, vol. 2, pp. 45–54, 2016.
4. Kim, Y., and S. White, “Stress Relaxation Behavior of 3501-6 Epoxy Resin during Cure,” *Polymer Engineering Science*, vol. 36, no. 23, pp. 2852–2862, 1996.

Tables

Table 1: Tool properties.

Young's modulus (GPa)	193
Poisson's ratio	0.29

Density (kg/m^3)	8000
Thermal expansion coefficient (K^{-1})	1.73×10^{-5}
Conductivity (W/m/K)	16.2
Specific heat (J/kg/K)	420

Table 2: Fiber properties.

Young's modulus (GPa)	76
Poisson's ratio	0.22
Density (kg/m^3)	2540
Thermal expansion coefficient (K^{-1})	4.9×10^{-6}
Conductivity (W/m/K)	1.0
Specific heat (J/kg/K)	800

Table 3: Kamal model coefficients.

Z (1/sec)	274906.7
E (J/mol)	54888.35
m	0.25
n	1.01
b	0.0

Table 4: Parameters in the DiBenedetto equation.

T_{g0} (K)	232.15
$T_{g\infty}$ (K)	409.15
λ	0.44

Table 5: Properties of resin (glassy state).

Bulk modulus (GPa)	3.6
Shear modulus (GPa)	0.94
Thermal expansion coefficient (K^{-1})	71×10^{-6}

Table 6: Properties of resin (rubbery state).

Bulk modulus (GPa)	3.6
Shear modulus (GPa)	0.0094
Thermal expansion coefficient (K^{-1})	178×10^{-6}

Table 7: Other resin properties.

Density (kg/m^3)	1100
Conductivity (W/m/K)	0.3

Specific heat (J/kg/K)	3.0 + 51.9
Cure shrinkage coefficient	−0.018

Figures

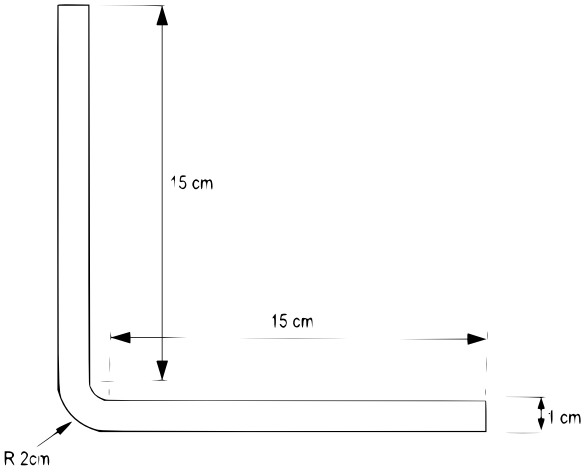


Figure 1: Dimensions of the cross-section of the angle bracket.

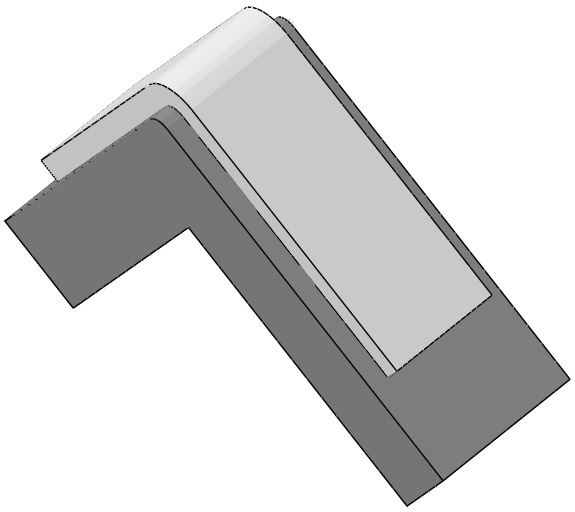


Figure 2: Bracket-tool assembly.

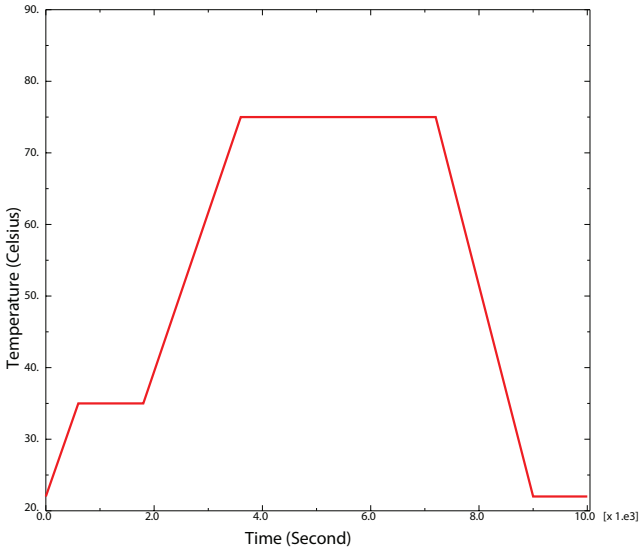


Figure 3: Applied temperature.

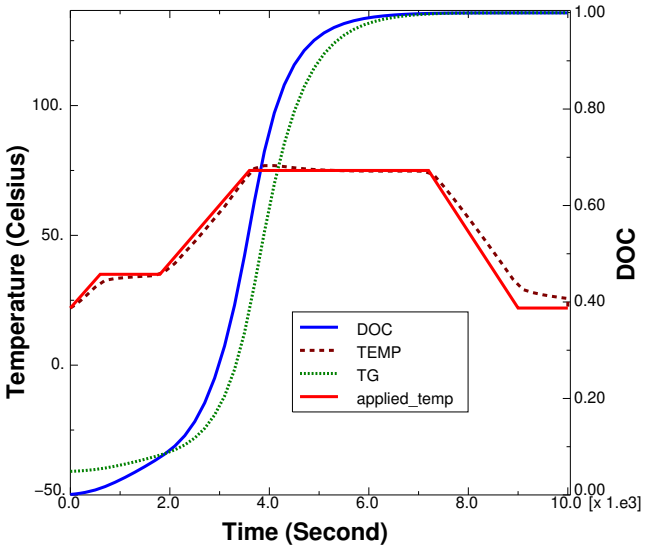


Figure 4: Time history of the temperature, degree of cure and glass transition temperature.

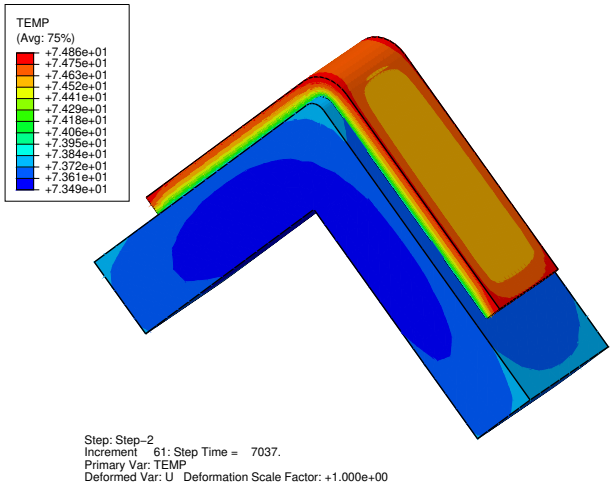


Figure 5: Temperature distribution of the bracket and tool assembly predicted by the fully coupled analysis.

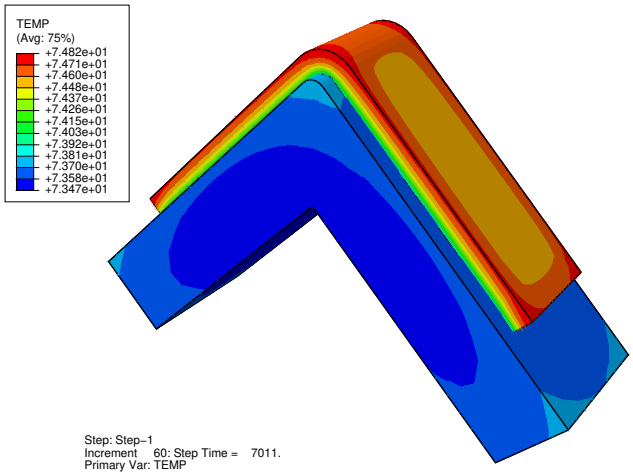


Figure 6: Temperature distribution of the bracket and tool assembly predicted by the sequentially coupled analysis.

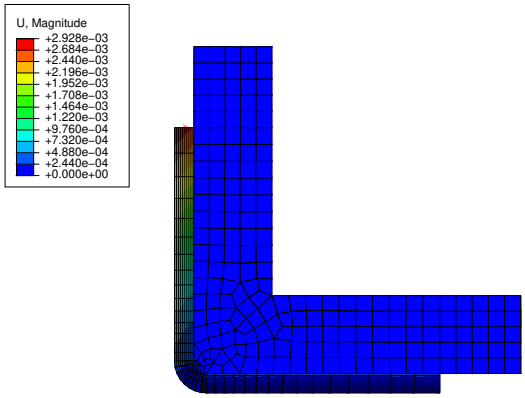


Figure 7: Shape distortion predicted by the fully coupled analysis (UD definition).

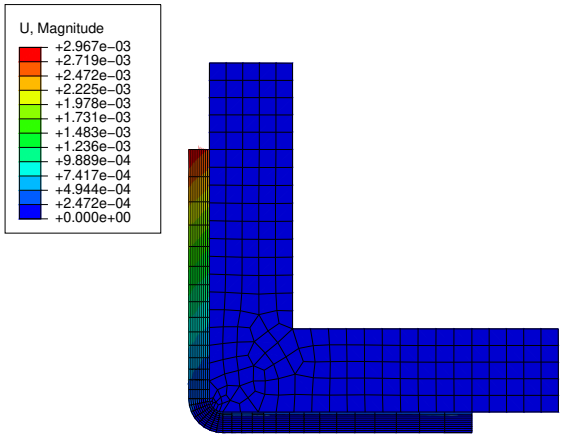


Figure 8: Shape distortion predicted by the sequentially coupled analysis (UD definition).

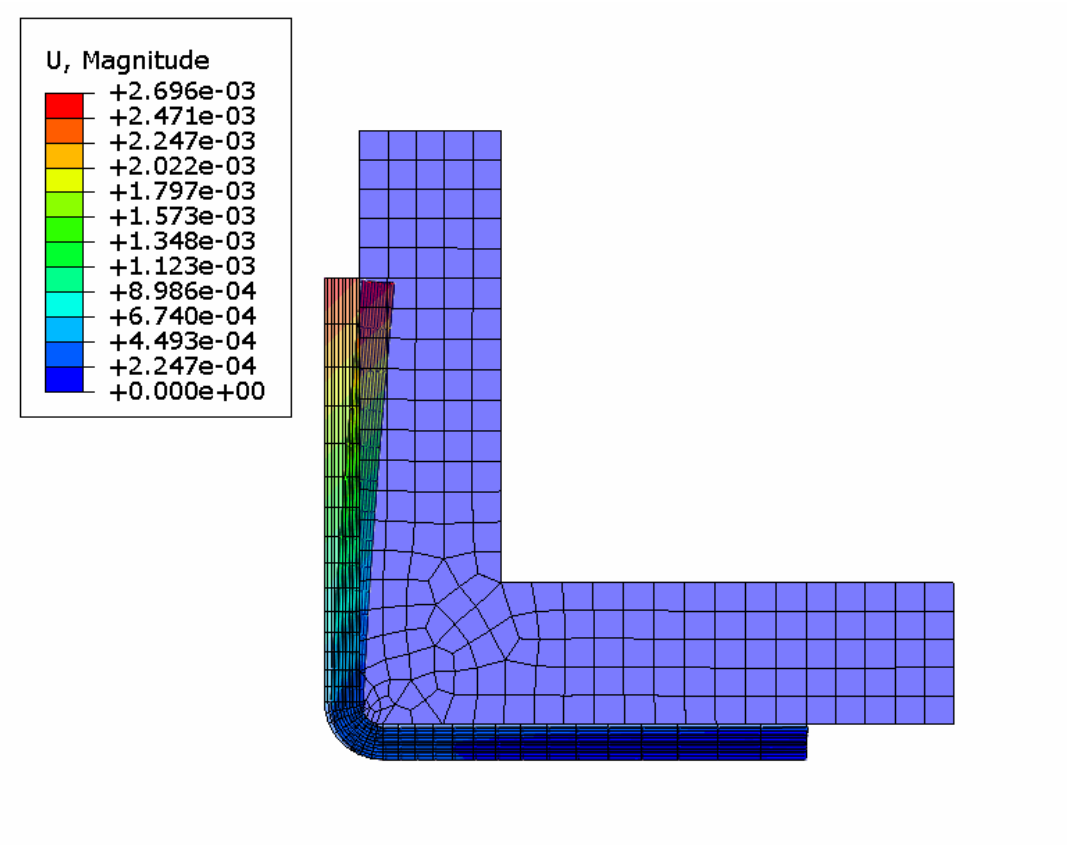


Figure 9: Shape distortion predicted by the fully coupled analysis (woven definition).

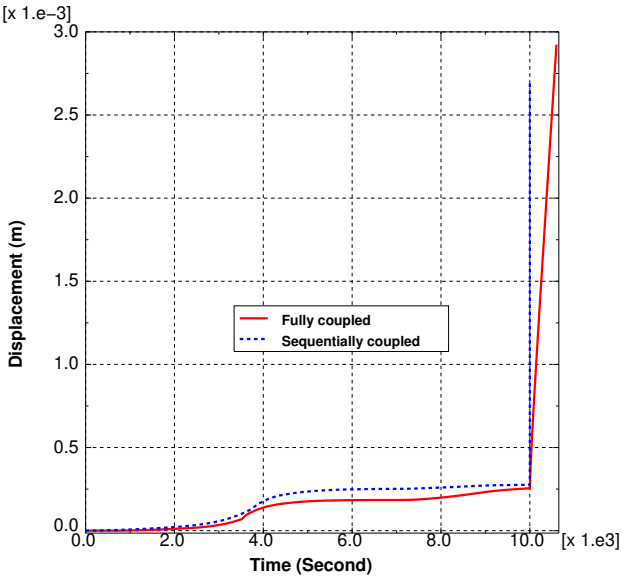


Figure 10: Displacement of the tip of the bracket.

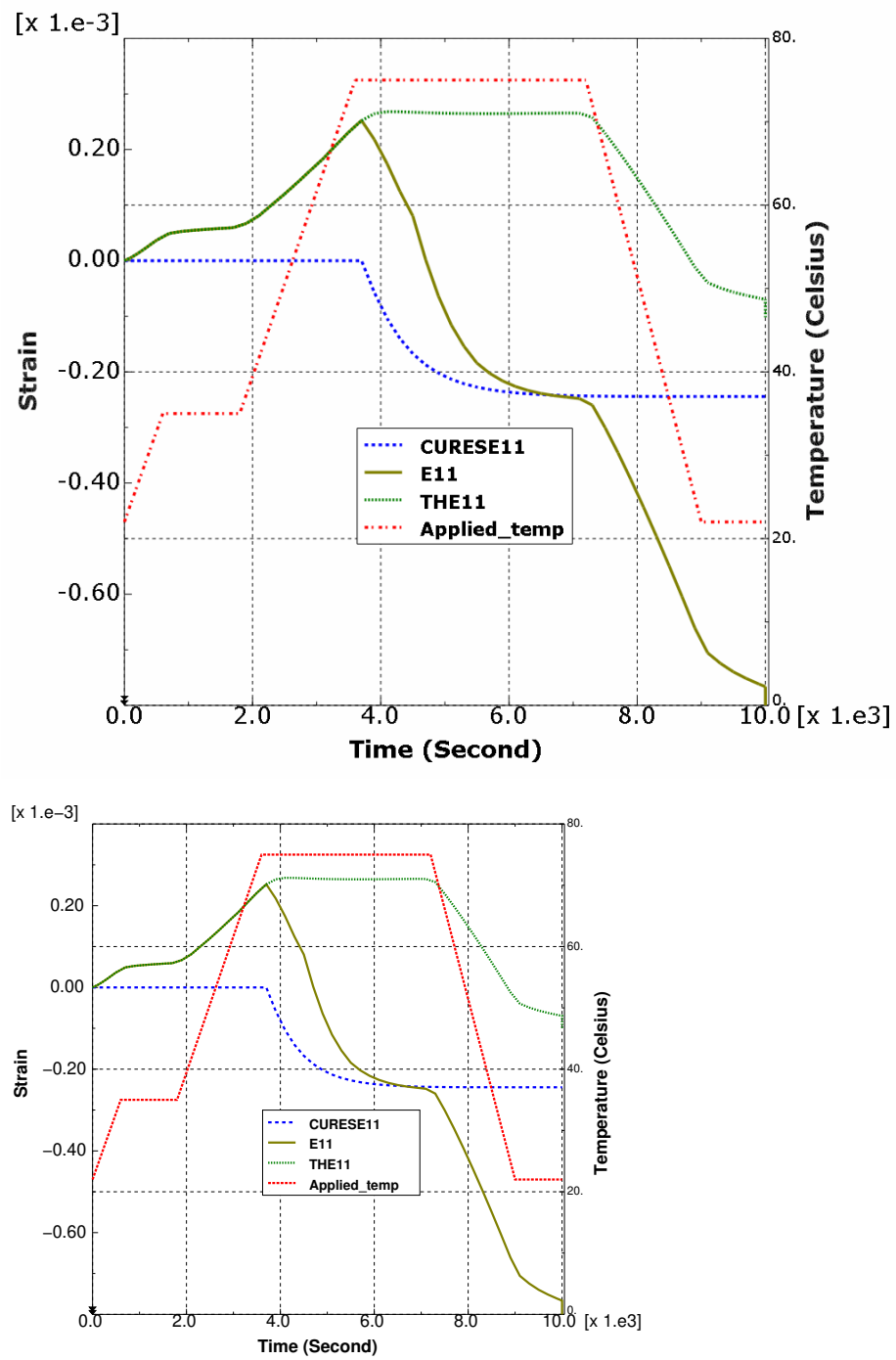


Figure 11: In-plane strains at element 4051.

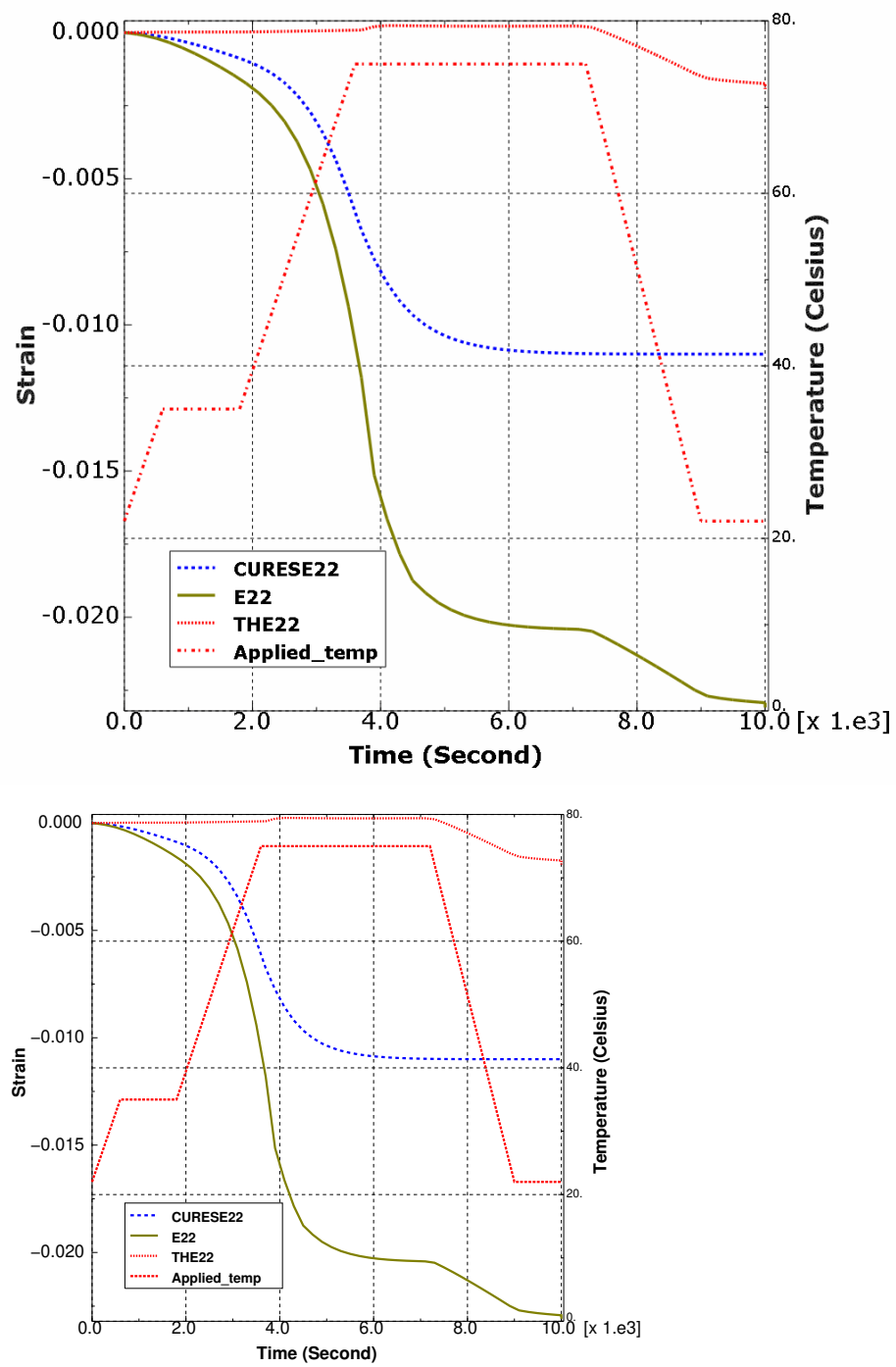


Figure 12: Out-of-plane strains at element 4051.

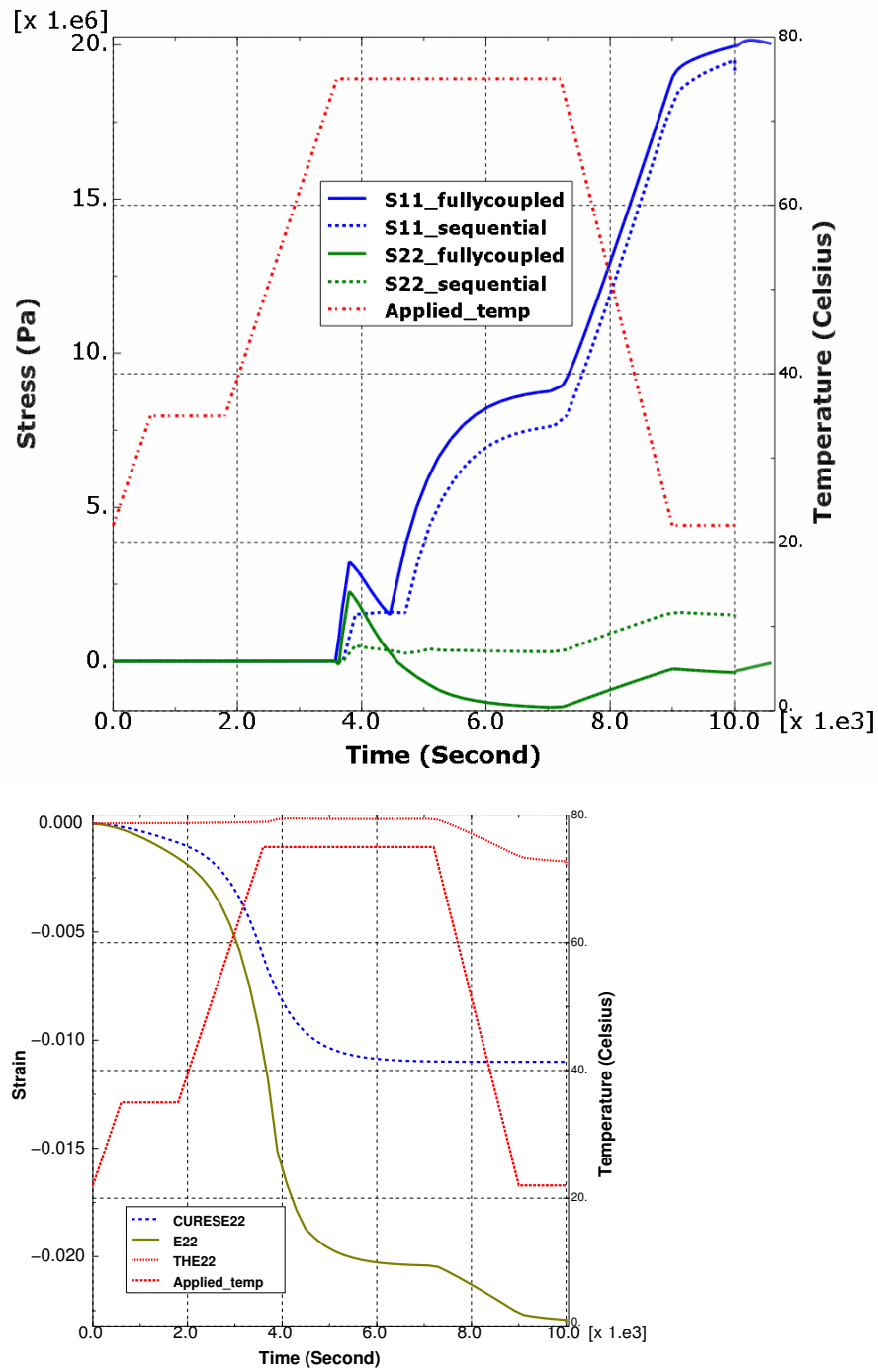


Figure 13: Residual stresses at element 3916.

Mass Diffusion and Soils Analyses

In this section:

- *Mass diffusion analyses*
- *Soils analyses*

Mass diffusion analyses

In this section:

- *Hydrogen diffusion in a vessel wall section*
- *Diffusion toward an elastic crack tip*

Hydrogen diffusion in a vessel wall section

Products: Abaqus/Standard

References:

- *Mass Diffusion Analysis*
- *Mass diffusion analysis*

Overview

This example uses a one-dimensional problem to verify the mass diffusion capability in Abaqus.
The physical problem considered here is that of a pressure vessel shell wall fabricated from 2 1/4 Cr–1 Mo steel alloy base metal with an internal weld overlay of Type 347 stainless steel. These vessels are typically used at high temperatures and under high pressure conditions. Under such service conditions hydrogen dissolves into the alloys (Fujii et al., 1982) and during cooldown may cause disbonding of the weld overlay from the base metal and, possibly, crack initiation and growth in the base metal due to hydrogen embrittlement. In this example we are concerned with the hydrogen diffusion aspect of the problem.

Problem description

The problem is shown in *Figure 1* and consists of a section of the vessel wall made up of a 200-mm thick base metal and a 5-mm thick weld metal. The problem is one-dimensional, the only gradient being through the thickness of the wall. The purpose of the analysis is to predict the evolution of hydrogen concentration through the wall thickness during cooling caused by a shutdown.

Geometry and model

Since the problem is one-dimensional, we use a plane mesh with only one element in the *y*-direction (see *Figure 2*). The mesh is graded, with more elements near the interface between the two materials because we expect very high concentration gradients in this vicinity.

The material properties of the two metals given by Fujii et al. (1982) are strongly dependent on temperature and can be written as follows.

Solubility in weld metal:

$$s_w = 1288e^{-1078/\tilde{\theta}} \text{ ppm}$$

Diffusivity in weld metal:

$$D_w = 9310\ e^{-6767/\bar{\theta}}\ \text{mm}^2/\text{h}$$

Solubility in base metal:

$$s_b = 4300e^{-3261/\tilde{\theta}} \text{ ppm mm N}^{-1/2}$$

Diffusivity in base metal:

$$D_b = \frac{274\ e^{-1157/\bar{\theta}}}{1+\left(1.05\times10^{-3}\ e^{3573/\bar{\theta}}\right)}\ \text{mm}^2/\text{h},$$

where $\tilde{\theta}$ is temperature in kelvins. These temperature-dependent properties are entered in Abaqus in tabulated form, as shown in the input listings.

The wall is initially at a uniform temperature of 727.5 K (454.4°C), and during the shutdown schedule it cools down to 298.15 K (25.0°C) at a constant rate over a period of 21.5 hours.

The boundary conditions are as follows. Under the initial steady-state conditions the exterior of the weld metal has a hydrogen concentration of 35.85 ppm, which corresponds to a normalized concentration of 0.1225 $N^{1/2}mm^{-1}$. Normalized concentration is used as the primary solution variable (continuous over the discretized domain) and is given as the concentration divided by the solubility. The exterior of the base metal has a zero hydrogen concentration. As the cooling period begins, the hydrogen concentration at the exterior of the weld metal is assumed to drop to zero instantaneously.

Time stepping

The problem is run in two parts. The first part consists of a step in which a single increment of steady-state uncoupled mass diffusion analysis is performed with an arbitrary time step to establish the initial steady-state hydrogen concentration distribution corresponding to the initial temperature.

The hydrogen diffusion during cooling is then analyzed in four subsequent mass diffusion transient analysis steps, using automatic time stepping. This need not be done in four separate steps. We do it here because the results given by Fujii et al. (1982), with which we compare the Abaqus results, are presented at four specific times during the transient: 2.7 h (673.15 K, 400.0°C), 5.2 h (623.15 K, 350.0°C), 10.2 h (523.15 K, 250.0°C), and 21.5 h (298.15 K, 25.0°C).

The accuracy of the time integration for the mass diffusion transient analysis steps, during which cooling occurs, is controlled by the DCMAX parameter. This parameter specifies the allowable normalized concentration change per time step. Even in a linear problem such as this, DCMAX controls the accuracy of the solution because the time integration operator is not exact (the backward difference rule is used). In this case DCMAX is chosen as 0.01 $N^{1/2}mm^{-1}$, which is a very tight value. This is necessary to obtain an acceptably accurate integration of the concentration because the solubility of the materials decreases significantly (by more than two orders of magnitude in the base metal) as the temperature decreases and, therefore, the changes in concentration become larger for a given change in normalized concentration.

An important issue in transient diffusion problems is the choice of initial time step. As in any transient problem, the spatial element size and the time step are related to the extent that time steps smaller than a certain size may lead to spurious oscillations in the solution and, therefore, provide no useful information. This coupling of the spatial and temporal approximations is always most obvious at the start of diffusion problems, immediately after prescribed changes in the boundary values. For the mass diffusion case the suggested guideline for choosing the initial time increment (see *Mass Diffusion Analysis*) is

$$\Delta t \geq \frac{1}{6D}(\Delta h)^2,$$

where Δh is a characteristic element size near the disturbance (that is, near the weld metal surface in our case), and D is the diffusivity of the material. For the weld metal in our model we choose a typical $\Delta h =$ 0.125 mm and we have $D =$ 0.85 mm^2/h at the initial temperature, which gives $\Delta t \geq$ 0.003 h. For the base metal in our model we choose a typical $\Delta h =$ 1.25 mm and we have $D =$ 4.88 mm^2/h at the initial temperature, which gives $\Delta t \geq$ 0.053 h. Based on these calculations an initial time step of 0.1 h is used, which gives an initial solution with no oscillations, as expected.

Results and discussion

Figure 3 shows hydrogen concentration distributions in the weld metal for the initial steady-state condition and four different times during the cooling period. *Figure 4* shows corresponding hydrogen concentration distributions

in the base metal. These results compare very well with those presented by Fujii et al. (1982) which are not plotted here since they would appear almost indistinguishable from the Abaqus results.

It can be observed that, although the primary solution variable (the normalized concentration) remains continuous across the material interface during the transient, the hydrogen concentration becomes increasingly discontinuous across the interface. During the cooling process the hydrogen concentration in the base metal decreases, whereas the hydrogen concentration in the weld metal increases very significantly, reaching a peak at the weld metal side of the interface.

Input files

hydrodiffvesselwall_2d.inp	Two-dimensional analysis.
hydrodiffvesselwall_3d.inp	Three-dimensional analysis.
hydrodiffvesselwall_3d_po.inp	**POST OUTPUT* analysis of hydrodiffvesselwall_3d.inp.
hydrodiffvesselwall_fick.inp	Two-dimensional analysis using Fick's law.
hydrodiffvesselwall_nonlinear.inp	Nonlinear version (including concentration dependence on the material properties) of the two-dimensional analysis.
hydrodiffvesselwall_heat.inp	Heat transfer analysis that writes temperatures to a results file for use in hydrodiffvesselwall_massdiff.inp.
hydrodiffvesselwall_massdiff.inp	Two-dimensional mass diffusion analysis that reads temperatures from the results file written in hydrodiffvesselwall_heat.inp.

References

- Fujii, T., T. Nazama, H. Makajima, and R. Horita, “A Safety Analysis on Overlay Disbonding of Pressure Vessels for Hydrogen Service,” *Journal of the American Society for Metals*, pp. 361–368, 1982.

Figures

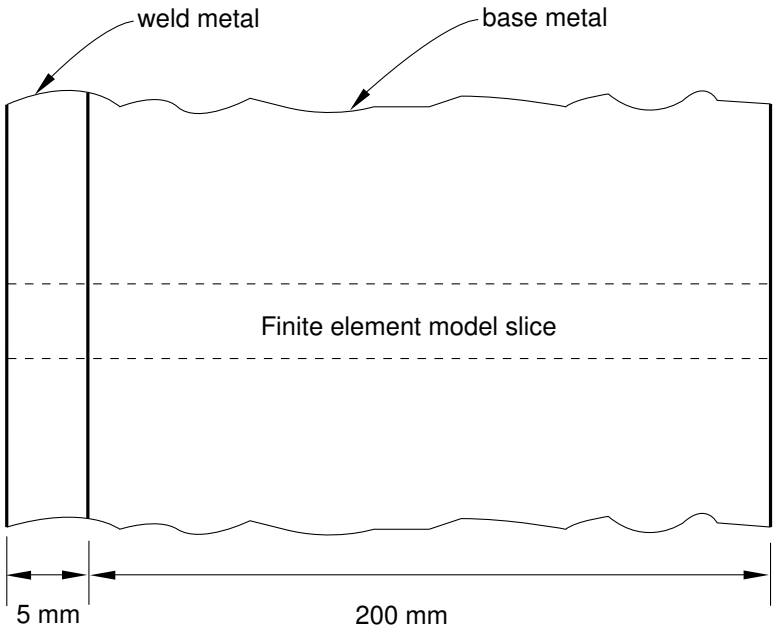


Figure 1: Pressure vessel shell wall section.

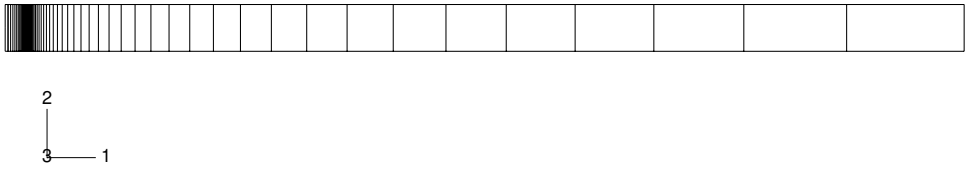


Figure 2: Finite element model of shell wall.

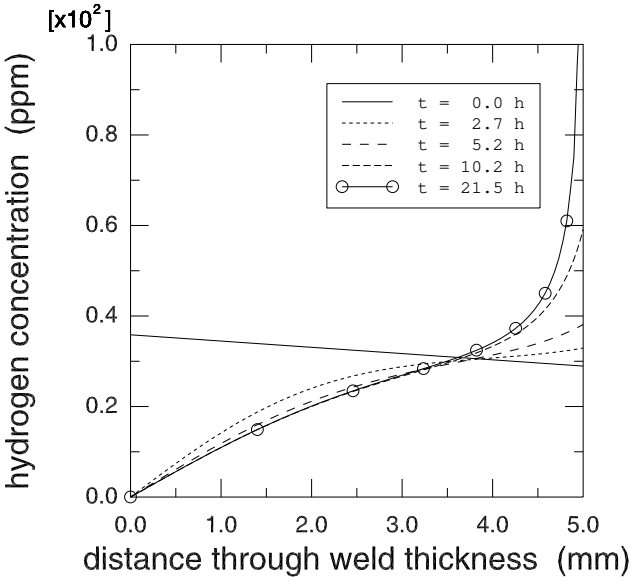


Figure 3: Hydrogen concentration distribution in weld metal.

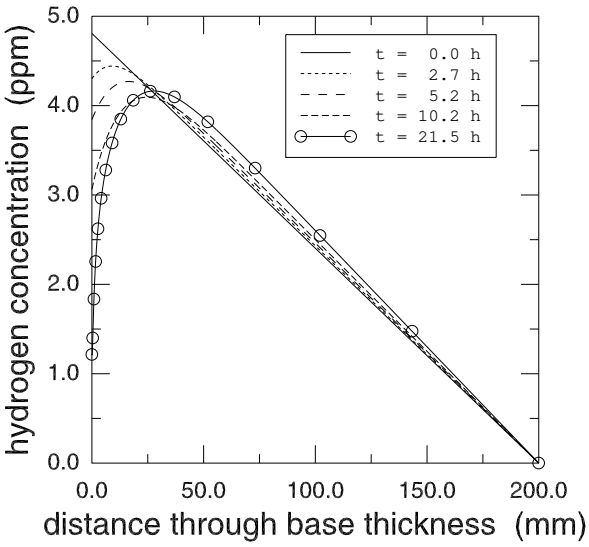


Figure 4: Hydrogen concentration distribution in base metal.

Diffusion toward an elastic crack tip

Products: Abaqus/Standard

References:

- *Mass Diffusion Analysis*
- *Mass diffusion analysis*

Overview

This examples uses a two-dimensional problem to verify the sequentially coupled, stress-assisted mass diffusion capability in Abaqus.
A center-cracked plate fabricated from 2 1/4 Cr–1 Mo steel alloy is subjected to end loading in a hydrogen-rich environment. Hydrogen is drawn to the crack-tip region by high hydrostatic stresses and may assist in crack growth resulting from hydrogen embrittlement. In this example we are concerned with the hydrogen diffusion aspect of the problem.

Geometry and model

The problem geometry and boundary conditions are shown in *Figure 1*. The specimen is 10-mm thick, 20-mm wide, and 80-mm high, with a 4-mm crack at its center. The mesh near the crack is focused at the crack tip, with the element size growing as the square of the distance to the crack tip (with the first bounding node set representing the crack tip). A very fine mesh (see *Figure 2*) is used to capture accurately the gradients of concentration and stress near the crack tip. Four combinations of stress and mass diffusion analyses are presented:

- Stress analysis with quadratic elements and quarter-point spacing at the crack tip, followed by a mass diffusion analysis with linear elements.
- Stress analysis with quadratic elements and quarter-point spacing at the crack tip, followed by a mass diffusion analysis with quadratic elements and quarter-point spacing at the crack tip.
- Stress analysis with quadratic elements (no quarter-point spacing), followed by a mass diffusion analysis with quadratic elements (no quarter-point spacing).
- Stress analysis with linear elements, followed by a mass diffusion analysis with linear elements.

The quarter-point spacing technique is used in fracture mechanics analyses to enforce a $1/\sqrt{r}$ singularity at the crack tip, where r is the distance from the crack tip.

The sequentially coupled mass diffusion analysis consists of a static stress analysis, followed by a mass diffusion analysis. Equivalent pressure stresses from the static analysis are written to the results file as nodal averaged values. Subsequently, these pressures are read in during the course of the mass diffusion analysis to provide a driving force for mass diffusion.

The material properties for mass diffusion given by Fujii et al. (1982) are as follows.

Solubility:

$$s = 4300e^{-3261/(\theta-\theta^z)}\ \text{ppm mm N}^{-1/2}$$

Diffusivity:

$$D = \frac{7611\times10^{-5}\ \ e^{-1157/(\theta-\theta^z)}}{1+\left(1.05\times10^{-3}\ \ e^{3573/(\theta-\theta^z)}\right)}\quad \text{mm}^2/\text{s},$$

where θ is the temperature in degrees Celsius and $\theta^z = -273$ is the absolute zero temperature.

Stress-assisted diffusion is specified by defining the pressure stress factor, κ_p, as

$$\kappa_p = \frac{\overline{V}_H \phi}{R(\theta - \theta^z)} \ \text{mm}\ \text{N}^{-1/2},$$

where $R = 8.31432$ Jmol^{-1}K^{-1} is the universal gas constant, $\overline{V}_H = 2.0 \times 10^3$mm^{3}mol^{-1} is the partial molar volume of hydrogen in iron-based metals, and ϕ is the normalized concentration. The concentration dependence of κ_p is entered in Abaqus in tabulated form as shown in the input listings. It is important to note that although κ_p is defined in terms of normalized concentration, ϕ, the tabular data must be entered in terms of concentration, $c = \phi s$.

The following properties are also used in the stress analysis: elastic modulus, $E = 2.0 \times 10^5$Nmm^{-2}, and Poisson's ratio, $\nu = 0.3$.

The specimen is maintained at a constant temperature of $\theta - \theta^z = 325$ K throughout the analysis. Under the initial steady-state conditions the specimen has a uniform concentration of 50 ppm, which corresponds to a normalized concentration of 265 N$^{1/2}$mm^{-1}. Normalized concentration is used as the primary solution variable (continuous over the discretized domain) and is given as the concentration divided by the solubility. The exterior of the specimen has a constant hydrogen concentration equal to the initial concentration. A 1 MPa distributed pressure is applied to the ends of the specimen, ramped linearly over the length of the step, and the steady-state distribution of hydrogen is obtained.

Results and discussion

The analytical solution for normalized concentration, presented by Liu (1970), has the form

$$\phi = \phi_o \exp\left(-\frac{\overline{V}_H\, p}{R(\theta - \theta^z)}\right),$$

where ϕ_o is the normalized concentration obtained in the unstressed state and p is the equivalent pressure stress. This solution dictates that for a crack-tip problem, the concentration follows the singularity of the stresses.

Figure 3 and *Figure 4* show the final distribution of equivalent pressure stress and concentration predicted by the Abaqus analysis in the region around the crack tip. The results shown represent the first case described above, using a quadratic, quarter-point mesh for stresses and a linear mesh for mass diffusion. The shapes of the contours show good agreement, since contours of constant pressure stress should be contours of constant concentration, as indicated by the analytical solution above.

Figure 5 and *Figure 6* show the pressures (in MPa) and concentrations (in ppm) ahead of the crack tip for all four combinations of stress and mass diffusion analyses. Results are presented as functions of the ratio of the distance to the crack tip, r, over the crack length, a. For the region immediately ahead of the crack, linear elastic fracture mechanics yields the analytical solution for equivalent pressure stress:

$$p = -\frac{\text{K}_\text{I}}{\sqrt{2\pi r}} \frac{(1+\nu)}{3} - \frac{(1+\nu)\sigma}{3},$$

where $\text{K}_\text{I} = \sigma\sqrt{\pi a}$ is the stress intensity factor for a Mode I crack of length a and σ is the externally applied distributed load.

As can be seen from the figures, the finite element results for all four combinations of element types are identical except at the first element, where the results are not expected to be valid. The results show good agreement with the analytically predicted solutions for both equivalent pressure stress and concentration as the distance to the crack tip, r, approaches zero. Farther from the crack tip, the deviation between the analytical solution and the

finite element solution increases. This deviation is consistent with the fact that the linear elastic crack-tip solution is valid only as r approaches zero.

No mesh convergence studies were conducted with respect to the number of elements in the crack-tip region. For comparison with the solutions presented here, an analysis was conducted with equally spaced elements approaching the crack tip. The results (not shown here) indicate that biasing the elements toward the crack tip is necessary to capture the gradients of concentration and equivalent pressure stress adequately. In addition, the equivalent pressure stress results demonstrate that the effect of using quarter-point positioning of the nodes at the crack tip is insignificant in this problem as long as the mesh is refined sufficiently.

Differences between the finite element and analytically predicted concentrations are a direct result of the differences between the finite element and analytically predicted values of pressure stress. If the analytical values of equivalent pressure stress are used to drive the Abaqus concentration solution, the resulting curve is indistinguishable from the analytical concentration shown.

Input files

difftocrack_quarterpstress.inp	Quadratic stress analysis with quarter-point spacing at the crack tip. This analysis writes the results file used in difftocrack_linearmassdiff1.inp and difftocrack_quarterpmassdiff.inp.
difftocrack_linearmassdiff1.inp	Linear mass diffusion analysis that reads results file data from difftocrack_quarterpstress.inp.
difftocrack_stress.inp	Stress analysis with quadratic elements (no quarter-point spacing). This analysis writes the results file used in difftocrack_massdiff.inp.
difftocrack_massdiff.inp	Mass diffusion analysis with quadratic elements that reads equivalent pressure stresses from the results file written in difftocrack_stress.inp.
difftocrack_quarterpmassdiff.inp	Mass diffusion analysis with quadratic elements and quarter-point spacing. This analysis reads equivalent pressure stresses from the results file written in difftocrack_quarterpstress.inp.
difftocrack_linearstress.inp	Stress analysis with linear elements. This analysis writes the results file used in difftocrack_linearmassdiff2.inp.
difftocrack_linearmassdiff2.inp	Mass diffusion analysis with linear elements that reads equivalent pressure stresses from the results file written in difftocrack_linearstress.inp.
difftocrack_node.inp	Node data for all the analyses.
difftocrack_quad_elements.inp	Element data for the analyses using quadratic elements.
difftocrack_linear_elements.inp	Element data for the analyses using linear elements.

References

- Fujii, T., T. Hazama, H. Nakajima, and R. Horita, “A Safety Analysis on Overlay Disbonding of Pressure Vessels for Hydrogen Service,” *Journal of the American Society for Metals*, pp. 361–368, 1982.
- Liu, H. W., “Stress-Corrosion Cracking and the Interaction Between Crack-Tip Stress Field and Solute Atoms,” Transactions of the ASME: Journal of Basic Engineering, vol. 92, pp. 633–638, 1970.

Figures

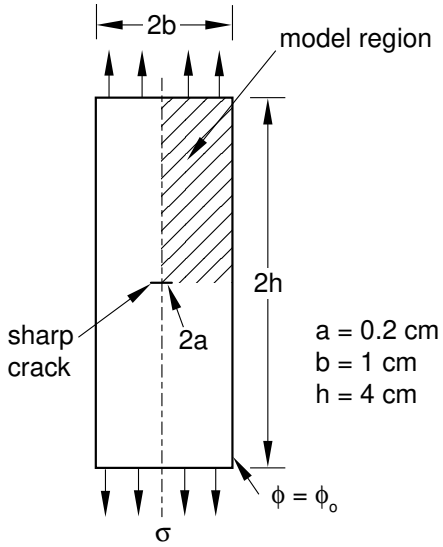


Figure 1: Center crack specimen geometry.

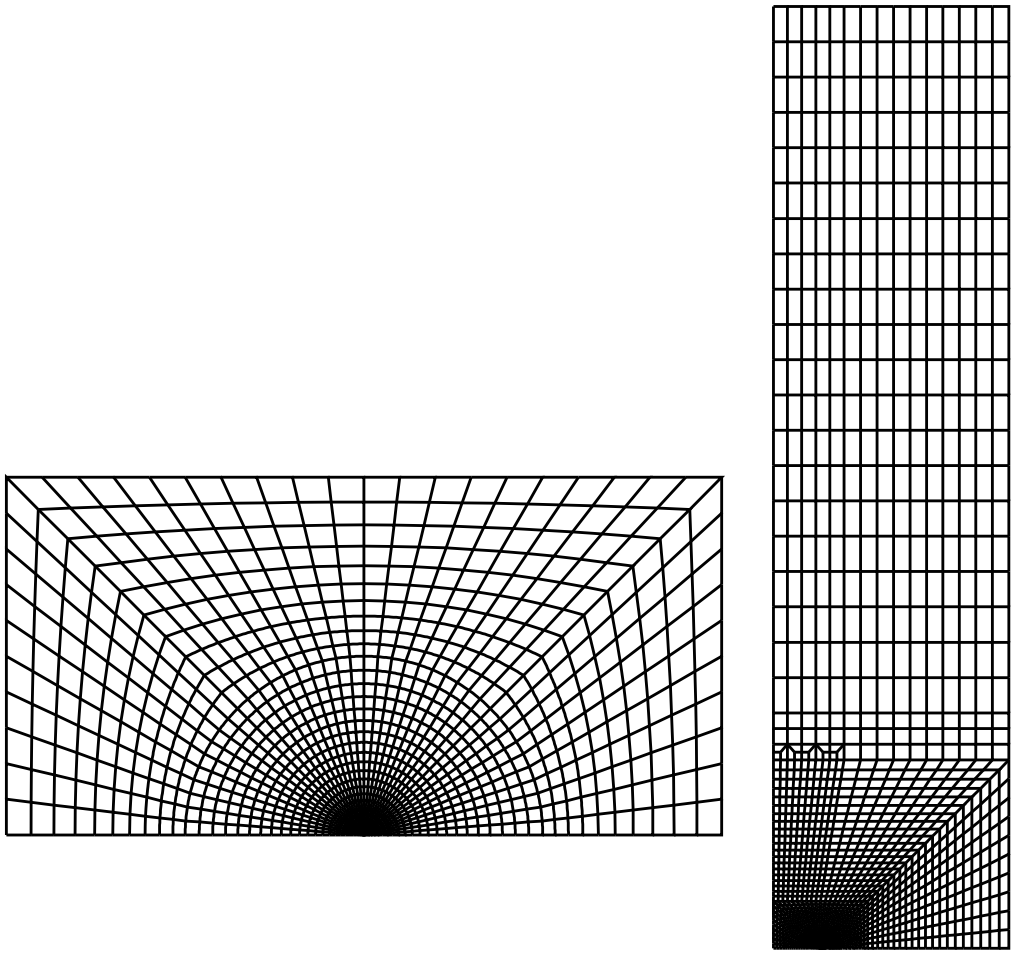


Figure 2: Finite element model of center crack specimen (with 1/4 symmetry) with detail of crack-tip mesh.

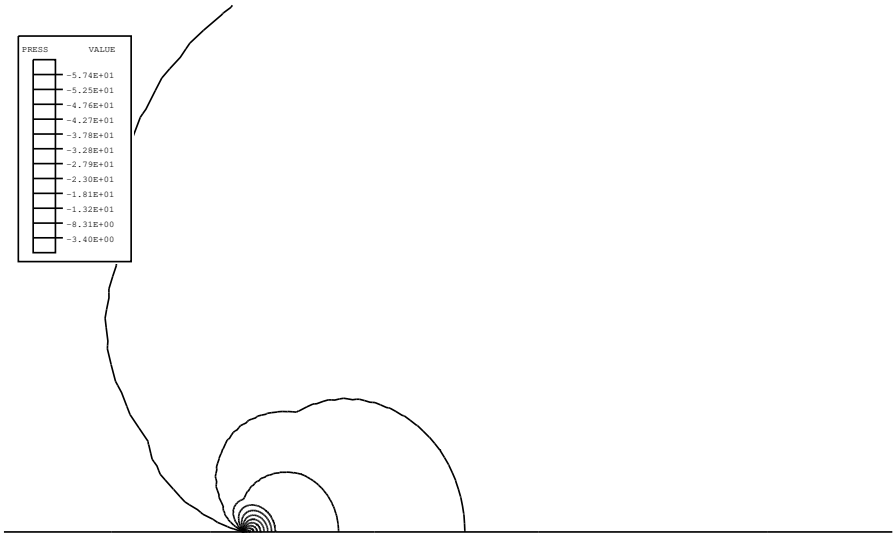


Figure 3: Contours of equivalent pressure stress at the crack tip.

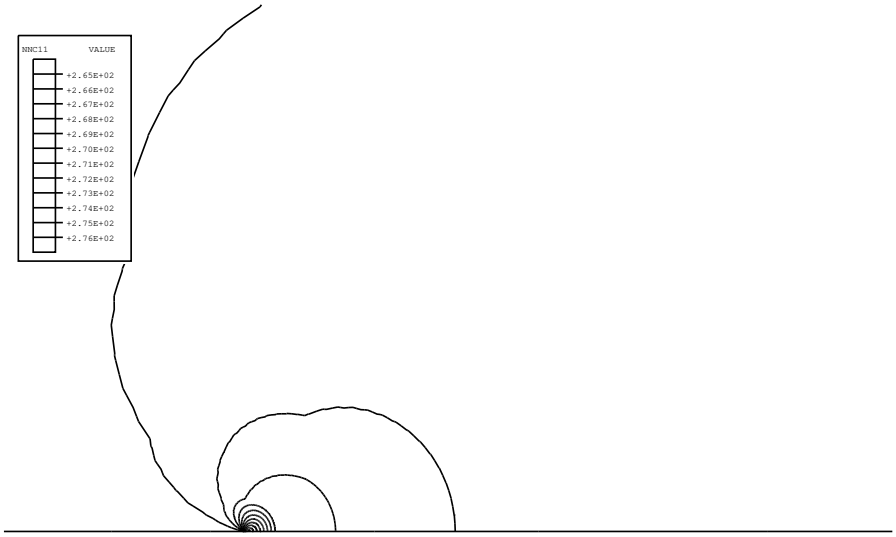


Figure 4: Contours of normalized hydrogen concentration at the crack tip.

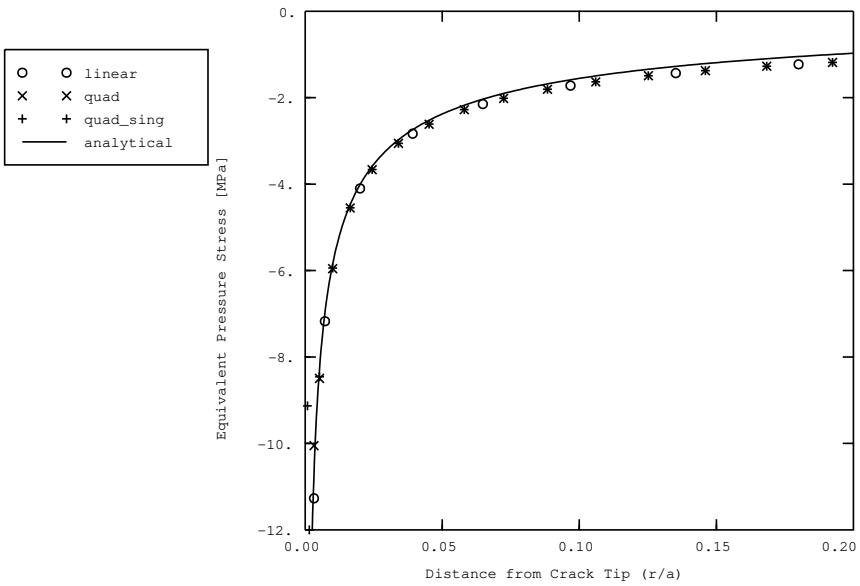


Figure 5: Distribution of pressure stress ahead of the crack tip.

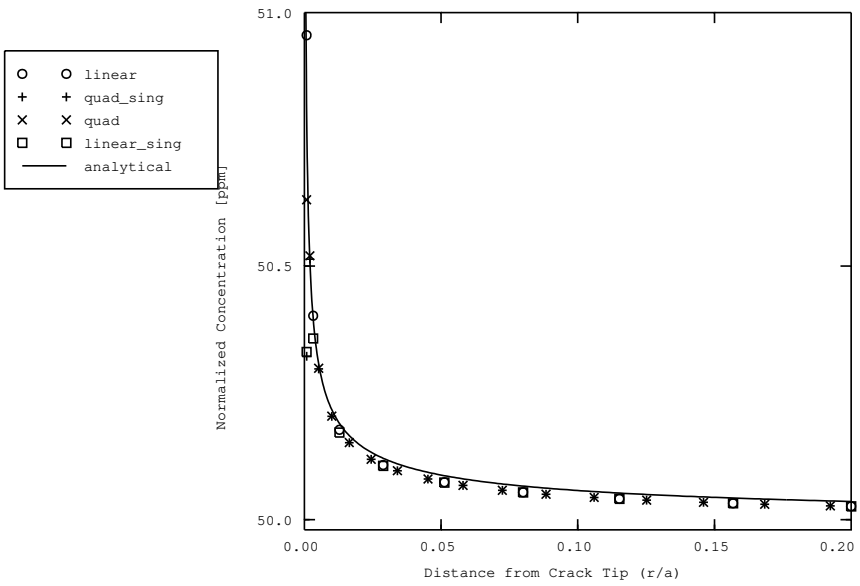


Figure 6: Hydrogen concentration distribution ahead of the crack tip.

Soils analyses

In this section:

- *Plane strain consolidation*
- *Calculation of phreatic surface in an earth dam*
- *Axisymmetric simulation of an oil well*
- *Analysis of a pipeline buried in soil*
- *Hydraulically induced fracture in a wellbore*
- *Permafrost thawing–pipeline interaction*

Plane strain consolidation

Products: Abaqus/Standard

References:

- *The Terzaghi consolidation problem*

Overview

This problem examines a linear, two-dimensional consolidation case: the settlement history of a partially loaded strip of soil.

Most consolidation problems of practical interest are two- or three-dimensional, so that the one-dimensional solutions provided by Terzaghi consolidation theory are useful only as indicators of settlement magnitudes and rates. This particular case is chosen to illustrate two-dimensional consolidation because an exact solution is available (Gibson et al., 1970), thus providing verification of this capability in Abaqus.

Geometry and model

The discretization of the semi-infinite, partially loaded strip of soil is shown in *Figure 1*. The loaded region is half as wide as the depth of the sample. The reduced-integration plane strain element with pore pressure, CPE8RP, is used in this analysis. Reduced integration is almost always recommended when second-order elements are used because it usually gives more accurate results and is less expensive than full integration. No mesh convergence studies have been done, although the reasonable agreement between the numerical results provided by this model and the solution of Gibson et al. (1970) suggests that the model used is adequate—at least for the overall displacement response examined. In an effort to reduce analysis cost while at the same time preserve accuracy, the mesh is graded from six elements through the height, under the load, to one element through the height at the outer boundary of the model, where a single infinite element (type CINPE5R) is used to model the infinite domain. This requires the use of two kinematic constraint features provided by Abaqus. Consider first the displacement degrees of freedom along line ***AC*** in *Figure 1*. The 8-node isoparametric elements used for the analysis allow quadratic variation of displacement along their sides, so the displacements of nodes *a* and *b* in elements *x* and *y* may be incompatible with the displacement variation along side ***AC*** of element *z*. To avoid this, nodes *a* and *b* must be constrained to lie on the parabola defined by the displacements of nodes *A*, *B*, and ***C.*** The QUADRATICMPC (“multi-point constraint”) is used to enforce this kinematic constraint: it must be used at each node where this constraint is required (see *planestrainconsolidation.inp*). Pore pressure values are obtained by linear interpolation of values at the corner nodes of an element. When mesh gradation is used, as along line ***AC*** in this example, an incompatibility in pore pressure values may result for the same reason given for the displacement incompatibility discussed above. To avoid this, the pore pressure at node *B* must be constrained to be interpolated linearly from the pore pressure values at *A* and ***C.*** This is done by using the P LINEARMPC.

The material properties assumed for this analysis are as follows: the Young's modulus is chosen as 690 GPa (10^8 lb/in^2); the Poisson's ratio is 0; the material's permeability is 5.08×10^{-7} m/day (2.0×10^{-5} in/day); and the specific weight of pore fluid is chosen as 272.9 kN/m^3 (1.0 lb/in^3).

The applied load has a magnitude of 3.45 MPa (500 lb/in^2). The strip of soil is assumed to lie on a smooth, impervious base, so the vertical component of displacement is prescribed to be zero on that surface. The left-hand side of the mesh is a symmetry line (no horizontal displacement). The infinite element models the other boundary.

Time stepping

As in the one-dimensional Terzaghi consolidation solution (see *The Terzaghi consolidation problem*), the problem is run in two steps. In the first transient soils consolidation step, the load is applied and no drainage is allowed across the top surface of the mesh. This one increment step establishes the initial distribution of pore pressures which will be dissipated during the second transient soils consolidation step.

During the second step drainage is allowed to occur through the entire surface of the strip. This is specified by prescribing the pore pressure (degree of freedom 8) at all nodes on this surface (node set `TOP`) to be zero. By default, in a transient soils consolidation step such boundary conditions are applied immediately at the start of the step and then held fixed. Thus, the pore pressures at the surface change suddenly at the start of the second step from their values with no drainage (defined by the first step) to 0.0.

Consolidation is a typical diffusion process: initially the solution variables change rapidly with time, while at the later times more gradual changes in stress and pore pressure are seen. Therefore, an automatic time stepping scheme is needed for any practical analysis, since the total time of interest in consolidation is typically orders of magnitude larger than the time increments that must be used to obtain reasonable solutions during the early part of the transient. Abaqus uses a tolerance on the maximum change in pore pressure allowed in an increment to control the time stepping. When the maximum change of pore pressure in the soil is consistently less than this tolerance, the time increment is allowed to increase. If the pore pressure changes exceed this tolerance, the time increment is reduced and the increment is repeated. In this way the early part of the consolidation can be captured accurately and the later stages are analyzed with much larger time steps, thereby permitting efficient solution of the problem. For this case the tolerance is chosen as 0.344 MPa (50 lb/in^2), which is 10% of the applied load. This is a fairly coarse tolerance but results in an economical and reasonable solution.

The choice of initial time step is important in consolidation analysis. As discussed in *The Terzaghi consolidation problem*, the initial solution (immediately following a change in boundary conditions) is a local, "skin effect" solution. Due to the coupling of spatial and temporal scales, it follows that no useful information is provided by solutions generated with time steps smaller than the mesh and material-dependent characteristic time. Time steps very much smaller than this characteristic time provide spurious oscillatory results (see Figure 3.1.5–2). This issue is discussed by Vermeer and Verruijt (1981), who propose the criterion

$$\Delta t \geq \frac{\gamma_\omega}{6Ek}(\Delta h)^2,$$

where Δh is the distance between nodes of the finite element mesh near the boundary condition change, E is the elastic modulus of the soil skeleton, k is the soil permeability, and γ_ω is the specific weight of the pore fluid. In this problem Δh is 8.5 mm (0.33 in). Using the material properties shown in *Figure 1*,

$$\Delta t_{\text{initial}} = 1 \times 10^{-5} \text{ days.}$$

We actually use an initial time step of 2×10^{-5} days, since the immediate transient just after drainage begins is not considered important in the solution.

Results and discussion

The prediction of the time history of the vertical deflection of the central point under the load (point P in *Figure 1*) is plotted in *Figure 2*, where it is compared with the exact solution of Gibson et al. (1970). There is generally good agreement between the theoretical and finite element solutions, even though the mesh used in this analysis is rather coarse.

Figure 2 also shows the time increments selected by the automatic scheme, based on the tolerance discussed above. The figure shows the effectiveness of the scheme: the time increment changes by two orders of magnitude over the analysis.

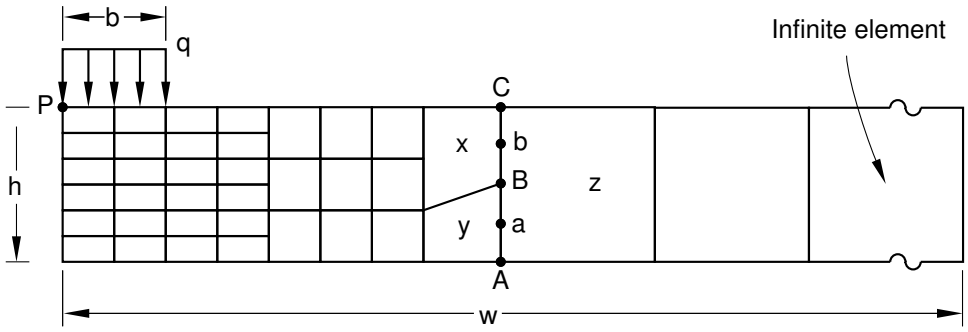
Input files

planestrainconsolidation.inp Input data for this example.

References

- Gibson, R. E., R. L. Schiffman, and S. L. Pu, “Plane Strain and Axially Symmetric Consolidation of a Clay Layer on a Smooth Impervious Base,” Quarterly Journal of Mechanics and Applied Mathematics, vol. 23, pt. 4, pp. 505–520, 1970.
- Vermeer, P. A., and A. Verruijt, “An Accuracy Condition for Consolidation by Finite Elements,” International Journal for Numerical and Analytical Methods in Geomechanics, vol. 5, pp. 1–14, 1981.

Figures



Geometry:

h = 50.8 mm (2.0 in)
b = 25.4 mm (1.0 in)
w = 533.4 mm (21.0 in)

Material:

E = 690 GPa (1.0×10^8 lb/in^2)
$\upsilon = 0.0$
k = 5.08×10^{-7} m/day (2.0×10^{-5} in/day)
$\gamma_w = 2.729 \times 10^5$ N/m^3 (1.0 lb/in^3)
$e_o = 1.5$

Loading:

q = 3.45 MPa (500.0 lb/in^2)

Figure 1: Plane strain consolidation example: geometry and properties.

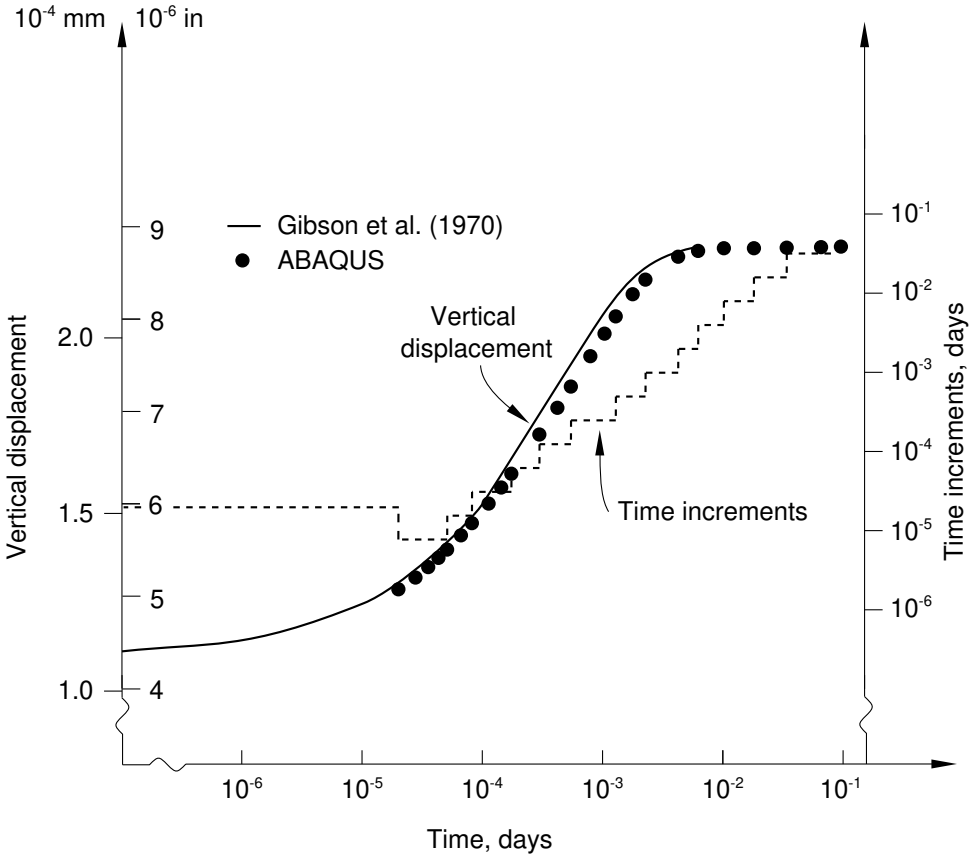


Figure 2: Consolidation history and time step variation history.

Calculation of phreatic surface in an earth dam

Products: Abaqus/Standard

Overview

This example illustrates the use of Abaqus to solve for the flow through a porous medium in which fluid flow is occurring in a gravity field and only part of the region is fully saturated, so the location of the phreatic surface is a part of the solution.
Such problems are common in hydrology (an example is the well draw-down problem, where the phreatic surface of an aquifer must be located based on pumping rates at particular well locations) and in some problems of dam design, as in this example. The basic approach takes advantage of the Abaqus capability to perform partially and fully saturated analysis: the phreatic surface is located at the boundary of the fully saturated part of the model. This approach has the advantage that the capillary zone, just above the phreatic surface, is also identified.

Boundary conditions

A typical dam is shown in *Figure 1*. We consider fluid flow only: deformation of the dam is ignored. Thus, although we use the fully coupled pore fluid flow-deformation elements, all displacement degrees of freedom are prescribed to be zero. A more general analysis would include stress and deformation of the dam.

The upstream face of the dam (surface S_1 in *Figure 1*) is exposed to water in the reservoir behind the dam. Since Abaqus uses a total pore pressure formulation, the pore pressure on this face must be prescribed to be $u_w = (H_1 - z)\, g\rho_w$, where H_1 is the elevation of the water surface, z is elevation, g is the gravitational acceleration, and ρ_w is the mass density of the water. ($g\rho_w$, the weight density of the water, must be given as the value of the specific weight of the wetting fluid as part of the definition of material permeability.) Likewise, on the downstream face of the dam (surface S_2 in *Figure 1*), $u_w = (H_2 - z)\, g\rho_w$.

The bottom of the dam (surface S_3) is assumed to rest on an impermeable foundation. Since the natural boundary condition in the pore fluid flow formulation provides no flow of fluid across a surface of the model, no further specification is needed on this surface.

The phreatic surface in the dam, S_4, is found as the locus of points at which the pore fluid pressure, u_w, is zero. Above this surface the pore fluid pressure is negative, representing capillary tension causing the fluid to rise against the gravitational force and creating a capillary zone. The saturation associated with particular values of capillary pressure for absorption and exsorption of fluid from the porous medium is a physical property of the material and is defined in the absorption/exsorption behavior under partially saturated flow conditions.

A special boundary condition is needed if the phreatic surface reaches an open, freely draining surface, as indicated on surface S_5 in *Figure 1*. In such a case the pore fluid can drain freely down the face of the dam, so $u_w = 0$ at all points on this surface below its intersection with the phreatic surface. Above this point $u_w < 0$, with its particular value depending on the solution. This example is specifically chosen to include this effect to illustrate the use of the Abaqus drainage-only flow boundary condition.

This drainage-only flow condition consists of prescribing the flow velocity on the freely draining surface in a way that approximately satisfies the requirement of zero pore pressure on the completely saturated portion of this surface (Pagano, 1997). The flow velocity is defined as a function of pore pressure, as shown in *Figure 2*. For negative pore pressures (those above the phreatic surface) the flow velocity is zero—the proper natural boundary condition. For positive pore pressures (those below the phreatic surface) the flow velocity is proportional to the pore pressure value. When this proportionality coefficient, k_s, is large compared to $k/\gamma_w c$—where k is

the permeability of the medium, γ_w is the specific weight of the fluid, and c is a characteristic length scale—the requirement of zero pore pressure on the free-drainage surface below the phreatic surface will be satisfied approximately. The drainage-only seepage coefficient in this model is specified as $k_s = 10^{-1}$ m^3/Nsec. This value is roughly 10^5 times larger than the characteristic value, $k/\gamma_w c$, based on the material properties listed below and an element length scale $\approx 10^{-1}$ m. This condition is prescribed using a pore fluid flow definition with the drainage-only flow type label (Q*n*D) as shown in *phreaticsurf_cpe8rp.inp*.

Geometry and model

The geometry of the particular earth dam considered is shown in *Figure 3*. This case is chosen because an analytical solution is available for comparison (Harr, 1962). The dam is filled to two-thirds of its height. Only a part of its base is impermeable. Since the dam is assumed to be long, we use CPE8RP coupled pore pressure/displacement plane strain elements (the mesh is shown in *Figure 4*). In addition, input files containing element types CPE4P and CPE6MP are included for verification purposes. Additional input files are included to demonstrate the use of contact pairs, tie constraints, and solution transfer capabilities in coupled pore pressure-displacement analyses.

Material

The permeability of the fully saturated earth of which the dam is made is 0.2117×10^{-3} m/sec. The default assumption is used for the partially saturated permeability: that it varies as a cubic function of saturation, decreasing from the fully saturated value to a value of zero at zero saturation. The specific weight of the water is 10 kN/m^3. The capillary action in the dam is defined by a single absorption/exsorption curve that varies linearly between a negative pore pressure of 10 kN/m^2 at a saturation of 0.05 and zero pore pressure at fully saturated conditions. This is not a very realistic model of physical absorption/exsorption behavior, but this will not affect the results of the steady-state analysis significantly insofar as the location of the phreatic surface is concerned. Accurate definition of this behavior would be required if definition of the capillary zone created by filling and emptying the dam at given rates were needed.

The initial void ratio of the earth material is 1.0. The initial conditions for pore pressure and saturation are assumed to be those corresponding to the dam being fully saturated to the upstream water level: the initial saturation is, therefore, 1.0; and the initial pore pressures vary between zero at the water level and a maximum value of 12.19 kN/m^2 at the base of the dam.

Loading and controls

The weight of the water is applied by GRAV loading, and the upstream and downstream pore pressures are prescribed as discussed above. A steady-state, coupled pore fluid diffusion/stress analysis is performed in five increments to allow Abaqus to resolve the high degree of nonlinearity in the problem.

Results and discussion

The steady-state contours of pore pressure are shown in *Figure 5*. The upper-right part of the dam shows negative pore pressures, indicating that it is partly saturated or dry. The phreatic surface is best shown in *Figure 6*, where we have chosen to draw the contours in the vicinity of zero pore pressure. This phreatic surface compares well with the analytical phreatic surface calculated by Harr (1962), shown in *Figure 2*. *Figure 7* shows contours of saturation that indicate a region of fully saturated material under the phreatic zone and decreasing saturation in and above the phreatic zone.

Input files

phreaticsurf_cpe8rp.inp	Phreatic surface calculation (element type CPE8RP).

phreaticsurf_cpe4p.inp	Element type CPE4P.
phreaticsurf_cpe6mp.inp	Element type CPE6MP.
phreaticsurf_cpe4p_contactpair.inp	Element type CPE4P using the **CONTACT PAIR* option.
phreaticsurf_cpe4p_mapsolution.inp	**MAP SOLUTION* continuation of thephreaticsurf_cpe4p_contactpair.inp analysis.
phreaticsurf_cpe4p_tie.inp	Element type CPE4P using the **TIE* option.

References

- Harr, M. E., Groundwater and Seepage, McGraw-Hill, New York, 1962.
- Pagano, L., “Steady State and Transient Unconfined Seepage Analyses for Earthfill Dams,” *Abaqus Users' Conference, Milan*, pp. 557–585, 1997.

Figures

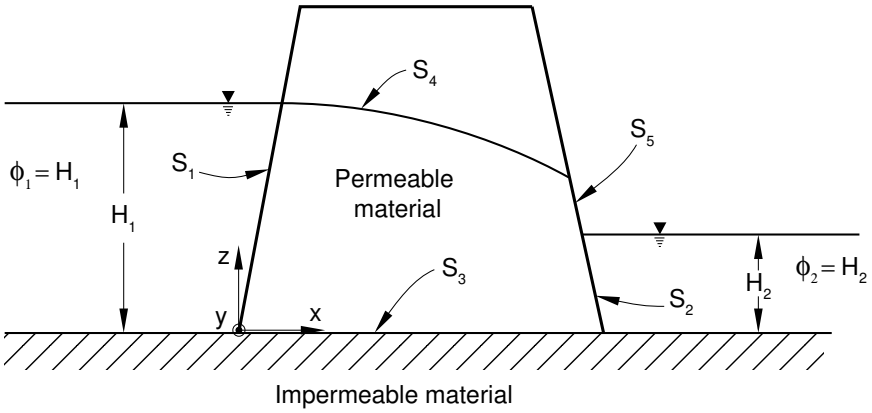


Figure 1: Phreatic surface problem.

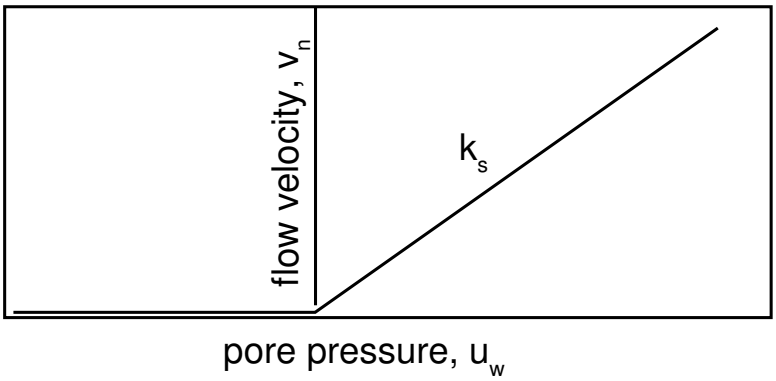


Figure 2: Pore pressure–flow velocity relationship defined on the drainage-only surface.

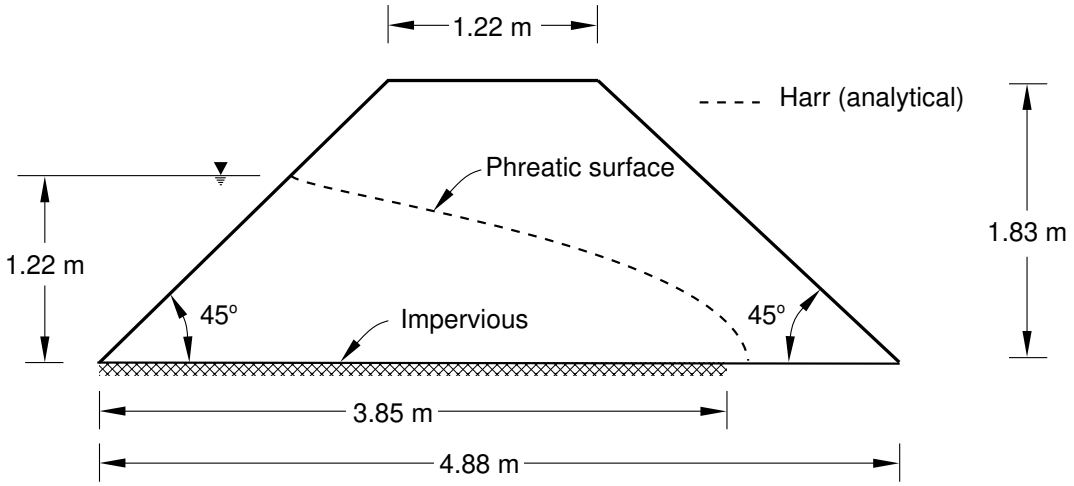


Figure 3: Configuration of earth dam and analytical phreatic surface.

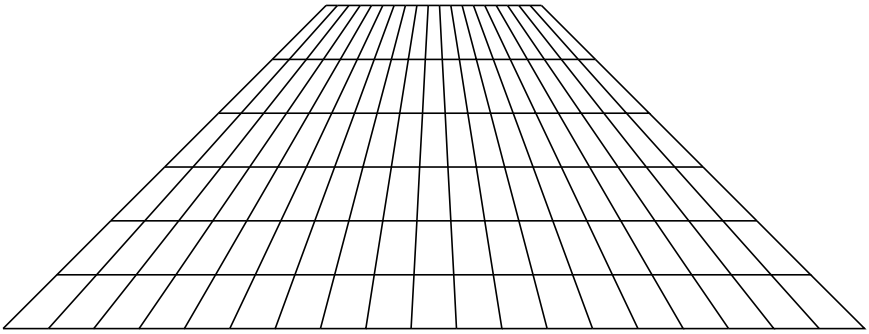


Figure 4: Finite element mesh.

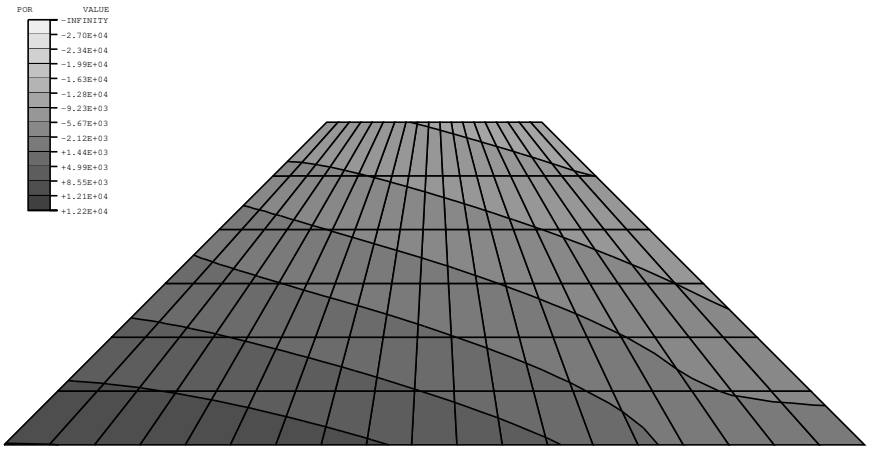


Figure 5: Pore pressure contours at steady state.

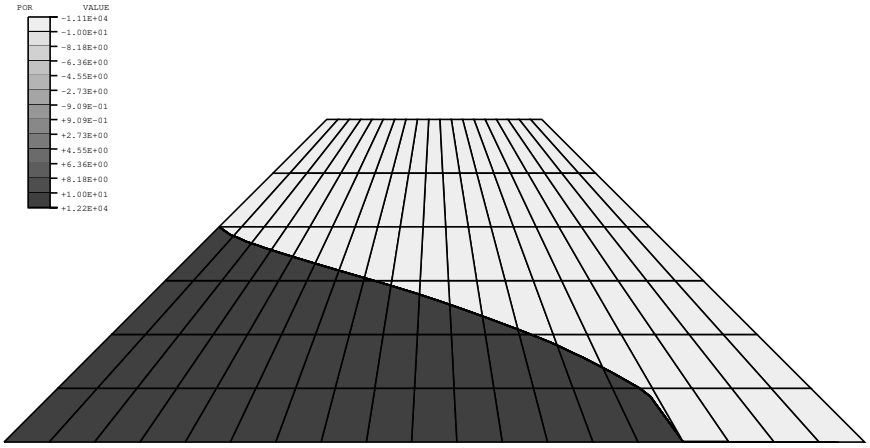


Figure 6: Pore pressure contours showing phreatic surface.

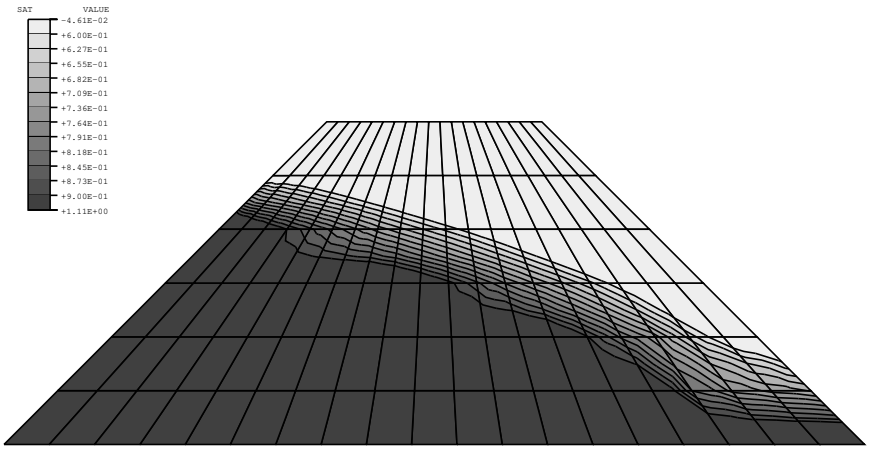


Figure 7: Saturation contours at steady state.

Axisymmetric simulation of an oil well

Products: Abaqus/Standard

Overview

This example simulates the settlement of soil near an oil well.
It is assumed that the oil in question is too thick for normal pumping. Therefore, steam is injected in the soil in the vicinity of the well to increase the temperature and decrease the oil's viscosity. As a result creep becomes an important component of the soil inelastic deformation and in the prediction of the effects of the oil pumping. Five years of oil pumping are simulated. This coupled displacement/diffusion analysis illustrates the use of Abaqus to solve problems involving fluid flow through a saturated porous medium, inelastic material properties with time-dependent creep behavior, and thermal loading. No experimental data exist to compare with the numerical results of this example.

Geometry and model

The example considers an axisymmetric model of an oil well and the surrounding soil, as shown in *Figure 1*. The radius of the well is 81 m (265 ft), and the well extends from a depth of 335 m (1100 ft) to 732 m (2500 ft). A depth of 1463 m (4800 ft) is modeled with 11 different soil layers. Reduced-integration axisymmetric elements with pore pressure, CAX8RP, are used to model the soil in the vicinity of the well. The far-field region is modeled with axisymmetric infinite elements, CINAX5R, to provide lateral stiffness. Reduced integration is almost always recommended when second-order elements are used, because it usually gives more accurate results and is less expensive than full integration. A coarse mesh is selected for the illustrative purpose of this example. No mesh convergence study has been performed.

Soil layers designated by S1, T1, U1, and L1 are modeled using the Drucker-Prager plasticity model. Both the elastic and inelastic material properties are tabulated in *Table 1*. The linear form of the Drucker-Prager model with no intermediate principal stress effect ($K = 1.0$) is used. The model assumes nonassociated flow; consequently, the material stiffness matrix is not symmetric. The use of unsymmetric matrix storage and an unsymmetric solution improves the convergence of the nonlinear solution significantly. The hardening/softening behavior is specified as an extension of the Drucker-Prager plasticity material definition, and the data are listed in *Table 1*. No creep data are provided for these layers since these are far removed from the loading. These layers are assumed to be saturated with water. A high permeability is assumed for the two top soil layers S1 and T1, while a low permeability is assigned to layers U1 and L1.

Layers D1 through D7 are modeled with the modified Drucker-Prager Cap plasticity model. The material property data are tabulated in *Table 2*. As required by the creep model, no intermediate principal stress effect is included (i.e., $K = 1.0$), and no transition region on the yield surface is defined (i.e., $\alpha = 0.0$). The material's volumetric strain-driven hardening/softening behavior is specified with a cap hardening curve definition, and the data are

listed in *Table 2*. The initial cap yield surface position, $\varepsilon_{vol}^{in}(0)$, is set to 0.02. Abaqus automatically adjusts the position of the cap yield surface if the stress lies outside the cap surface. Consolidation creep is modeled with a Singh-Mitchell type creep model. The creep material data are specified using the cap creep model and are dependent on temperature. The following creep data are specified:

- A=2.2E−7 1/day, α=3.05 1/MPa (0.021 1/psi), t_1=1.0 day, n=1.0 at 10°C (50°F)
- A=3.5E−4 1/day, α=3.05 1/MPa (0.021 1/psi), t_1=1.0 day, n=1.0, at 100°C (212°F)

These layers consist of rich organic matter and are saturated with oil. The permeability data are specified as temperature dependent.

A uniform thermal expansion coefficient of 5.76E−6 1/°C (3.2E−6 1/°F) and a constant weight density 1.0 metric ton/m^3 (64.6 lbs/ft^3) are assumed for all layers.

For a coupled diffusion/displacement analysis care must be taken when choosing the units of the problem. The coupled equations may be numerically ill-conditioned if the choice of the units is such that the numbers generated by the equations of the two different fields differ by many orders of magnitude. The units chosen for this example are inches, pounds, and days.

Initial conditions

An initial geostatic stress field is defined for the analysis and is based on the soil weight density integrated over the depth. A coefficient of lateral stress of 0.85 is assumed. An initial void ratio of 1.5 is used throughout all soil layers with an initial uniform temperature field of 10°C (50°F).

Loading

The problem is run in five steps. The first step of the analysis is a geostatic step to equilibrate geostatic loading of the finite element model. This step also establishes the initial distribution of pore pressure. Since gravity loading is defined with distributed load type BZ and not with gravity load type GRAV, the pore fluid pressure reported by Abaqus is defined as the pore pressure in excess of the hydrostatic pressure required to support the weight of pore fluid above the elevation of the material point.

The second step is a transient soils consolidation analysis step to equilibrate any creep effects induced from the initial geostatic loading step. The choice of the initial time step is important in a consolidation analysis. Because of the coupling of spatial and temporal scales, no useful information is provided by solutions generated with time steps that are smaller than the mesh and material-dependent characteristic time. Time steps that are very much smaller than this characteristic time provide spurious oscillatory results. For further discussion on calculating the minimum time step, refer to *Coupled Pore Fluid Diffusion and Stress Analysis*. For this example a minimum initial time step of one day was selected.

The third step of the analysis models the injection of steam into the well region between a depth of 366 m to 732 m (1200 ft to 2400 ft). The region is indicated by the shaded area in *Figure 1*. The nodes in this region are heated to 100°C (212°F) during a transient soils consolidation analysis. Creep effects are not considered. The injection of the steam increases the permeability of the oil and increases the soil creep behavior.

The fourth step simulates the pumping of oil by prescribing an excess pore pressure of −1.2 MPa (−170 psi) at nodes located at the depth of 427 m to 550 m (1400 ft to 1800 ft) below the surface. The pressure produces a pumping rate of approximately 172.5 thousand barrels per day at the end of the fifth year.

The final step consists of a consolidation analysis performed over a five-year period to investigate the settlement that results from pumping and creep effects in the vicinity of the well.

Results and discussion

The two initial steps show negligible deformations, indicating that the model is in geostatic equilibrium. *Figure 2* shows a contour plot of the soil settlement resulting from consolidation after the five-year period. A settlement of 0.13 m (0.4 ft) is expected at the surface. A maximum soil dislocation of 0.24 m (0.78 ft) occurs above the pump intake. *Figure 3* shows a contour plot of the excess pore pressure. The negative pore pressure represents the suction of the pump. During the five-year period, a total of 313.5 million barrels of oil are pumped (as determined from nodal output variable RVT). *Figure 4* through *Figure 6* show contour plots of the vertical stress components, plastic strains, and creep strains, respectively. Plastification occurs in soil layers D3 through D5. Significant creep occurs in the area in which steam is injected.

Input files

axisymoilwell.inp	Finite element analysis.
axisymoilwell_thermalexp.inp	Same as axisymoilwell.inp except that the thermal expansion of the pore fluid is also included.

Tables

Table 1: Soil data using Drucker-Prager model.

Soil layer	Elastic properties	Inelastic properties	Hardening behavior
S1	$E =$ 124 MPa	$\beta =$ 42.0°	0.075 MPa, 0.0
	$\nu =$ 0.3	K = 1.0	0.083 MPa, 0.058
		$\varphi =$ 0.0°	0.075 MPa, 0.116
T1	$E =$ 2068 MPa	$\beta =$ 36.0°	0.48 MPa, 0.0
	$\nu =$ 0.25	K = 1.0	0.62 MPa, 0.058
		$\varphi =$ 0.0°	0.48 MPa., 0.116
U1	$E =$ 468.8 MPa	$\beta =$ 38.0°	1.97 MPa, 0.0
	$\nu =$ 0.22	K = 1.0	3.17 MPa, 0.0037
		$\varphi =$ 0.0°	2.47 MPa, 0.04
L1	$E =$ 2482 MPa	$\beta =$ 38.0°	1.97 MPa, 0.0
	$\nu =$ 0.29	K = 1.0	3.17 MPa, 0.0037
		$\varphi =$ 0.0°	2.47 MPa, 0.04

Table 2: Soil data using modified Drucker-Prager cap model.

Soil layer	Elastic properties	Inelastic properties		Hardening behavior
D1	$E =$ 328 MPa	d = 1.38 MPa	$\alpha =$ 0.0	2.75 MPa, 0.0
	$\nu =$ 0.17	$\beta =$ 36.9°	K = 1.0	4.14 MPa, 0.02
		R = 0.33	$\varepsilon_{vol}^{in}(0) =$ 0.02	5.51 MPa, 0.05
				6.20 MPa, 0.09
D2	$E =$ 434 MPa	d = 1.38 MPa	$\alpha =$ 0.0	1.38 MPa, 0.0
	$\nu =$ 0.17	$\beta =$ 39.4°	K = 1.0	4.14 MPa, 0.02
		R = 0.33	$\varepsilon_{vol}^{in}(0) =$ 0.02	6.89 MPa, 0.04
				55.1 MPa, 0.1
D3	$E =$ 546 MPa	d = 1.38 MPa	$\alpha =$ 0.0	1.38 MPa, 0.0
	$\nu =$ 0.19	$\beta =$ 42.0°	K = 1.0	3.45 MPa, 0.02

Soil layer	Elastic properties	Inelastic properties		Hardening behavior
		R = 0.34	$\varepsilon_{vol}^{in}(0) = 0.02$	13.8 MPa, 0.04
				62.0 MPa, 0.06
D4	$E =$ 411 MPa	d = 1.2 MPa	$\alpha =$ 0.0	1.38 MPa, 0.0
	$\nu =$ 0.2	$\beta =$ 40.1°	K = 1.0	5.03 MPa, 0.02
		R = 0.3	$\varepsilon_{vol}^{in}(0) = 0.02$	6.90 MPa, 0.10
				62.0 MPa, 0.3
D5	$E =$ 494 MPa	d = 1.38 MPa	$\alpha =$ 0.0	2.75 MPa, 0.0
	$\nu =$ 0.17	$\beta =$ 40.4°	K = 1.0	4.83 MPa, 0.02
		R = 0.3	$\varepsilon_{vol}^{in}(0) = 0.02$	5.15 MPa, 0.04
				62.0 MPa, 0.08
D6	$E =$ 775 MPa	d = 17 MPa	$\alpha =$ 0.0	2.76 MPa, 0.0
	$\nu =$ 0.17	$\beta =$ 50.2°	K = 1.0	4.14 MPa, 0.005
		R = 0.23	$\varepsilon_{vol}^{in}(0) = 0.02$	7.58 MPa, 0.02
				62.0 MPa, 0.05
D7	$E =$ 1,121 MPa	d = 1.7 MPa	$\alpha =$ 0.0	3.44 MPa, 0.0
	$\nu =$ 0.17	$\beta =$ 58.5°	K = 1.0	4.14 MPa, 0.006
		R = 0.23	$\varepsilon_{vol}^{in}(0) = 0.02$	7.58 MPa, 0.012
				67.6 MPa, 0.03

Figures

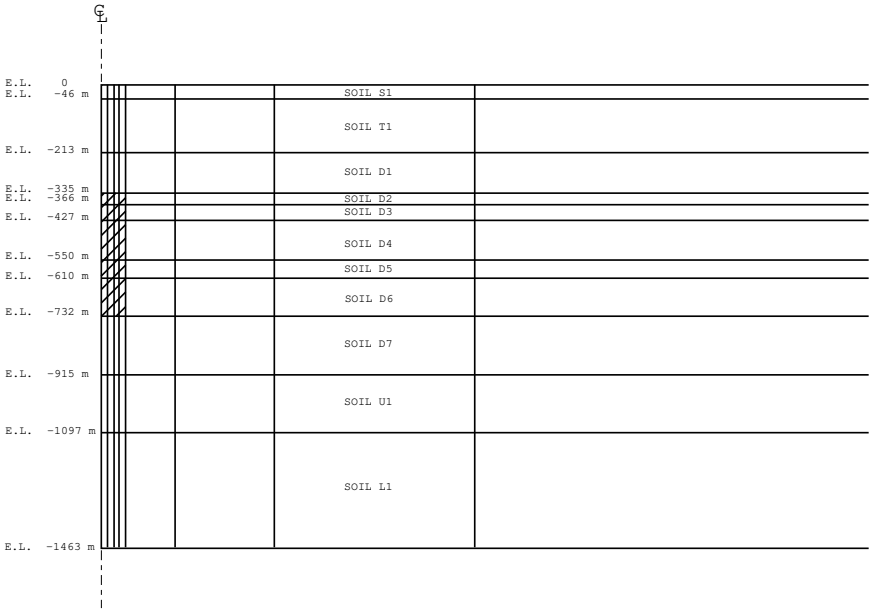


Figure 1: Axisymmetric model of oil well and surrounding soil.

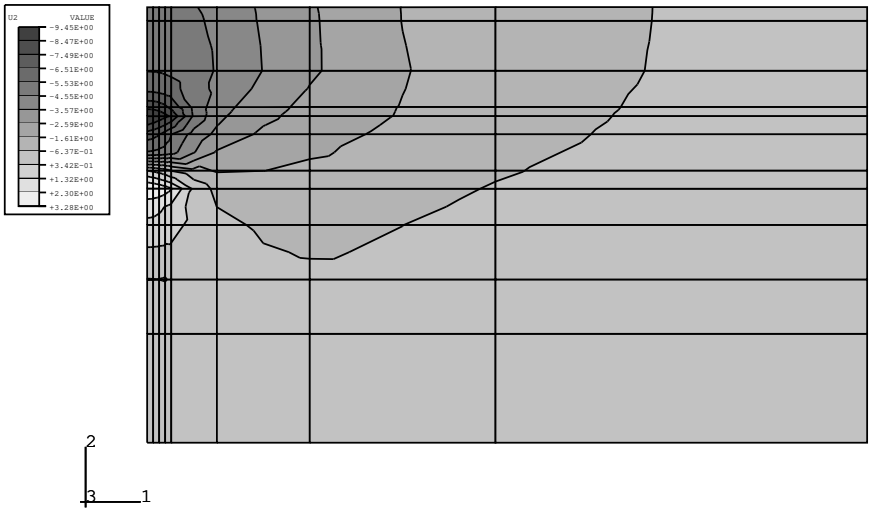


Figure 2: Soil settlement after five-year period.

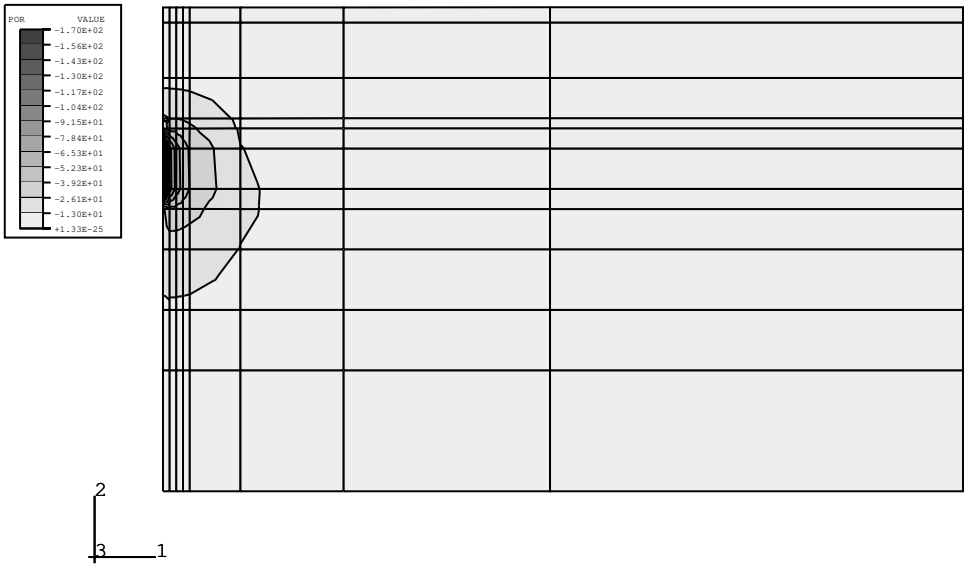


Figure 3: Contour plot of the pore pressure.

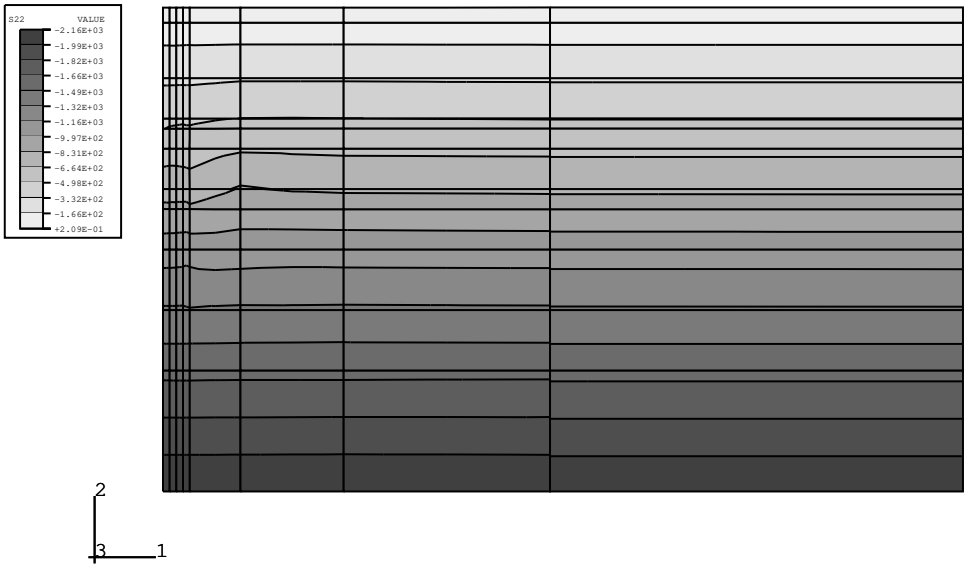


Figure 4: Contour plot of the vertical stress components.

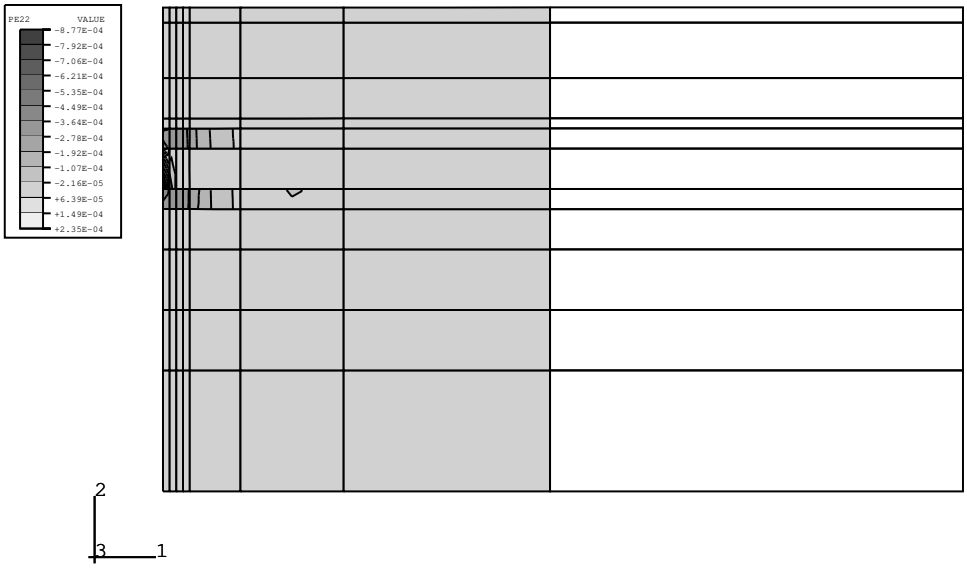


Figure 5: Contour plot of the vertical plastic strain components.

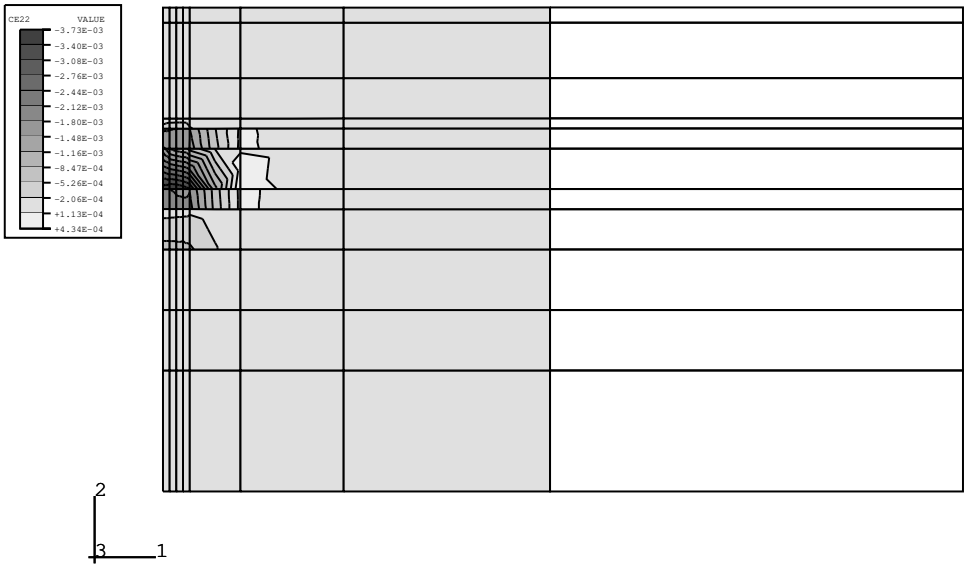


Figure 6: Contour plot of the vertical creep strain components.

Analysis of a pipeline buried in soil

Products: Abaqus/Standard

Overview

This example verifies the use of pipe-soil interaction elements to model a buried pipeline subjected to large ground movement.

Oil and gas pipelines are usually buried in the ground to provide protection and support. Buried pipelines may experience significant loading as a result of relative displacements of the ground along their length. Such large ground movement can be caused by faulting, landslides, slope failures, and seismic activity.

Abaqus provides a library of pipe-soil interaction (PSI) elements to model the interaction between a buried pipeline and the surrounding soil. The pipeline itself is modeled with any of the beam, pipe, or elbow elements in the Abaqus/Standard element library. The ground behavior and soil-pipe interaction are modeled with the pipe-soil interaction elements. These elements have only displacement degrees of freedom at their nodes. One side or edge of the element shares nodes with the underlying beam, pipe, or elbow element that models the pipeline. The nodes on the other edge represent a far-field surface, such as the ground surface, and are used to prescribe the far-field ground motion. The elements are described in detail in *Pipe-Soil Interaction Elements*.

The purpose of this example is to determine the stress state along the length of a infinitely long buried pipeline subjected to large fault movement of 1.52 m (5.0 ft), as shown in *Figure 1*. The pipeline intersects the fault at 90.0°. The results are compared with results from an independent analysis, as described below.

Problem description

The problem consists of an infinitely long pipeline buried at a depth of 6.1 m (20.0 ft.) below the ground surface. Only a 610.0 m (2000.0 ft.) long section of the pipeline is modeled. The outside radius of the pipe is 0.61 m (24.0 in), and the wall thickness is 0.0254 m (1.0 in). The pipeline is modeled with 50 first-order PIPE21 elements. A nonuniform mesh, with smaller elements focused near the fault, is used.

The pipe-soil interaction behavior is model with PSI24 elements. The PSI elements are defined so that one edge of the element shares nodes with the underlying pipe element, and the nodes on the other edge represent a far-field surface where ground motion is prescribed. The far-field side and the side that shares nodes with the pipeline are defined by the element connectivity.

A three-dimensional model that uses PIPE31 and PSI34 elements is also included for verification purposes.

Material

The pipeline is made of an elastic–perfectly plastic metal, with a Young's modulus of 206.8 GPa (30×10^6 lb/in^2), a Poisson's ratio of 0.3, and a yield stress of 413.7 MPa (60000 lb/in^2).

The pipe-soil interaction behavior is elastic–perfectly plastic. The nonlinear constitutive model is used to define the interaction model. The behavior in the vertical direction is assumed to be different from the behavior along the axial direction. It is further assumed that the pipeline is buried deep below the ground surface so that the response is symmetric about the origin. Abaqus also allows a nonsymmetric behavior to be defined in any of the directions (this is usually the case in the vertical direction when the pipeline is not buried too deeply). The ultimate force per unit length in the axial direction is 730.0 N/m (50.0 lb/ft), and in the vertical direction it is 1460.0 N/m (100.0 lb/ft). The ultimate force is reached at 0.0304 m (0.1 ft) in both the horizontal and vertical directions.

The loading occurs in a plane (axial-vertical), so the properties for the pipe-soil interaction behavior in the transverse horizontal direction are not important.

Loading

The loading on the pipeline is caused by a relative vertical displacement 1.52 m (5.0 ft) along the fault line. It is assumed that the effect of the vertical ground motion decreases linearly over a distance of 91.4 m (300.0 ft.) from the origin of fault, as shown in *Figure 1*.

This linear distribution of ground motion is prescribed as follows. Rigid (R2D2) elements are connected to the far-field edges of the PSI to create two rigid surfaces, one on each side of the fault line. These surfaces extend a distance of 91.4 m (300.0 ft.) from the origin of the fault. The rigid body reference nodes are also placed a distance of 91.4 m (300.0 ft.) from the fault on the ground surface. The fault movement is modeled by prescribing a rotation to each of the rigid body reference nodes so that a positive vertical displacement of 0.76 m (2.5 ft) is obtained on one side of the fault and a negative vertical displacement of 0.76 m (2.5 ft) is obtained on the other side of the fault, as shown in *Figure 2*. All degrees of freedom on the remaining far-field nodes are fully fixed. In addition, the two end points of pipeline are fully fixed. *Figure 2* does not show the PSI elements or any of the remaining nodes on the ground surface.

Reference solution

The reference solution is obtained by using JOINTC elements between the pipeline and ground nodes to model the pipe-soil interaction. These elements provide an internal stiffness, which is modeled with linear or nonlinear springs; nonlinear springs are used in this example. The behavior of the nonlinear spring is elastic in the sense that reversed loading does not result in permanent deformation. This behavior is different from the behavior provided by the nonlinear PSI elements. However, this is not a limitation in this example since the loading is monotonic.

Another distinct difference between JOINTC elements and PSI elements is that the spring behavior associated with JOINTC elements is defined in terms of total force, whereas the constitutive behavior for PSI elements is defined as a force/unit length. This difference requires us to define a separate stiffness for each JOINTC element or to use a uniform mesh with JOINTC elements spaced at unit length intervals along the pipeline. A unit length mesh is used in this example.

Results and discussion

Figure 3 and *Figure 4* show the axial and vertical forces per unit length applied to the pipeline due to relative ground motion. The figures show that permanent deformation occurs in the pipe-soil interaction model near the fault along the axial and horizontal directions, with purely elastic behavior further from the fault.

Figure 5 compares the axial stress in the bottom wall of the pipeline with the reference solution. The figure shows that the pipeline behavior is purely elastic. The figure also shows close agreement with the reference solution. The small differences between the solutions can be accounted for by the different mesh densities. The reaction forces at the pipeline edges and the maximum pipeline displacements are also in close agreement with the reference solution.

Input files

buriedpipeline_2d.inp	Two-dimensional model using PSI24 elements.
buriedpipeline_3d.inp	Three-dimensional model using PSI34 elements.
buriedpipeline_ref.inp	Reference solution using JOINTC elements.

References

- Audibert, J. M. E., D. J. Nyman, and T. D. O'Rourke, "Differential Ground Movement Effects on Buried Pipelines," *Guidelines for the Seismic Design of Oil and Gas Pipeline Systems, ASCE publication*, pp. 151–180, 1984.

Figures

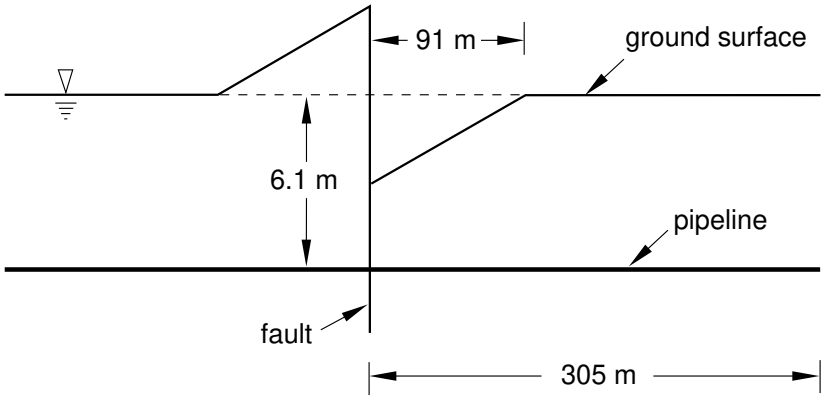


Figure 1: Pipe with fault motion.

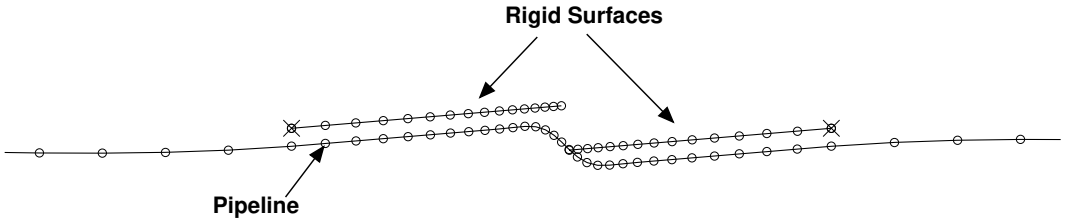


Figure 2: Displaced shape (magnification factor=10.0).

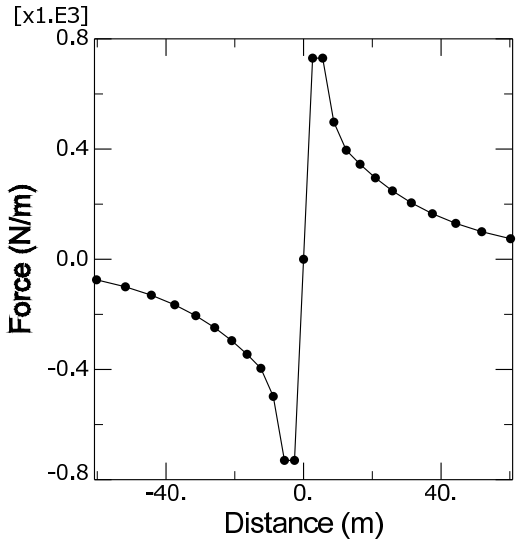


Figure 3: Axial force/unit length applied along the pipeline.

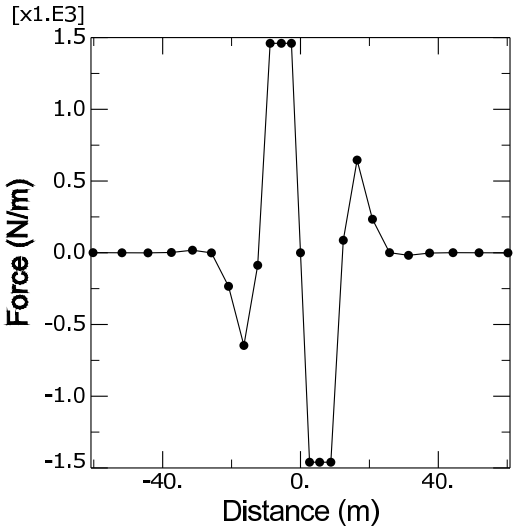


Figure 4: Vertical force/unit length applied along the pipeline.

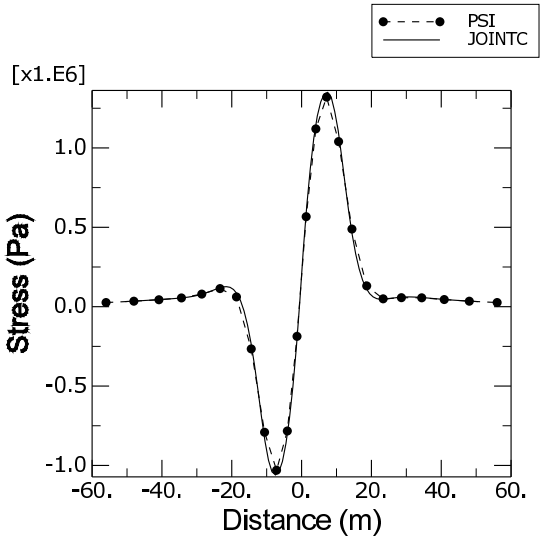


Figure 5: Axial stress along the bottom of the pipeline.

Hydraulically induced fracture in a wellbore

Products: Abaqus/Standard

Overview

This example demonstrates the use of pore pressure cohesive elements and fluid pipe elements to model the initiation and opening of hydraulically induced cracks near an oil well borehole.
With the technique illustrated in this section you can assess the quantitative impact of the hydraulic fracture process on wellbore productivity.

Problem description

The hydraulic fracture process is commonly used in the production of oil and natural gas reservoirs as a means of increasing well productivity and extending the production lifetime of the reservoir. The objectives of a hydraulic fracture treatment are

1. to create more surface area that is exposed to the hydrocarbon bearing rocks, and
2. to provide a highly conductive pathway that allows the hydrocarbons to flow easily to the wellbore.

The productivity of a hydraulically fractured oil or gas well is directly related to the extent of the fracture and to how well the wellbore is connected to the fracture. Some rock formations contain natural fracture systems that can further increase a well's productivity, provided that the generated hydraulic fracture can grow such that it intersects these natural fractures.

A hydraulic fracture job consists of pumping fluids into a well at very high pressures so that the tractions created on the well-bore face reduce the in-situ (compressive) stress in the rock so much that the rock fractures. Once a fracture initiates in the rock formation it is possible, given enough hydraulic fluid, to propagate the fracture for a considerable distance, sometimes as far as a hundred meters or more.

Pumping stages

The initial stage often involves pumping a rather small amount of polymer laden fluid, typically 1–20 barrels (.15 to 3.2 cubic meters) so that data can be gathered on the pressure needed to fracture the formation and the rate at which the fluid will "leakoff" from the fracture into the pore space of the rock. The data gathered are used to plan subsequent stages of the job. The main stage of the job might consist of anywhere from one hundred to several thousands of barrels of hydraulic fluid. The size of this stage is determined by the target fracture size, the leak-off rate, and the capacity (rate) of the pumps.

During the next stage of the fracture job, solid material, known as proppant, is added to the injected fluid and is carried into the fracture volume. Chemicals, typically polymers, are added to the fluid in each stage of a fracture job to produce the necessary properties in the fluid (viscosity, leakoff, density). In the last stage of the job, chemicals are pumped into the fracture that help breakdown the polymers used in the previous stages and make it easier to flow fluid back through the fracture without disrupting the proppant material.

Geometry and model

The domain of the problem considered in this example is a 50 m (1969 in) thick slice of oil-bearing rock with the well borehole modeled. The domain has a diameter of 400 m (15,748 in). Three sections of rock are considered: a region where production oil recovery is targeted, and two surrounding regions of shale, as shown in *Figure 1*.

Due to symmetry only a half of the domain is modeled. *Figure 2* shows the finite element model. The rock is modeled with C3D8RP elements, and the borehole casing is modeled with M3D4 elements.

The unopened fracture is modeled along the entire height of the model domain. Cohesive elements (COH3D8P) are used to model a vertical fracture surface.

The flowing fluid in the wellbore is modeled using fluid pipe and fluid pipe connector elements. The fluid pipe elements are tied to the formation to induce hydraulic fracture in a more realistic fashion. The valve behavior is activated on the connector element with user subroutine *UFLUIDCONNECTORVALVE* to turn off flow to the formation after the pumping stage is completed.

Rock constitutive model

A linear Drucker-Prager model with hardening is chosen for the rock, while the casing is linear elastic.

Fracture constitutive model

The fracture model consists of the mechanical behavior of the fracture itself and the behavior of the fluid that enters and leaks through the fracture surfaces.

Fracture mechanical behavior

The elastic properties of the bonded interface are defined using a traction-separation description, with stiffness values of $K_{nn}=K_{ss}=K_{tt}= 8.5 \times 10^4$ MPa. The quadratic traction-interaction failure criterion is chosen for damage initiation in the cohesive elements; and a mixed-mode, energy-based damage evolution law is used for damage propagation. The relevant material data are as follows: $N_0 = S_0 = 0.32$ KPa, $G_{1C}= G_{2C}=G_{3C}= 28$ N/mm, and $\eta= 2.284$.

Fluid model

Tangential and normal flow are both modeled in the fracture zone cohesive elements. The following parameters are specified:

- Gap flow is specified as Newtonian with a viscosity of 1×10^{-6} kPaS (1 centepoise), roughly the viscosity of water.
- Fluid leakoff is specified as 5.879×10^{-10} m/(kPa s) for the early stages. In the final stage, when the polymer is dissolved, the fluid leak-off coefficient is increased to 1×10^{-3} m/(kPa s). This step-dependent fluid leak-off coefficient is set in user subroutine *UFLUIDLEAKOFF*.

Initial conditions

An initial geostatic stress field is defined using the user subroutines *SIGINI* and *UPOREP*. A depth varying initial void ratio is specified using user subroutine *VOIDRI*. Gravity loading is specified; and an orthotropic overburden stress state is imposed, with the maximum principle stress in the formation aligned orthogonal to the cohesive element fracture plane.

Loading and boundary conditions

The analysis consists of four steps:

1. A geostatic step is performed where equilibrium is achieved after applying the initial pore pressure to the formation and the initial in-situ stresses. The bottom hole shut-in pressure is applied as a traction to the well-bore face.

2. The next step represents the hydraulic fracture stage where the main volume of fluid is being injected into the well. Flow at a rate of 2.4 m^3 (15 barrels) per minute is injected along a targeted 8 m extent in the target formation in the model, and the cohesive elements adjacent to the wellbore along this length are defined as initially open to permit entry of fluid. The duration of this stage is 20 minutes.
3. Following the hydraulic fracture, another transient soils consolidation analysis is conducted. The injection into the well is terminated, and the built-up pore pressure in the fracture is allowed to bleed off into the formation. An additional boundary condition is applied at this stage, fixing the fracture surface open to simulate the behavior of the proppant material that was injected into the fracture.
4. In the final step a drawdown pressure of 20 kPa is applied to the well-bore nodes of the fracture cohesive elements. This step ends after 100 days of production or when steady-state conditions prevail, defined as pore pressure transients below 0.05 kPa/sec in the model.

When the fluid pipe elements are used to simulate fluid flow in the wellbore, the loading conditions in the analysis are modified to specify the flow as an inlet condition at the top of the wellbore. The bottom of the wellbore is tied to the formation to enable automatic injection of the fluid. User subroutine *UFLUIDCONNECTORVALVE* is used to shut off flow to the formation during and after the third step of the analysis as described above.

Results and discussion

The flow injected during Step 2, the pumping stage, initiated and grew a crack extending outward from the wellbore. *Figure 3* shows the resulting geometry of the fracture at the end of the 20 minute pumping period. These results show that a fracture initiated within the target formation zone tends to avoid the lower shale region, where the compressive stresses are higher, but does infiltrate the upper shale region, where it can decrease well yields.

Figure 4 shows the fracture opening profile at various times during the pumping stage; the final profile, shown at 1200 s, is then fixed in the subsequent steps, holding the fracture open with the proppant material. *Figure 5* shows a similar history of the pore pressure across the fracture face and indicates that the pore flow has stabilized.

Figure 6 shows the resulting well-bore yield following the hydraulic fracture process. This is compared to an equivalent model where the hydraulic fracturing did not occur. In this simple example the hydraulically fractured wellbore shows a marked improvement, with a flow rate more than 100 times the unfractured configuration.

Input files

exa_hydfracture.inp	Hydraulic fracture analysis.
exa_hydfracture.f	User subroutines *DISP*, *DLOAD*, *SIGINI*, *UPOREP*, *UFLUIDLEAKOFF*, and *VOIDRI*.
exa_hydfracture-flpipe.inp	Hydraulic fracture analysis with flow in the wellbore modeled with fluid pipe elements.
exa_hydfracture-pipe.f	User subroutines *DISP*, *DLOAD*, *SIGINI*, *UPOREP*, *UFLUIDLEAKOFF*, *VOIDRI*, and *UFLUIDCONNECTORVALVE*.

Figures

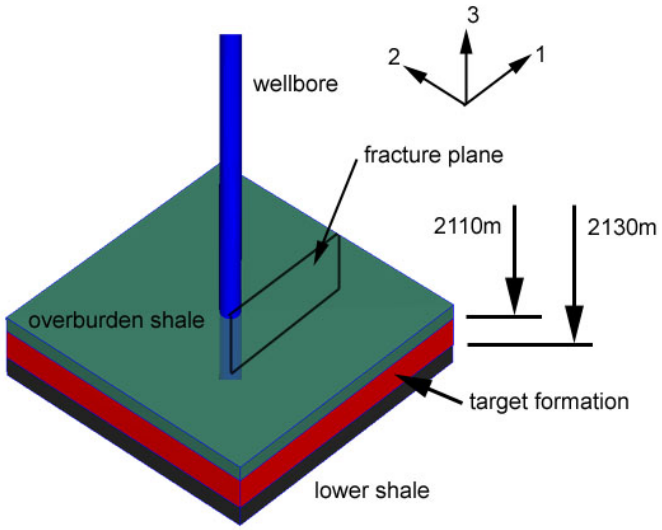


Figure 1: Position of the target formation and surrounding shale.

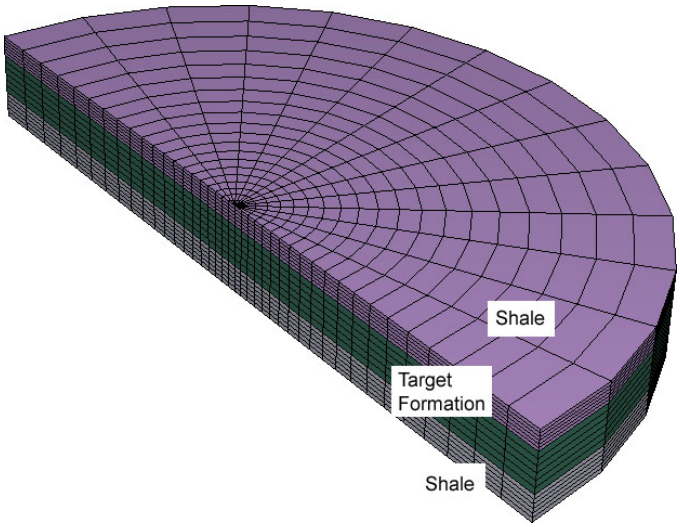


Figure 2: Near well-bore mesh.

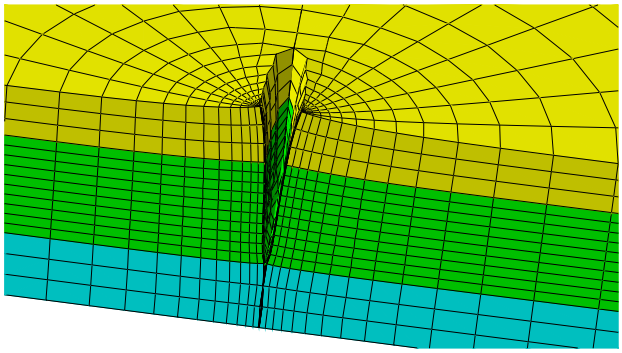


Figure 3: Fracture geometry following the injection stage, with the deformation magnified 50 times.

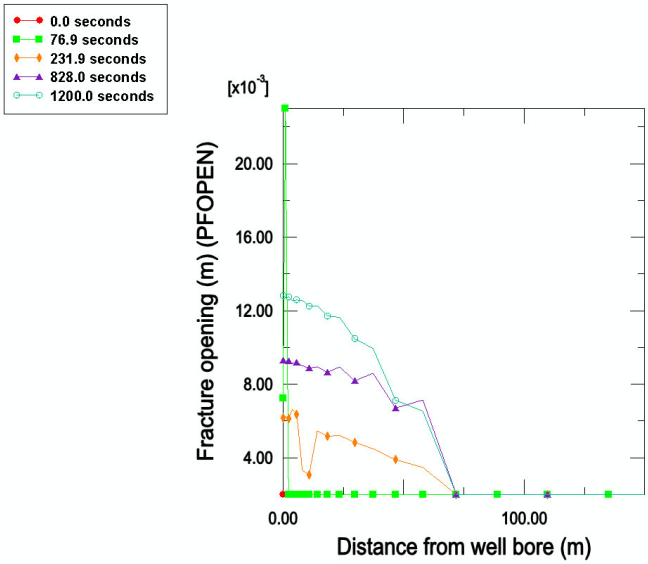


Figure 4: History of the fracture opening profile.

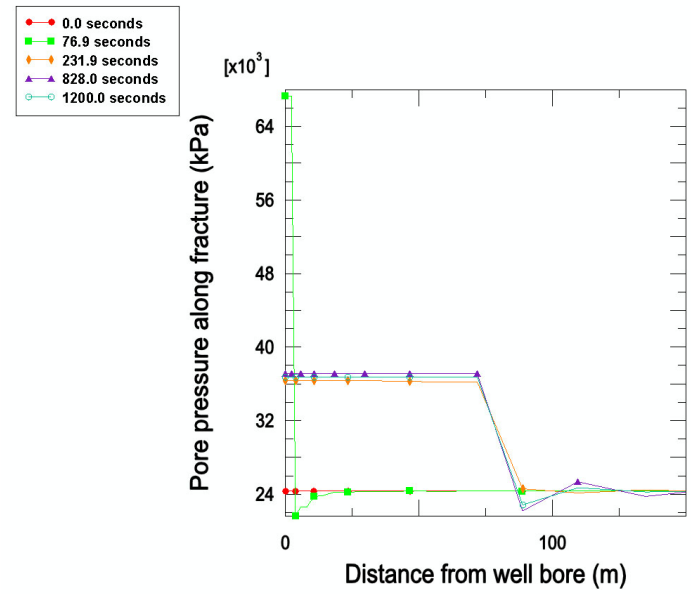


Figure 5: History of the fracture pore pressure profile.

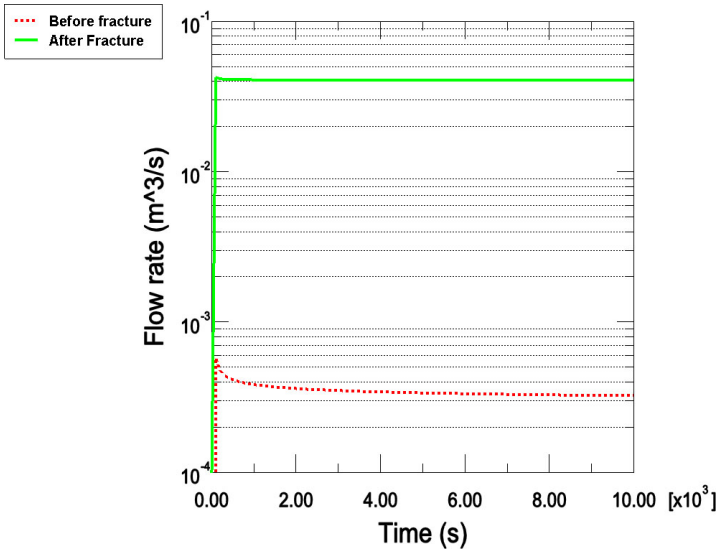


Figure 6: Well yield before and after the hydraulic fracture process.

Permafrost thawing–pipeline interaction

Products: Abaqus/Standard

References:

- *Coupled Pore Fluid Diffusion and Stress Analysis*
- *Geostatic Stress State*
- **GEOSTATIC*
- **SOILS*

Overview

This example illustrates the use of the coupled temperature–pore pressure displacement modeling capability in Abaqus/Standard to model the effects of thaw settlement that commonly occurs in relatively warm pipelines buried in permafrost.

Application description

Thaw settlement generally occurs in shallow waters and at shore crossings where soil ice-bonded permafrost underlies the pipeline (Xu et al., 2009). Due to the heat released from the relatively warm pipeline, the surrounding permafrost may gradually thaw over years of operation and create a permafrost thaw bulb. The load-carrying capacity of the soil is generally believed to be reduced as a result of the thaw formation, which could in turn (as a result of the weight of the soil above it) lead to excessive deformation and stress in the pipeline leading to its eventual damage.

Two primary mechanisms control the behavior of the problem: heat transfer from the warm pipeline to the surrounding permafrost leading to thawing of the permafrost and diffusion of the pore fluid and associated consolidation of the thawed soil mass surrounding the pipeline. The analysis is carried out utilizing two approaches. The first approach is a sequential one, in which the permafrost melting process is modeled as a pure heat transfer analysis in Abaqus, and the resulting temperature is used to drive a soils consolidation analysis. The second approach is fully coupled, in which the heat transfer and soils consolidation problems are solved in a fully coupled manner in a single analysis. This approach uses the coupled temperature–pore pressure elements that solve for the temperature field as a nodal degree of freedom in addition to the displacement and the pore pressure fields. The results obtained using these two approaches are compared, and the advantages of using a fully coupled approach are discussed. The model presented here has been adapted from Xu et al. (2009), where the first approach alone was used to investigate this problem.

Geometry

Figure 1 shows the geometry of the model consisting of two outer sections of sand separated by an inner section of clay, which approximates the nonuniform soil properties that are often encountered in practice along a pipeline. The geometry assumes symmetry about the right vertical plane in the figure, and the dimensions of the soil mass are as shown. The pipe diameter and thickness are assumed to be 0.4 m (15.76 in) and 0.096 m (3.8 in), respectively, and it is assumed to be buried 2.5 m below the soil surface.

Materials

The material properties assumed for this analysis are shown in *Table 1*, *Table 2*, *Table 3*, and *Table 4* for clay, sand, steel, and water (pore fluid), respectively. The symbols E, ν, K, ρ, c, L, α, k, and e represent the Young's modulus, Poison's ratio, thermal conductivity, density, specific heat, latent heat, thermal expansion coefficient,

permeability, and void ratio, respectively. The elastic and the pore fluid flow properties of the soil are assumed to depend on whether the soil is frozen (more precisely, whether the water in the soil is frozen) or thawed. The soil is assumed to be significantly stiff and impermeable when frozen compared to the corresponding properties when thawed. The properties in the frozen state and the thawed state are distinguished from each other using the symbols F and T, respectively, next to a property symbol. This approach approximately captures the effects of phase change of the pore fluid on the mechanical properties of the soil and can be improved upon with a more detailed modeling of the physics of the phase change process. In the heat transfer part of the sequential approach, the thermal properties are assumed to be weighted averages of the properties of the individual constituents (clay or sand, and water). The latent heat of fusion from ice to water is utilized to define the phase change of the soil. The pipe is assumed to be made out of steel, with the mechanical behavior defined using a linear elastic model along with thermal expansion. Inelastic response of the pipe material, such as plasticity and creep, can be easily included in the model if needed. Both the soil and the pore fluid are also assumed to undergo thermal expansion associated with the temperature variation. Thermal expansion is an important part of the overall physics of the problem, with the differential expansion of the pore fluid relative to pores in the soil often determining the initial rise in the pore pressure.

Initial conditions

The initial temperature is assumed to be −5°C everywhere in the model. The initial void ratio, pore pressure, and saturation are assumed to be 1.8, 0.0, and 1.0, respectively, everywhere in the model. Thus, the pore fluid flow is assumed to occur under fully saturated conditions. Initial effective geostatic stress in the vertical direction is defined as a function of the depth below the soil surface. The stress in the horizontal direction is assumed to be a fraction of the stress in the vertical direction.

Boundary conditions and loading

All the outer surfaces of the model (except the top surface) are assumed to be restrained from moving in a perpendicular direction. Following Xu et al. (2009), the model assumes the flow of heat and fluid across all the outer surfaces (except the top surface) to be zero. The top surface allows free drainage (zero pore pressure). The axial displacement at both ends of the pipe is restrained. This simple example model does not include gravity in the loading. In general, gravity should be taken into account for more realistic consolidation analyses.

Abaqus modeling approaches and simulation techniques

In the consolidation part of the analysis in the sequential approach, the soil is modeled using pore pressure elements with displacement and pore pressure degrees of freedom, while the pipe is modeled using solid continuum elements using displacement degrees of freedom only. In the coupled analysis, the soil is modeled using the coupled elements with displacement, pore pressure, and temperature degrees of freedom, while the pipe is modeled using coupled elements with displacement and temperature degrees of freedom only. The soil and the pipe share nodes at the interface. An alternate approach would model this interface utilizing contact constraints; however, this approach has not been used because contact is not supported with the coupled temperature–pore pressure elements.

Analysis types

The sequential approach utilizes a heat-transfer analysis first to determine the temperature profile. This temperature profile is subsequently used to drive the mechanical part of the simulation that does not model heat transfer. The latter consists of two steps: a geostatic analysis, followed by a soils consolidation analysis. On the other hand, the coupled approach solves the thermal and mechanical problem in a fully coupled manner and consists of two steps: a geostatic analysis, followed by a soils consolidation analysis.

Mesh design

The heat transfer part of the sequential approach uses first-order brick (DC3D8) elements, while the consolidation part of the sequential approach uses first-order reduced-integration (C3D8RP and C3D8R) elements for the soil and pipe regions, respectively. The coupled analysis uses first-order reduced-integration fully coupled (C3D8RPT and C3D8RT) elements for the soil and pipe regions, respectively.

Loads

Inside the pipe the fluid is assumed to be flowing at 20°C. A constant convection coefficient on the inner pipe wall is assumed. The top surface of the soil is subjected to a sinusoidal temperature variation ranging between −5°C and 5°C. The above boundary condition approximately represents the season variation of temperature at the soil surface. The weight of the pipe is considered as a load and is modeled as a body force in the vertical direction.

While the physics of this problem will likely span two or more decades, the present analysis is carried out for a period of one year in order to present the basic framework for an analysis of this type.

Solution controls

The simulations use solution controls to specify a nondefault initial value of the time average pore fluid flux. The default choice may not be appropriate in situations such as those encountered in the present problem where the fluid velocities are, in relative terms, lower compared to typical flux values encountered for other fields (such as displacements or rotations). Without the above specification, the increments would be treated as linear from the viewpoint of the continuity equation. In other words, without using solution controls to specify a nondefault initial value of the time average pore fluid flux, the pore fluid part of the incrementation will be treated as linear. Consequently, the continuity equation would be assumed to have been satisfied at the first iteration itself, without performing any further iterations to compute corrections to pore pressure.

Results and discussion

Figure 2 shows a contour of the temperature field around the buried pipe after a period of one year. The temperature profile can be used to determine the thaw bulb around the pipe. The results are obtained from the heat transfer part of the sequential approach; the coupled analysis produces similar results. The thaw bulb represents the region where the permafrost has thawed and the temperature is above freezing. *Figure 2* shows the mesh for the whole model along with the contour plot of the fluid velocities. The zero pore pressure boundary condition on the top surface allows the seepage of pore fluid across this surface. Clay is significantly less permeable compared to sand. The effect can be observed in *Figure 3*: the flow across the top surface in the clay section is less compared to the flow in the sand section.

Figure 4 shows the temperature history at a point (node number 1591) in the soil in the vicinity of the pipe. The two curves were obtained from the uncoupled (sequential) and coupled analysis. The temperature results are, in relative terms, close to each other before thawing. However, upon thawing the results from the coupled analysis predict a higher temperature. The latter may be explained by recalling that unlike the coupled analysis, the heat transfer part of the uncoupled (sequential) analysis does not account for heat transfer associated with pore fluid flow. The results from both the sequential and coupled analyses predict that the temperature does not increase significantly for a certain period of time after it reaches 0°C. This behavior is due to phase change of the pore fluid as a result of thawing; all the available heat is used up as the latent heat of fusion.

Figure 5 shows the normalized (by the maximum vertical displacement from the uncoupled analysis) vertical displacement as a function of length along the span of the pipe for both the uncoupled (sequential) and coupled analyses. The uncoupled analysis predicts a higher peak displacement compared to the coupled analysis. This can be attributed to the difference in temperatures at points in the vicinity of the pipe as predicted by the two analyses and the associated effects on the elastic stiffness of the surrounding soil mass. It is particularly interesting

to note that a fully coupled analysis can be used as a basis for a less conservative design as the predicted displacements and associated stresses are lower.

Input files

buried_pipe_permafrost_ht.inp	Heat transfer part of the sequential analysis.
buried_pipe_permafrost_consol.inp	Consolidation part of the sequential analysis.
buried_pipe_permafrost_ctup.inp	Fully coupled analysis.

References

- Xu, J., B. Abdalla, A. Eltaher, and P. Jukes, "Permafrost Thawing-Pipeline Interaction Advanced Finite Element Model," Proceedings of the ASME 2009 28th International Conference on Ocean, Offshore, and Arctic Engineering, OMAE2009–79554, 2009.

Tables

Table 1: Properties for clay (F-frozen state, T-thawed state).

Property	Value
E (F)	1.03421×10^{11} Pa
E (T)	6.89476×10^{5} Pa
ν	0.30
K	2.0 J/sec m°C
ρ	1602.0 kg/m^3
c	1381.0 J/kg °C
α	9.67×10^{-6} /°C
k (F)	1.0×10^{-14} m/sec
k at $e = 1.0$ (T)	1.0×10^{-8} m/sec
k at $e = 1.8$ (T)	2.0×10^{-8} m/sec

Table 2: Properties for sand (F-frozen state, T-thawed state).

Property	Value
E (F)	1.03421×10^{11} Pa
E (T)	1.03421×10^{7} Pa
ν	0.30
K	2.0 J/sec m°C
ρ	1602.0 kg/m^3
c	1381.0 J/kg °C

Property	Value
α	5.60×10^{-6} /°C
k (F)	1.0×10^{-13} m/sec
k at $e = 1.0$ (T)	1.0×10^{-7} m/sec
k at $e = 1.8$ (T)	2.0×10^{-7} m/sec

Table 3: Properties for steel.

Property	Value
E	2.0×10^{11} Pa
ν	0.30
K	52.0 J/sec m°C
ρ	7832.0 kg/m^3
c	434.0 J/kg °C
α	12.0×10^{-6} /°C

Table 4: Properties for water (pore fluid).

Property	Value
K	0.58 J/sec m°C
ρ	1000.0 kg/m^3
c	4186.0 J/kg °C
α (at −5°C)	51.0×10^{-6} /°C
α (at 0°C)	51.0×10^{-6} /°C
α (at 1°C)	-16.67×10^{-6} /°C
α (at 4°C)	0.0
α (at 10°C)	29.33×10^{-6} /°C

Figures

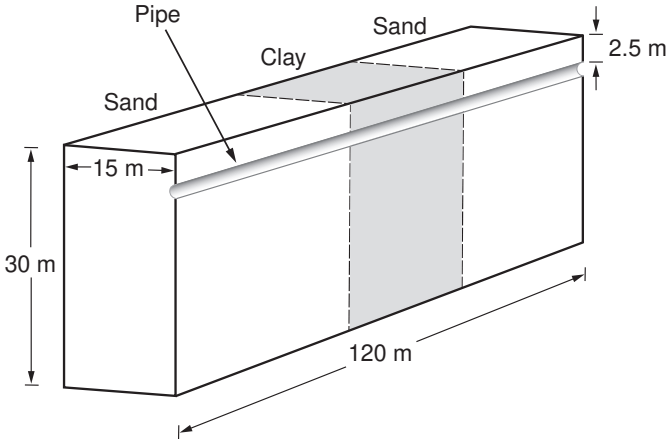


Figure 1: Pipeline buried in permafrost (half-symmetry model).

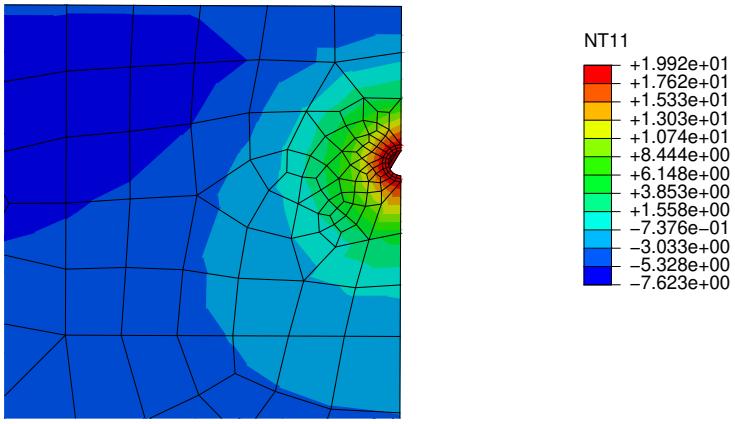


Figure 2: Thaw bulb around the relatively warm pipe after one year.

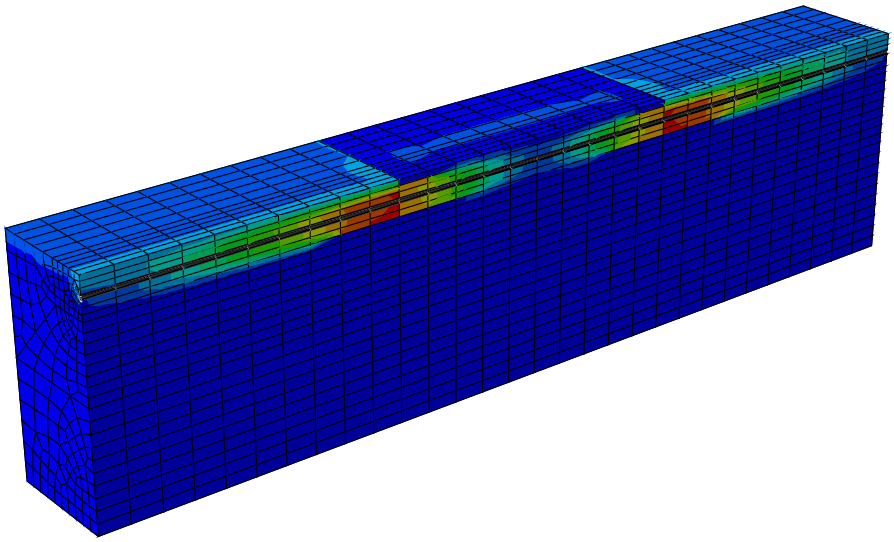


Figure 3: The meshed buried pipeline model showing the contour of fluid velocities. The relatively impermeable clay region exhibits low fluid velocities (blue-colored contour in top center section).

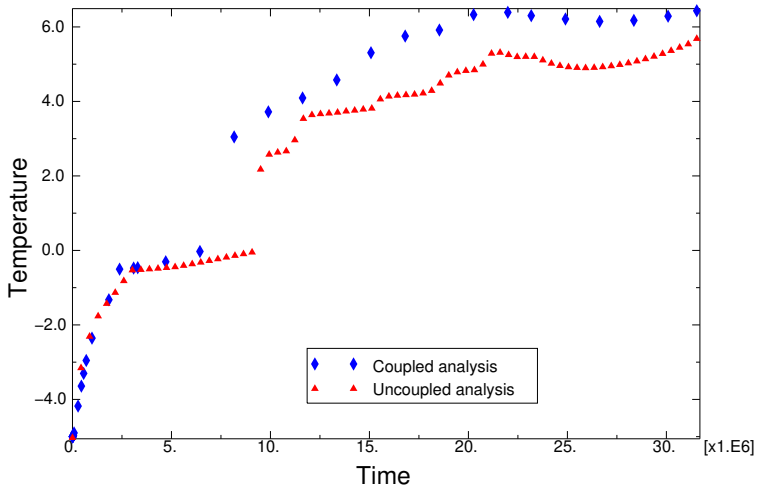


Figure 4: Temperature history in the soil near the pipe.

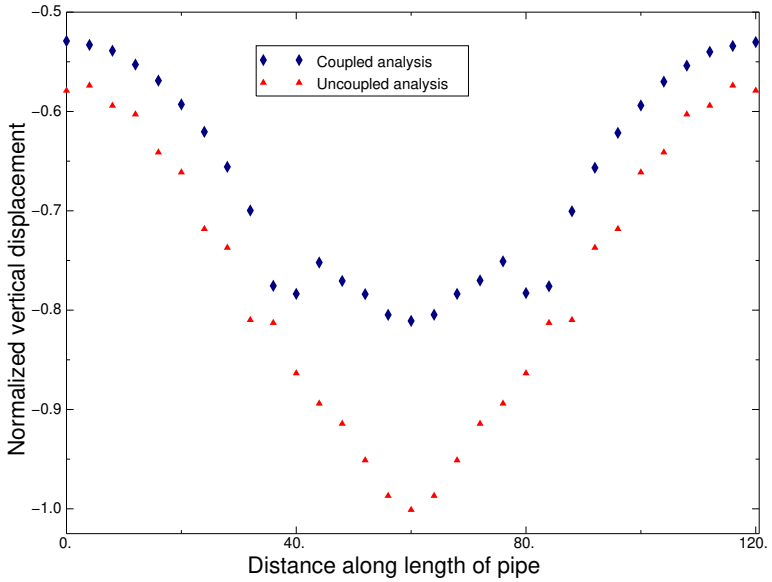


Figure 5: Normalized vertical displacement along the length of the pipe.

Acoustic and Shock and Abaqus/Aqua Analyses

In this section:

- *Acoustic and shock analyses*
- *Abaqus/Aqua analyses*

Acoustic and shock analyses

In this section:

- *Fully and sequentially coupled acoustic-structural analysis of a muffler*
- *Coupled acoustic-structural analysis of a speaker*
- *Analysis of a speaker using Abaqus-Dymola co-simulation*
- *Response of a submerged cylinder to an underwater explosion shock wave*
- *Convergence studies for shock analyses using shell elements*
- *UNDEX analysis of a detailed submarine model*
- *Coupled acoustic-structural analysis of a pick-up truck*
- *Long-duration response of a submerged cylinder to an underwater explosion*
- *Deformation of a sandwich plate under CONWEP blast loading*

Fully and sequentially coupled acoustic-structural analysis of a muffler

Products: Abaqus/Standard Abaqus/Explicit

References:

- *Coupled Acoustic-Structural Analysis*
- *Node-Based Submodeling*

Overview

This example demonstrates the solution of the acoustic field in the vicinity of a muffler in air caused by the vibrations of the muffler shell.
Steady-state and transient dynamic computations are done using both the fully coupled and sequentially coupled acoustic-solid interaction procedures in Abaqus. In the fully coupled case the solid medium of the muffler is directly coupled to the enclosed and surrounding air in a single analysis. In the sequentially coupled case the muffler vibrations are considered to be independent of the loading effects of the surrounding air, while the acoustic vibrations of the surrounding air are forced by the motion of the muffler. This allows the muffler vibration and acoustic radiation problems to be solved in sequence, using the submodeling procedure in Abaqus. The results for the sequentially coupled model are verified by comparing them to the results from the fully coupled procedure.

Full modeling vs. submodeling in Abaqus

The fully coupled model includes the effect of the acoustic pressure in the surrounding air loading the muffler body during vibration of the system. When modeling the acoustics of metal structures in air, such as in this case, such acoustic pressure loading is often negligible in comparison with other forces in the structure. The submodeling capability can be used in this situation. The part of the interacting system that is unaffected by the other is treated as the "global" model, while the part whose solution depends strongly on the solution of the other is treated as the "submodel." In the case of an acoustic analysis, of course, this nomenclature refers to the hierarchy of the solutions, not the geometric sizes of the models.

When sequential coupling is physically appropriate, its use offers a performance advantage over a fully coupled solution. Two problems, each smaller than the fully coupled problem, are less computationally expensive. If the applicability of the sequentially coupled solution method is uncertain, the user should make characteristic test computations in the frequency range of interest. If these computations show little difference between the fully and sequentially coupled solutions, the less expensive sequentially coupled method can be used.

Geometry and model

The system considered here consists of a cylindrical muffler and the interacting air. The muffler is a simple tube 180 mm in diameter and 1 m in length, with inlet and outlet pipes 70 mm in diameter and 100 mm in length. The muffler structure is made from stainless steel sheeting, 0.75 mm in thickness. A porous packing material, which dampens the acoustic field, surrounds the inner pipe.

Although this problem is in essence axisymmetric, a narrow three-dimensional wedge (subtending an angle of 10°) of the coupled system is modeled because Abaqus has a limitation on the use of submodeling with axisymmetric shells. Appropriate boundary conditions are applied to the three-dimensional model so that the axisymmetric solution is captured. The meshes of the surrounding air, the exterior muffler shell, and the air inside the muffler are shown in *Figure 1*, *Figure 2*, and *Figure 3*, respectively.

The air inside the muffler is meshed with AC3D10 elements (second-order tetrahedra) in Abaqus/Standard and with AC3D4 elements in Abaqus/Explicit. The innermost column of fluid elements models the undamped air.

The adjacent annulus models the air in the region of the packing material. These two regions are highlighted in *Figure 3*, where the annulus is shown as the darker region. The effect of the packing material is modeled using the volumetric drag coefficient for the acoustic medium. The muffler is meshed with S4R shell elements.

The exterior fluid is shown in *Figure 1*. Its outer boundary is made up of spherical and cylindrical segments, on which spherical and cylindrical absorbing boundary conditions are imposed. The cylindrical and spherical absorbing boundary conditions can be combined in Abaqus, allowing the external mesh to conform to the geometry of the radiating object more closely. Combinations of different boundary condition types are most effective when the boundaries are continuous in slope as well as displacement. Second-order hexahedral acoustic elements (AC3D20) are used in Abaqus/Standard and reduced-integration acoustic brick elements (AC3D8R) are used in Abaqus/Explicit to fill in the volume of the exterior fluid region.

In Abaqus/Explicit the possibility of using acoustic infinite elements to model the effect of the exterior fluid is explored. The use of acoustic infinite elements removes the need of impedance-type absorbing boundary conditions on the outer boundary. Acoustic infinite elements are used in two different ways. In the first approach the mesh modeling the exterior fluid is replaced by a single row of AC3D8R elements, and acoustic infinite elements ACIN3D4 are defined on the outer boundary of this row. In the second approach ACIN3D4 elements are defined directly on the outer boundary of the muffler and tied to the muffler surface.

In the submodeling procedure performed in Abaqus/Standard the interface between the surrounding air and the muffler is meshed with 8-node acoustic interface elements (ASI8); in the Abaqus/Explicit submodeling analysis the tie constraint is used to define this coupling. The choice of mesh density (element size) is discussed in *Coupled Acoustic-Structural Analysis*. In both cases the inner boundary of the exterior air mesh conforms to the muffler shell and to rigid baffles, which isolate the exterior field from the exhaust and inlet noise. These baffle pipes are the same diameter as the inlet and exhaust pipes but are modeled simply by imposing no boundary condition on the acoustic elements in this region. This is equivalent to imposing the condition that the acceleration on this boundary is zero, which is correct for a rigid baffle.

In Abaqus/Standard we are most interested in performing a frequency sweep about the first resonant frequency of the fully coupled system. For problems involving air and metal structures, the structure usually dominates the behavior of the system. Therefore, an estimate of the first important resonance of the coupled system is found by performing a frequency sweep in the vicinity of the first eigenfrequency of the muffler shell, computed without any interaction with the interior or exterior air. This occurs at $f = 172$ Hz. Although the resonant frequencies of the fully coupled system do not coincide with the resonant frequencies of the muffler shell alone, they are close, especially at lower frequencies.

Using the Abaqus/Standard direct-solution steady-state dynamic procedure to search around 172 Hz, we find that the first resonant frequency for the fully coupled system occurs at approximately 180 Hz. A frequency sweep of both the fully coupled and the sequentially coupled models from 179.0 Hz to 181.0 Hz at 0.2 Hz increments is performed. A pressure wave of unit magnitude is applied to the muffler inlet at each frequency, and a plane wave absorbing boundary condition is applied at the muffler outlet.

A transient dynamic analysis is performed in Abaqus/Explicit over the period of time that corresponds to the first resonant frequency of 180 Hz found in Abaqus/Standard. The pressure boundary conditions applied at the muffler inlet have a sinusoidal variation over time to simulate the steady-state dynamic procedure performed in Abaqus/Standard. The absorbing boundary conditions are imposed in the same way as in the steady-state dynamic procedure.

The material properties for the air are a bulk modulus K_f of 0.142 MPa and a density ρ_f of 1.2 kg/m^3, yielding a characteristic sound speed of 344 m/s. The volumetric drag, γ, specified for the air in the packing material region is 1.2 N s/m. Volumetric drag values are considered “small” if they are small compared to $2\pi\rho_f f$, a condition satisfied by $\gamma = 1.2$ N s/m for the frequency range of interest. The muffler is made of stainless steel with Young's modulus E of 190 GPa, Poisson's ratio ν of 0.3, and density ρ_s of 7920 kg/m^3.

Material properties affect the mesh parameters appropriate for wave problems. The characteristic wavelength of air at $f =$ 180 Hz, $\Omega = 2\pi \times 180 \approx$ 1131 rad/sec, is $\lambda_a = \sqrt{\frac{K_f}{\rho_f f^2}} \approx$ 1.91 m, which is long compared to the overall system geometry. The internodal spacing of roughly 40 mm used in the surrounding acoustic mesh and 30 mm in the interior acoustic mesh is adequate for this frequency. The acoustic wavelength must also be considered in selecting the overall size of the exterior domain. Accuracy of the solution requires placement of the radiating boundary at least one-quarter wavelength from the acoustic sources; in this problem a standoff distance of approximately 700 mm is selected. The characteristic flexural wavelength λ_p of the steel plating can be computed using the thickness h and the formula $\lambda_p = \frac{2\pi}{\sqrt{\Omega}} \left(\frac{Eh^2}{12\rho_s(1-\nu^2)} \right)^{1/4} \approx$ 203 mm. The discretization requirements of the finite element method in wave problems require at least six nodes per wavelength; here, we use an internodal distance of approximately 30 mm for the shells.

The fully coupled model consists of all three meshes shown in *Figure 1*, *Figure 2*, and *Figure 3* constrained at their abutting surfaces using a tie constraint.

The sequentially coupled analysis is performed in two jobs. The “global” model job consists of the meshes shown in *Figure 2* and *Figure 3*. The shell displacements, and displacement phases in Abaqus/Standard, are saved from this analysis and drive the second “submodel” analysis through the use of a submodel boundary condition. In Abaqus/Standard the second model consists of the exterior air mesh (*Figure 1*) used in the fully coupled case, with ASI8 elements placed on the boundary that abuts the shell surface. These elements convert the displacements from the “global” analysis to the appropriate boundary conditions for acoustic elements. In this analysis the ASI8 elements conform to the acoustic submodel mesh but not to the shell mesh of the global model. The nodes of the ASI8 elements are placed in a node set, specified in the model data of the submodel. The global elements used to drive the submodel must be specified to ensure that only the displacements of the ASI8 elements are driven by the shell elements. If this global element set is not specified, Abaqus may attempt to drive the acoustic pressure of the ASI8 elements by the interior acoustic elements, since those elements share the shell nodes in the “global” model. In Abaqus/Explicit the tie constraint is used in both the global and submodel analyses to couple the muffler structure with the surrounding acoustic medium.

Results and discussion

It is good practice to check the absorbing boundary conditions used on a particular mesh at a desired frequency by analyzing only the exterior fluid mesh with some test forcing on the boundary where acoustic excitations are expected. If the forcing is at a single point, the pressure phase angles should show a pattern of concentric circles, minimally distorted by the radiating boundary. While not a rigorous numerical test, such a result usually coincides with a properly offset radiating boundary. As shown in *Figure 4*, this criterion is met by the mesh used in this analysis.

Figure 5 is a plot of the radial displacement of the muffler inlet as a function of frequency for both the fully coupled and the global models. The resonant peak for the fully coupled model at 179.9 Hz is clearly illustrated. In contrast, the resonant peak for the “global” model (without the acoustic medium) occurs at approximately 180.0 Hz. The difference in the two peaks can be accounted for by the fact that the exterior air on the fully coupled model adds a small amount of damping due to radiation as well as mass to the system, which results in a lower natural frequency, as well as a slightly lower peak response. It is clear from *Figure 5* that for the frequency range of interest the coupling between the exterior air and the muffler is most important at 179.9 Hz.

Figure 6 and *Figure 7* contain contour plots of the pressure magnitude and phase for the muffler interior at 181.0 Hz for both the “global” model and the fully coupled model. In both cases the results indicate that the modeling assumptions of the sequentially coupled analysis appear to be valid for the solutions in the muffler interior.

Contour plots of the pressure magnitude and phase for the muffler exterior at 181.0 Hz are shown in *Figure 8* and *Figure 9*. The resulting pressure magnitude in the exterior air is small in both cases. The differences in the pressure amplitudes and phase as computed by the two analyses are not considered to be significant. Two factors

that account for the small differences are the different modeling methods (fully coupled vs. sequentially coupled) and the different techniques used to couple the muffler to the exterior air (tie constraints vs. acoustic interface elements).

Figure 10 and *Figure 11* contain contour plots of the pressure magnitude and phase for the muffler interior at 179.9 Hz for both the "global" model and the fully coupled model. It is clear that at 181.0 Hz, the modeling assumptions of the sequentially coupled analysis are less valid than they are at 179.9 Hz for the solutions in the muffler interior. This result is anticipated by *Figure 5*. However, the solutions are still reasonably close to one another, indicating that the sequentially coupled analysis is still a reasonable approximation for this system even at a resonant peak.

Contour plots of the pressure magnitude and phase for the muffler exterior at 179.9 Hz are shown in *Figure 12* and *Figure 13*. Again, the resulting pressure magnitude in the exterior air is small in both cases. The differences in the pressure amplitudes and phase as computed by the two analyses are less evident in the exterior than they were in the interior.

The pressure magnitudes along the muffler centerline at both 179.9 Hz and 181.0 Hz are shown in decibels in *Figure 14*. The reference pressure is chosen as one unit for convenience. The plot illustrates the variation of acoustic pressure in the muffler near resonance.

Table 1 shows comparative solution times and memory requirements for the fully and sequentially coupled analyses. The total computational time for the sequentially coupled case is lower, and the peak memory requirements are significantly lower. These differences will be greater for larger models. Optimal speed increases occur when global and submodels have nearly equal numbers of degrees of freedom. Here, solving the fully coupled system does not impose as much of a speed penalty as might be expected, because the sparse solver used by Abaqus exploits the extreme sparsity of the fluid-solid coupling term. When the number of system nodes involving fluid-solid coupling is a large percentage of the total number of nodes, the sparsity of the coupling term decreases, favoring the sequentially coupled procedure. Sequentially coupled analyses are even more advantageous than fully coupled analyses when many different submodels need to be analyzed, driven by a single set of global results.

Abaqus issues a series of warning messages in this example, because the narrow wedge domain results in some three-dimensional acoustic elements with bad aspect ratios. These messages can be ignored in this study, since the solutions are essentially axisymmetric and the gradient of the solution in the circumferential direction is nearly zero. Moreover, elements with scalar degrees of freedom, such as the acoustic elements used in this example, are much less sensitive to geometric distortion than elements with vector degrees of freedom, such as continuum stress/displacement elements.

The results obtained in Abaqus/Explicit agree well with the Abaqus/Standard results. For the fully coupled analysis the pressure variation in time at the muffler outlet centerline is shown in *Figure 15* (for a clear comparison the Abaqus/Standard analysis is also performed as a transient simulation). The Abaqus/Explicit models using acoustic infinite elements give results that agree well with the results using the impedance-type absorbing boundary. In *Figure 15* we include the results for the test using acoustic infinite elements, where the mesh modeling the exterior fluid is replaced by a single row of AC3D8R elements and acoustic infinite elements ACIN3D4 are defined on the outer boundary of this row. For the Abaqus/Explicit submodeling analysis the inside air pressure in the global model and the outside air pressure of the submodel compare well with the air pressures obtained in these regions in the fully coupled problem.

Input files

Abaqus/Standard input files

muffler_full.inp	Three-dimensional fully coupled model.
muffler_globl.inp	Muffler and internal air global model.
muffler_innerair_freq.inp	Internal air eigenanalysis model, Lanczos.

muffler_innerair_freq_ams.inp	Internal air eigenanalysis model, AMS.
muffler_submo.inp	Exterior air submodel.
muffler_shell_nodes.inp	Nodal coordinates for muffler shell mesh.
muffler_intair_nodes.inp	Nodal coordinates for interior air mesh.
muffler_extair_nodes.inp	Nodal coordinates for surrounding air mesh.
muffler_shell_elem.inp	Element definitions for muffler shell mesh.
muffler_intair_elem.inp	Element definitions for interior air mesh.
muffler_extair_elem.inp	Element definitions for surrounding air mesh.
muffler_freq.inp	Natural frequency extraction for shell mesh.
muffler_bctest.inp	Radiating boundary condition test.

Abaqus/Explicit input files

muffler_full_xpl.inp	Three-dimensional fully coupled transient analysis.
muffler_full_acoinfxpl.inp	Three-dimensional fully coupled transient analysis using acoustic infinite elements.
muffler_full_acoinftiexpl.inp	Three-dimensional fully coupled transient analysis using acoustic infinite elements tied to the muffler outer surface.
muffler_global_xpl.inp	Muffler and internal air global model, transient analysis.
muffler_submodel_xpl.inp	Muffler and exterior air submodel, transient analysis.
muffler_shell_nodes.inp	Nodal coordinates for muffler shell mesh.
muffler_shell_elem.inp	Element definitions for muffler shell mesh.
muffler_intair_nodes_xpl.inp	Nodal coordinates for interior air mesh.
muffler_intair_elem_xpl.inp	Element definitions for interior air mesh.
muffler_extair_nodes_xpl.inp	Nodal coordinates for surrounding air mesh.
muffler_extair_elem_xpl.inp	Element definitions for surrounding air mesh.
muffler_extair_elem_ainxpl.inp	Element definitions for surrounding air mesh for model using acoustic infinite elements.

Tables

Table 1: Comparison of relative CPU times (normalized with respect to the CPU time for the sequential analysis) and approximate problem size for the frequency sweep excluding preprocessing.

	Memory	DOF	Relative CPU Time
Global model	10 Mb	10030	0.325
Submodel	15 Mb	19030	0.675
Fully coupled model	29 Mb	29060	1.086
Sequential analysis			1.000

Figures

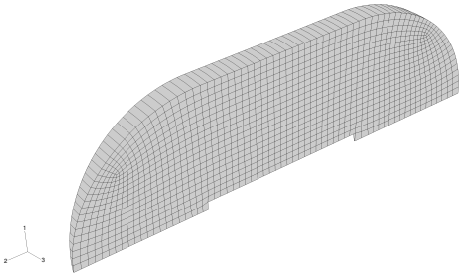


Figure 1: Mesh of surrounding air.

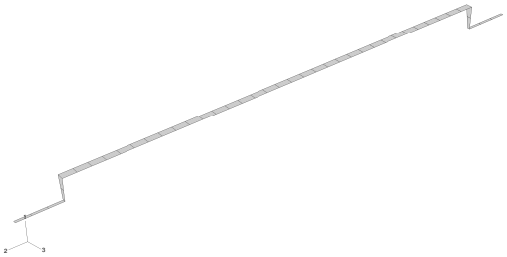


Figure 2: Mesh of muffler.

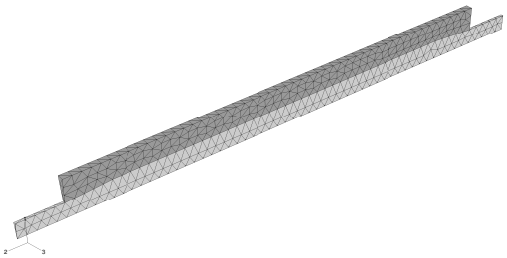


Figure 3: Mesh of interior air.

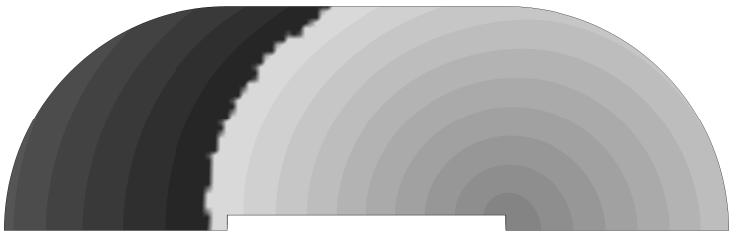


Figure 4: Radiating boundary condition test at 165 Hz.

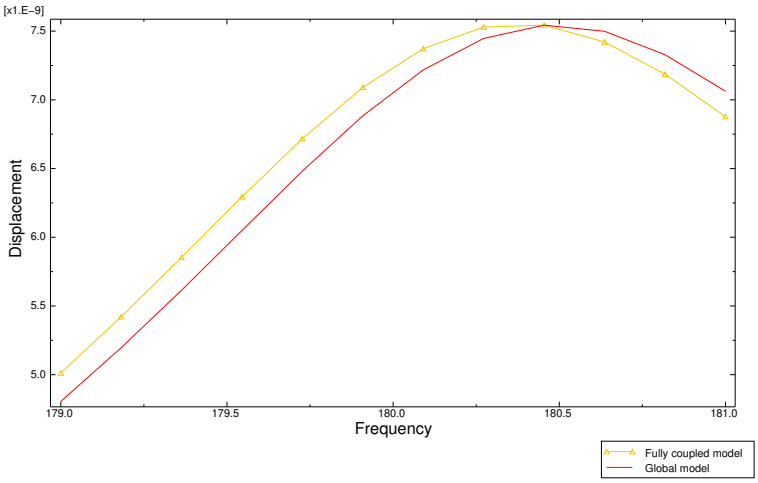


Figure 5: Radial displacement of the muffler inlet as a function of frequency.

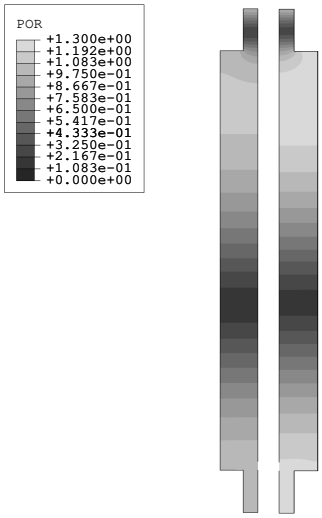


Figure 6: Muffler internal pressure magnitudes at 181.0 Hz, muffler inlet at top: fully coupled solution on left, "global" model (without the exterior acoustic medium) on right.

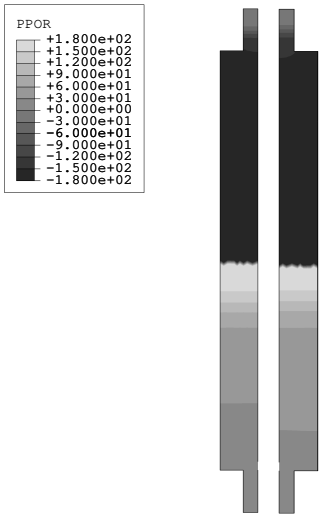


Figure 7: Muffler internal pressure phase at 181.0 Hz, muffler inlet at top: fully coupled solution on left, "global" model (without the exterior acoustic medium) on right.

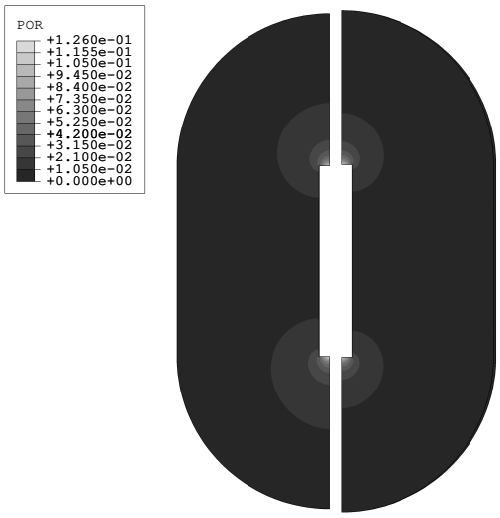


Figure 8: Muffler external pressure magnitudes at 181.0 Hz, muffler inlet at top: fully coupled solution on left, “submodel” on right.

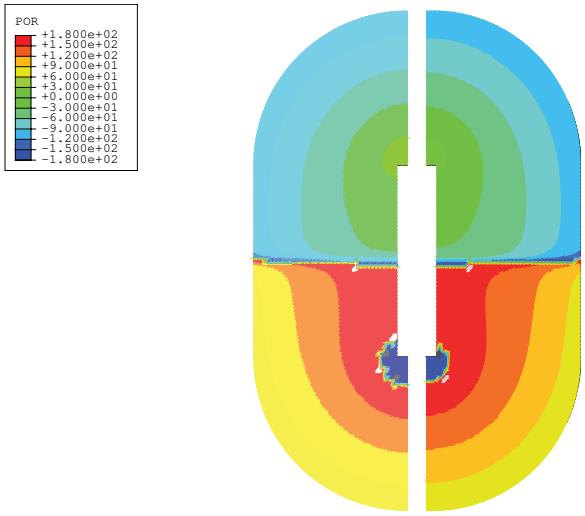


Figure 9: Muffler external pressure phase at 181.0 Hz, muffler inlet at top: fully coupled solution on left, “submodel” on right.

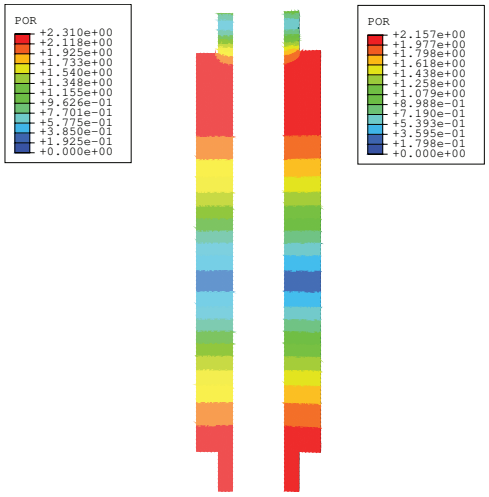


Figure 10: Muffler internal pressure magnitudes at 179.9 Hz, muffler inlet at top: fully coupled solution on left, "global" model (without the exterior acoustic medium) on right.

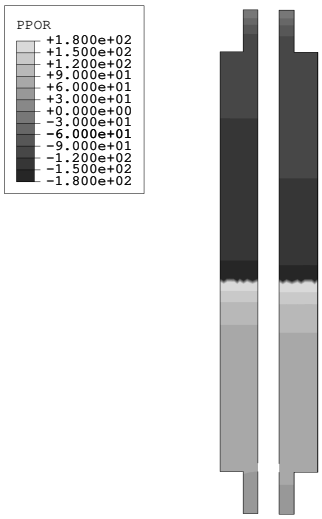


Figure 11: Muffler internal pressure phase at 179.9 Hz, muffler inlet at top: fully coupled solution on left, "global" model (without the exterior acoustic medium) on right.

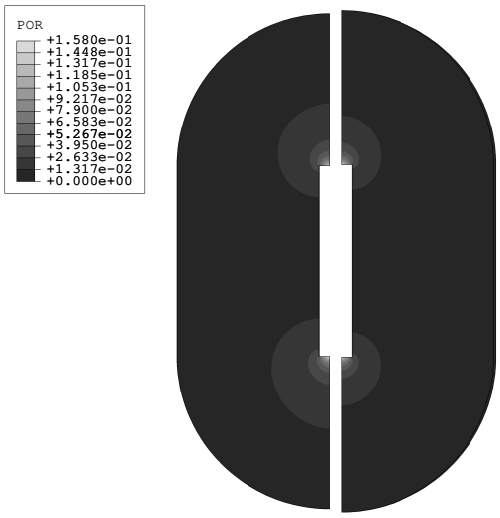


Figure 12: Muffler external pressure magnitudes at 179.9 Hz, muffler inlet at top: fully coupled solution on left, “submodel” on right.

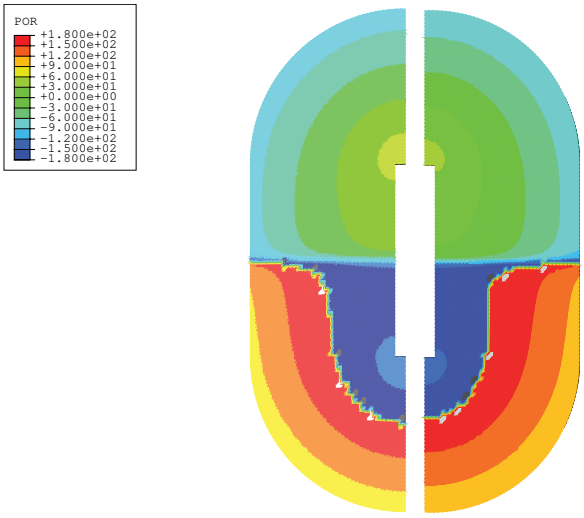


Figure 13: Muffler external pressure phase at 179.9 Hz, muffler inlet at top: fully coupled solution on left, “submodel” on right.

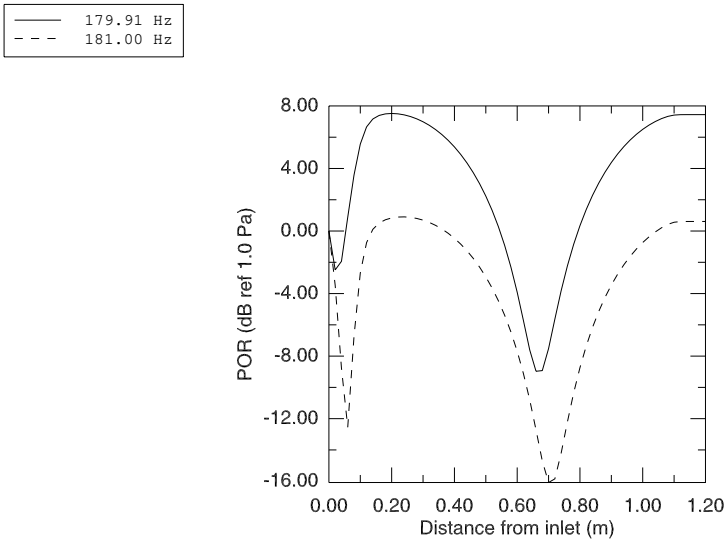


Figure 14: Muffler internal pressure magnitude at 179.9 and 181.0 Hz: dB along muffler centerline.

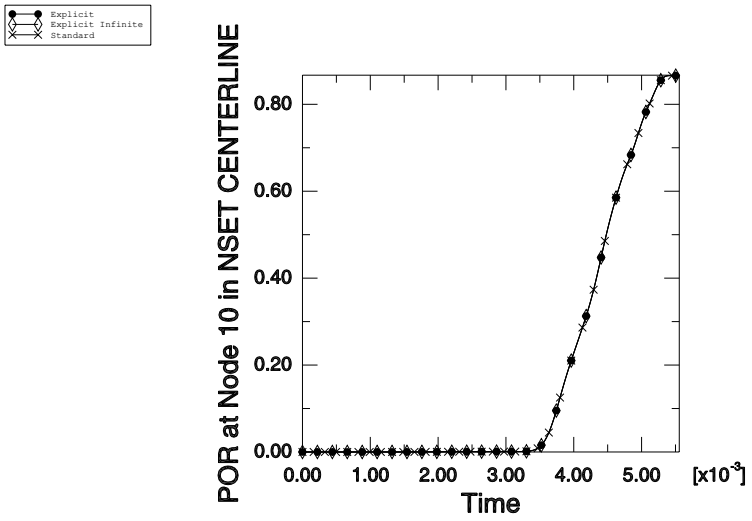


Figure 15: Internal pressure at the muffler outlet for the transient analysis.

Coupled acoustic-structural analysis of a speaker

Products: Abaqus/Standard

References:

- *Coupled acoustic-structural medium analysis*

Overview

This example illustrates the effect of coupling between a structure and an acoustic medium.
Coupling between a structure and an acoustic medium can produce problems when the solid-fluid interaction is fundamental to the overall vibrational behavior of the body or of the acoustic fluid. Typical examples of such problems include loudspeaker enclosures, fluid-filled tanks, muffler systems, and vehicle cabin enclosures.

Geometry and model

The model is shown in *Figure 1*. The system considered here consists of a speaker box, a speaker cone, and the interacting interior. To simplify the problem, the effect of interacting air at the outside of the speaker box is neglected. The width, depth, and height of the speaker box are 0.5 m, 0.4 m, and 0.6 m, respectively. Its thickness is 0.005 m. The speaker box is made of wood with a Young's modulus, E, of 11.6 GPa; a Poisson's ratio, ν, of 0.3; and a density, ρ_s, of 562 kg/m^3. At the center of the front speaker box, there is a cone-shaped speaker 0.345 m in diameter, 0.04 m in height, and 0.001 m in thickness. No mass or impedance of the speaker is considered. The speaker is made of polyethylene with a Young's modulus, E, of 3.4 GPa; a Poisson's ratio, ν, of 0.3; and a density, ρ_s, of 450 kg/m^3. The air has a density, ρ_f, of 1.11 kg/m^3 and a bulk modulus, K_f, of 0.134 MPa. Volumetric drag of the air is assumed to have a negligible effect in this problem, so it is ignored in this analysis.

First-order hexahedral acoustic elements (AC3D8) and first-order acoustic triangular prism elements (AC3D6) are used to fill in the volume of the interior air region. The speaker box and speaker are meshed with S4R and S3R elements, respectively. No mesh convergence study has been done since the example is intended only as an illustration. The choice of mesh density (element size) is discussed in *Coupled Acoustic-Structural Analysis*.

The surface-based tie constraint is used to couple the structure with the air inside. Surfaces are defined at the inside of the speaker box and the speaker cone and at the free surface of the air. To constrain the structure, four corner points of the bottom panel are simply supported.

A substructure analysis is performed as well. The entire speaker model is turned into a dynamic coupled structural-acoustic substructure using all the eigenmodes up to 800 Hz. A substructure load case is generated to be used in the forced response analyses.

Results and discussion

The results for each of the analyses are discussed below.

Natural frequency analyses

If the eigenvalues of the structure alone or the acoustic medium alone are not in the range of interest, it is not necessary to consider the whole system simultaneously. Thus, it is recommended that you understand the modal characteristics of each part separately before analyzing the whole system. The eigenfrequency extraction procedure takes acoustic-structural coupling effects into account by default if an acoustic medium and a structure are joined using a tie constraint. To ignore this effect, you can compute uncoupled mode sets for acoustic and structural/solid

regions using the eigenfrequency extraction procedure or perform an eigenvalue extraction analysis that uses the SIM architecture.

In the SIM-based analysis, uncoupled modes are extracted and the structural-acoustic coupling is imposed in the subsequent modal steady-state dynamic procedure. To compensate for the approximation introduced by using uncoupled modes, the maximum frequency of interest for acoustic modes is increased by setting the acoustic range factor to a value greater than 1.0. In addition, residual modes are requested to reduce the modal truncation error. To request residual modes in the Lanczos eigensolver, you must first compute the static perturbation response of the load that will be applied in the subsequent mode-based analysis by specifying that load in a static perturbation step that immediately precedes the frequency extraction step. The residual modes are computed in both the coupled and uncoupled eigenvalue extraction analyses.

The results of the natural frequency analysis of the uncoupled system are summarized in *Table 1*. The natural frequencies for both the structure and the air span the same range in this example, which shows that the two parts can affect each other. Thus, the coupled approach should be adopted to understand the characteristics of the whole speaker system in this example.

Table 2 shows the results of the natural frequency analysis of the coupled system. Due to the coupling effect, the eigenfrequencies shift mode by mode. Each mode shape is also more complex than those in the uncoupled case, so that each mode has nonzero components on both the structural and acoustic parts. The substructure analysis based on coupled modes yields eigenfrequencies identical to those from the analysis without substructures.

Coupled forced response analyses

The response of the system is obtained by using mode-based and direct-solution steady-state dynamic analyses. The eigenvalue extraction analysis must be performed prior to the modal analysis. Two types of mode-based steady-state dynamic analyses are performed: an analysis that is based on coupled modes and a SIM-based analysis that utilizes uncoupled modes. The analyses are performed as frequency sweeps from 200 to 400 Hz. The system is excited by a concentrated force of 1.0 N at the center point of the speaker cone. The results for the coupled frequency response using each of the three methods are shown in *Figure 2*. This figure illustrates the acoustic pressure near the center point of the speaker cone, plotted as a function of frequency. The results for the mode-based steady-state dynamic analyses based on coupled and uncoupled modes show good agreement with the direct-solution steady-state dynamic results. Because the coupling between the structure of the speaker and the air inside is relatively strong, additional acoustic eigenmodes are extracted for the modal analysis based on uncoupled modes to improve accuracy. The substructure analysis produces results that are virtually identical to the results from the equivalent analysis without substructures.

Input files

speaker_direct.inp	Direct-solution steady-state dynamic analysis.
speaker_coupled.inp	Natural frequency extraction of coupled eigenmodes and mode-based steady-state dynamic analysis using coupled modes.
speaker_sim.inp	Natural frequency extraction of uncoupled eigenmodes and mode-based steady-state dynamic analysis using uncoupled modes.
speaker_gen.inp	Substructure generation for the coupled structural-acoustic model and steady-state dynamic solution without substructure.
speaker_use.inp	Direct-solution and mode-based steady-state dynamic analyses using the substructure.
speaker_ams.inp	Mode-based steady-state dynamic analysis using uncoupled eigenmodes extracted with the AMS eigensolver.
speaker_model.inp	Model definition.

Tables

Table 1: Uncoupled frequency analysis. Modes are arranged in order of increasing frequency starting from 201 Hz.

Mode	Frequency [Hz]	Description
1	201	Structure
2	211	Structure
3	212	Structure
4	232	Structure
5	247	Structure
6	263	Structure
7	283	Air
8	283	Structure
9	313	Structure
10	314	Structure
11	339	Air
12	352	Structure
13	361	Structure
14	382	Structure

Table 2: Coupled frequency analysis. Modes are arranged in order of increasing frequency starting from 204 Hz.

Mode	Frequency [Hz]	Description
1	204	Coupled mode
2	227	Coupled mode
3	228	Coupled mode
4	235	Coupled mode
5	245	Coupled mode
6	269	Coupled mode
7	292	Coupled mode
8	307	Coupled mode
9	322	Coupled mode
10	348	Coupled mode
11	352	Coupled mode
12	367	Coupled mode
13	370	Coupled mode
14	399	Coupled mode

Figures

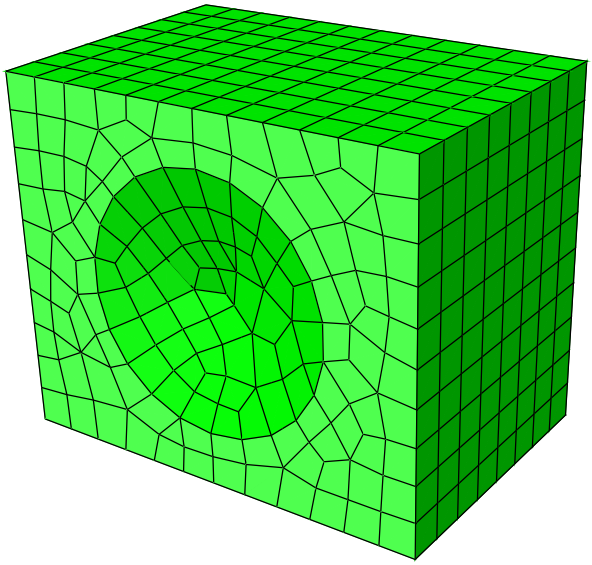


Figure 1: Three-dimensional model of speaker system.

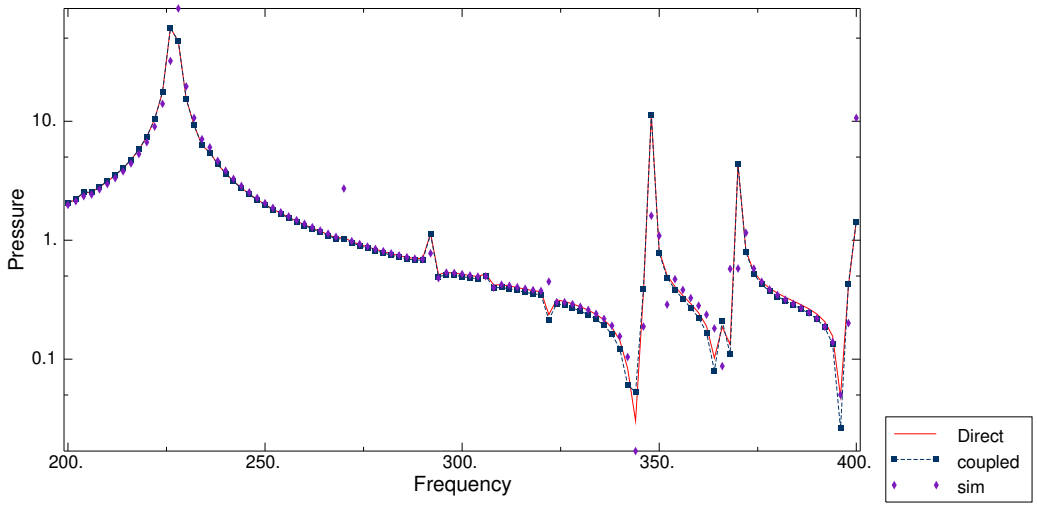


Figure 2: Acoustic pressure near the center of speaker cone.

Analysis of a speaker using Abaqus-Dymola co-simulation

Products: Abaqus/Explicit

References:

- *System-Level Modeling between Logical and Physical Interactions*

Overview

This example illustrates the use of the co-simulation technique to couple system-level logical models and functional-level models in Dymola with a physical model in Abaqus/Explicit.

Application description

Mobile devices like tablets and smart phones are increasingly becoming an important part of our lifestyle. One of the critical features that determines the overall quality of such a device is the audio quality of the speaker. The most commonly used speaker in mobile devices is based on the principle of moving coil transduction, as described by Jackman et al. (2009). An electromagnetic field applied on the voice coil generates a mechanical driving force (known as the Lorentz force) that generates the sound by imparting motion to the diaphragm. There are several key design challenges in the selection of the transducer and its placement. Some of the important issues to be addressed in the audio system design are discussed below.

Audio system design

The mobility and lightness of the device are major considerations, and space is at a premium. Providing many features in a mobile device requires numerous components, and packaging plays a critical role as these components compete for the limited available space. The smaller the audio speaker, the poorer its bass response. The enclosure behind the diaphragm in the speaker (referred to as the “back volume”) has a major influence on the speaker performance; it affects the audio quality by introducing cavity resonances at high frequencies and by reducing the compliance at low frequencies. In other words, the smaller the back volume, the lower the quality of the low frequency (bass) sound output. The designer must take measures to maximize the back volume, conforming to the packaging needs of the device in addition to providing an excellent bass audio quality.

The audio quality is reduced due to the inherent nonlinearities such as the total harmonic distortion. Even for microspeakers with very low diaphragm motions or excursions, the nonlinearities due to the mechanical resistance at the restraints (in addition to acoustic losses in ports) tend to be dominant. The nonlinearities are guided not so much by the displacement but by the velocity of the diaphragm excursion (Klippel and Knobloch, 2013). Hence, there is a need to model the nonlinearities that can be appropriately modeled in the time domain to reproduce the physically accurate solution. Simulating the audio response as a harmonic analysis in the frequency domain may provide inaccurate results, especially at higher levels of nonlinearity.

For transducer design it is important to have a flat frequency response while making sure that the sound output is as large as possible without distortions. The first resonant mode (along with the damping) sets the maximum for the sound output. The flat frequency response (at least in the mid-frequency range; e.g., 1000–6000 Hz) is ensured by shifting the second resonance peak as far away as possible from the first resonance peak.

Interaction through the co-simulation technique

This example discusses a methodology that simulates the diaphragm excursion (and the subsequent acoustic response) due to electromagnetic excitation as a transient problem. The electromagnetic circuit of the speaker is assumed to have no direct contribution to the structural response of the system and, hence, is modeled

independently as a reduced-order logical system in Dymola. The structural-acoustic response of the diaphragm in the speaker assembly is modeled with its full three-dimensional finite element representation in Abaqus/Explicit. The Abaqus model and the Dymola model interact with each other through the co-simulation technique in the time domain.

Abaqus model

The Abaqus model of the speaker is shown in *Figure 1*. Features such as very short edges or small faces, although important for machining and packaging of a component, have almost no bearing on the mechanics of the problem. Including such features in the numerical analysis could result in a very fine mesh density, leading to increased computation time. Such minor geometric details are excluded by combining the feature with an adjacent larger feature. Nodes and elements are created to conform to the original geometry. The diaphragm occupies the region in between the front volume and the speaker's back volumes, as shown in *Figure 1*. Symmetric notches and the racetrack-like separation between the center and the annulus region in the diaphragm (shown in *Figure 2*) are designed to achieve the flat frequency response and the first resonant peak, respectively. The other speaker components that hold the diaphragm in place are considered rigid and are not modeled. The radius of the hemispherical acoustic domain corresponds to one-third of the largest acoustic wavelength. Additional details on the modeling of the speaker are described in Jackman et al. (2009).

Dymola model

The electromagnetic component of the speaker, which functions primarily in the low frequency range, can be idealized with a lumped-parameter model in Dymola, as shown in *Figure 3*. The advantage of using a lumped-parameter model is twofold; it simplifies the modeling and improves the performance dramatically. The electromagnetic part of the model is assumed to have a linear response to the electromagnetic excitation; hence, it is modeled with a resistor and an inductor. The current sensor output from Dymola is used as an actuator in Abaqus, and the velocity output of the sensor from Abaqus is used as an actuator in calculating the back electromotive force (EMF) in the system-level functional model.

Abaqus modeling approaches and simulation techniques

Two cases are studied. The first case analyzes loading with a white noise signal to check the drop in the signal amplitude over the audible frequency range. The second case analyzes loading with a fixed frequency signal to check the fidelity of the response of the speaker.

Summary of analysis cases

Case 1	White noise analysis.
Case 2	Response fidelity analysis.

The sections that follow discuss the analysis considerations that are applicable for both cases.

Analysis Techniques

Abaqus-Dymola co-simulation provides a convenient way to couple system-level logical and functional-level models in Dymola with a physical model in Abaqus. The Abaqus user subroutines *UAMP* and *VUAMP* can be used to design such system-level models interacting with physical models, but the user is required to manually code the user subroutine. Dymola provides a host of libraries that have components belonging to different engineering domains (such as fluid, thermal, and electromagnetic) as well as mathematical operators (such as controllers and Boolean operators) that can be easily dragged and dropped as icons into a user interface to build the system-level model. Co-simulation with Abaqus involves passing the state of the physical system in Abaqus as sensor data to Real Input interfaces in Dymola while reading the Real Output interfaces from Dymola into Abaqus as actuators. For more details about co-simulation, refer to *System-Level Modeling between Logical and Physical Interactions*.

Mesh design

The diaphragm is modeled with reduced-integration conventional shell elements (S3R and S4R), and its rim is meshed with first-order reduced-integration hexahedral continuum elements (C3D8R). Modeling the diaphragm with solid elements instead of shell elements allows the acoustic domain to be cut away more easily. The acoustic domain, both in the back volume and the volume exterior to the diaphragm, is modeled with three-dimensional continuum acoustic linear tetrahedral elements (AC3D4). The mesh of the acoustic domain is nonuniform, with the largest element size being less than one-eighth of the shortest acoustic wavelength.

Materials

The units used in the Abaqus model are mm-tonnes-sec, and SI units are used in Dymola. The diaphragm material in Abaqus is polyimide (Young's modulus=3677; density=1.4×10^{-9}; structural damping=1.5); the voice coil is made of steel (Young's modulus=110,000; Poisson's ratio=0.3; density=3.3×10^{-9}); while the air is modeled as the acoustic medium (bulk modulus= 0.142; density=1.2×10^{-12}). The electromagnetic circuit in Dymola has a resistor of 5 ohms, an inductor of 5×10^{-5} H, and a coupling factor of 3.14159 N/A.

Boundary conditions

Boundary conditions are not applied to the acoustic regions that are in contact with the surfaces of the speaker components to enforce the rigid assumption that an Abaqus acoustic domain without any boundary condition is assumed to be a rigid termination. The surface impedance for the nonreflecting spherical boundary of the acoustic medium is defined as 550.

Constraints

The structural and acoustic media are coupled through tie constraints. The diaphragm's rim is rendered rigid with its reference point at the center of the rim.

Output requests

The component of the translational velocity of the reference node of the rigid rim of the diaphragm along the direction perpendicular to the hemispherical flat surface of the acoustic medium is declared as a sensor. This sensor output from the diaphragm is passed into Dymola as an input signal. The pressure at a certain location in the front volume acoustic medium is also stored as history output.

Run procedure

The Abaqus model can be run on any supported platform (Windows/Linux). For more details on how to submit an Abaqus-Dymola co-simulation job, refer to *System-Level Modeling between Logical and Physical Interactions*.

You can use the **abaqus fetch** utility to obtain the DymolaFMU files for the examples using the following commands:

```
abaqus fetch job=EMsystem_cosim_white_noise.fmu
```

```
abaqus fetch job=EMsystem_cosim_frequency.fmu
```

Case 1: White noise analysis

This case analyzes loading with a white noise signal to check the drop in the signal amplitude over the audible frequency range.

Analysis Techniques

A concentrated force with amplitude as white noise is applied on a rigidly fixed dummy node, and a sensor of this concentrated force at this node is passed from Abaqus into Dymola as an input value.

Loads

A white noise signal voltage of unit magnitude is applied through a sensor on a dummy node in Abaqus, as mentioned above.

Solution controls

This analysis is run for a duration of 0.03 s, thereby setting a threshold frequency of 33 Hz as the lower limit for the frequency range of the audio signal to obtain an accurate response from the speaker. The models's stable time increment of ~10^{-8} determines the Nyquist frequency of 1×10^{6} Hz as the higher limit for the frequency range of the audio signal to obtain an accurate response.

Output requests

An additional sensor for a concentrated force on a dummy node along the direction of loading is defined for transferring the white noise to Dymola from Abaqus, as mentioned above.

Case 2: Response fidelity analysis

This case analyzes loading with a fixed frequency signal to check the fidelity of the response of the speaker.

Loads

A signal voltage of 2 mV with a frequency of 2000 Hz is applied. In Abaqus the reference node of the rigid rim of the diaphragm is actuated by the Lorentz force that was computed in Dymola.

Solution controls

This analysis with a signal frequency of 2000 Hz must be run for at least 0.005 s to obtain a steady response of the speaker for a few wavelengths of the signal.

Discussion of results and comparison of cases

For the response fidelity analysis, the acoustic signal is reciprocated in the diaphragm and in the air at almost 2000Hz, as shown in the velocity and pressure plots in *Figure 4* and *Figure 5*, respectively.

Files

Case 1: White noise analysis

speaker_white_noise.inp	Abaqus input file.
speaker_white_noise_config.xml	Co-simulation configuration file.
speaker_white_noise.mo	Dymola modelica file.
white_noise.inp	Include file containing white noise signal data.

Case 2: Response fidelity analysis

speaker_frequency.inp	Abaqus input file.

speaker_frequency_config.xml	Co-simulation configuration file.
speaker_frequency.mo	Dymola modelica file.

References

- Jackman, C., M. Zampino, D. Cadge, R. Dravida, V. Katiyar, and J. Lewis, “Estimating Acoustic Performance of a Cell Phone Speaker Using Abaqus,”SIMULIA Customer Conference, 2009.
- Klippel, W., and D. Knobloch, “Nonlinear Losses in Electro-Acoustical Transducers,” The Association of Loudspeaker Manufacturers & Acoustics International (ALMA), Winter Symposium, 2013.
- Radcliffe, C., and S. Gogate, “Velocity Feedback Compensation of Electromechanical Speakers for Acoustic Applications,” International Federation of Automatic Control, Triennial World Congress, 1996.

Figures

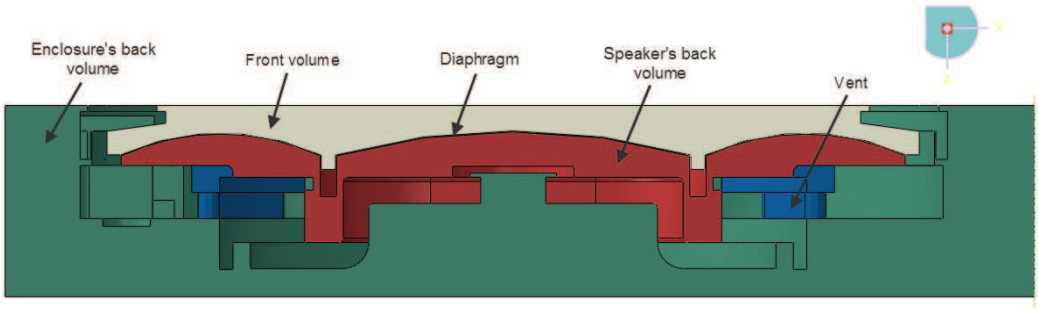


Figure 1: Abaqus model of the speaker.

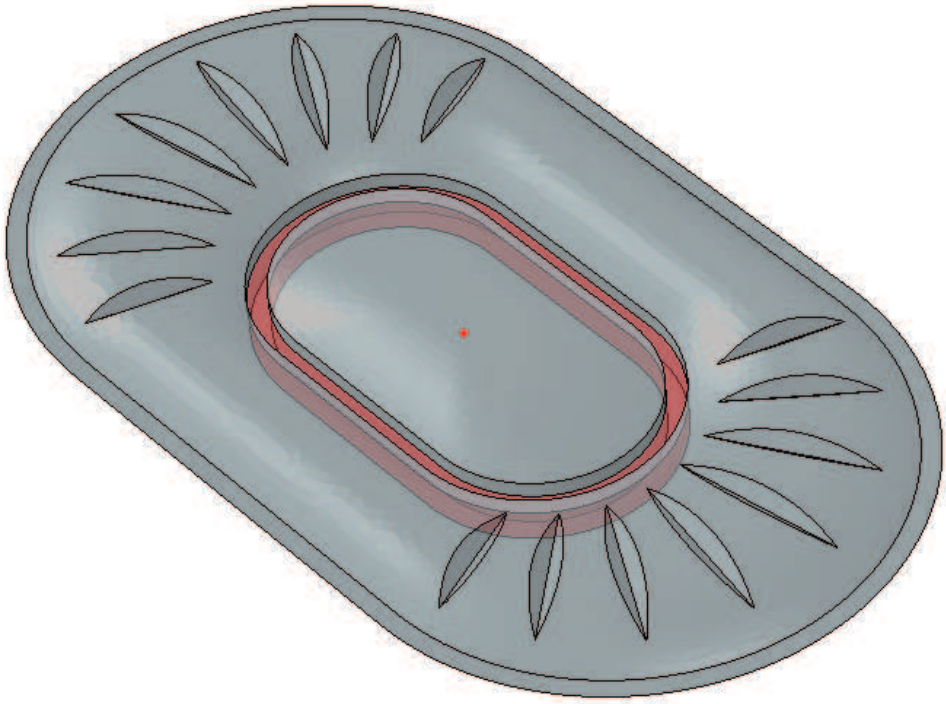


Figure 2: Diaphragm of the speaker in the Abaqus model. The highlighted red region is rendered rigid with its reference node at the center highlighted in red.

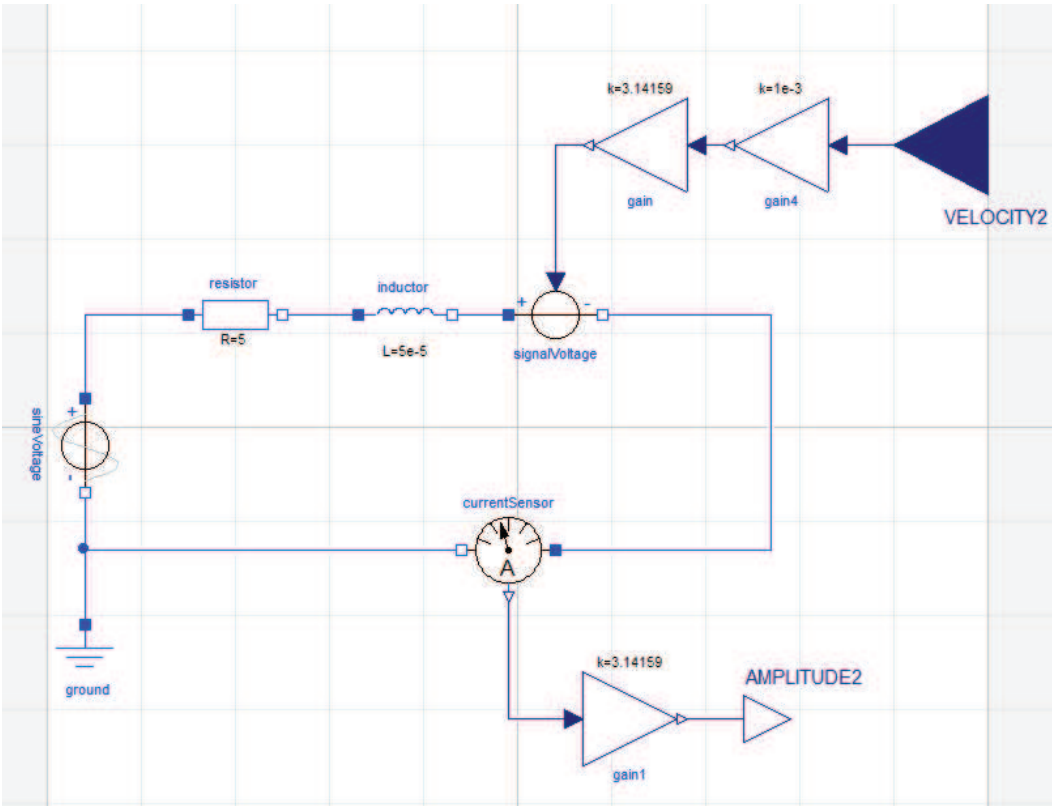


Figure 3: Dymola model showing the electromagnetic circuit of the speaker.

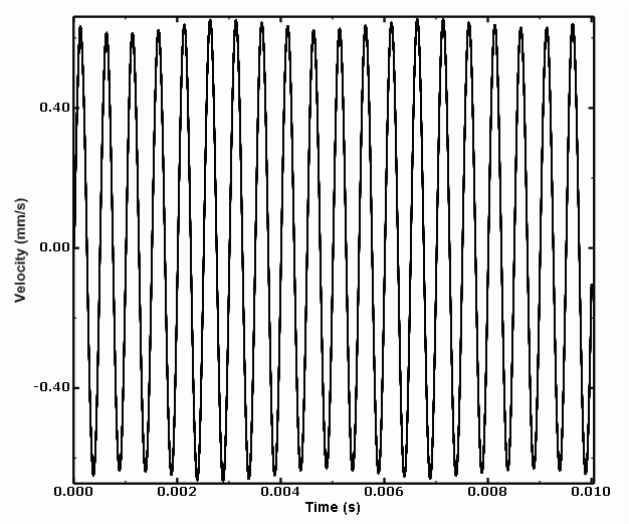


Figure 4: Response fidelity analysis: velocity of the reference node of the rigid part of the diaphragm.

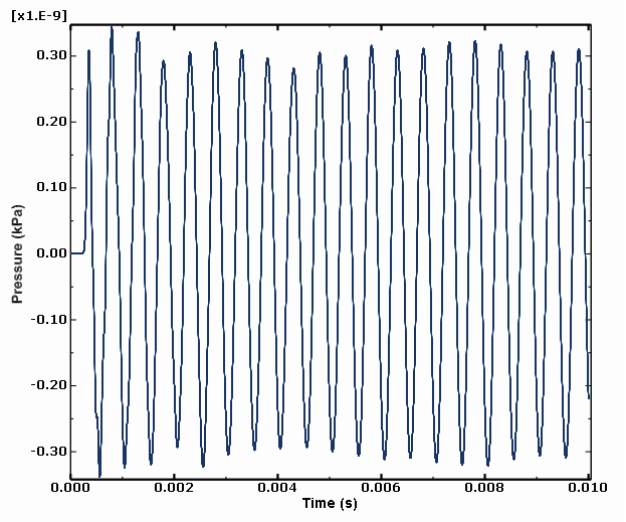


Figure 5: Response fidelity analysis: pressure at a point in air in front of the diaphragm.

Response of a submerged cylinder to an underwater explosion shock wave

Products: Abaqus/Explicit

Overview

This example demonstrates how Abaqus/Explicit can be used to predict the transient response of submerged structures that experience loading by an acoustic pressure shock wave resulting from an underwater explosion (UNDEX).

This class of problem is characterized by a strong coupling between the structural motions and acoustic pressures on the wetted interface between the external fluid and the structure. The structural response in a strongly coupled acoustic-structural system can be described as a combination of the following:

- Low-frequency response characterized by structural wavelengths that are significantly shorter than the associated acoustic wavelengths. The external fluid on the structure adds an effective mass to the structure on the wetted interface.
- High-frequency response characterized by structural wavelengths that are significantly longer than the associated acoustic wavelengths. The external fluid on the structure acts as a simple damping mechanism, where energy is transported away from the structure via acoustic radiation.
- Intermediate-frequency response characterized by structural wavelengths that are similar in length to the associated acoustic wavelengths. In this frequency regime the external fluid has both an added mass and a radiation damping influence on the structure.

The spherical pressure wave associated with an underwater explosion (UNDEX) shock loading is characterized by a very steep front where the maximum pressure is attained over an extremely short time duration (rise time). The pressure then drops off exponentially over a significantly longer period of time. Therefore, UNDEX shock loads can be expected to excite submerged structures over a large frequency range that will include low, high, and intermediate response frequencies. The boundaries of the external fluid must be located a sufficient distance from the structure to ensure proper low-frequency response, while the size of the acoustic elements must be small enough to accurately represent the propagation of high-frequency acoustic waves away from the submerged structure.

Problem and geometry description

This example problem is based upon an UNDEX experiment in which a submerged test cylinder is exposed to a pressure shock wave produced by a 60 lb HBX-1 explosive charge. Kwon and Fox originally described the experiment along with a set of selected experimental results. The objective of this class of analysis is to evaluate the behavior and integrity of a structure under UNDEX loading conditions.

The test cylinder is made of T6061-T6 aluminum. It has an overall length of 1.067, an outside diameter of 0.305, a wall thickness of 6.35 mm, and 24.5 mm thick welded endcaps. The cylinder is suspended horizontally in a 40 m deep fresh water test quarry. The 60 lb HBX-1 explosive charge and the cylinder are both placed at a depth of 3.66 m. The charge is centered off the side of the cylinder and located 7.62 m from the cylinder surface. The suspension depths, charge offset, and duration of the test are selected such that cavitation of the fluid is not significant and no bubble pulse occurs. Strain gauges are placed at several locations on the outer surface of the test cylinder, as shown in *Figure 1*. The strain gauge experimental data are filtered at 2000 Hz. The experimental data presented here are obtained by digitizing the Kwon and Fox strain history curves.

When the acoustic fluid behavior is linear (i.e., no cavitation), the total acoustic pressure within the fluid consists of an incident wave and a scattered wave component. For this example the incident wave is the shock wave produced by the UNDEX charge. The scattered wave is the acoustic field generated by the interaction of the incident wave and the submerged structure. The nature of the incident wave can be determined from either empirical formulas or experimental data. Therefore, the spherical incident shock wave is applied as a transient

load active on both the acoustic and structural meshes at their common surfaces (the wetted interface), and the external fluid pressure degrees of freedom represent only the unknown scattered component of the total acoustic pressure. The type of incident wave loading is either a scattered wave or a total wave formulation. The scattered wave formulation described above is the default condition for Abaqus/Explicit analyses. The total wave formulation is used for cases where nonlinear fluid response is expected or where the total acoustic pressure history is prescribed at an acoustic fluid boundary.

During the UNDEX test two pressure transducers are positioned 7.62 m from the charge, away from the cylinder but at the same depth as the cylinder. These transducers provide an experimental determination for the pressure vs. time history of the spherical incident shock wave as it travels by the point on the cylinder closest to the charge (strain gauge location B1). *Figure 2* shows a time history curve of the incident pressure wave recorded by the transducers. The input file shock-pulse.inp contains this time history curve as an amplitude table used to define the incident wave loading.

Abaqus/Explicit model

Figure 3 shows the S4R finite element shell mesh used to represent the test cylinder. The mesh consists of 2402 nodes (14412 DOF) and 2400 elements with 40 circumferential divisions and 53 axial divisions. The element connectivity is such that each shell normal is directed into the external fluid. The nodes are positioned on the outside surface of the test cylinder. The S4R elements adjacent to the endcaps are dummy elements with reduced mass and stiffness used only to provide surfaces that correspond to the thickness of the endcaps. BEAM type MPCs are used to tie the endcaps to the main cylinder body. The local coordinate system is used to define the shell element material axes for postprocessing, such that the local 1-direction is aligned with the cylinder's axis for the main body and is radially directed for the endcaps. The local 2-direction is in the circumferential (hoop) direction for both the cylinder main body and the endcaps.

The external fluid is meshed with 4-node AC3D4 acoustic tetrahedral elements. The outer boundary of the external fluid is represented by a cylindrical surface with spherical ends. The characteristic radius of the outer boundary is 0.915 m. The outer boundary must be placed a sufficient distance from the cylinder so that the added mass associated with the low-frequency beam bending modes of the cylinder is represented adequately. The beam bending modes correspond to an $N = 1$ sinusoidal translation of the cylinder's cross-section through the fluid. For evaluating added mass effects when using a simple plane wave radiation impedance boundary for the external fluid, the outer boundary of the fluid can be considered rigid (nonradiating). Therefore, an analytical solution for the added mass associated with the translation of an infinite cylinder of radius R_i located within a fluid-filled infinite cylinder of radius R_0 can be used to determine an appropriate characteristic radius for the external fluid. Results for the analytical solution presented by Blevins are listed in *Table 1*. The characteristic radius is based upon an outer boundary (R_0) to cylinder radius (R_i) ratio of 6.0, which corresponds to an added mass error of about 6% for infinite cylinders. When using enhanced surface impedance models, the outer fluid boundary location can be placed at about half of the distance required when using the plane wave radiation impedance model. However, for this example the R_0/R_i ratio was maintained at 6.0 even when using the source-based surface impedance models for the external fluid. Comparable results for the structural response can be obtained when the source-based boundaries were located half as far from the structure. For the low-frequency beam bending modes, system losses (damping) caused by hydrodynamic drag and/or fluid viscosity are not accounted for by acoustic radiation. Therefore, mass-proportional damping applied to the test cylinder mesh is used to approximate these types of losses.

Figure 4 shows the combined external fluid and test cylinder meshes. One quarter of the fluid mesh is omitted from this figure to allow an inspection of the acoustic element mesh inside the external fluid domain. The mesh is generated with Abaqus/CAE. The nodal seeding on the fluid outer boundary is set at 0.10 m, corresponding to 9.7 element divisions per acoustic wavelength at a response frequency of 1500 Hz. The nodal seeding on the fluid wetted interface with the test cylinder is set at 0.04 m, corresponding to 24.4 element divisions per acoustic wavelength at 1500 Hz. The radiation boundary condition is applied on the fluid outer-boundary surfaces.

Fluid-structure coupling and shock wave loading

The acoustic structural coupling between the fluid mesh acoustic pressures and the test cylinder structural displacements at their common surfaces (the wetted interface) is accomplished with a surface-based tie constraint. *Figure 5* shows the surface mesh at the acoustic-structure wetted interface associated with the external fluid (*Figure 3* shows the test cylinder surface). Since the acoustic mesh is coarser than the structural mesh, the surface of the external fluid at the wetted interface is designated as the main surface. This pairing creates an internal coupling of the acoustic pressure and structural displacements at the test cylinder (secondary) surface nodes and ties the cylinder's acoustic pressures to the fluid mesh acoustic pressures at the wetted interface.

Figure 5 also illustrates the concept of a source point and a standoff point as they relate to an incident acoustic wave loading. For this example the source point represents the actual physical location of the explosive charge relative to the structure. The standoff point represents the location of the incident wave (shock front) at the start of the analysis (total time = 0.0) and is the point at which the pressure history of the incident wave is provided. For solution efficiency the standoff point should be placed at the location on the fluid-structure interface that is closest to the source point. The standoff point can be placed away from the structure closer to the source point, but this will only delay the onset of the transient response. Under no circumstances should the standoff point be located within or behind the structure being analyzed.

The incident wave can either be planar or spherical and requires the location of the standoff point and the source point. For a spherical shock wave, as in this example, the relative positions of the standoff point and source point determine how the wave's pressure will decay with distance from the source point. For a planar wave, which does not decay, the relative positions of the standoff point and source point are used to define the direction of incident wave travel. The fluid properties associated with the incident wave includes wave speed. Defining the incident wave properties independent of the acoustic mesh allows incident wave loading to be used in the analysis of weakly coupled or uncoupled acoustic-structural systems (i.e., air blast analyses). For these cases the incident wave loading can be applied to a structure when no acoustic medium is directly modeled.

The pressure history at the standoff point is used to drive the incident wave. The amplitude definition specifies the surface name to which the incident wave loading is applied and a reference magnitude for the pressure curve. For acoustic-structural systems where the fluid and structure are both modeled and coupled, the incident wave loading must be defined to act upon both the fluid and structural surfaces at the wetted interface. Acoustic volumetric acceleration loads corresponding to the incident wave are then applied to the fluid surface, while the incident wave pressures are applied to the structural surface.

Results and discussion

The Abaqus/Explicit model for this UNDEX example has a total of 23337 active degrees of freedom and requires approximately 160 MB of memory. The transient analysis is run for 0.008 seconds with a 1.69×10^{-6} critical time increment (~4733 solution increments). *Figure 6* shows the time history of axial displacement (U3) for the center nodes of the endcaps. These curves clearly show the periodic response associated with a dominant axially directed mode of the cylinder–endcap structure. *Figure 7* shows the 1-direction translation (U1) of the endcap center nodes. The 1-direction is also the primary direction of shock wave propagation. The response curves clearly illustrate that there is a rigid body translation of the cylinder, and the oscillations are representative of the fundamental beam bending mode of the cylinder. *Figure 8* shows the time history of vertical (U2) displacement for nodes located at the top and bottom midplane of the test cylinder. These curves suggest that a dominant N=2 ovalization mode of vibration occurs at about 170 Hz (based on an estimated period of 0.0059 seconds). The frequency for the first ovalization mode of the test cylinder in a vacuum is 330 Hz, based upon an Abaqus/Standard eigenvalue extraction analysis. This shift in the N=2 response mode frequency illustrates the added mass effect of the external fluid on the response of the submerged cylinder.

Figure 9 through *Figure 11* contain time history plots of the test cylinder strains obtained from the Abaqus/Explicit analysis with experimental data for locations B1, C1, and A2. The experimental curves are obtained by digitizing the response plots published by Kwon and Fox. The digitized curves are shifted to the left by 0.0002 seconds

on the time axis to account for an apparent time differential between the experiment and the Abaqus/Explicit solution. *Figure 9* contains history plots of the axially directed strains at location B1. The analytical-experimental correlation at an early time (peak strain prediction) is very good, as is the prediction for the dominant response frequency of the test cylinder. The predicted strain oscillations at longer times suggest that the modeling of hydrodynamic drag damping and viscous losses by applying mass damping to the cylinder mesh could be improved. *Figure 10* contains the history plots for the axially directed strains at location C1. The initial peak response (high frequency) contained in the Abaqus/Explicit solution is not present in the experimental data. This may be due to the sampling rate and filtering techniques used to obtain the data or to high strain gradients being averaged over the effective length of the strain gauge. Otherwise, the Abaqus/Explicit solution closely tracks the experimental data and provides a conservative estimate for the peak response. *Figure 11* contains the history plots for the hoop-directed strains at location A2. As in *Figure 10*, the initial peak response (high frequency) contained in the Abaqus/Explicit solution is not present in the experimental data. Otherwise, the Abaqus/Explicit solution closely tracks the experimental data. *Figure 9* through *Figure 11* indicate that the overall UNDEX analysis model provides a conservative estimate of the cylinder's peak response and is, therefore, appropriate for meeting the analysis objective.

Figure 12 shows a contour plot of accumulated equivalent plastic strain (PEEQ) on the outer surface of the test cylinder. To obtain the plot, an averaging threshold of 100% is used and the maximum contour value is specified as 9.16E−3. The plot corresponds to the end of the transient analysis, which is well after the last increment of plastic strain is detected from a plot of the cylinder's total plastic strain energy vs. solution time. The slight degree of solution nonsymmetry exhibited about the cylinder's midplane is due to the nonsymmetric nature of the free tetrahedron acoustic element mesh of the external fluid.

Input files

submerged_cyl_driver.inp	Abaqus/Explicit analysis of a submerged cylinder subjected to an UNDEX shock wave.
submerged_cyl_cylinder.inp	The finite element mesh data for the test cylinder, including element and node set definitions for output requests.
submerged_cyl_water.inp	The finite element mesh data for the external water, including element and node set definitions for surface creation and output requests.
submerged_cyl_pulse.inp	The time history of the shock wave pressure at the standoff point defined by the **AMPLITUDE* option.

References

- Kwon, Y. W., and P. K. Fox, "Underwater Shock Response of a Cylinder Subjected to a Side-On Explosion," *Computers and Structures, Vol. 48, No. 4, 1993.*
- Blevins, R. D., Formulas for Natural Frequencies and Mode Shapes, Robert E. Fruger Publishing Co., 1979.

Tables

Table 1: Added mass for N=1 translation mode of an infinite cylinder (fluid between concentric cylinders).

Cylinder Radius Ratio (R_0/R_i)	Added Mass Ratio (External Boundary/Infinite Domain)
1.5	2.600
2.0	1.667
4.0	1.133

Cylinder Radius Ratio (R_0/R_i)	Added Mass Ratio (External Boundary/Infinite Domain)
6.0	1.057
8.0	1.032
16.0	1.008
24.0	1.004

Figures

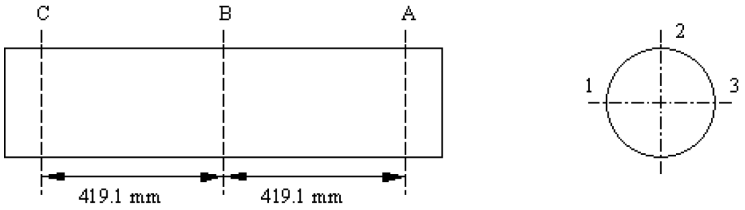


Figure 1: Strain gauge locations (A1, A2, B1, B2, B3, C1, C2) with B1 closest to the charge.

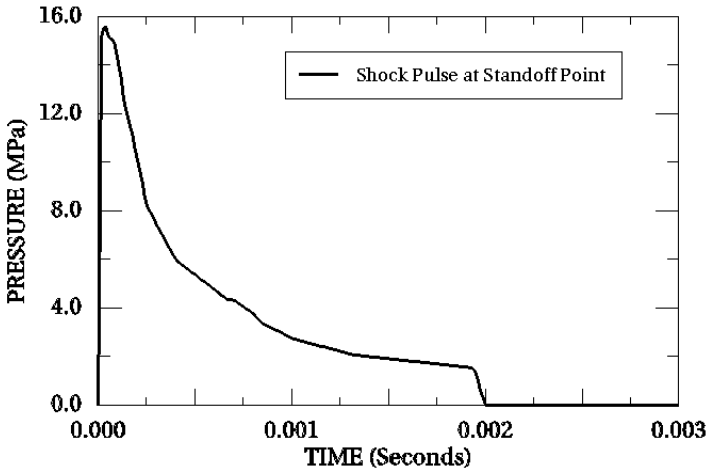


Figure 2: Incident pressure wave transient (shock pulse).

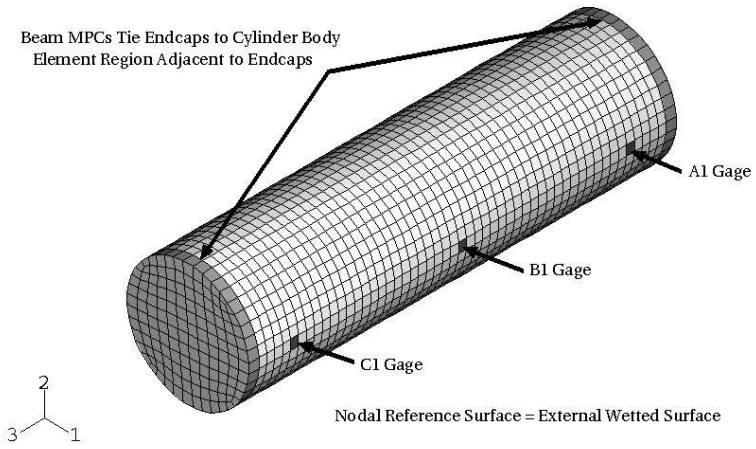


Figure 3: Test cylinder model.

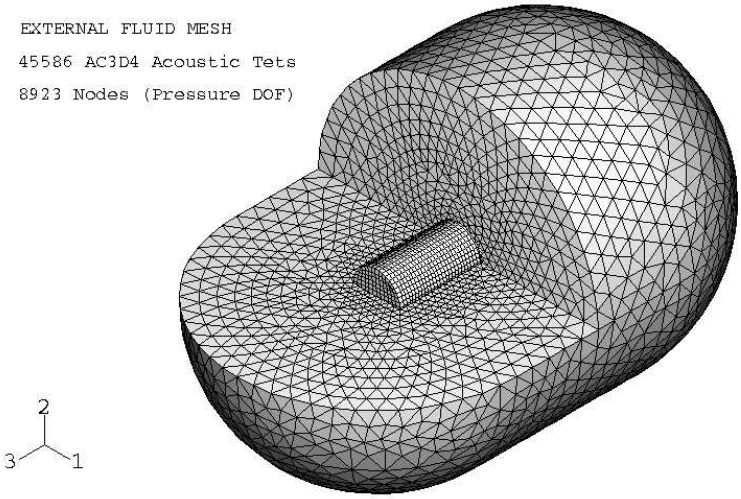


Figure 4: Test cylinder and external fluid acoustic mesh.

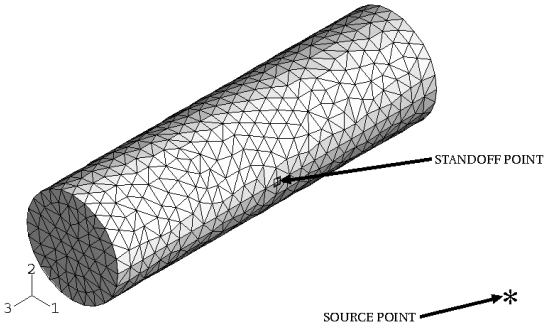


Figure 5: External fluid surface mesh at the acoustic-structure wetted interface.

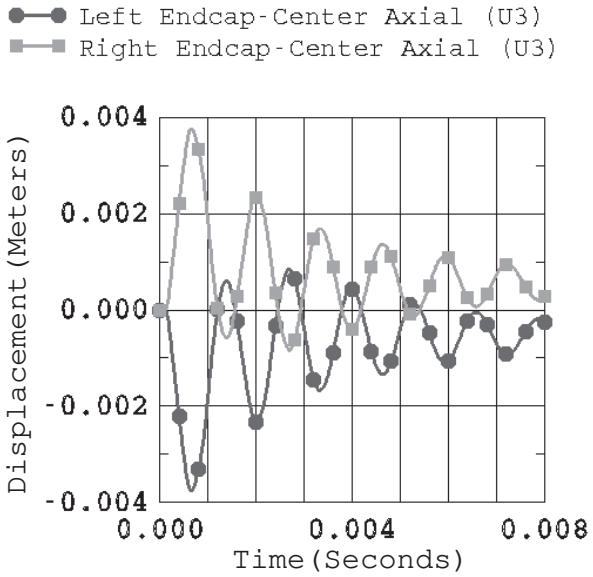


Figure 6: Axially directed displacements (U3) at the center of the endcaps.

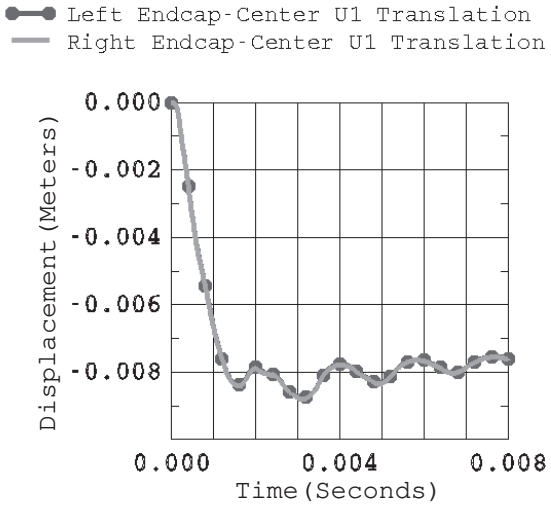


Figure 7: Displacements at the center of the endcaps.

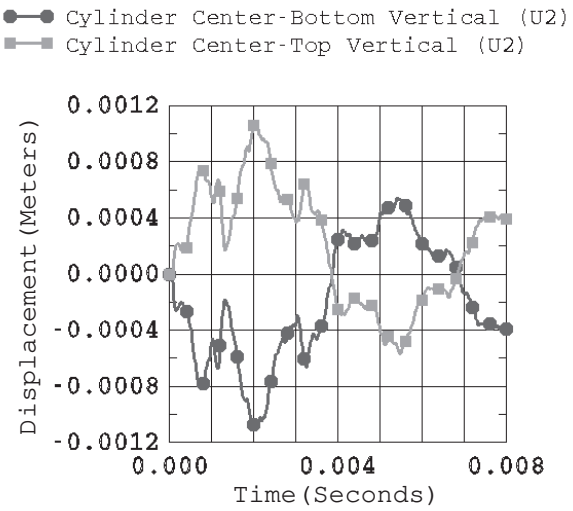


Figure 8: Vertical (U2) displacements at the cylinder midplane (top and bottom).

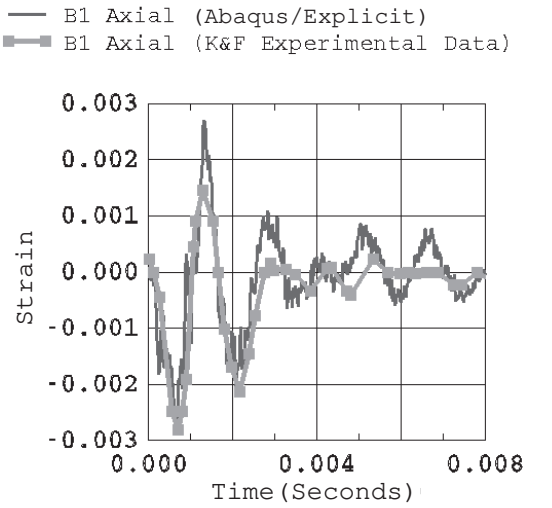


Figure 9: Axially directed strains at location B1.

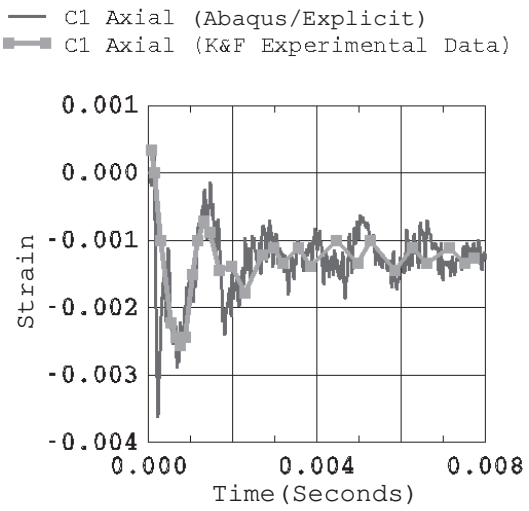


Figure 10: Axially directed strains at location C1.

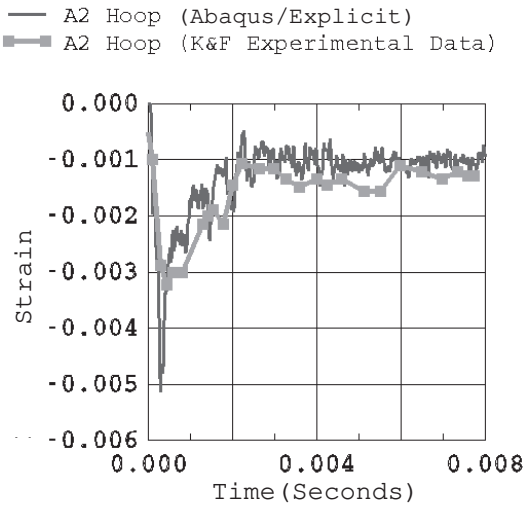


Figure 11: Hoop-directed strains at location A2.

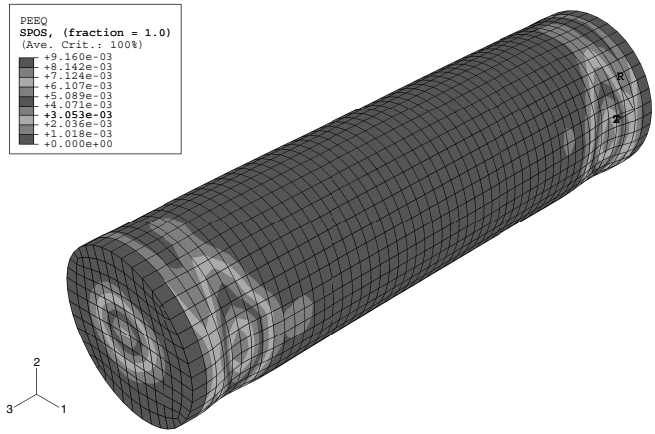


Figure 12: Accumulated equivalent plastic strains (PEEQ).

Convergence studies for shock analyses using shell elements

Products: Abaqus/Explicit

Overview

This example illustrates a methodology you can use to design a computationally efficient model of a structure subjected to shock loading.

Structures subject to shock loads require careful design because of the high-frequency content of the loading signal and the low-pass filtering property of a finite element mesh. To address these issues, a simplified model is useful to estimate optimal parameters for an accurate, efficient, and realistic model.

The goal of the present study is to propose a methodology for designing a computationally efficient structural model, by answering the following questions:

- Given a load signal $f(t)$, what is the maximum element size for a given structure to obtain a reasonably accurate response?
- Given a minimum possible element size, what is the best signal $f'(t)$ to emulate the structural response to $f(t)$ with reasonable accuracy?

Model description

For the purpose of this study, consider a cylinder made of a linear elastic material with E=2.11 × 10^{11} N/m^2 and ν=0.3. It has an overall length of 10 m, a diameter of 3.5 m, and a wall thickness of 0.04 m. Quarter symmetry is employed by applying the appropriate boundary conditions on the corresponding geometry edges. The model is shown in *Figure 1*. Incident wave loading is applied on the outer surface of the cylinder, using the Kwon & Fox (KF) signal described in *Response of a submerged cylinder to an underwater explosion shock wave*. Abaqus/Explicit is used with a fixed time increment of 1 × 10^{-6} s, chosen to provide a suitable integration scheme for the load. The transient response (U, V, A) is analyzed at a standoff point over a period of 6 ms. Uniform meshes of S4R elements with various sizes are employed.

Results and discussion

First, convergence of the structural response at the standoff point is examined, applying the original KF loading signal. Errors are reported in a L^2-norm over the entire 6 ms duration of the response. The converged solution is formed by successively applying finer meshes until a desired relative accuracy is obtained. From *Figure 2* and *Table 1* we conclude that a mesh with h=0.025 m is a reference solution. The solution with h=0.05 m can also be considered converged; therefore, it can become the reference solution for subsequent studies (the acceleration response differs by only 1.1% with respect to the solution with h=0.025 m).

Figure 3 and *Figure 4* depict the relative accuracy of the displacement and velocity responses with respect to the solution with h=0.05 m. For example, *Figure 3* shows that if an accurate velocity response is of interest, element size restrictions can be relaxed to h=0.25 m (6.3% relative error versus the reference configuration); if the displacement response is of interest, the element size restrictions can be relaxed further to $h > 0.5$ m (*Figure 4*).

Figure 5 is constructed by defining a mesh characteristic frequency, Ω_h, used as x-data for the error curve. For a given number of elements, N, per excitation wavelength, the mesh characteristic frequency is

$$\Omega_h = \frac{c_s}{N \cdot h},$$

where c_s is the characteristic wave speed (here the shell flexural wave speed) and h is the element diameter. The structural wave speed is dispersive; i.e., it varies with the excitation frequency. In *Figure 5*, however, a constant reference value of c_s=1200 m/s is used. The previously defined mesh characteristic frequency also depends on the user's choice of the number of elements per wavelength, N. To preserve accuracy and to address the pollution effect (Ihlenburg and Babuska, 1995), N should increase with the excitation frequency. The choice of h, N, and the wavespeed in the medium, c_s, determines the value of the mesh characteristic frequency, Ω_h, which can be interpreted as a cutoff frequency for designing the filtered signal.

Figure 5 shows that by considering a mesh with h=0.05 m and N=15, the mesh characteristic frequency becomes Ω_h=1600 Hz. If the number of elements per wavelength is relaxed to N=5, Ω_h will increase to 4800 Hz. If the latter value is applied as the cutoff frequency, the filtered signal will capture the true signal more accurately; however, it is likely to produce more noise if the waves are not resolved properly.

The study presented above may lead to meshes that are impractical for realistic models, especially if the acceleration response is of primary interest. For this model of 10 m length, an element of 0.05 m provides sufficient accuracy at the standoff point. Since this may lead to prohibitive computational costs for realistic models, the next studies address ways to relax the element size restraints by modifying the shock signal applied to the structure. These following studies examine the impact of filtering and modified rise time upon the element size restrictions, in an attempt to minimize the noise at coarse meshes.

Filtered signals are applied successively to the structure, in an attempt to quantify the dependence of the optimal cutoff frequency on the element size. The sine-Butterworth second-order filter is used. Although filter performance is very important for optimal results, a comparison of various filtering techniques is beyond the purpose of this study.

The filtered signals are postprocessed via linear scaling with the parameter a such that the final and original signals are equivalent in a n-norm:

$$a_n = \left(\frac{\int_0^T (f_{KF})^n dt}{\int_0^T (f_f)^n dt} \right)^{1/n}; a_1 = \frac{\int_0^T f_{KF} dt}{\int_0^T f_f dt}; a_\infty = \frac{max(f_{KF})}{max(f_f)}.$$

The norm can be varied to design a final signal suitable for various responses: since initially the KF is applied as a load signal, the infinity norm is suitable for obtaining a more accurate acceleration response, while the 1-norm (impulse conservation) is suitable for obtaining a more accurate velocity response.

For this study filtered signals are designed with scaling parameters suitable for acceleration response. Sample postprocessed filtered signals are presented in *Figure 6*. By applying these as incident wave loads on the mesh with h=0.05 m, the results shown in *Figure 7*, *Figure 8*, and *Figure 9* are obtained for the acceleration, velocity, and displacement, respectively. There is a clear distinction between *Figure 7*—a well-captured acceleration response—and *Figure 8* and *Figure 9*—which are poor estimates of the velocity and displacement responses—since the applied filtered load has a significantly higher impulse than the unfiltered signal.

The results presented in *Figure 10* and *Figure 11* are obtained by applying the same load signals to coarser meshes. A summary is presented in *Table 1* and *Figure 14*. The expected trend is visible: as the mesh coarsens, the acceleration response when applying filtered signals is considerably less noisy than the response for unfiltered signals. As the mesh is refined, the response converges to the “wrong” solution given by the filter of choice. Thus, when computational resources are scarce, there is an obvious benefit to applying prefiltered signals, which attenuate the noise due to insufficient spatial discretization.

As an alternative to the sine-Butterworth filter, the signal can be idealized in a heuristic manner by using a linear rise, followed by an exponential decay. By keeping the decay constant, you can study the sensitivity of the structural element size to the rise time of the shock signal.

The original rise time, Δt, of the KF signal is multiplied by a rise time factor (rtf= 2, 5, 10, 15, 20) to obtain new signals, which are then applied to the structure. To account for noise contributions only, the error of the acceleration response is reported on the decay portion [T_{max}, T], where T_{max} is the time stamp of the peak amplitude of the load signal.

The results using this technique are presented in *Figure 12* and *Figure 13* and summarized in *Table 2* and *Figure 15*. By comparison with *Figure 14*, the smoothing strategy appears considerably more effective for the acceleration response than using a sine-Butterworth filter. For example, the case with h=0.125 m and rtf=20 (scaling the upper bound of the spectrum at ~2250 Hz) yields an error of 9%; for the same element size with h=0.125 m, the sine-Butterworth filter with a cutoff frequency of 3000 Hz yields a 32% error.

Input files

step_data.inp	Step data for all models.
cyl_h0025.inp	Model data for submerged cylinder with an element size of 0.025 m.
cyl_h005.inp	Model data for submerged cylinder with an element size of 0.05 m.
cyl_h0125.inp	Model data for submerged cylinder with an element size of 0.125 m.
cyl_h025.inp	Model data for submerged cylinder with an element size of 0.25 m.
cyl_h05.inp	Model data for submerged cylinder with an element size of 0.5 m.
sf_unfilt.inp	Amplitude data for unfiltered KF signal.
sf_300.inp	Amplitude data for KF signal filtered at 300 Hz.
sf_600.inp	Amplitude data for KF signal filtered at 600 Hz.
sf_900.inp	Amplitude data for KF signal filtered at 900 Hz.
sf_1200.inp	Amplitude data for KF signal filtered at 1200 Hz.
sf_1500.inp	Amplitude data for KF signal filtered at 1500 Hz.
sf_2000.inp	Amplitude data for KF signal filtered at 2000 Hz.
sf_3000.inp	Amplitude data for KF signal filtered at 3000 Hz.
sf_5000.inp	Amplitude data for KF signal filtered at 5000 Hz.
sr_2.inp	Amplitude data for KF signal with a rise time factor of 2.
sr_5.inp	Amplitude data for KF signal with a rise time factor of 5.
sr_10.inp	Amplitude data for KF signal with a rise time factor of 10.
sr_15.inp	Amplitude data for KF signal with a rise time factor of 15.
sr_20.inp	Amplitude data for KF signal with a rise time factor of 20.
driver_h0025.inp	Driver file for the mesh with h=0.025 m and all signals.
driver_h005.inp	Driver file for the mesh with h=0.05 m and all signals.
driver_h0125.inp	Driver file for the mesh with h=0.125 m and all signals.
driver_h025.inp	Driver file for the mesh with h=0.25 m and all signals.
driver_h05.inp	Driver file for the mesh with h=0.5 m and all signals.

References

- Ihlenburg, F., and I. Babuska, "Finite Element Solution of the Helmholtz Equation with High Wave Numbers. Part 1: The h-version of the FEM," *Computers & Mathematics with Applications*, no. 30(9), pp. 9–37, 1995.
- Kwon, K. W., and P. K. Fox, "Underwater Shock Response of a Cylinder Subjected to a Side-On Explosion," *Computers and Structures*, vol. 48, no. 4, 1993.

Tables

Table 1: Percent relative error in acceleration response for pre-filtered signals using sine-Butterworth filter. Reference solution is the unfiltered load signal applied to a mesh with h=0.025 m.

Cutoff (Hz)	h=0.05 m	h=0.125 m	h=0.25 m	h=0.5 m
300	142	105	105	104
600	116	102	102	102
900	91	90	95	90
1200	72	75	88	90
1500	64	64	85	101
2000	49	49	47	110
3000	33	32	43	129
5000	16	16	32	116
Unfiltered	1.2	18	91	103

Table 2: Percent relative error in acceleration response for smoothed rise time signals. Reference solution is the unfiltered load signal applied to a mesh with h=0.025 m. Errors are measured in L^2-norm over $[T_{max}, T]$, where T_{max} is the time stamp of the peak amplitude.

Rise time factor	h=0.05 m	h=0.125 m	h=0.25 m	h=0.5 m
2	0.7	12	67	93
5	2.1	4.0	55	93
10	4.2	4.4	20	85
15	7	7	14	70
20	9	9	20	44

Figures

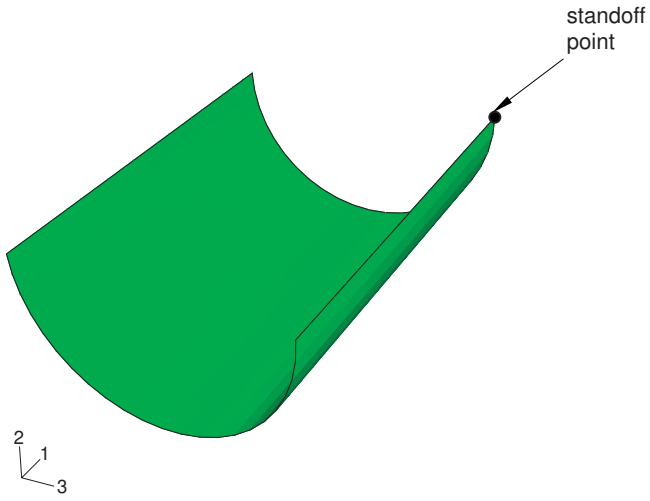


Figure 1: Test problem geometry.

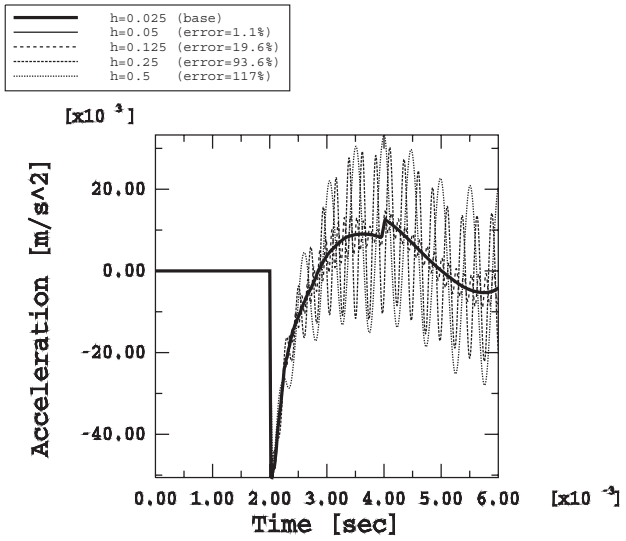


Figure 2: Convergence of athwartship acceleration at standoff point. Element size h [m].

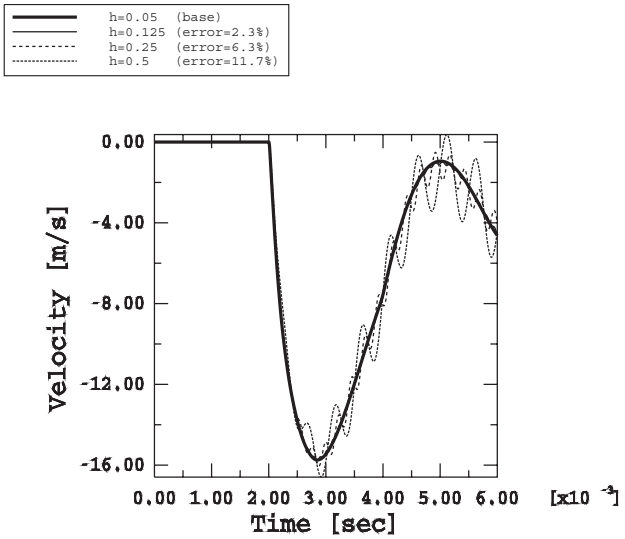


Figure 3: Convergence of athwartship velocity at standoff point. Element size h [m].

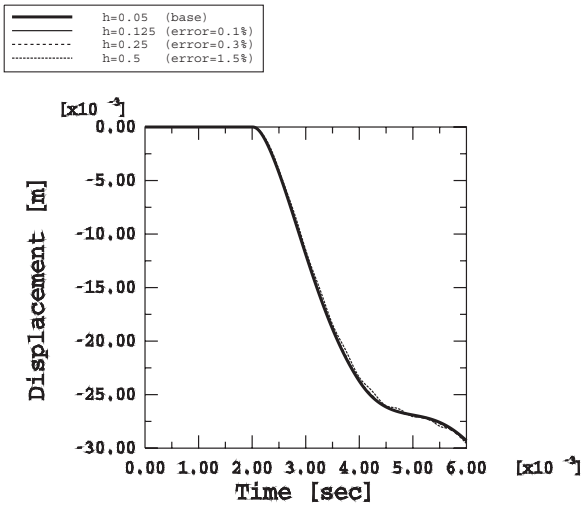


Figure 4: Convergence of athwartship displacement at standoff point. Element size h [m].

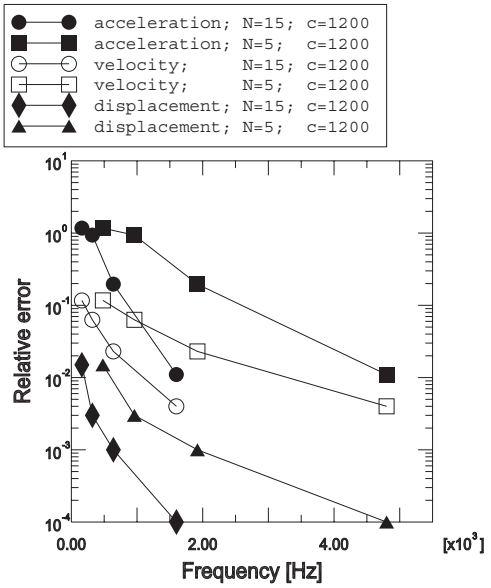


Figure 5: Shock cutoff frequency based on structure response. Wave speed c [m/s].

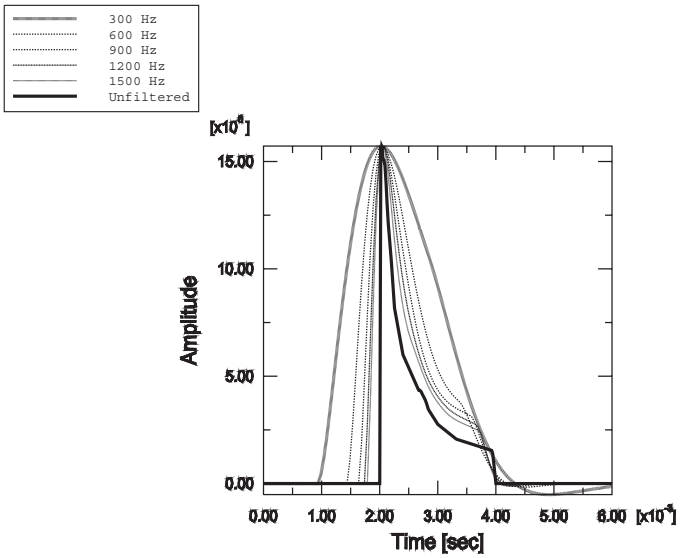


Figure 6: Sample signals obtained after applying sine-Butterworth filter to the original KF signal.

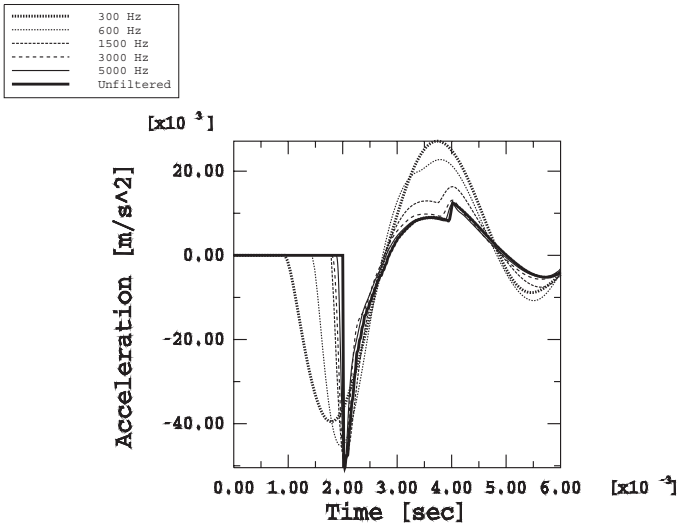


Figure 7: Acceleration response at standoff point for filtered load signals and mesh with h=0.05 m.

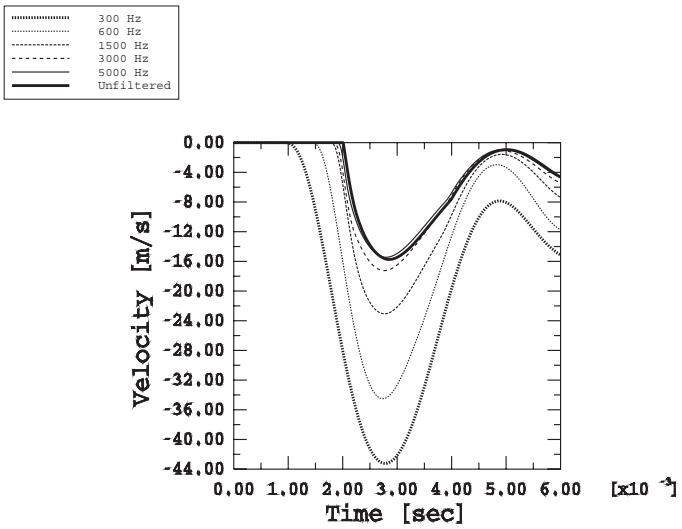


Figure 8: Velocity response at standoff point for filtered load signals and mesh with h=0.05 m.

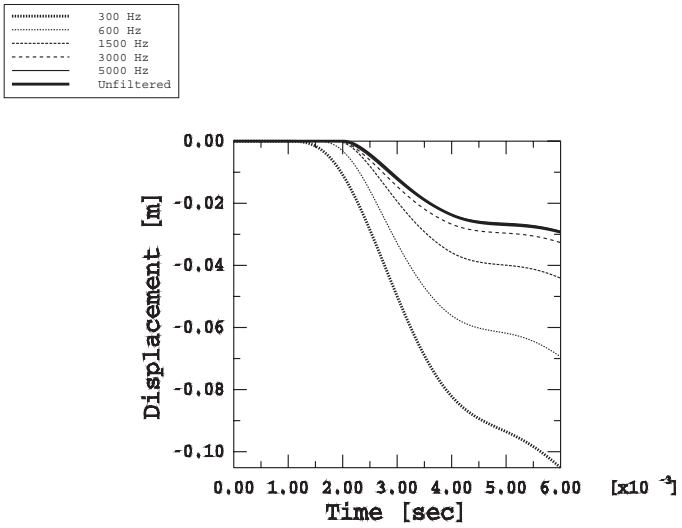


Figure 9: Displacement response at standoff point for filtered load signals and mesh with h=0.05 m.

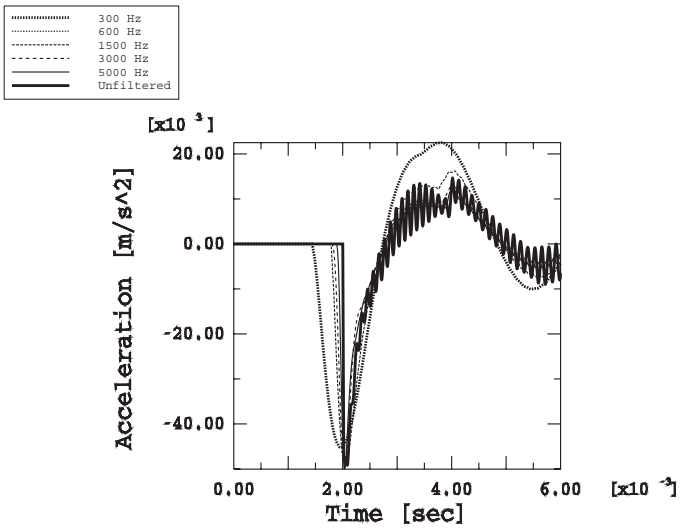


Figure 10: Acceleration response at standoff point for filtered load signals and mesh with h=0.125 m.

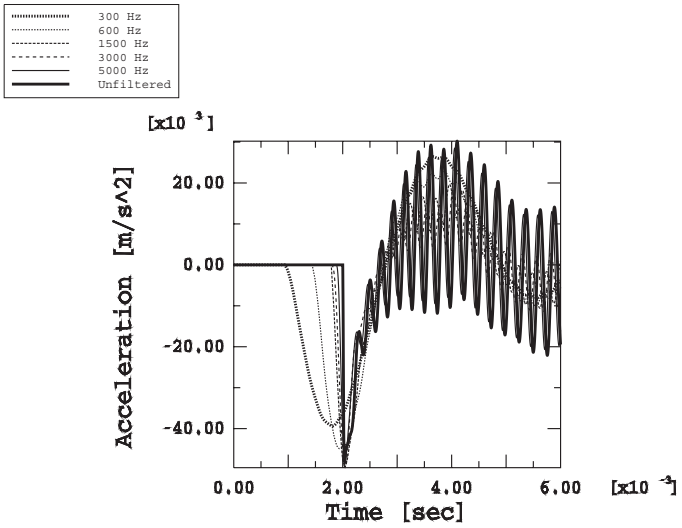


Figure 11: Acceleration response at standoff point for filtered load signals and mesh with h=0.25 m.

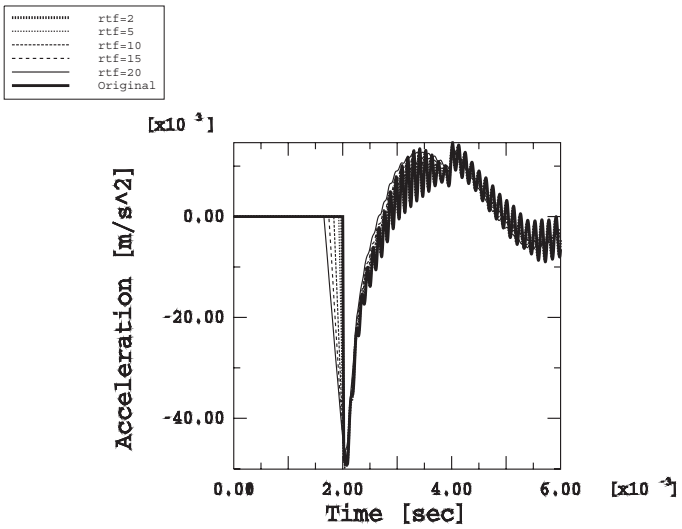


Figure 12: Acceleration response at standoff point for smoothed load signals and mesh with h=0.125 m.

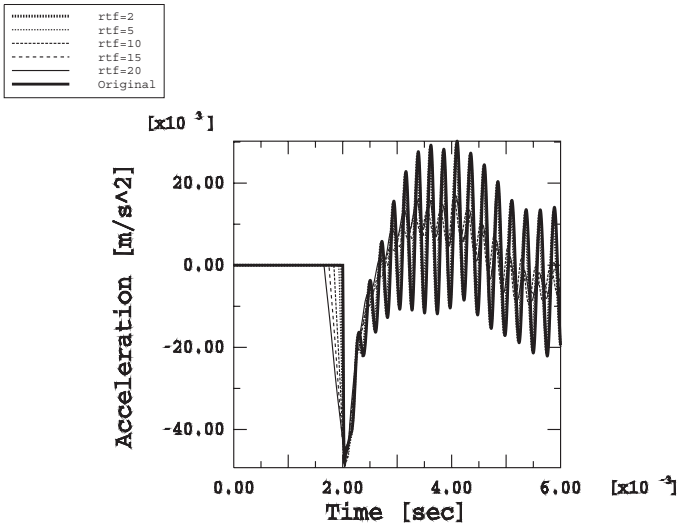


Figure 13: Acceleration response at standoff point for smoothed load signals and mesh with h=0.25 m.

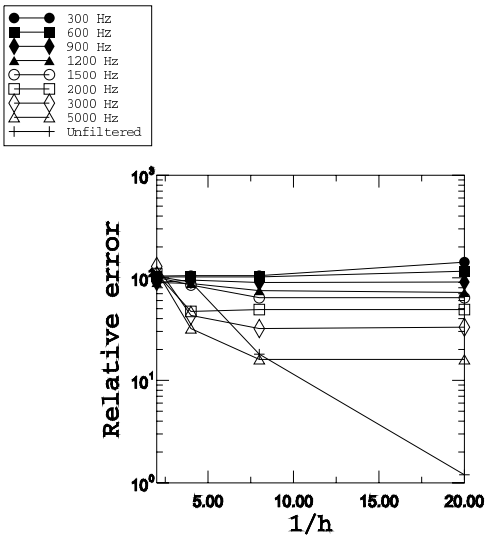


Figure 14: Error in acceleration response for sine-Butterworth filter with various cutoff frequencies. Element size h [m].

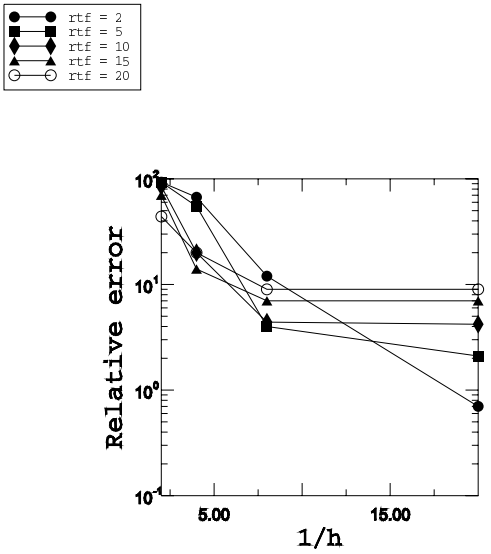


Figure 15: Error in acceleration response for various rise time factors. Element size h [m].

UNDEX analysis of a detailed submarine model

Products: Abaqus/Explicit

Overview

This example illustrates how Abaqus/Explicit can be used to predict the transient response of a large and complicated structure subject to a shock wave loading resulting from an underwater explosion (UNDEX).

Modeling large underwater structures subject to shock loads typically leads to computationally intensive numerical models that require considerable computing resources. A publicly available full submarine model is modified via a modeling approach that minimizes the computational costs and obtains an accurate response in a particular region of interest. Consequently, particular attention is given to structural details in the respective region of interest, while simplifications meant to reduce analysis costs are assumed elsewhere.

The structure is loaded by an incident wave with a shock profile amplitude. If structural integrity is analyzed, a reasonable assumption is that most affected regions will be around the standoff point; thus, more attention to modeling details should be paid to the front part of the submarine (*Figure 1* and *Figure 5*).

Problem and geometry description

The model presented was created based on specifications provided by Forschungsanstalt der Bunderwehr für Wasserschall und Geophysik (FWG), Kiel, Germany (Fiedler and Schneider, 2002). This model is known as the Benchmark Target Strength Simulation (BeTSSi) model and is a reasonably sophisticated replica of a Kilo-class Soviet submarine. The complexity of the structure is considered suitable for testing target-strength simulation codes for more realistic problems (Schneider et al., 2003). In this example problem the BeTSSi model is adapted to test Abaqus/Explicit for a realistic underwater explosion (UNDEX) simulation.

Structural details include flooded bow compartments, sonar array, torpedo tubes, flooded sail compartments, sail access tubes, and an aft chamber (*Figure 1* through *Figure 4*). The pressure hull is enveloped by an external hull only along the upper side and is in direct contact with the fluid throughout the bottom side. The intra-hull space is also flooded along the length of the submarine.

The entire structure is hit by a spherical shock wave due to the explosion of a charge located 16.5 m away from the submarine (*Figure 1*). The pressure-time signal corresponds to a 60lb HBX-1 charge, as in *Response of a submerged cylinder to an underwater explosion shock wave* and taken from Kwon & Fox (1993).

Run procedure

The following procedure summarizes the sequence to run the input files:

1. Run the following beam section property generation procedures to generate the corresponding *jobname*`.bsp` files:
   - `abaqus job=undex_beam_section_front.`
   - `abaqus job=undex_beam_section_sail.`
   - `abaqus job=undex_beam_section_back1.`
   - `abaqus job=undex_beam_section_back2.`
   - `abaqus job=undex_beam_section_back3.`
   - `abaqus job=undex_beam_section_back4.`
   - `abaqus job=undex_beam_section_back5.`

- `abaqus job=undex_beam_section_back6`.
- `abaqus job=undex_beam_section_back7`.
- `abaqus job=undex_beam_section_back8`.
- `abaqus job=undex_beam_section_back9`.
- `abaqus job=undex_beam_section_back10`.
- `abaqus job=undex_beam_section_back11`.
- `abaqus job=undex_beam_section_back12`.
- `abaqus job=undex_beam_section_back13`.

2. Run the following driver input file:

   - `abaqus job=undex_driver_xpl`

   which uses not only the previously generated *jobname*`.bsp` files in Step 1 but also the following included files:

   - `undex_parts.inp`.
   - `undex_outwater_h01.inp`.
   - `undex_innerwater_h01.inp`.
   - `undex_tapered_beam_elsets.inp`.
   - `undex_tapered_beam_sections.inp`.
   - `undex_subbody_h005.inp`.
   - `undex_ties.inp`.
   - `undex_assembly.inp`.
   - `undex_materials_s.inp`.
   - `undex_ampl.inp`.
   - `undex_acoustics_s.inp`.
   - `undex_boundary_conditions.inp`.
   - `undex_output_requests.inp`.
   - `undex_step.inp`.

Reduced model

Since the Kwon & Fox (KF) load signal carries a large excitation spectrum, the outside water has to extend to a large distance, corresponding to the lowest end of spectrum, and both the submarine structure and the outside water have to be discretized with a small element size, corresponding to the wavelength of the high end of the spectrum. This generates a considerably large computational model, due also to the significant difference between the length of the submarine model (62 m) and the low wavelength of the high end of the spectrum. Consequently, a modeling approach is employed in this example problem where a “region of interest” is defined around the shock-wave standoff point. The reduced model still includes details such as flooded bow compartments, torpedo tubes, a sonar dome, and a portion of the pressure and exterior hulls, with the fluid domain defined in the intra-hull space. In addition, stiffeners are added for the detailed region of the pressure hull. The rest of the submarine model is simplified using beam elements, coupled with the region of interest using kinematic coupling constraints. The section behavior of the beams is defined using meshed cross-sections to approximate the cross-section inertia of the real structure (*Figure 6*, *Figure 7*, and *Figure 8*). The tapered region at the back of the submarine is

approximated via 13 stepwise-constant circular cross-sections. Finally, the inertia effects of the outside water are included.

The exterior acoustic domain (water) envelops only the region of interest and extends to a distance of approximately 1.5 m from the submarine structure, corresponding to half the wavelength inwater at a frequency of 500 Hz. Impedance boundary conditions are applied on truncation surfaces to reduce spurious reflections.

Results and discussion

Applying the above simplifications and element sizes of $h_s = 0.05$ for the structure and $h_w = 0.1$ for both the inner and outer water domains, the aggregate model size is 1.2 million nodes and 5.1 million elements, including AC3D4, S4R, S3R, and B31 elements. The analysis is stopped after 6 ms, enough for the wavefront traveling through the acoustic medium to sweep the region of interest.

The pressure for the inner and outer acoustic domains is depicted in *Figure 9* at the end of the 6 ms response. Observe the high-frequency content of the acoustic pressure inside the bulkheads, due to the excitation from the walls. The high-frequency content of the waves in the bulkheads' walls can be seen in *Figure 13* and *Figure 14*.

Figure 10, *Figure 11*, and *Figure 12* illustrate the history of acceleration, velocity, and displacement responses at the standoff point, respectively. A kick-off acceleration of almost 8 km/s^2 is reached due to the initial shock, followed by a rapid decay. By the end of the shock duration (4 ms), it is almost stabilized around zero. The kick-off velocity reaches a value of almost 8 m/s, while the athwartship displacement shows a maximum drift of about 10 mm.

Structural integrity is one of the major concerns of such an analysis. Almost the entire structure is assumed to be made of perfect elastic-plastic steel plates, with position-dependent thicknesses. The only exceptions are the stiffeners, defined as beams with T cross-sections. The material is the same as for the rest of the submarine model, a perfect elastic-plastic steel. High stress concentration regions are analyzed by plotting von Mises stresses (*Figure 15*); permanent deformations are monitored via the equivalent plastic strain (*Figure 16*). High stress concentration regions as well as permanent deformations are observed for regions around the joints. The maximum stresses are observed on the upper panels of the sonar dome, while the largest permanent deformation is obtained around the joint between the pressure hull, the exterior hull, and the horizontal bow compartment.

Input files

undex_driver_xpl.inp	Driver input file.
undex_parts.inp	Parts definition input file.
undex_assembly.inp	Assembly input file.
undex_outwater_h01.inp	Outer water mesh data.
undex_innerwater_h01.inp	Inner water mesh data.
undex_subbody_h005.inp	Submarine structure mesh data.
undex_tapered_beam_elsets.inp	Element set definitions for the tapered region of the submarine body.
undex_tapered_beam_sections.inp	Section data for the element sets defined for the tapered region of the submarine body.
undex_ampl.inp	Amplitude data.
undex_acoustics_s.inp	Impedance and incident wave model data.
undex_materials_s.inp	Material data.
undex_ties.inp	Tie definitions.
undex_step.inp	Step data.
undex_boundary_conditions.inp	Boundary condition data.

undex_output_requests.inp	Output request data.
undex_beam_section_front.inp	Section mesh data for generating cross-section beam properties for the front region of the submarine.
undex_beam_section_sail.inp	Section mesh data for generating cross-section beam properties for the sail region of the submarine.
undex_beam_section_back1.inp	Section mesh data for generating cross-section beam properties for the 1st back region.
undex_beam_section_back2.inp	Section mesh data for generating cross-section beam properties for the 2nd back region.
undex_beam_section_back3.inp	Section mesh data for generating cross-section beam properties for the 3rd back region.
undex_beam_section_back4.inp	Section mesh data for generating cross-section beam properties for the 4th back region.
undex_beam_section_back5.inp	Section mesh data for generating cross-section beam properties for the 5th back region.
undex_beam_section_back6.inp	Section mesh data for generating cross-section beam properties for the 6th back region.
undex_beam_section_back7.inp	Section mesh data for generating cross-section beam properties for the 7th back region.
undex_beam_section_back8.inp	Section mesh data for generating cross-section beam properties for the 8th back region.
undex_beam_section_back9.inp	Section mesh data for generating cross-section beam properties for the 9th back region.
undex_beam_section_back10.inp	Section mesh data for generating cross-section beam properties for the 10th back region.
undex_beam_section_back11.inp	Section mesh data for generating cross-section beam properties for the 11th back region.
undex_beam_section_back12.inp	Section mesh data for generating cross-section beam properties for the 12th back region.
undex_beam_section_back13.inp	Section mesh data for generating cross-section beam properties for the 13th back region.

References

- Fiedler, Ch., and H. G. Schneider, “BeTSSi-Sub—Benchmark Target Strength Simulation Submarine,” *Technical Report, Forschungsanstalt der Bundeswehr für Wasserschall und Geophysik, Kiel*, 2002.
- Kwon, K. W., and P. K. Fox, “Underwater Shock Response of a Cylinder Subjected to a Side-On Explosion,” *Computers and Structures*, vol. 48, no. 4, 1993.
- Schneider, H. G. et al., “Acoustic Scattering by a Submarine: Results from a Benchmark Target Strength Simulation Workshop,” *Proceedings of Tenth International Congress on Sound and Vibration, Stockholm, Sweden*, 2003.

Figures

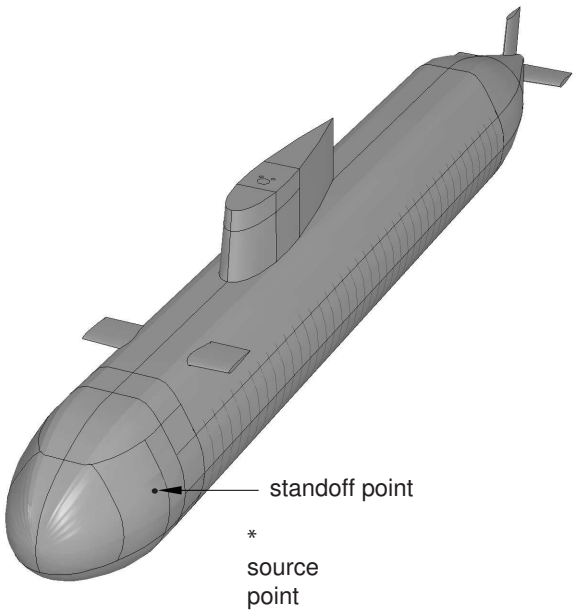


Figure 1: BeTSSi full submarine model.

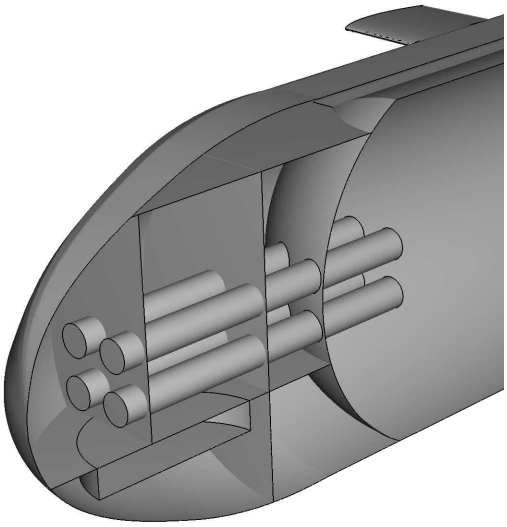


Figure 2: Cut view through front part of the model including details such as bulkheads, torpedo tubes, and the sonar dome.

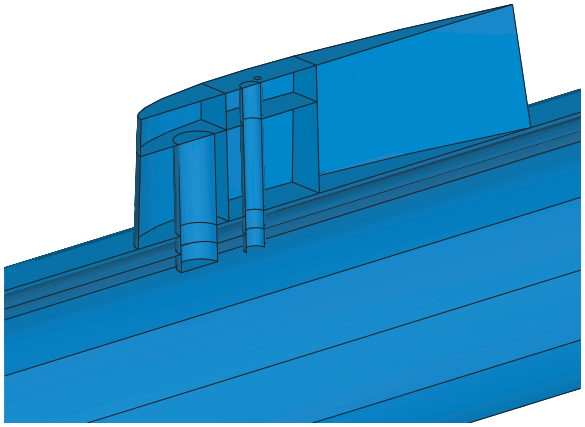


Figure 3: Cut view through sail; sail compartments and access tubes are visible.

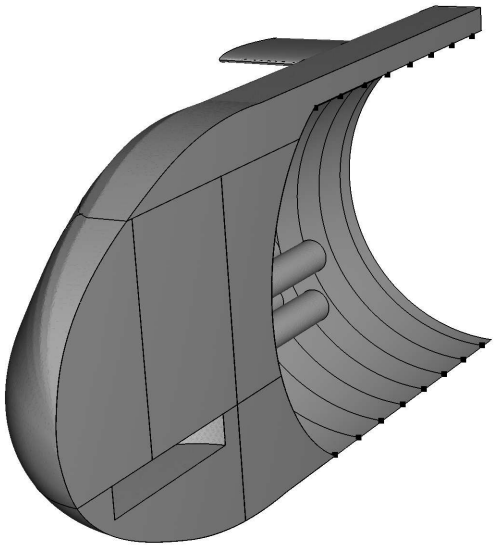


Figure 4: Flooded bow compartments.

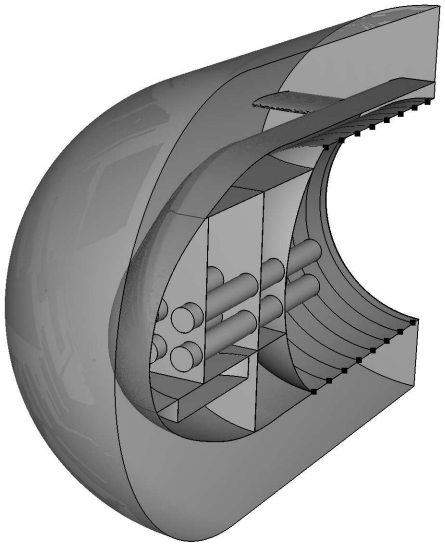


Figure 5: Cut view of reduced model, including exterior fluid domain.

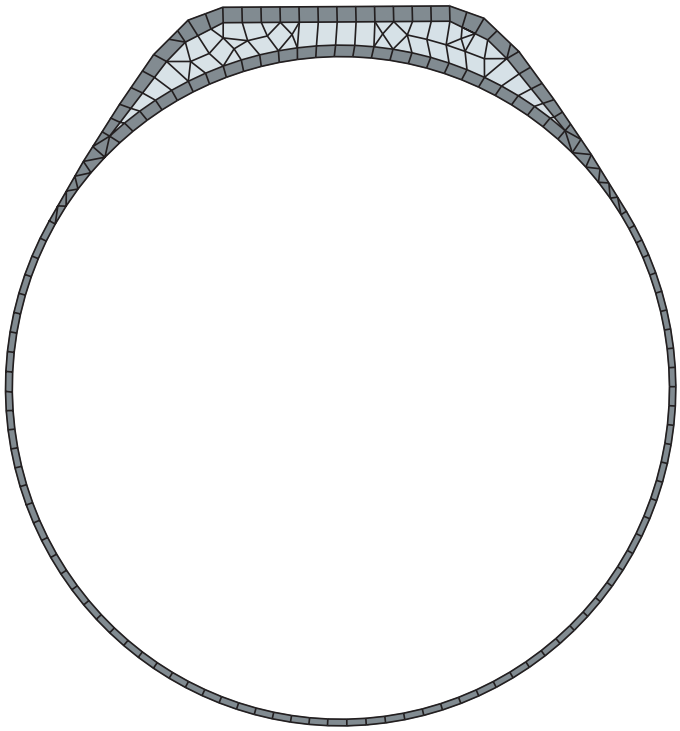


Figure 6: Meshed cross-section of the front beams.

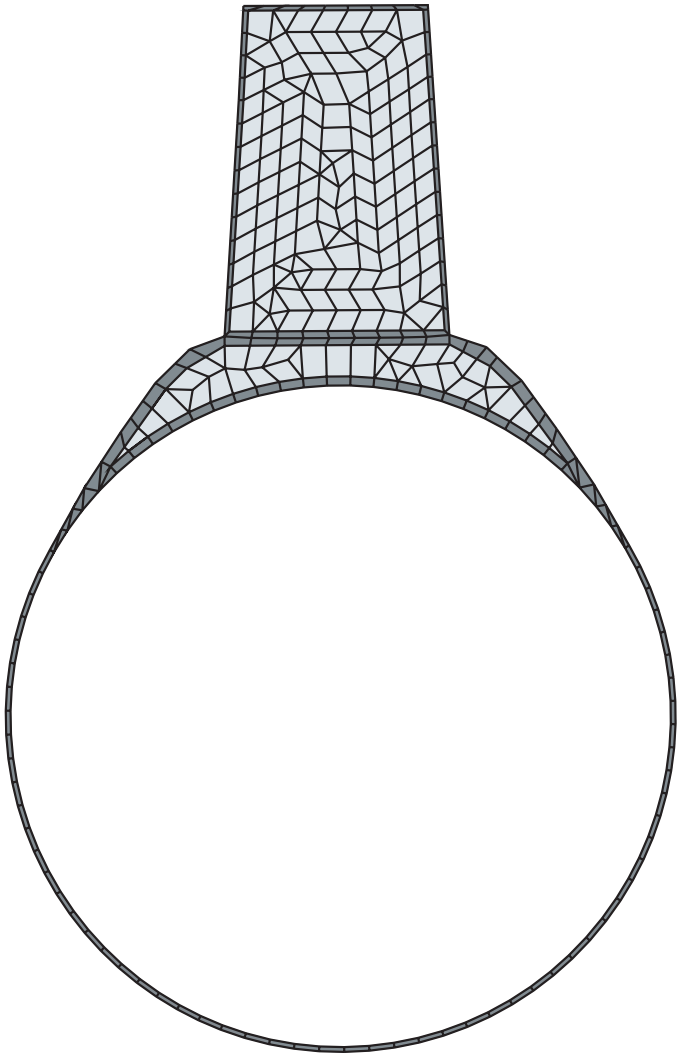


Figure 7: Meshed cross-section of the sail beams.

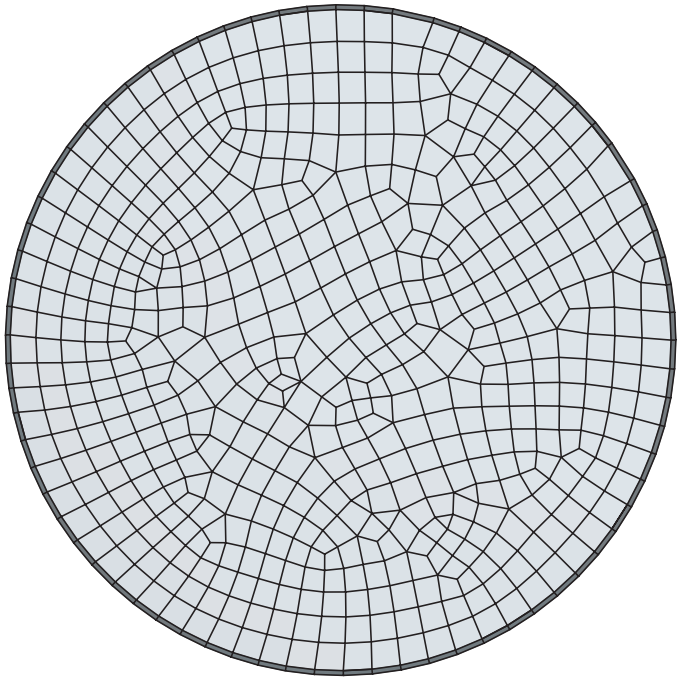


Figure 8: Meshed cross-section of the back beams.

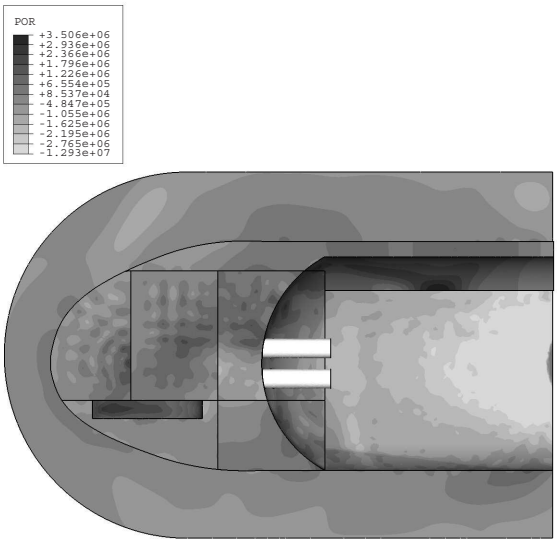


Figure 9: Pore pressure contours of the inner and outer water.

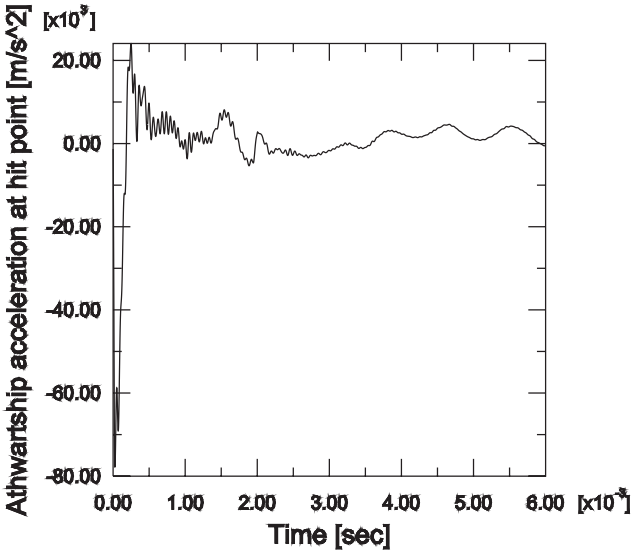


Figure 10: Athwartship acceleration at the standoff point.

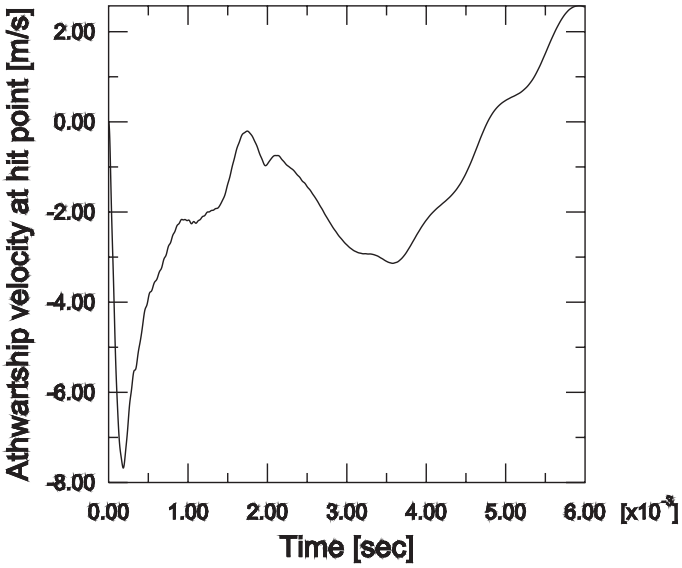


Figure 11: Athwartship velocity at the standoff point.

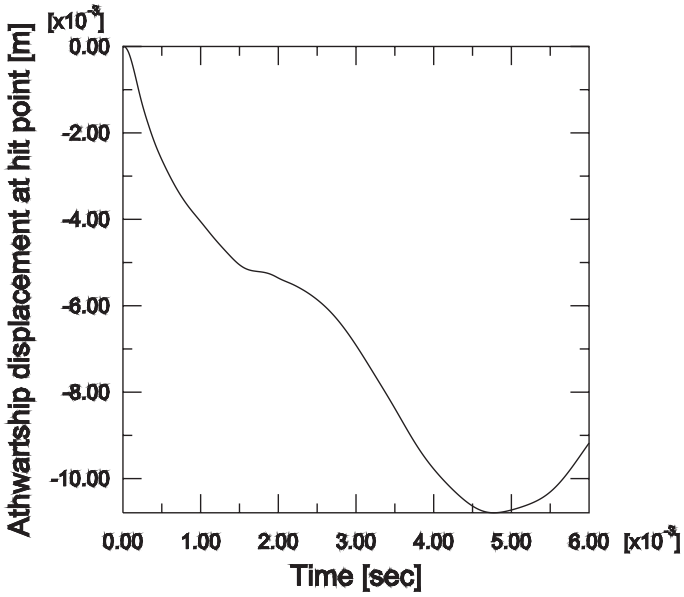


Figure 12: Athwartship displacement at the standoff point.

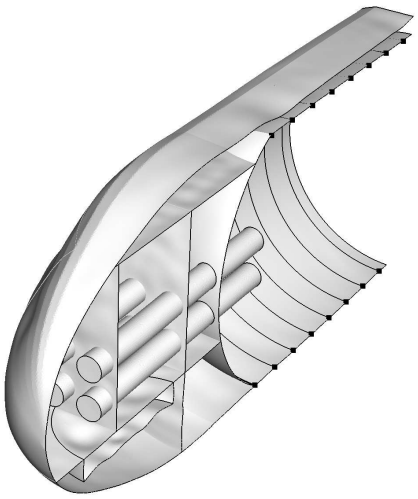


Figure 13: Cut view of deformed configuration.

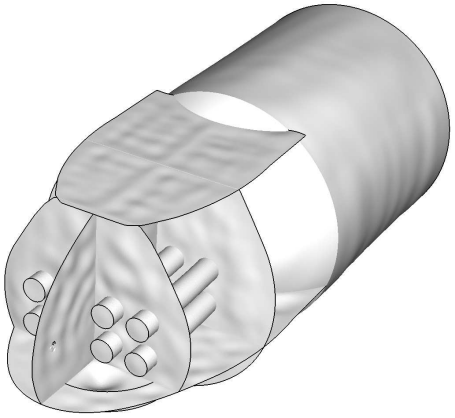


Figure 14: Deformed configuration of pressure hull, bulkheads, torpedo tubes, and sonar dome: whole view.

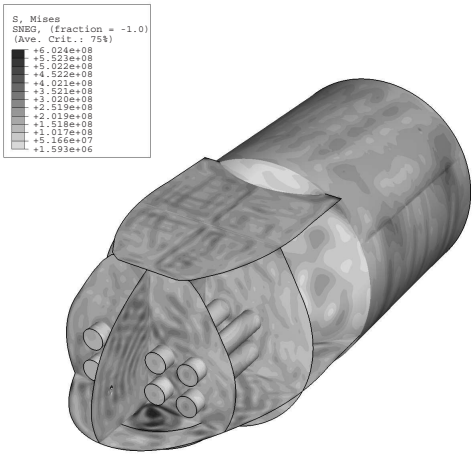


Figure 15: Von Mises stress for bow compartments, pressure hull, torpedo tubes, and sonar array.

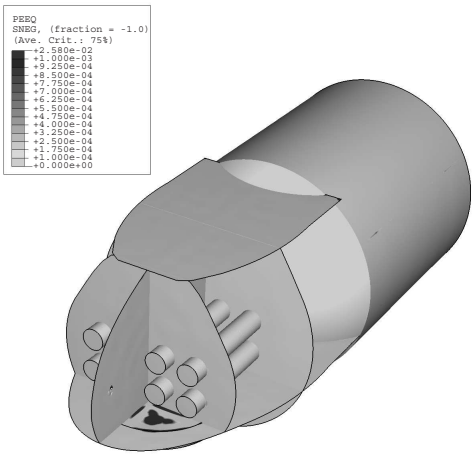


Figure 16: Equivalent plastic strain for bow compartments, pressure hull, torpedo tubes, and sonar array.

Coupled acoustic-structural analysis of a pick-up truck

Products: Abaqus/Standard Abaqus/CAE

Overview

This example illustrates the capability in Abaqus to perform fully coupled acoustic-structural analyses of a pick-up truck model.

This type of analysis has become critically important in the automotive industry and it provides essential benefits toward designing vehicles for ride comfort and quietness. This example uses the pick-up truck model geometry described in *Inertia relief in a pick-up truck*. Only a portion of the pick-up truck is modeled, including the chassis, the cabin, and the air inside the cabin. Structural elements are used to model the cabin and the chassis, and acoustic elements are used to model the air interior. Connector elements are used to connect the various structural parts together. The coupling between the structure and acoustic medium is modeled by applying a tie constraint. Frequency domain analyses are performed for both regular models (without substructures) and for models using coupled structural-acoustic substructures.

Geometry and materials

The pick-up truck model (1994 Chevrolet C1500) discussed here is depicted in *Figure 1* through *Figure 3*.

The air is modeled only inside the cabin, and the process of constructing the air mesh in this particular case is worth a brief discussion. Normally, if the solid geometry of the structural part (the cabin) were available, a Boolean subtraction could be performed in Abaqus/CAE to obtain the solid geometry of the included space (in this case the air inside the cabin). Unfortunately, the solid geometry of the cabin is not available since the structural model is based on a public-domain mesh, as discussed in *Inertia relief in a pick-up truck*. To overcome this issue, the following strategy is adopted. An approximate air geometry is created in Abaqus/CAE to follow roughly the contour of the cabin interior, including the dashboard, the doors, the cabin floor, the cabin top, the back wall, and the seat. The mesh created from this geometry does not conform exactly to the geometry of the cabin interior. However, on the tie constraint that connects the air mesh to the structural parts, you can adjust the nodes so that the nodes belonging to the air surface will be pushed onto the inside cabin surface or onto the seat surface to conform to the structural mesh (see *Figure 2*).

The materials used for the cabin and chassis are described in *Inertia relief in a pick-up truck*. The air properties used inside the cabin are: air density of 1.2 kg/m^3 and air bulk modulus of 1.39×10^5 Pa, which produce a sound speed of 340 m/s.

Models both with and without damping are constructed. In the models where damping is considered, two forms of damping are modeled, as follows. Rayleigh stiffness proportional damping, governed by the parameter β, is used in the structural materials in the model. For a given value of β applied to all materials in the structure (mostly steel), the damping fraction ξ_α for a mode α with natural circular frequency ω_α is given by the formula $\xi_\alpha = \beta\omega_\alpha/2$. The value of β in the model is chosen to give approximately 1% critical damping for the modes whose natural frequencies are in the middle of the range of excitation (at about 80 Hz). Surface impedance is also specified on the cabin floor to model the acoustic damping effect of a carpet by using an impedance boundary condition. The impedance properties on this surface are chosen such that about 67% of a planar wave incident to this surface in the normal direction would be reflected.

Models

Four different models are considered.

Model 1

In the first model only the cabin and the interior air are considered. The structural part of the finite element model is shown in *Figure 1* with the doors removed for illustration purposes. The cabin has six connection points with the chassis: two in the front and four under the seat. The outermost four of these connection points are fixed with boundary conditions, and a natural frequency extraction is performed. Several steady-state dynamic analyses (mode-based, direct-solution, and subspace-based) follow the eigenfrequency extraction step. Separate analyses with and without damping are conducted. There are 43,663 structural elements (mostly shells) and 12,171 acoustic elements in this model for a total of 207,994 degrees of freedom. The average structural element size is about 90 mm, and the average acoustic element size is approximately 325 mm. Considering that at least 5–6 elements are needed per wavelength for accurate representation of the dynamics, the highest excitation frequency for which results can be computed accurately is about 175 Hz. Two excitation cases are considered: harmonic point loading (at the two hook-up points that have not been constrained) and incident wave loading on the bulkhead below the dashboard. The latter case models engine compartment noise propagating through an air path. This airborne load is modeled both as an incident plane wave and as a diffuse field.

Model 2

In the second model the cabin-air model is reduced to a fully coupled structural-acoustic substructure. The four connection points where boundary conditions are applied in the first model are retained using the retained degrees of freedom. In addition, 200 coupled structural-acoustic eigenmodes are extracted. The eigenmodes are then retained using eigenmode selection to better represent the dynamics of the substructure in the frequency range of interest. Consequently, the substructure is represented by a total of 224 degrees of freedom to represent the 207,994 degrees of freedom in the first model. The substructure is then used in a separate, one-element natural frequency extraction analysis. The results are recovered from the substructure and compared to the results obtained from the first model.

Model 3

In the third model both the cabin and the chassis are considered on the structural side (*Figure 3*), while the air is modeled inside the cabin only. Since no air mesh is used to model the ambient air, this type of model can be used to study the structural path contributions to the noise inside the cabin. In the steady-state dynamic analyses of this model the excitation is provided by point loads applied to the engine mounts, while the chassis is supported with fixed boundary conditions at its ends. There are 53,897 structural elements in this model, while the number of acoustic elements is the same as in the first model.

Model 4

Finally, the cabin-chassis model is reduced to two substructures: a fully coupled structural acoustic cabin-air substructure and a structural-only chassis substructure. As in the previous substructure model both the retained degrees of freedom and the eigenmode selection are used to generate the two substructures. The two substructures are represented by 236 and 284 degrees of freedom, respectively. The substructures are then used in a separate, two-element natural frequency extraction analysis; and the results are compared to the results from the third model.

Results and discussion

Some of the steady-state dynamic results for the undamped cabin-air model analyses are shown in *Figure 4* and *Figure 5*. In all analyses 200 sampling points are selected in the frequency range of interest (35–120 Hz). This frequency range corresponds to engine-induced vibrations in the range of 2100–7200 RPM. Of particular interest in these analyses is the sound pressure level at a location in the vicinity of the driver's ear. The response is shown in *Figure 4* and is calculated from the acoustic pressure using the following equation

$$SPL = 20.0 \times log_{10} \left(p_{rms} / p_{ref} \right),$$

where $p_{rms} = p/\sqrt{2}$ and $p_{ref} = 2 \times 10^{-5}$ Pa. *Figure 5* shows the displacement response of one of the nodes on the cabin floor where the harmonic load is applied. The results from the subspace-based and the mode-based steady-state dynamic analyses are virtually identical (as expected), and they compare quite well with the results from the direct steady-state dynamics analysis (*Figure 4* and *Figure 5*). The sound pressure level as computed in these analyses is very high since no damping is considered (neither structural nor impedance-like at the structural-acoustic interface).

Figure 6 shows the noise level for the cabin-air model when damping is considered. Since the mode-based steady-state dynamics analysis would not take into account the forms of damping considered here, only results from the subspace projection and direct analyses are computed. The results compare quite well. Notably, the subspace projection analysis is approximately 20 times faster than the direct analysis. While the sound pressure level is significantly lower in this analysis when compared to the analysis with no damping, the level is still very high. This suggests that the damping considered in this model is still quite low. Impedance-type damping is considered only on the cabin floor; thus, 100% of the acoustic waves would be reflected from the cabin walls, roof, doors, and windows to produce a higher sound pressure level.

Figure 7 shows the real part of the acoustic pressure in the cabin volume at 120 Hz for both the plane wave (left) and diffuse field (right) excitations. The same damping is used as in the previous case—an impedance defined on the cabin floor. The same source location, a point on the radiator, and the same standoff, a point on the bulkhead, are used for both excitations. These figures illustrate the typical effect of the diffuse excitation option—it produces a more even distribution of pressure in the cabin. This occurs because the incident pressure is divided into many waves, striking the bulkhead from different directions and resulting in a degree of cancellation and averaging of the pressure load on the surface. The results shown are obtained using the direct steady-state dynamics procedure; using the subspace projection steady-state dynamics procedure, the results appear nearly identical.

The frequency analysis performed on the cabin-air substructure generates eigenvalues identical to those from the model without substructures. Moreover, the eigenmodes obtained from the regular non-substructure model (*Figure 8*) and those recovered from the substructure model (*Figure 9*) compare very well (shown here for the air pressure for the 25th eigenmode).

The frequency response obtained for the cabin-air-chassis model is shown in *Figure 10* and *Figure 11*. Given the size of the model, the direct steady-state dynamics analysis is computationally less efficient and, thus, is not performed. In addition to the natural frequency extraction procedure for the whole structure, a frequency analysis is performed on the equivalent cabin-air-chassis model using two substructures. While the eigenfrequencies are not identical to those obtained from the regular non-substructure model, the differences are quite small for the range of interest, as shown in *Figure 12*. Once the substructures are generated, the analysis to extract eigenfrequencies from the two-element substructure model is hundreds of times faster than the analysis to extract them from the regular non-substructure model.

The mode-based and the subspace projection steady-state dynamics procedures in Abaqus demonstrate significant improvements in computational efficiency when compared to the direct steady-state dynamics approach. When damping is small or if it can be well approximated using modal damping coefficients, the mode-based procedures are extremely efficient. When damping is more complex, the subspace projection method also demonstrates significant computational advantage in comparison with the direct-integration approach.

The use of substructures is also demonstrated to produce significant gains in computational efficiency. The reduction of the acoustic volume and of its bounding structure to a substructure has clear advantages. The low-dimensional coupled acoustic-structural substructures are very computationally efficient, and the data for the acoustic response inside the substructure can be recovered when the global analysis is completed.

Input files

tr_acous_cabin_mode.inp	Mode-based and subspace projection steady-state dynamic analysis of the cabin-air model without damping.

tr_acous_cabin_mode_ams.inp	Mode-based steady-state dynamic analysis of the cabin-air model without damping and using Abaqus/AMS.
tr_acous_cabin_direct.inp	Direct steady-state dynamic analysis of the cabin-air model without damping.
tr_acous_cabin_sp_impedance.inp	Subspace projection steady-state dynamic analysis of the cabin-air model with damping.
tr_acous_cabin_direct_impedance.inp	Direct steady-state dynamic analysis of the cabin-air model with damping.
tr_acous_cabin_subspace_bulkhead.inp	Subspace projection steady-state dynamic analysis of the cabin-air model with damping and incident wave excitation on the bulkhead.
tr_acous_cabin_direct_bulkhead.inp	Direct steady-state dynamic analysis of the cabin-air model with damping and incident wave excitation on the bulkhead.
tr_acous_cabin_gen.inp	Cabin-air coupled substructure generation for the second model.
tr_acous_cabin_sub_freq.inp	Frequency analysis of the cabin-air model using one substructure.
tr_acous_cabin_chassis_gen.inp	Cabin-air substructure generation analysis for the fourth model.
tr_acous_chassis_gen.inp	Chassis substructure generation analysis for the fourth model.
tr_acous_cabin_chassis_sub.inp	Frequency analysis of the cabin-air-chassis model using two substructures.
tr_materials_acous.inp	All material definitions.
tr_cabin_air_w.inp	Interior air model.
tr_acous_chassis_coup.inp	Coupling definitions for the chassis.
tr_cabin_elements.inp	Element definitions for the cabin.
tr_cabin_elsets.inp	Element set definitions for the cabin.
tr_cabin_nodes.inp	Node definitions for the cabin.
tr_cabin_nsets.inp	Node set definitions for the cabin.
tr_cabin_sections.inp	Section definitions for the cabin.
tr_parameters_inphase.inp	Parameter definitions.
tr_parameters.inp	Parameter definitions.
tr_all_nodes.inp	All node definitions.
tr_cabin_elts.inp	Element definitions for the cabin.
tr_cabin_coup_steer_col.inp	Coupling definitions.
tr_cabin_mpc.inp	**MPC* definitions for the cabin.
tr_seat_elts.inp	Element definitions for the seat.
tr_seat_coup.inp	Coupling definitions for the seat.
tr_conn_seat.inp	Connector definitions for the seat.
tr_door_left_elts.inp	Element definitions for the left door.
tr_door_left_coup.inp	Coupling definitions for the left door.
tr_door_left_mpc.inp	**MPC* definitions for the left door.
tr_conn_door_left_nobehav.inp	Connector definitions for the left door.
tr_door_right_elts.inp	Element definitions for the right door.
tr_door_right_coup.inp	Coupling definitions for the right door.
tr_door_right_mpc.inp	**MPC* definitions for the right door.
tr_chassis_elts.inp	Element definitions for the chassis.

Figures

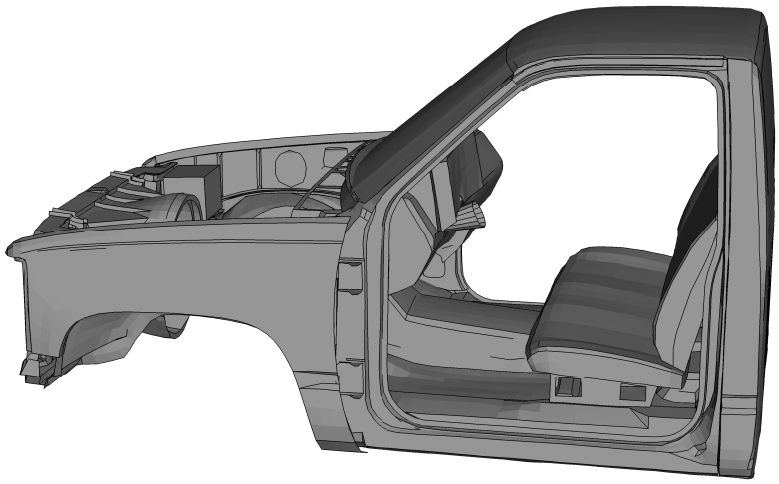


Figure 1: The cabin model (doors removed for clearer illustration).

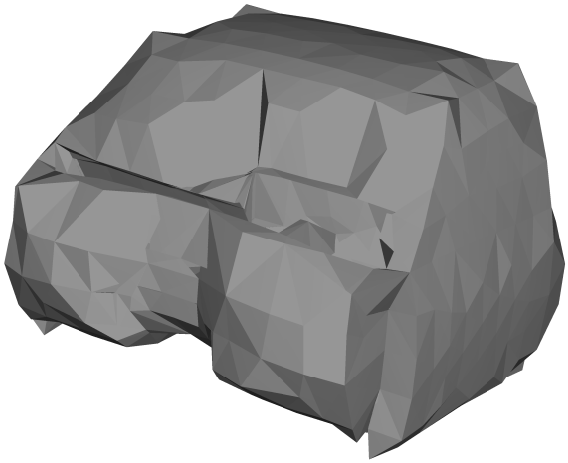


Figure 2: The air mesh inside the cabin.

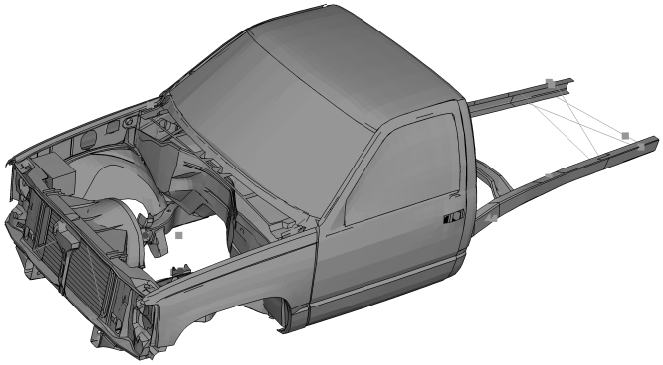


Figure 3: The cabin-air-chassis model.

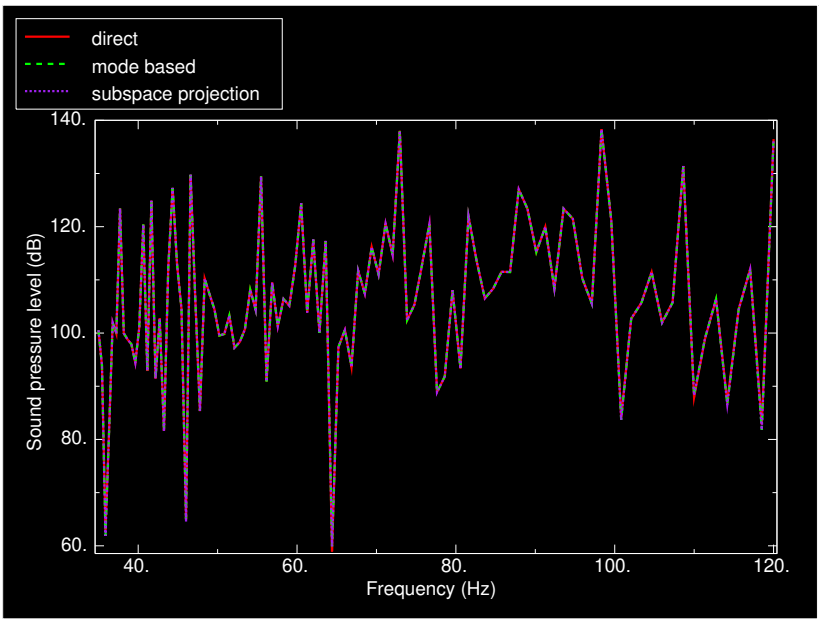


Figure 4: Sound pressure level at the ear position for the cabin-air model (no damping).

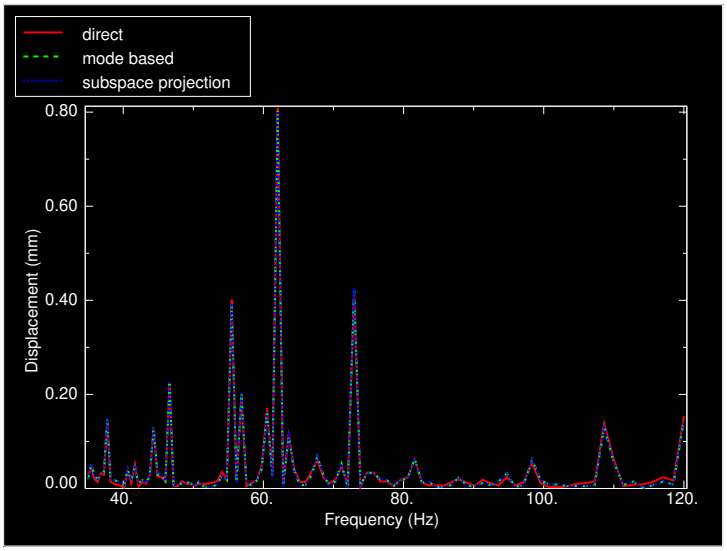


Figure 5: Displacement at one of the harmonically excited cabin floor points (no damping).

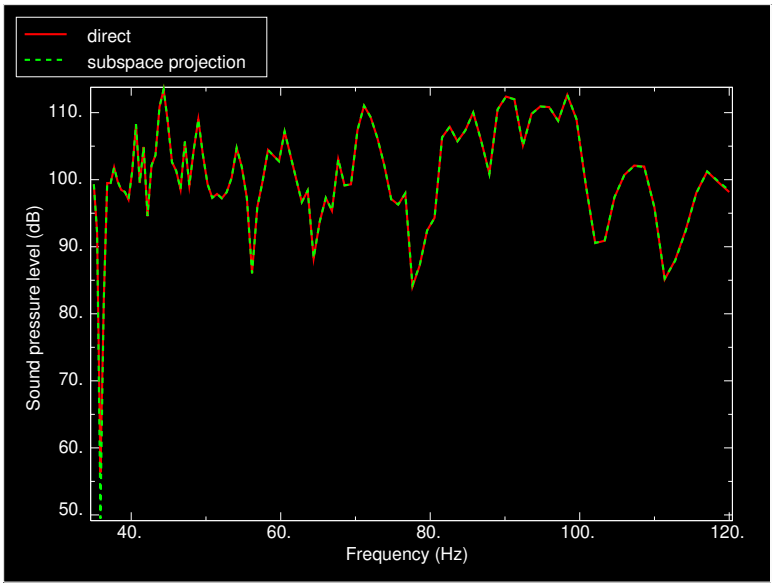


Figure 6: Sound pressure level at the ear position for the cabin-air model when damping is considered.

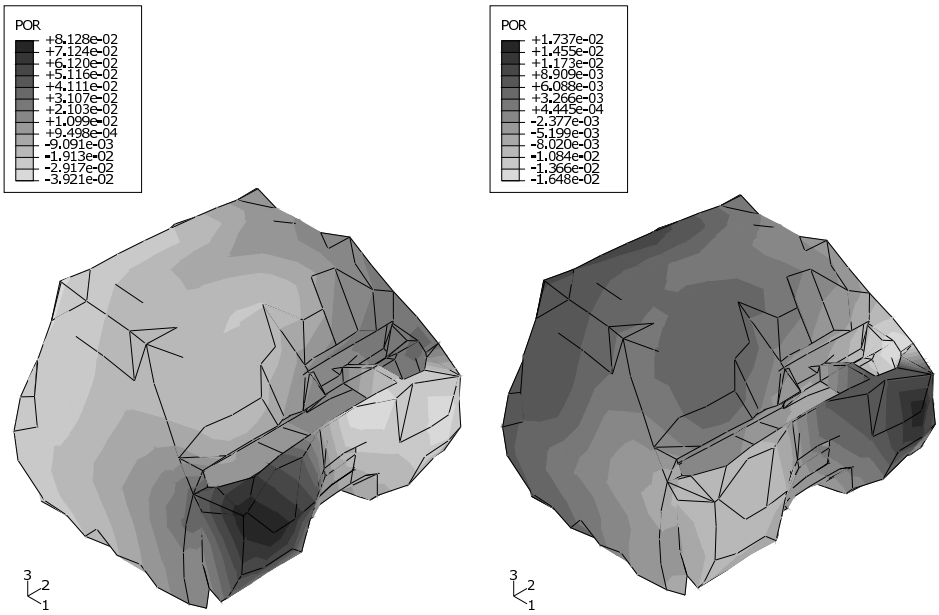


Figure 7: Real part of the acoustic pressure in the cabin-air model when damping is considered and when the lower bulkhead is excited with an incident pressure field (planar wave case, left; diffuse field case, right).

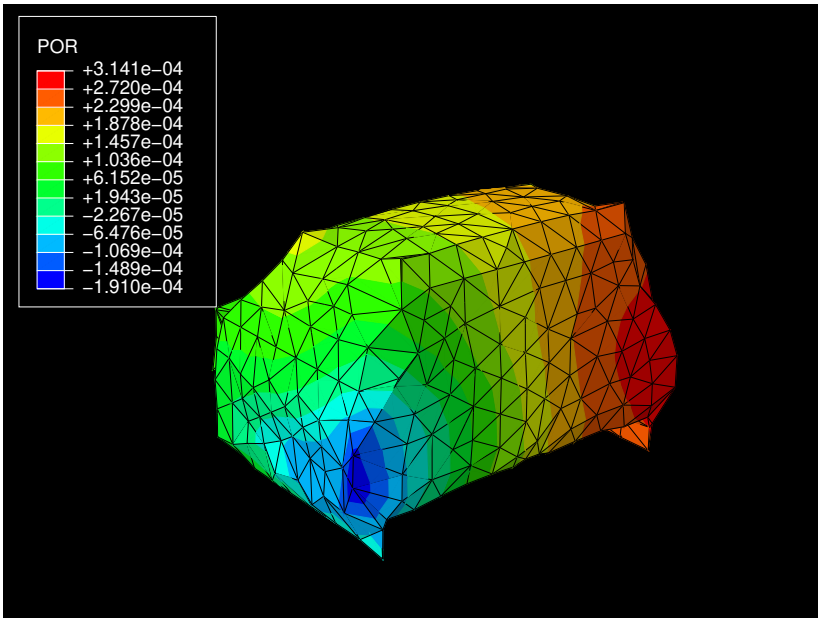


Figure 8: Air pressure for the 25th eigenmode (35.131 Hz) from the cabin-air model.

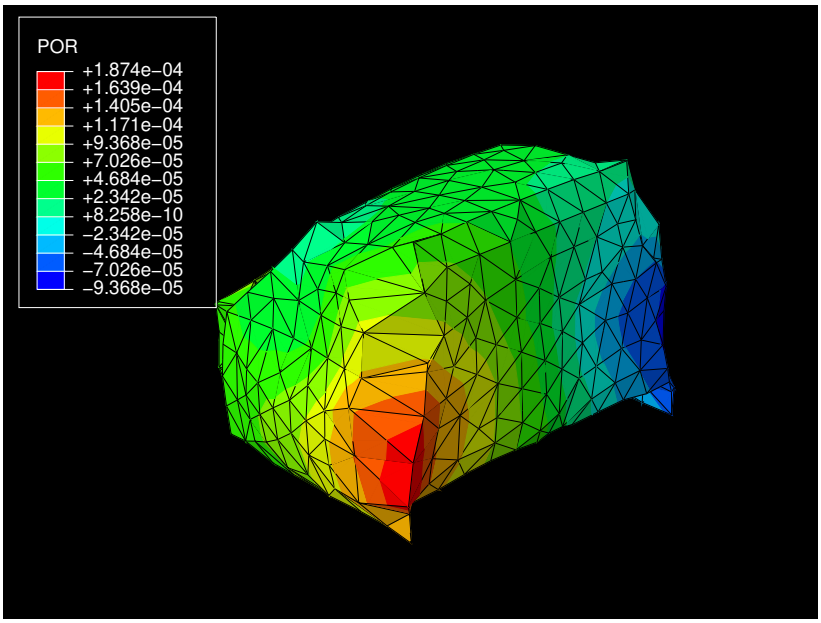


Figure 9: Air pressure for the 25th eigenmode (35.131 Hz) from the cabin-air substructure model.

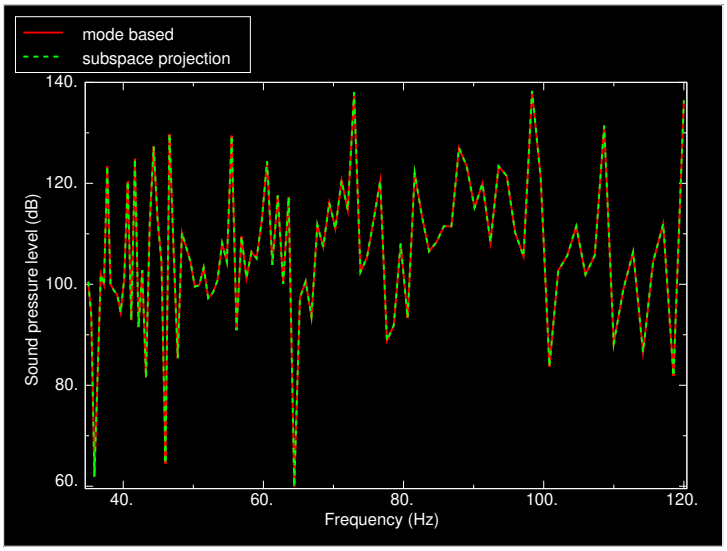


Figure 10: Sound pressure level at ear level from the cabin-air-chassis model (no damping).

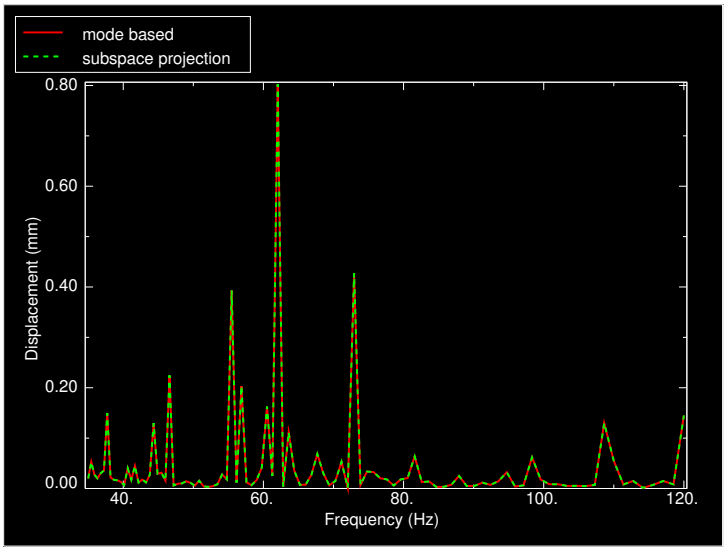


Figure 11: Displacement at one of the cabin floor points for the cabin-air-chassis model (no damping).

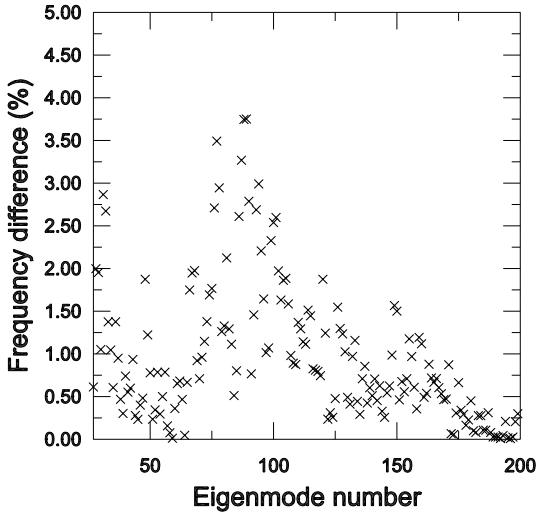


Figure 12: Eigenfrequency differences between the cabin-air-chassis model and the equivalent model with substructures for the range of interest (35–120 Hz).

Long-duration response of a submerged cylinder to an underwater explosion

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example demonstrates how Abaqus can be used to predict the long-duration response of submerged structures that experience loading by a wave resulting from an underwater explosion (UNDEX).

An emphasis on structural dynamic motions naturally leads to the use of beams to model the structure, rather than solid or shell elements. This class of problem is characterized by structural dynamic motions at speeds much slower than the acoustic wave speed in the fluid, so that the fluid can be modeled as an incompressible medium. The implication of modeling the fluid in this manner is to reduce its effect on the structure to an “added mass” on the beam.

The spherical pressure waves associated with an underwater explosion (UNDEX) event are characterized by two distinct phases. The first, very short, phase is the initial wave produced by the detonation. It involves a very steep rise to a characteristic pressure value, followed by a more gradual decay. In the second phase of the loading the gas produced by the explosive expands to a maximum volume, at which the pressure of the surrounding fluid forces it back upon itself. At some minimum volume the gas and fluid system emits another pressure pulse, and the gas bubble expands again. This process may repeat many times, causing several pressure pulses. As the gas bubble oscillates, it also acts under the effects of buoyancy, causing an unsteady motion opposite to the force of gravity.

Problem and geometry description

This problem involves a structural-dynamics model of a submerged submarine under athwartships attack from an underwater explosion. The ship is 100 m in length and 50 m below the surface; and the detonation point of the charge is centered along the ship's length, 15 m to one side and 15 m below the ship. The response of the ship to the initial direct and reflected shock waves as well as to the first few bubble pulses is of interest, so dynamic simulations are carried out to 5 seconds.

Model

The model is made of 100 B31 beam elements, arrayed along a line. Their (uniform) section properties are defined using a general beam section. The structure has an overall length of 50 m. Point mass elements of 10000 kg are defined at each node to simulate the effect of internal equipment on the beam structural dynamics. The effect of the entrained fluid is simulated using beams with additional inertia, with which a fluid mass density of 1025 kg/m^3, an outside radius of 5 m, and a fluid drag coefficient of 1.0 are specified. Structural damping is specified using a material damping definition. No additional acoustic fluid elements or absorbing boundary conditions are required.

Fluid-structure coupling and shock wave loading

The loading specification for this problem includes descriptions of the explosive charge, the fluid medium in which the wave propagates to the structure, and the geometry of the charge with respect to the structure.

In Abaqus the time histories of pressure, its derivatives, and the motion of the explosive gas bubble are defined using the Geers-Hunter model. This model is invoked using the bubble-type amplitude. Under this option material properties of the explosive, its mass, its distance from the free surface, and some other control parameters are specified. The data on this option are used to govern a separate bubble dynamics time integration operation, performed as part of the preprocessing. Parameters defined on this option do not affect the rest of the analysis. Here a charge of 100 kg is used, with model parameters set to suppress wave loss effects within the bubble

simulation. An initial depth of 65 m is specified: this affects the oscillation of the gas bubble and the duration of the bubble dynamics, since the solution naturally terminates when the bubble reaches the free surface. In this analysis, however, the bubble simulation time is cut off at 0.6 seconds. The bubble migration is defined to be along the z-axis. Default values for the bubble dynamics time integration parameters are used.

The actual loads on the structure are defined using the incident wave and the associated incident wave reflection. The incident wave defines the distributed time-varying loads within an analysis step on the structural surface, due to the specified parameters. Only the surface defined for the beam elements and the reference load magnitude need to be indicated, as is the case for most distributed loads in Abaqus. The incident wave reflection defines any planes outside the computational domain for the purpose of calculating additional incident wave loads due to reflections. Here a "soft" (zero total pressure) reflecting plane is defined, located 65 m from the original position of the source and oriented normal to the z-axis. The original position of the source point is defined as (50, 15, −15), and the standoff point is defined as (50, 3.536, −3.536). The fluid properties, used for propagation of the wave across the structure, are given as mass density ρ_f = 1025 kg/m^3 and bulk modulus K_f = 2.30635 GPa.

Results and discussion

The model for this UNDEX example has a total of 606 active degrees of freedom and requires approximately 15 MB of memory and 267 KB of disk space.

Figure 1 shows the time history of vertical displacement (V3) for the center node of the structure using a logarithmic time axis. This curve clearly shows the initial shock-induced velocity peak, the velocity peak caused by the reflected path, and the decaying periodic response associated with a structural motion after the loading ceases at t = 0.6. The response curve clearly illustrates that there is a rigid body translation of the cylinder, due to the velocity induced by the initial shocks. The peaks due to the direct and reflected shocks are of the same sign, since the reflected wave, with a negative sign, is traveling in the opposite vertical direction from the direct wave.

Figure 2 shows the strain along the axis of the beam for the section points oriented along the 1-direction of the section. The 1-direction is also the y-direction in the global system. The curves suggest that a dominant mode of vibration occurs at about 2.1 Hz (based on an estimated period of 0.48 seconds). *Figure 3* shows the axial strain for section points oriented in the section 2-direction, or the global z-direction. Again, the two peaks corresponding to the incident shocks are evident, followed by the decaying oscillation at roughly 2.1 Hz.

Input files

iw_exa_whip_std.inp	Abaqus/Standard analysis of a submerged cylinder subjected to an UNDEX shock wave.
iw_exa_whip_xpl.inp	Abaqus/Explicit analysis of a submerged cylinder subjected to an UNDEX shock wave.

References

- Hicks, A. N., "The Theory of Explosion Induced Hull Whipping," *Naval Construction Research Establishment, Dunfermline, Fife, Scotland, Report NCRE/R579, March 1972.*

Figures

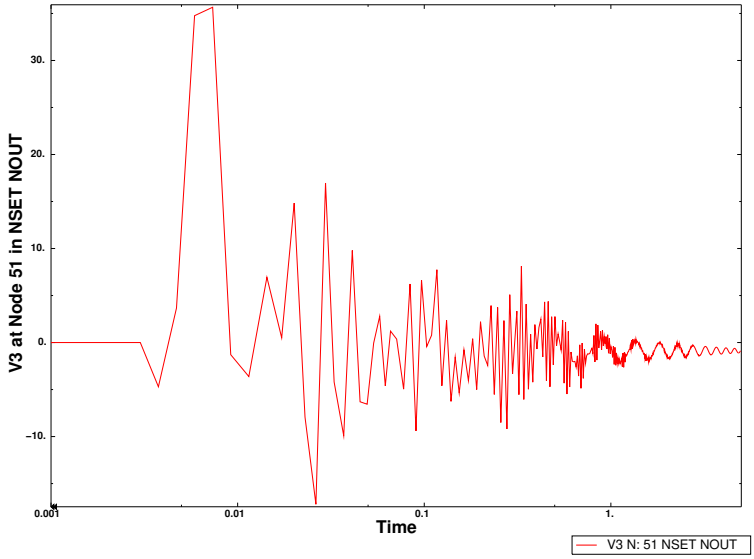


Figure 1: Vertical velocity at ship midpoint.

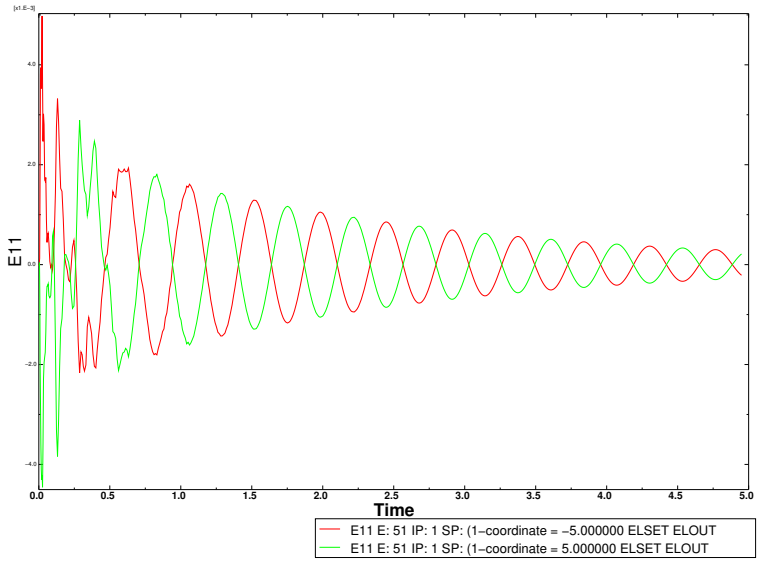


Figure 2: Strain at horizontal extrema at ship midpoint.

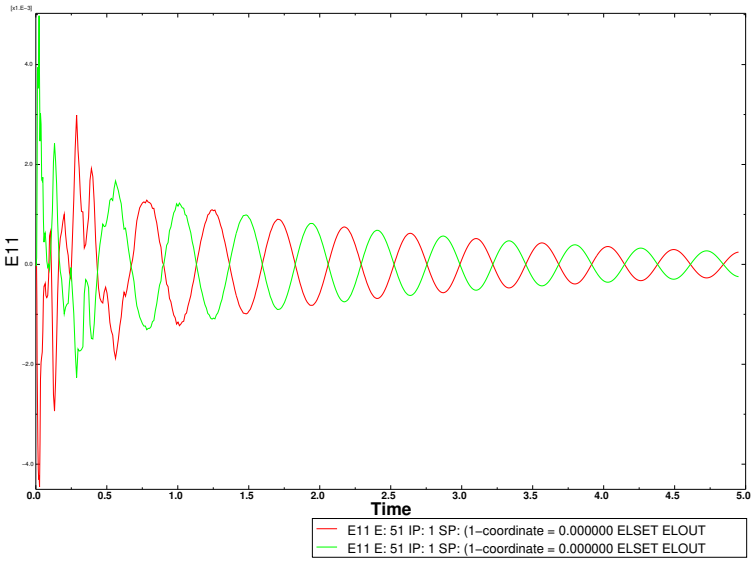


Figure 3: Strain at vertical extrema at ship midpoint.

Deformation of a sandwich plate under CONWEP blast loading

Products: Abaqus/Explicit

References:

- *Coupled Acoustic-Structural Analysis*
- *Acoustic and Shock Loads*
- **CONWEP CHARGE PROPERTY*
- **INCIDENT WAVE INTERACTION*
- **INCIDENT WAVE INTERACTION PROPERTY*
- *CONWEP blast loading pressures*
- *Blast loading of a circular plate using the CONWEP model*

Overview

This example shows how to build better structures that resist blast loads by using the blast loading capability in Abaqus/Explicit.

The following Abaqus features are demonstrated:

- applying CONWEP blast loading,
- comparing computational results using Abaqus/Explicit and experimental measurements of deformation of sandwich structures under blast loads, and
- demonstrating a generic example of a nonlinear analysis of a sandwich structure.

Application description

Modern structural designs can be used to protect various systems that are vulnerable to large blast loads. Sandwich structures are a particular class of modern structures that could be explored for such applications.

Geometry

Two models are used in this example. One model is the sandwich plate structure described by Dharmasena et al. (2008) and shown in *Figure 1*. The sandwich structure consists of a square honeycomb core with vertical webs welded to top and bottom plates. The dimensions of the overall sandwich plate structure are 610 × 610 × 61 mm. The sandwich structure lies in the *X*–*Y* plane, while the blast source is 100 mm vertically above (along the *z*-direction) the center of the top plate of the sandwich structure. The top and bottom plates are 5 mm thick, and the square honeycomb core webs are 0.76 mm thick. The spacing between the honeycomb webs is 30.5 mm measured from the midplane.

The second model is a solid plate, which was chosen for comparison with Dharmasena et al.

Materials

The top and bottom plates and honeycomb core of the sandwich structure and the solid plate are all made of a high ductility stainless steel alloy (Al-6XN) comprised of 49% Fe, 24% Ni, 21% Cr, and 6% Mo by weight as described by Nahshon et al. (2007).

Initial conditions

All structures are initially at their undeformed state at 273 K temperature.

Boundary conditions and loading

All the edges of the sandwich structure and the solid plate are fixed. One-quarter of the full plate is modeled, assuming symmetry of the solution.

Interactions

In the sandwich structure the honeycomb web is welded to the inner surfaces of the plates.

Abaqus modeling approaches and simulation techniques

This example demonstrates the usage of CONWEP blast loading using Abaqus/Explicit. This example was chosen based on experiments reported by Dharmasena et al. (2008) that discuss the deformation of a particular sandwich structure and an equivalent solid plate subjected to CONWEP blast loads due to 1, 2, and 3 kg of TNT. The solid plate with equivalent material is presented as a simple example. The solid plate is modeled using shell elements and three-dimensional continuum elements. Finally, the main problem of interest, the sandwich structure described by Dharmasena et al., is analyzed. A tonne-millimeter-second-kelvin unit system was chosen for all simulations.

Summary of analysis cases

Case 1	Solid plate modeled with S4R shell elements under 1 kg TNT blast load.
Case 2	Solid plate modeled with S4R shell elements under 2 kg TNT blast load.
Case 3	Solid plate modeled with S4R shell elements under 3 kg TNT blast load.
Case 4	Solid plate modeled with C3D8R continuum elements under 1 kg TNT blast load.
Case 5	Solid plate modeled with C3D8R continuum elements under 2 kg TNT blast load.
Case 6	Solid plate modeled with C3D8R continuum elements under 3 kg TNT blast load.
Case 7	Sandwich structure under 1 kg TNT blast load.
Case 8	Sandwich structure under 2 kg TNT blast load.
Case 9	Sandwich structure under 3 kg TNT blast load.

Analysis types

Dynamic analysis using Abaqus/Explicit is performed for all cases. Solutions are computed up to 1.5 milliseconds, where no further permanent deformation is observed for all load values.

Materials

The mechanical properties of the steel alloy as described in Nahshon et al. (2007) are specified as follows: Young's modulus of 1.61×10^5 MPa, Poisson's ratio of 0.35, density of 7.85×10^{-9} metric tons/mm^3, and coefficient of expansion of 452×10^6 Nmm/metric tons·K.

A Johnson-Cook model is used to model the elastic-plastic behavior with the following coefficients and constants: $A = 400$ MPa, $B = 1500$ MPa, $C = 0.045$, $n = 0.4$, $m = 1.2$, and $\dot{\varepsilon_0} = 0.001$ s^{-1}. The transition temperature is 293 K, and the melting temperature is 1800 K.

Initial conditions

The temperature is initialized to 273 K for all nodes in each model.

Boundary conditions

Symmetry behavior is assumed. Only one-quarter of the structure is modeled for all cases, with the center of the plate positioned at the origin of the *X*–*Y* plane. The boundaries at $X = 305$ mm and $Y = 305$ mm are fixed for all degrees of motion (ENCASTRE). Symmetry conditions about the *x*-axis (XSYMM) are applied at the plane $X = 0$. Similarly, symmetry conditions about the *y*-axis (YSYMM) are applied at the plane $Y = 0$.

Loads

The CONWEP blast load is applied on the top surface of the plate. The source of the blast is at a standoff distance of 100 mm vertically above the center of the top surface of the plate. The property of the blast load is specified using the incident wave interaction property and the CONWEP charge property at the model level and the incident wave interaction at the step level.

Analysis steps

Each analysis consists of a single dynamic explicit step.

Output requests

Translational degrees of freedom (UT) are requested at the center of the plate.

Results and discussion

The center displacement after 1.5 milliseconds was monitored to compare each case with experimental results. Single and double precision job execution gave similar results for all cases.

The animation of the deformed plate over the entire time period of 1.5 milliseconds shows large deformations at the center. The plate stabilizes after a few oscillations. Comparison of the total work done history (ALLWK) and the total plastic dissipation history (ALLPD) indicates that most of the work done by the blast load is dissipated in plastic deformation.

Cases 1–3: Solid plate with shell elements

The solid plate model is modeled with shell elements and is subjected to CONWEP blast loading using different charge masses.

Mesh design

The plate surface is discretized using 31 × 31 S4R elements with nine integration points through the thickness of each element.

Boundary conditions

All degrees of freedom including the rotational degrees of freedom are fixed at the $X = L$ and $Y = L$ edges boundaries, where $L = 305$ mm. Symmetry boundary conditions (XSYMM) are applied at the $X = 0$ edge, and symmetry boundary conditions (YSYMM) are applied at the $Y = 0$ edge.

Loads

1, 2, and 3 kg TNT loads are used in Cases 1, 2, and 3, respectively. The blast source is kept at a standoff distance of 100 mm from the midsection of the shell elements.

Results and discussion

The artificial energy history (ALLAE) is significantly lower than the total internal energy (ALLIE), indicating that the solution is trustworthy with minimal artificial effects.

Cases 4–6: Solid plate with 3D continuum elements

The solid plate is modeled with three-dimensional continuum elements and is subjected to CONWEP blast loading using different charge masses.

Mesh design

The plate surface is discretized using 31×31 C3D8R elements with five layers of elements through the thickness of the plate.

Boundary conditions

All degrees of freedom including the rotational degrees of freedom are fixed at the $X = L$ and $Y = L$ face boundaries, where $L = 305$ mm. *X*-symmetry boundary conditions are applied at the $X = 0$ face, and *Y*-symmetry boundary conditions are applied at the $Y = 0$ face.

Loads

1, 2, and 3 kg TNT loads are used in Cases 4, 5, and 6, respectively. The blast source is kept at a standoff distance of 100 mm from the top surface of the plate.

Output requests

History output of the center deflection at the top and the bottom surfaces of the plate is requested at every increment.

Results and discussion

Similar behavior using C3D8R elements was observed as that observed using S4R elements. The top surface center deflection and the bottom surface center deflection were quite close to each other. The mean of the two values was used to compare the midsection deflection of the solid plate.

Cases 7–9: Sandwich plate structure

The sandwich plate structure is modeled using three-dimensional continuum elements for the top and bottom plates and shell elements for the square honeycomb core and is subjected to CONWEP blast loading using different charge masses.

Mesh design

The plate surface is discretized using 31×31 C3D8R elements with five layers of elements through the thickness of the plate. The honeycomb core is meshed using 30 layers of S4R shell elements along the height of the core with five integration points through their thickness.

Boundary conditions

All degrees of freedom including the rotational degrees of freedom are fixed at the $X = L$ and $Y = L$ face boundaries, where $L = 305$ mm. *X*-symmetry boundary conditions are applied at the $X = 0$ face, and *Y*-symmetry boundary conditions are applied at the $Y = 0$ face.

Loads

1, 2, and 3 kg TNT loads are used in Cases 7, 8, and 9, respectively. The blast source was kept at a standoff distance of 100 mm from the top surface of the plate.

Constraints

The edges of the shell elements at the top and bottom of the core are attached to the inner surfaces of the top plate and the bottom plate, respectively, using tie constraints. The shell element edges form the node-based secondary surface to the main plate surfaces. The tie constraint is defined without allowing any adjustments and using the node set of the edges to identify the nodes on the secondary surface that will be tied to the main surface.

Interactions

General contact is specified at the step level, including all exterior surface contact interactions.

Output requests

History output of the center deflection at the top and the bottom surfaces of the plate is requested at every increment.

Results and discussion

The analysis demonstrates significant buckling of the honeycomb webs involving self-contact near the center of the plate, as shown in *Figure 2* for the three loading cases. *Figure 3* shows the internal view for the 1 kg TNT blast charge case, revealing the deformation of the honeycomb core. It also confirms that tie constraints using node-based surfaces capture appropriately the weld between the webs and the inner surfaces of the plates.

Additional cases

Studies of similar structures using different parameters have been documented in the literature. Alternative material models and loading cases are mentioned here to provide examples for additional analyses. Different material models have been used to simulate the material behavior under such loads. The material model for the following atypical strain-rate hardening (Dharmasena et al., 2008) can be used for simulation of the above cases with user subroutine *VUHARD*:

$$\sigma = \begin{cases} E\varepsilon, & \varepsilon \le \dfrac{\sigma_Y}{E}\left(1+\left(\dfrac{\dot{\varepsilon_p}}{\dot{\varepsilon_0}}\right)^m\right) \\ \sigma_Y\left(1+\left(\dfrac{\dot{\varepsilon_p}}{\dot{\varepsilon_0}}\right)^m\right) + E_t\left(\varepsilon - \dfrac{\sigma_Y}{E}\left(1+\left(\dfrac{\dot{\varepsilon_p}}{\dot{\varepsilon_0}}\right)^m\right)\right), & \varepsilon > \dfrac{\sigma_Y}{E}\left(1+\left(\dfrac{\dot{\varepsilon_p}}{\dot{\varepsilon_0}}\right)^m\right) \end{cases}$$

The material properties used by Dharmasena et al. and Rathbun et al. (2006) are as follows: Young's modulus of $E = 2.00 \times 10^5$ MPa, Poisson's ratio of $\nu = 0.30$, density of $\rho = 7.85 \times 10^{-9}$ metric tons/mm^3, yield stress of $\sigma_Y = 300.0$ MPa, tangent modulus of $E_t = 2.00 \times 10^3$ MPa, $\dot{\varepsilon_0} = 4916$ s^{-1}, and $m = 0.154$.

In addition, the following approximate loading, which was applied on the sandwich structure described by Dharmasena et al., can be implemented with user subroutine *VDLOAD*:

$$p(t) = (p_s - p_a)\left(\frac{t_d - t}{t_d - t_a}\right) e^{-\left(\frac{t-t_a}{\theta}\right)} e^{-\left(\frac{d}{d_0}\right)^2},$$

where p_s is the shock wave pressure (360 MPa), p_a is the ambient pressure (0), t_d is the time when the shock decays to a value close to zero, t_a is the shock arrival time, d is the distance from the center of the plate (in this example, also the origin), and d_0 is a reference distance of 120 mm. All variable values are chosen based on Dharmasena et al.

Discussion of results and comparison of cases

The Abaqus/Explicit results for the solid plate using the Johnson-Cook model for the steel alloy (Cases 1–6) are shown in *Table 1*. These results compare well with the experimental results reported by Dharmasena et al. (2008), shown in *Table 2*, for both the shell and continuum elements with CONWEP blast loading. The Abaqus/Explicit results for the sandwich structure compare within reasonable error with the experimental results. The simulation and experimental results are presented graphically in *Figure 4*. The solution differs significantly for higher loads, which Dharmasena et al. attribute to boundary conditions. At higher loads, it is likely that the edges of the sandwich panel used in the test arrangement were actually more flexible than the clamped condition used in the Abaqus model, causing differences between the numerical and experimental results. Differences in the results could also be due to debonding of the honeycomb core webs from the top and bottom plates in the experimental setup.

The sandwich structure was modeled for the same CONWEP loading with a different material model. An isotropic bilinear model with an atypical strain-rate hardening used user subroutine *VUHARD* with the material properties as described in "Additional cases" above. The displacements, shown in *Table 3*, were found to be much higher than the Johnson-Cook model and the experimental results, probably due to lower yield stress values.

In addition, the solid plate and the sandwich structure were modeled with the Johnson-Cook material model with an approximate loading, as described in "Additional cases" above, with user subroutine *VDLOAD*. The displacements, shown in *Table 4*, were found to be lower than the CONWEP blast loading and the experimental results, probably because less work was done on the model by the approximate load over the total time.

Input files

Cases 1–3 Solid plate with shell elements

exa_aco_conwep_sol_1kg.inp	Solid plate model with S4R elements and 1 kg TNT blast load.
exa_aco_conwep_sol_2kg.inp	Solid plate model with S4R elements and 2 kg TNT blast load.
exa_aco_conwep_sol_3kg.inp	Solid plate model with S4R elements and 3 kg TNT blast load.

Cases 4–6 Solid plate with continuum elements

exa_aco_conwep_spl_1kg.inp	Solid plate model with C3D8R elements and 1 kg TNT blast load.
exa_aco_conwep_spl_2kg.inp	Solid plate model with C3D8R elements and 2 kg TNT blast load.
exa_aco_conwep_spl_3kg.inp	Solid plate model with C3D8R elements and 3 kg TNT blast load.

Cases 7–9 Sandwich structure

exa_aco_conwep_snd_1kg.inp	Sandwich structure model with 1 kg TNT blast load.
exa_aco_conwep_snd_2kg.inp	Sandwich structure model with 2 kg TNT blast load.
exa_aco_conwep_snd_3kg.inp	Sandwich structure model with 3 kg TNT blast load.

Additional cases

exa_aco_conwep_snd_vuhard_1kg.inp	Sandwich structure model with user-defined material model and 1 kg TNT blast load.
exa_blast_vuhard.f	User subroutine *VUHARD* with user-defined strain-rate hardening model.
exa_aco_conwep_sol_vdload_1kg.inp	Solid plate model with user-defined approximate loading equivalent to 1 kg TNT blast.
exa_aco_conwep_snd_vdload_1kg.inp	Sandwich structure model with user-defined approximate loading equivalent to 1 kg TNT blast.
exa_blast_vdload.f	User subroutine *VDLOAD* with the approximate loading equivalent to 1 kg TNTCONWEP blast load.

References

- Dharmasena, K. P., H. N. G. Wadley, Z. Xue, and J. W. Hutchinson, "Mechanical Response of Metallic Honeycomb Sandwich Panel Structures to High-Intensity Dynamic Loading," Journal of Impact Engineering, vol. 35, pp. 1063–1074, 2008.
- Nahshon, K., M. G. Pontin, A. G. Evans, J. W. Hutchinson, and F. W. Zok, "Dynamic Shear Rupture of Steel Plates," Journal of Mechanics of Materials and Structures, vol. 2–10, pp. 2049–2066, December 2007.
- Rathbun, H. J., D. D. Radford, Z. Xue, M. Y. Hu, J. Yang, V. Deshpande, N. A. Fleck, J. W. Hutchinson, F. W. Zok, and A. G. Evans, "Dynamic Shear Rupture of Steel Plates," International Journal of Solids and Structures, vol. 43, pp. 1746–1763, 2006.

Tables

Table 1: Center deflection computed by Abaqus/Explicit for Cases 1–9.

Model	Charge mass	Center deflection (mm)		
		Top surface	Midsection	Bottom surface
Solid plate (S4R)	1 kg TNT	–	48.54	–
	2 kg TNT	–	88.38	–
	3 kg TNT	–	109.75	–
Solid plate (C3D8R)	1 kg TNT	–	47.09	–
	2 kg TNT	–	85.72	–
	3 kg TNT	–	108.38	–
Sandwich structure	1 kg TNT	69.15	–	26.15
	2 kg TNT	110.68	–	66.14
	3 kg TNT	141.37	–	96.63

Table 2: Center deflection measurements from experiments.

Model	Charge mass	Center deflection (mm)		
		Top surface	Midsection	Bottom surface
Solid plate	1 kg TNT	–	37.65	–

Model	Charge mass	Center deflection (mm)		
		Top surface	Midsection	Bottom surface
	2 kg TNT	–	71.37	–
	3 kg TNT	–	132.94	–
Sandwich structure	1 kg TNT	47.06	–	15.29
	2 kg TNT	98.82	–	53.73
	3 kg TNT	158.04	–	127.45

Table 3: Center deflection computed by Abaqus/Explicit with user subroutine `VUHARD`.

Model	Charge mass	Center deflection of top surface (mm)	Center deflection of bottom surface (mm)
Sandwich structure with *VUHARD*	1 kg TNT	96.18	50.74
	2 kg TNT	156.76	114.79
	3 kg TNT	200.03	155.08

Table 4: Center deflection computed by Abaqus/Explicit with user subroutine `VDLOAD`.

Model	Charge mass	Center deflection (mm)		
		Top surface	Midsection	Bottom surface
Solid plate (S4R) with *VDLOAD*	1 kg TNT	–	32.10	–
Sandwich structure with *VDLOAD*	1 kg TNT	36.31	–	18.89

Figures

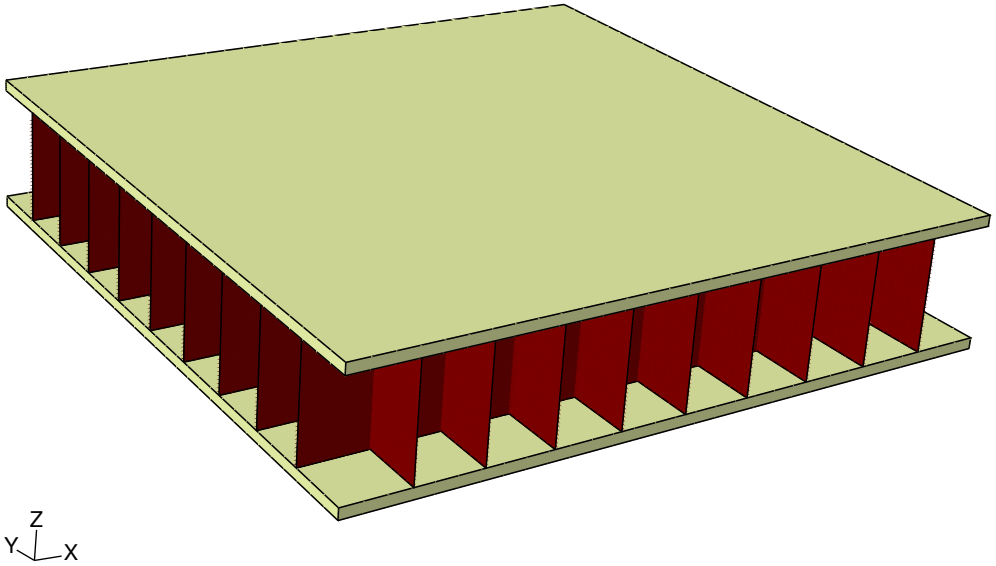


Figure 1: The sandwich structure model.

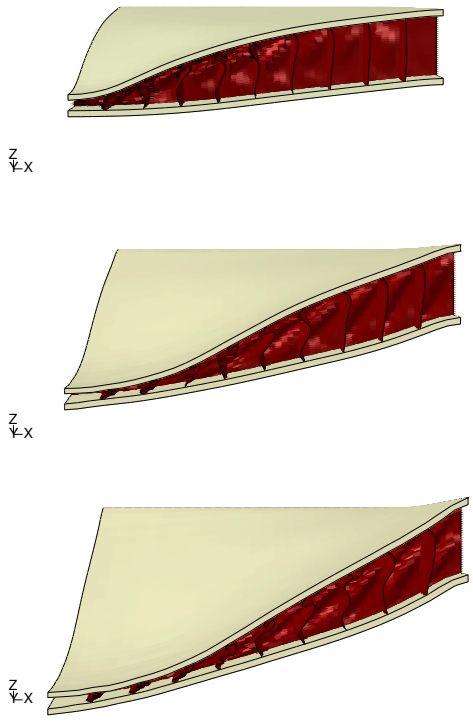


Figure 2: Deformed sandwich structure under CONWEP blast loads due to (from top to bottom) 1, 2, and 3 kg TNT.

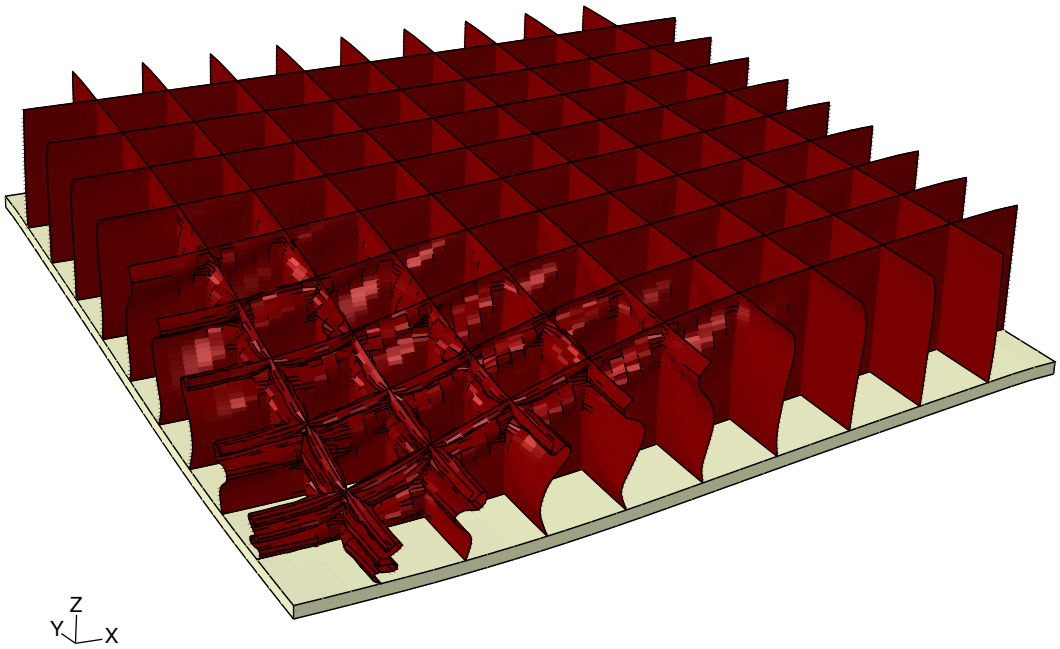


Figure 3: Deformation of the honeycomb core for a CONWEP charge of 1 kg TNT.

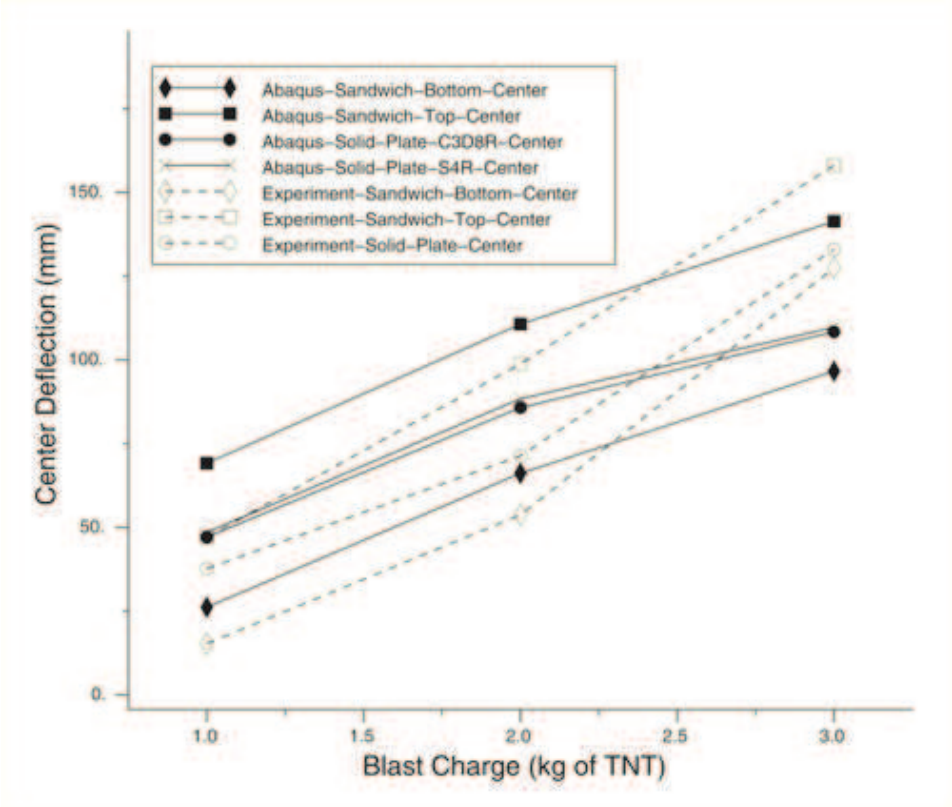


Figure 4: Comparison of center deflection between Abaqus/Explicit and experimental results for different blast charges.

Abaqus/Aqua analyses

In this section:

- *Jack-up foundation analyses*
- *Riser dynamics*

Jack-up foundation analyses

Products: Abaqus/Standard Abaqus/Aqua

Overview

This example simulates a jack-up rig on a sand foundation subjected to alternating wind loading.

Geometry and model

The model—a simplified planar model for the analysis of a multiple leg, portal frame-type structure—is intended for the analysis of 3-leg jack-up rigs with shallow foundation supports. *Figure 1* is a schematic of a 3-leg jack-up, as represented by the model. The jack-up hull is assumed to be rigid and triangular, and the connection between the hull and the legs is also taken to be rigid. The jack-up has two windward legs and one leeward leg; the model is projected onto the vertical symmetry plane that passes through the leeward leg and between the windward legs. Elastic beam columns are used to model both the upper and lower segment of each leg. The soil model is chosen to be macro-yield sand. Three degrees of freedom—vertical, horizontal, and rotational—are assumed at each spud can at the base of each leg. Mass is assumed to be concentrated at the center of the hull. The horizontal degree of freedom at the center is assumed to represent the motion of the rig for analysis purposes. Wind loading on the rig is applied as a horizontal force above the center of gravity of the hull.

The leg segments are modeled using B21 elements, and the general beam section is used to define the structural properties of the beam. The interaction between the spud can and the soil is modeled through JOINT2D elements and the joint elasticity and joint plasticity definitions. Rigid beam elements, RB2D2, are used to model the rigid hull.

The dimensions of the rig and the material properties of the sand and the spud can are as follows (force units are in kN, and length units are in meters):

Leg length upper segment	49.4
Leg length lower segment	13.5
Leg EI upper segment	2.7×10^8
Leg EI lower segment	2.7×10^9
Leg AE upper segment	2.2×10^8
Leg AE lower segment	2.2×10^9
Leg GA upper segment	8.1×10^7
Leg GA lower segment	8.1×10^8
Horizontal distance from platform center of gravity to leeward leg	23.4
Horizontal distance from platform center of gravity to windward leg	11.7
Spud can diameter	10.9
Spud can cone angle	180°
Foundation preload per spud can	50600
Foundation tensile capacity	0
Operational vertical load (weight)	62700
Vertical distance from center of gravity	

to load application point	7.1
Soil submerged unit weight	10.0
Soil friction angle	33°
Soil Poisson's ratio	0.2
Foundation elastic shear moduli, G_{vv}	5.14×10^4
G_{hh}	3.87×10^3
G_{rr}	2.04×10^3
Constant coefficient, Λ_1	1.0
Constant coefficient, Λ_2	0.5

Boundary conditions and loading

The base nodes of the JOINT2D elements are always fixed. The required preload is applied to each spud can using an initial condition. In the first step the weight loading is applied at the center of gravity of the hull. The rig is then subjected to an alternating horizontal wind loading applied at the specified location above the center of gravity of the hull.

The rig is loaded from zero to 5370 kN, unloaded to zero and then to 6440 kN in the opposite direction, reloaded to 9130 kN in the initial direction, unloaded and reloaded to 9770 kN in the opposite direction, and unloaded to zero again. Each of these loadings is done in a separate step and is ramped from zero to the specified magnitude at the end of the step.

Results and discussion

The estimated load path for the leeward spud can foundation is plotted in a graph of equivalent horizontal load, $R = \sqrt{(M/D)^2 + \Lambda_1 H^2}$, versus V/V_c. The plot is shown in *Figure 2* and is in good agreement with the load path predicted by an independent analysis, as detailed in the reference below. The moment-horizontal load response (i.e., M/D versus H) for the leeward spud can foundation, shown in *Figure 3*, compares well with the independent analysis.

Input files

jackup.inp	Input data for this example.

References

- Wong, P. C. and J. D. Murff, “Dynamic Analysis of Jack-Up Rigs Using Advanced Foundation Models,” Proceedings, 13th International Conference on Offshore Mechanics and Arctic Engineering (OMAE), vol. 2 - Safety and Reliability, Houston, pp. 93–109, February 1994.

Figures

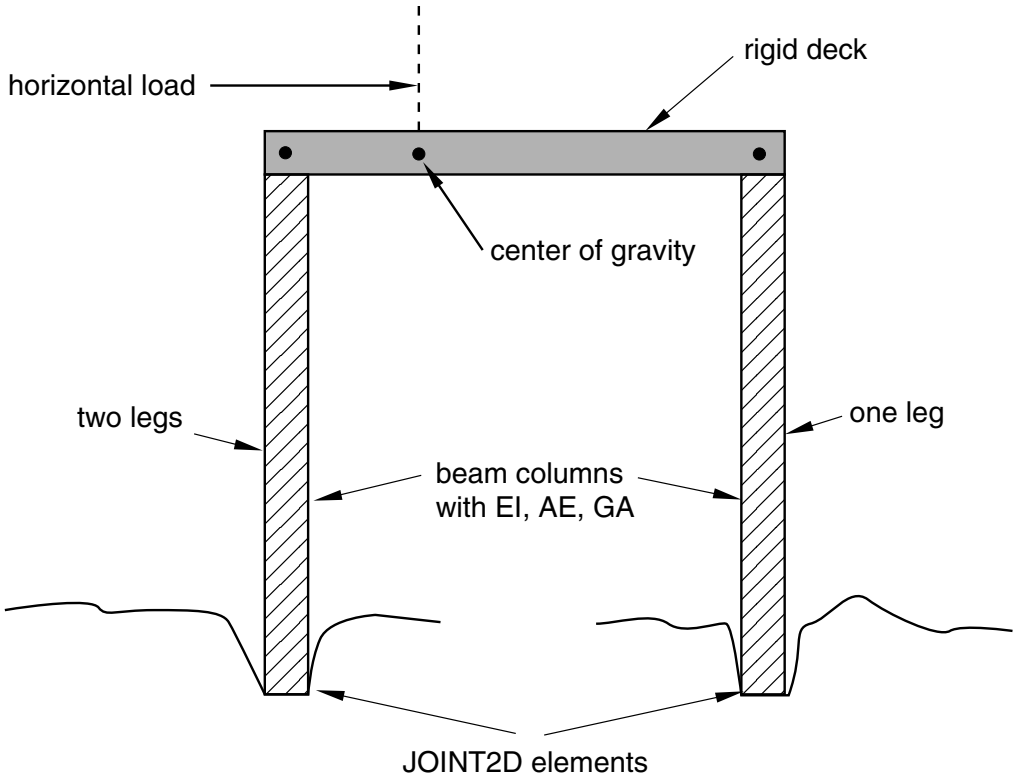


Figure 1: Schematic representation of jack-up rig.

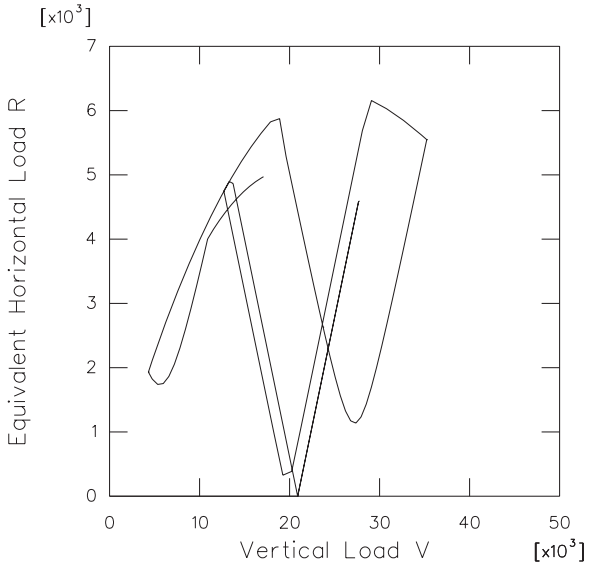


Figure 2: Load path for leeward spud can.

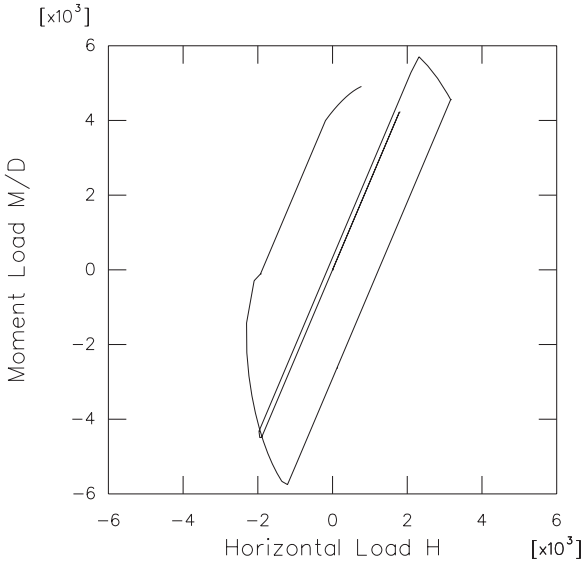


Figure 3: Moment versus horizontal load for leeward spud can.

Riser dynamics

Products: Abaqus/Standard Abaqus/Aqua

Overview

This example verifies the response of a riser under different types of load conditions.

Pipelines extending from the sea floor to the ocean surface (risers) are subject to many types of load: self-weight, buoyancy, internal and external pressure, tensile forces arising from surface moorings, current drag, and oscillatory loads resulting from wave motion. The response of a riser to these loads is complex, and the difficulty of such analysis is heightened by the relative length of such pipelines (deep water risers). In this example a riser is analyzed under conditions specified by the American Petroleum Institute for comparison of drilling riser analyses (API BULLETIN2J, 1977), and the results are compared with the results shown in that publication.

Geometry and model

The riser is shown in *Figure 1*. Its length is 463.3 m (1520 ft), and it stands in 448.1 m (1470 ft) of water. The outer diameter of the riser is 405 mm (1.33 ft), and it has a wall thickness of 15.88 mm (0.0521 ft). The pipeline is made of steel, with a Young's modulus of 206.8 GPa (4.32×10^9 lb/ft^2) and a density of 11508.685 kg/m^3 (22.332 lb-s^2/ft^4). The riser is modeled with 10 beam elements of type B21. No mesh convergence studies have been performed; hence, more elements may be required for accurate prediction of the stress in the riser.

Loading

The riser has a weight of 2575 N/m (176.36 lb/ft) and is loaded by a top tension of 2.224 MN (5×10^5 lb). Drag loading is applied by a steady current flowing by the riser with a velocity distribution varying linearly from 0.257 m/s (0.844 ft/s) at the mean water level to zero at the base of the riser. The coefficients in Morison's equation are transverse drag coefficient (C_D) 0.7, tangential drag coefficient (C_T) 0.0, and transverse inertia coefficient (C_M) 1.5.

The effective outer diameter for the drag calculations is 0.66 m (2.167 ft). Waves of peak to trough height 6.1 m (20 ft) travel across the water surface with a period of 9 seconds; these are modeled with the Airy wave theory provided in Abaqus/Aqua (*Abaqus/Aqua Analysis*). The density of the fluid is taken to be 1021 kg/m^3 (1.982 lb–s^2/ft^4). In Abaqus/Aqua, user subroutine *UWAVE* can be used to specify user-defined wave kinematics. We illustrate this capability by repeating this analysis with a user-specified Airy wave theory that is identical to the built-in Airy wave option in Abaqus/Aqua.

Boundary conditions

The base of the riser is “gimballed,” supporting no moments. The top of the riser has two motions prescribed: an initial offset of 13.716 m (45 ft) from the vertical position of the riser and a sinusoidal motion about this static configuration, representing the surge of a vessel attached to the riser, with peak-to-peak amplitude of 1.22 m (4 ft) and a period of 9 seconds. The vessel surge angle is 15° out of phase with the surface waves. The phase angle, ϕ_N, for the Airy wave definition provides an arbitrary choice of origin in time for the vertical displacement of a fluid particle. Based on the initial offset and vessel surge angle, this angle is set to −54.026°, where the negative sign indicates that the wave lags behind the vessel surge.

Analysis

The analysis is done in two steps. The first is the static step, in which the top tension is applied and the riser is moved from the vertical to its offset position by specifying the necessary horizontal displacement at the top of the pipeline. The top tension is 2.224 MN (5×10^5 lb).

In the second step, which is a dynamic step, the time increment is chosen as a fixed value of 0.125 second. The prescribed displacement at the top of the riser has a 9-second period, so this time step should provide reasonably accurate time integration once the higher modes are damped out by the fluid drag. The "half-increment residual" values calculated by Abaqus provide a measure of accuracy of the solution, and these values are typically of order 4.4 kN (1000 lb). Since these values are smaller than typical actual forces, they suggest that the time integration is reasonably accurate.

Results and discussion

The initial static step, which moves the riser to its offset position and applies the static loads, is completed in four increments. The first increment requires more iterations than subsequent increments, which is typical of this class of problem: the riser is initially unstressed and, therefore, is highly flexible. After some loading is applied, the axial tension stabilizes the system, and convergence is more rapid.

At the end of the static step the top of the riser makes an angle of 1.17° with the vertical. This value agrees well with the value of 1.20° presented in API BULLETIN2J (1977). The angle predicted at the base of the riser is 2.48°, which compares to 2.55° reported in the API bulletin. The slight discrepancies are attributed to the relative coarseness of the model.

The dynamic solution is carried out for 18 seconds of response. Typically one equilibrium iteration is required in each of the time increments. Half-increment residual values for the first few increments are of order 178 MN (4.0×10^7 lb), and at the end of the run they are of order 4.4 kN (1000 lb). This result is typical: initially there is much high frequency content in the solution, which is reflected in the larger half-increment residual values. As the analysis proceeds, the fluid drag dissipates this "noise," the solution becomes smoother, and the half-increment residual values drop accordingly.

The envelope of pipeline excursions during the second cycle of the dynamic excitation is plotted in *Figure 2*, and the envelope of bending stress is shown in *Figure 3*. These results are in basic agreement with those given in the API bulletin.

As expected, the results obtained by the model with the Airy wave theory implemented in user subroutine *UWAVE* are identical to those due to the built-in Airy wave option.

Input files

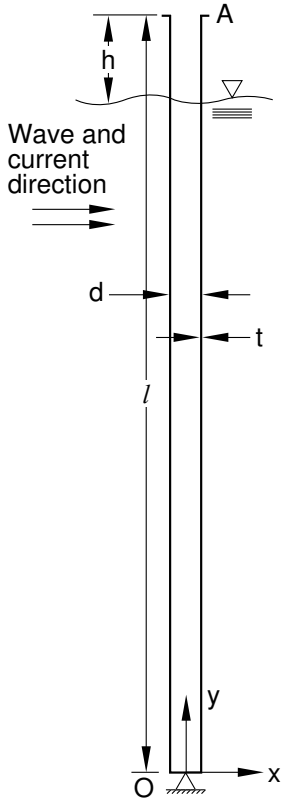
riserdynamics_airy_disp.inp	Analysis with the Airy wave theory. User subroutine *DISP* is used to prescribe the sinusoidal surge motion. This motion could be prescribed instead through the use of the **AMPLITUDE* option. User subroutine *DISP* is used to illustrate the use of this routine to prescribe a nonzero boundary condition value.
riserdynamics_airy_disp.f	User subroutine *DISP* used in riserdynamics_airy_disp.inp.
riserdynamics_wavedata.inp	Wave data for use in riserdynamics_airy_disp.inp.
riserdynamics_stokes_disp.inp	Analysis with the Stokes wave theory.
riserdynamics_stokes_disp.f	User subroutine *DISP* used in riserdynamics_stokes_disp.inp.
riserdynamics_airy_disp_uwave.inp	Analysis with the Airy wave theory implemented in user subroutine *UWAVE*.

riserdynamics_airy_disp_uwave.f	User subroutines *UWAVE* and *DISP* used in riserdynamics_airy_disp_uwave.inp.
exa_riserdynamics_stokes_disp_uwave.inp	Analysis with the Stokes wave theory implemented in user subroutine *UWAVE*.
exa_riserdynamics_stokes_disp_uwave.f	User subroutines *UWAVE* and *DISP* used in exa_riserdynamics_stokes_disp_uwave.inp.

References

- American Petroleum Institute, "Comparison of Marine Drilling Riser Analyses," API Bulletin 2J, Washington, DC, January 1977.

Figures



Geometry:
l = 463.3 m (1520.0 ft)
h = 15.2 m (50.0 ft)
d = 405 mm (1.33 ft)
t = 15.8 mm (0.0521 ft)

Material:
Young's modulus = 206.8 GPa (4.32×10^9 lb/ft^2)
density = 11508.68 kg/m^3 (22.332 lb-s^2/ft^4)

Boundary conditions:
at O => $u_x = u_y = 0$
at A => static: $u_x = a$
dynamic: $u_x = a + b \sin(\frac{2\pi}{9} t)$
a = 13.716 m (45.0 ft)
b = 610 mm (2.0 ft)

Loading:
self weight 2575.0 N/m (176.36 lb/ft)
top tension = 2.224 MN (5.0×10^5 lb)

Morison coefficients:
transverse drag coefficient = 0.7
tangential drag coefficient = 0.0
transverse inertia coefficient = 1.5
effective outer diameter = 0.66 m (2.167 ft)

Airy wave definition:
wave height (peak to trough) = 6.1 m (20.0 ft)
wave period = 9.0 s

Fluid velocity:
v_x = 0.257 m/s (0.844 ft/s) at mean water level
v_x = 0.0 at base of riser
v_y = 0.0

Fluid density:
ρ_w = 1021.0 kg/m^3 (1.982 lb-s^2/ft^4)

Figure 1: Riser problem definition.

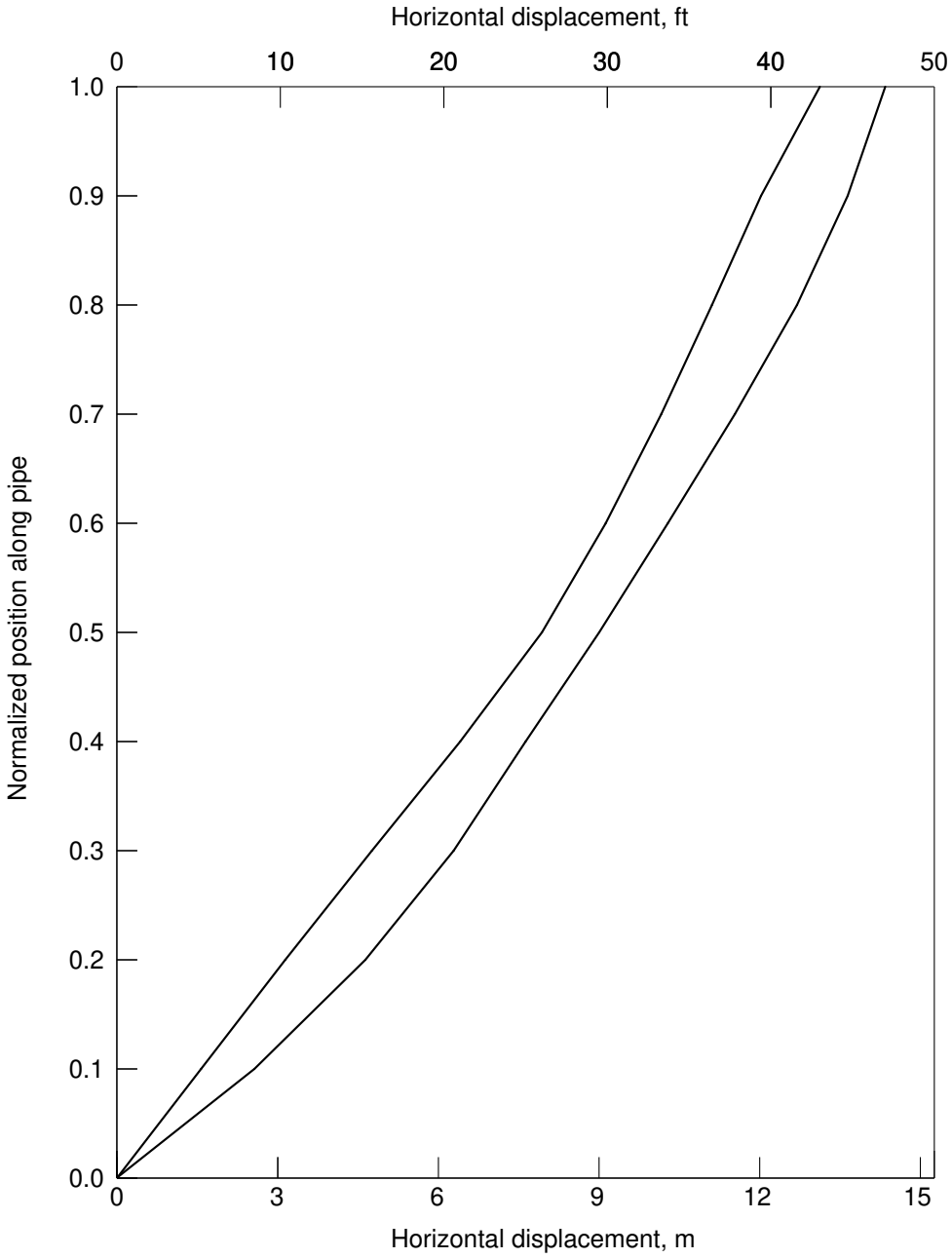


Figure 2: Horizontal displacement envelope during the second cycle of the dynamic excitation.

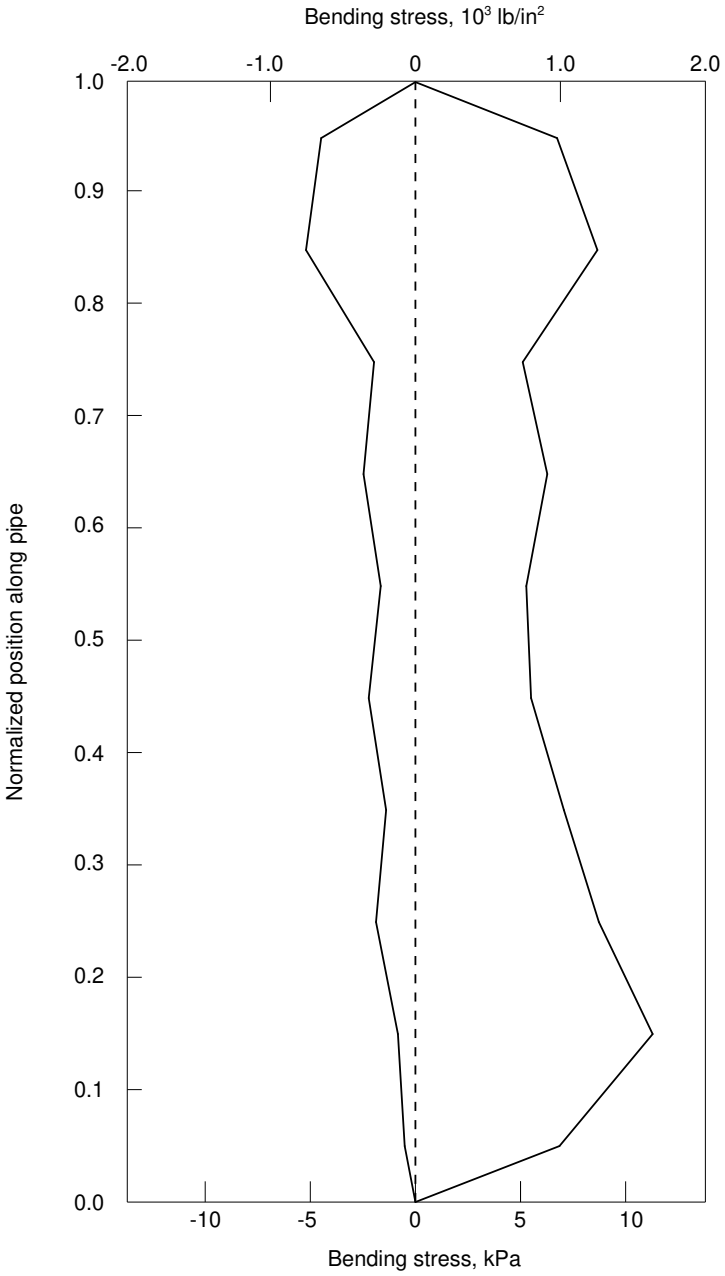


Figure 3: Bending stress envelope during the second cycle of the dynamic excitation.

Structural Optimization Analyses

In this section:

- *Topology, shape, and bead optimization analyses*
- *Sizing optimization analyses*

Topology, shape, and bead optimization analyses

In this section:

- *Topology optimization of an automotive control arm*
- *Shape optimization of a connecting rod*
- *Bead optimization of a plate*

Topology optimization of an automotive control arm

Products: Abaqus/Standard Abaqus/CAE

References:

- *Optimization Techniques*
- *Topology Optimization*
- *The Optimization module*
- *Understanding optimization processes*

Overview

This example uses the Optimization module to optimize the design of an automotive control arm by reducing the volume of the control arm while maximizing its stiffness.

Application description

This example illustrates topology optimization of an automotive control arm. During a topology optimization, the material properties of the elements in the design area are modified (effectively removing elements from or adding elements to the Abaqus analysis) until the optimal solution is achieved. For more information, see *Topology Optimization*.

Geometry

The control arm model is a single orphan mesh part that was meshed with quadratic tetrahedral (C3D10) elements. The control arm is symmetric about the *X*–*Y* plane, and only one half of the model is studied.

Materials

The control arm is made of an elastic material with a Young's modulus of 210 GPa and a Poisson's ratio of 0.3.

Boundary conditions and loading

The center of the model is constrained to be symmetric about the *Y*–*Z* plane. The upper left and upper right ends of the control arm are outside the design area and are fixed in all three translation degrees of freedom. The lower bearing center is also outside the design area, and its translation is constrained along the *z*-direction.

The center node of the lower bearing is loaded with a concentrated force of 70000 N in the *x*-direction and −70000 N in the *y*-direction.

Optimization features

The topology optimization is configured as described in the following sections.

Optimization task

This example creates a topology optimization task that is governed by the condition-based optimization algorithm.

Design area

The design area of the model is the region that will be modified during the optimization, as shown in *Figure 1*. Some regions are excluded from the design area because they are required for fixtures and for applying loads. The material properties of the elements excluded from the design area remain unchanged.

Design responses

A design response is created that calculates the sum of the strain energy over all the elements in the design area. A second design response calculates the volume of the design area.

Objective function

Objective functions define the objective of the optimization. In this example a single objective function attempts to minimize the sum of the strain energy of the design area. Since compliance is defined as the sum of the strain energy, and stiffness is the reciprocal of compliance, the objective function is equivalent to maximizing the stiffness of the design area.

Constraint

Optimization constraints constrain the optimization process from making changes to the topology of the model. Constraints must allow the optimization to arrive at a solution that is both feasible and acceptable. In this example a single constraint is created that specifies that the optimized model should contain 57% of the initial volume of the original control arm.

Geometry restrictions

You can apply geometric restrictions to further constrain the topology optimization process to consider only designs that can be manufactured using standard techniques, such as casting or forging. The control arm is manufactured by forging. The demold control geometric restriction shown in *Figure 1* ensures that the structure formed by the topology optimization can be removed from the forging die and does not contain undercuts. This example also introduces a frozen area geometric restriction to limit the material that is removed from the upper arm of the structure, as shown in *Figure 2*.

Abaqus modeling approaches and simulation techniques

This example imports the model in the form of an orphan mesh from an input file. The input file contains the element sets that define the regions of the model that are used by the optimization, such as the design area and the frozen area. The example creates an optimization process with a global stop criterion of 17 design cycles.

Analysis types

A static stress analysis is performed.

Constraints

The center node is connected to the bearing surface through a kinematic coupling.

Run procedure

A Python script is included that reproduces the model using the Abaqus Scripting Interface in Abaqus/CAE. The Python script (*control_arm_topology_optimization.py*) imports the input file (*control_arm.inp*) and builds the optimization model. The Python script can be run interactively or from the command line. Both the script and the input file must be available from your working directory.

When the script completes, you can use the Optimization module to review the topology optimization model that was created in Abaqus/CAE. To run the optimization, you can submit the optimization process from the **Optimization Process Manager** in the Job module. You can use the **Optimization Process Manager** to monitor the progression of the optimization and to view the results of the topology optimization in the Visualization module.

Results and discussion

The results are available in the output database file created by the optimization process. The step contains 14 optimization iterations that correspond with the 14 design cycles of the optimization process. *Figure 3* shows a history output plot of the strain energy and volume design responses over the 14 design cycles. The control arm is optimized such that the maximum stiffness is achieved while satisfying the specified target volume. Although the strain energy design response increases (the overall stiffness decreases) as the volume of the control arm is reduced, the optimized design achieves a topology with only 57% of the initial volume. *Figure 4* shows how the topology optimization progressively removes material from the control arm while it seeks an optimized solution.

Files

control_arm_topology_optimization.py	Python script to import the orphan mesh from the input file and create the topology optimization.
control_arm.inp	Input file to create the orphan mesh control arm and the element sets that are used by the optimization.

References

- Bendsøe, M. P., E. Lund, N. Ohloff, and O. Sigmund, “Topology Optimization - Broadening the Areas of Application,” Control and Cybernetics, vol. 34, pp. 7–35, 2005.

Figures

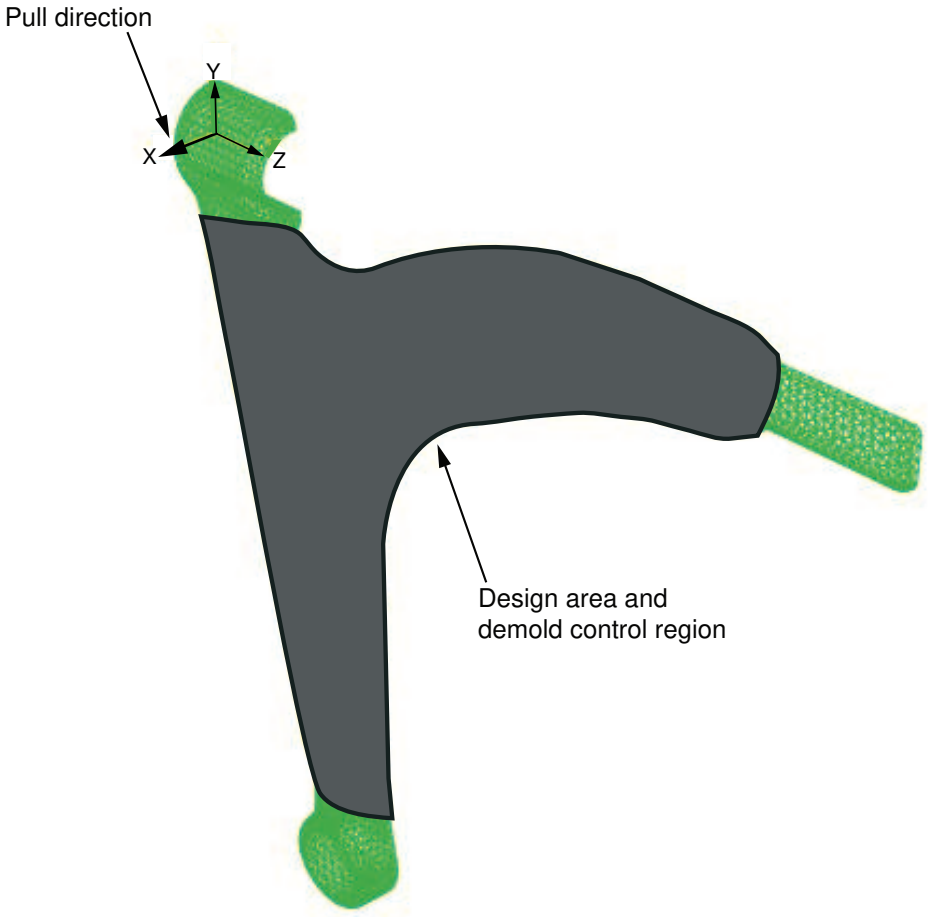


Figure 1: Design area and demold control region.

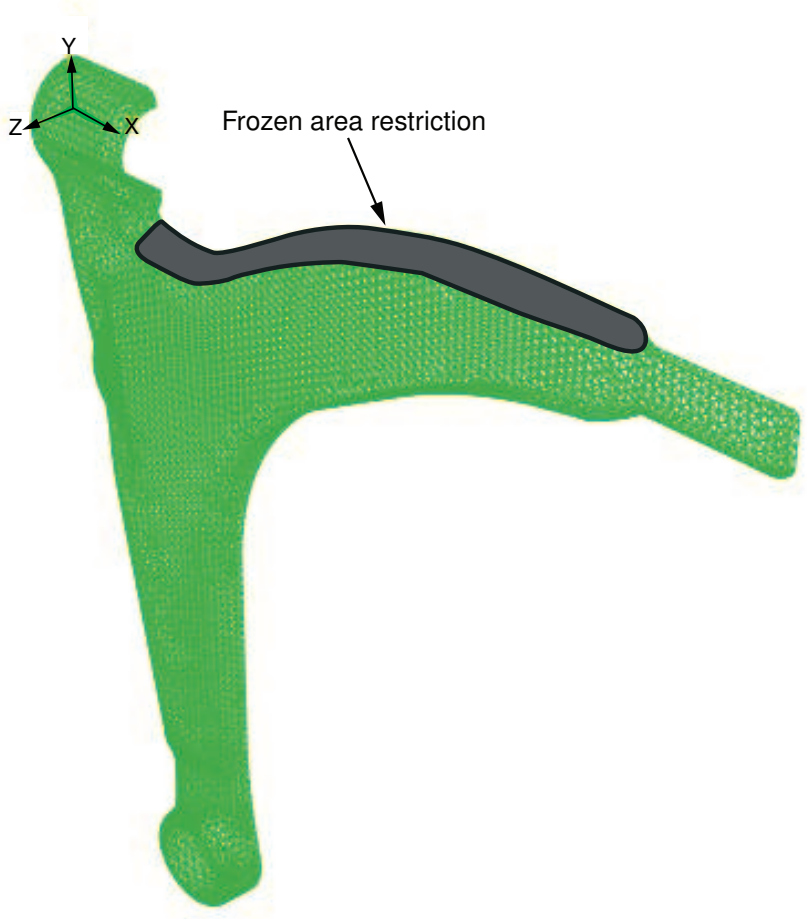


Figure 2: Frozen area geometric restriction.

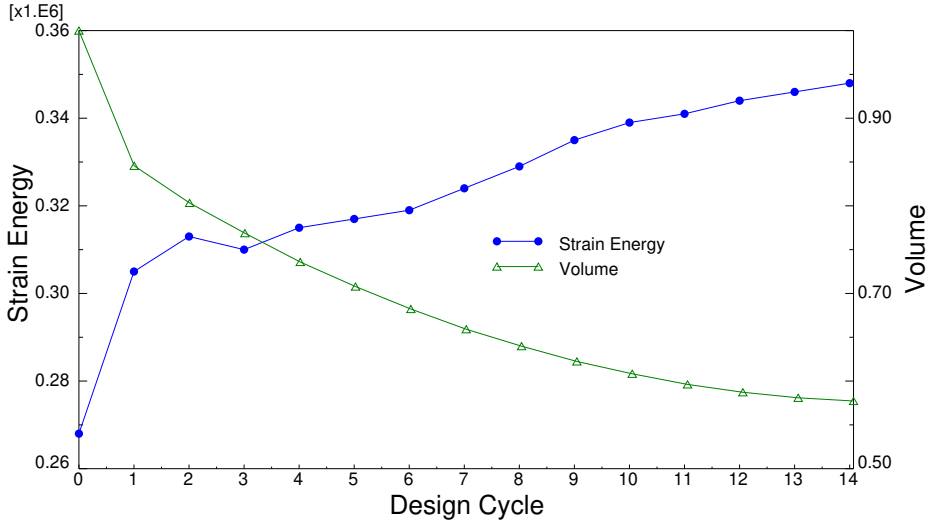


Figure 3: Design responses (strain energy and volume).

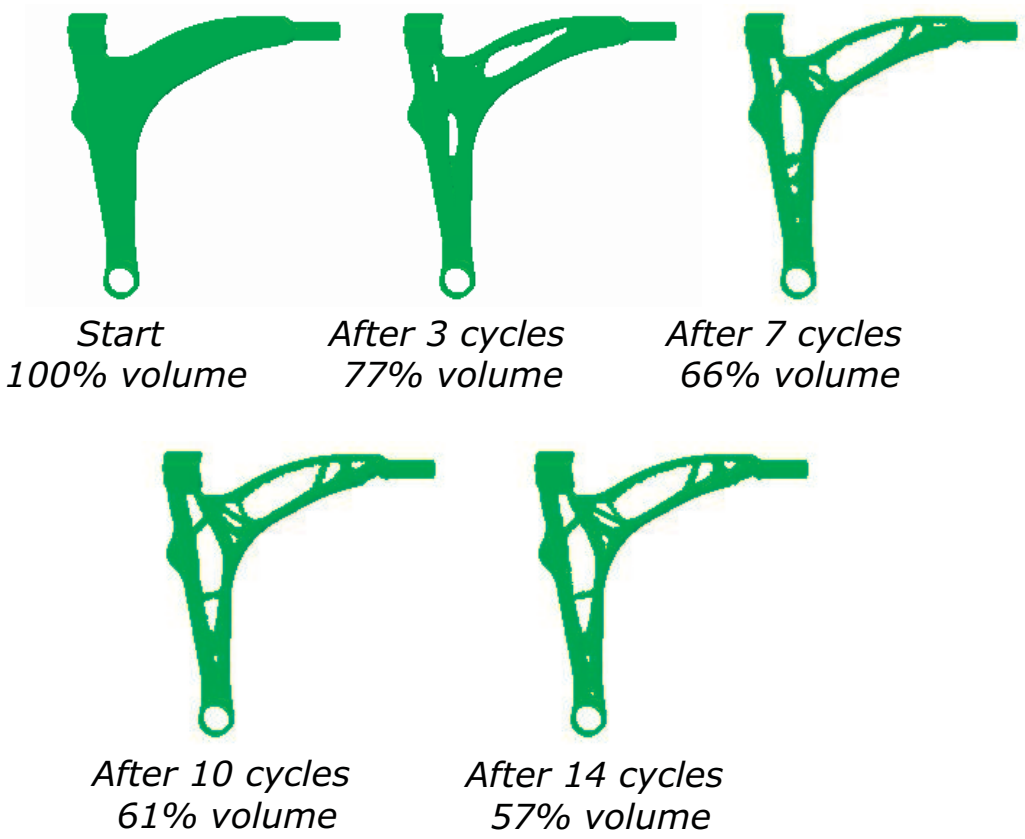


Figure 4: Progression of the topology optimization.

Shape optimization of a connecting rod

Products: Abaqus/Standard Abaqus/CAE

References:

- *Optimization Techniques*
- *Shape Optimization*
- *The Optimization module*
- *Understanding optimization processes*

Overview

This example uses the Optimization module to minimize the stress concentrations in a connecting rod without changing the volume of the connecting rod.

Application description

This example illustrates shape optimization of a connecting rod. Shape optimization makes slight modifications to the position of the surface nodes in the design area to achieve the optimized solution. Shape optimization is typically applied at the end of the design process when the general outline of a component is fixed and only minor changes are allowed. For more information, see *Shape Optimization*.

Geometry

The connecting rod model is a single orphan mesh part that was meshed with linear tetrahedral (C3D4) elements. The connecting rod is symmetric about the *X*–*Z* plane.

Materials

The connecting rod is made of an elastic material with a Young's modulus of 210 GPa, a Poisson's ratio of 0.3, and a density of 7800 kg/m^3.

Boundary conditions and loading

The displacement of the center node of the small end is constrained in the *x*- and *y*-directions, and the rotation of the center node is constrained in the *y*- and *z*-directions. For the center node of the big end, the displacement in the *x*- and *z*-directions and the rotation in the *y*- and *z*-directions are constrained.

During the first step a 25000 N load is applied to the center node of the small end of the connecting rod in the *z*-direction.

During the second step a −2000 N load is applied to the center node of the small end of the connecting rod in the *z*-direction and a 1750 N load is applied to the center node of the big end of the connecting rod in the *y*-direction. In addition, a rotation of 0.004 radians about the *x*-axis is applied to the center node of the big end.

Optimization features

The shape optimization is configured as described in the following sections.

Optimization task

This example creates a shape optimization task. To ensure good quality elements in the final design, mesh smoothing is applied to the elements in the design area.

Design area

The design area of the model is the region that will be modified during the optimization, as shown in *Figure 1*. Regions are excluded from the design area because they are required for fixtures and for applying loads.

Design responses

A design response is created for each step that determines the maximum von Mises stress in the design area. A third design response calculates the volume of the design area.

Objective function

The objective function determines which of the two design responses results in the largest von Mises stress in the design nodes. The objective function then tries to minimize the maximum von Mises stress for that design response.

Constraint

The volume design response is configured as the single optimization constraint such that the total volume of the design area remains unchanged during the optimization process.

Geometry restrictions

The example introduces a casting restriction by defining demold control geometry restrictions that are applied to the nodes in each half of the design area in the positive x- and negative x-directions. *Figure 2* illustrates the demold control geometry restriction that is applied to one half of the design area in the positive x-direction.

Abaqus modeling approaches and simulation techniques

This example imports the model in the form of an orphan mesh from an input file. The input file contains the element sets that are used to define the regions of the model that are used by the optimization, such as the design area and the demold control region. The example creates an optimization process with a global stop criterion of 15 design cycles. To maintain the quality of the surface elements, mesh smoothing is applied to the design area, which adjusts the position of the inner nodes in relation to the movement of the surface nodes. For more information, see *Applying Mesh Smoothing to a Shape Optimization*.

Analysis types

The analysis includes two static, general steps.

Constraints

The center nodes at both the ends of the connecting rod are connected to the bearing surfaces of the connecting rod with kinematic couplings.

Run procedure

A Python script is included that reproduces the model using the Abaqus Scripting Interface in Abaqus/CAE. The Python script (*conrod_shape_optimization.py*) imports the input file (*connecting_rod.inp*) and builds the optimization model. The script can be run interactively or from the command line. Both the Python script and the input file must be available from your working directory.

When the script completes, you can use the Optimization module to review the shape optimization model that was created in Abaqus/CAE. To run the optimization, you can submit the optimization process from the **Optimization Process Manager** in the Job module. You can use the **Optimization Process Manager** to monitor the progression of the optimization and to view the results of the shape optimization in the Visualization module.

Results and discussion

The results are available in the output database file created by the optimization process. The step contains 15 optimization iterations that correspond with the 15 design cycles of the optimization process. *Figure 3* shows a history output plot of the von Mises stress design response for each of the two load cases over the 15 design cycles. The history output plot also shows the volume design response, which is kept constant throughout the optimization.

The position of the surface nodes in the design area is optimized such that for the specified volume constraint and geometry restrictions the von Mises stress is minimized during the load case that results in the maximum stress. In this example the 25000 N load that is applied to the small end in the first step results in the highest von Mises stress in the connecting rod. Therefore, the objective function adjusts the surface nodes to reduce the maximum von Mises stress during the first step, although the resulting optimized shape allows the von Mises stress to increase slightly under the loading conditions in the second step.

Figure 4 shows the initial von Mises stress distribution during the first load case and the change in the stress distribution after 15 design cycles. *Figure 5* shows similar von Mises stress distribution plots before and after optimization during the second load case.

Files

conrod_shape_optimization.py	Script to create the model and the optimization attributes using connecting_rod.inp.
connecting_rod.inp	Input file to create the orphan mesh connecting rod and the element sets that are used by the optimization.

References

- Bakhtiary, N., and P. Allinger, “A New Approach for Size, Shape and Topology Optimization,”SAE International Congress and Exposition, Detroit, Michigan, USA, February 26–29, 1996.

Figures

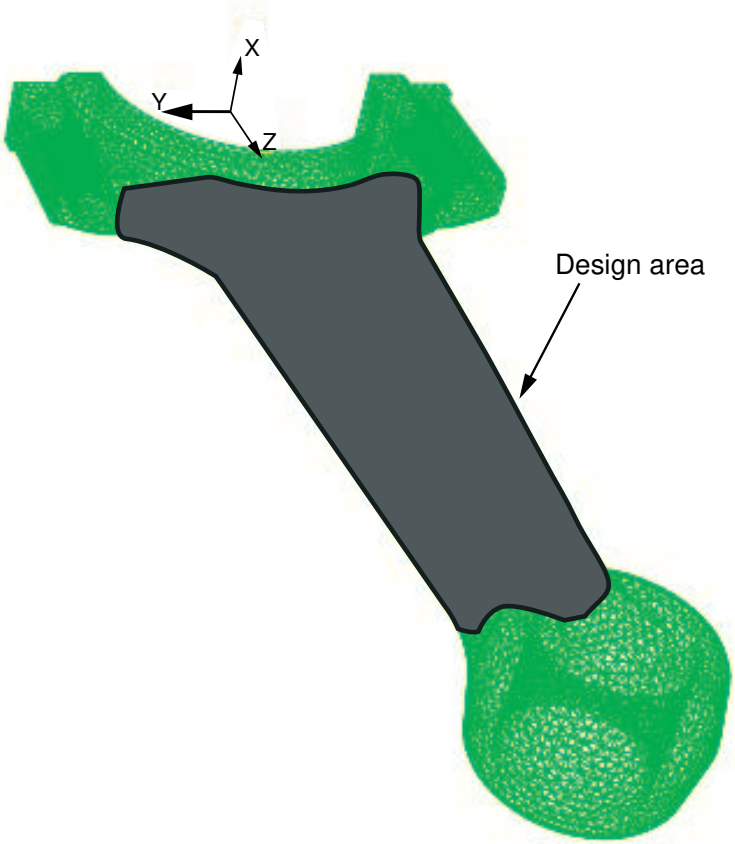


Figure 1: Design area.



Figure 2: Demold control geometric restriction.

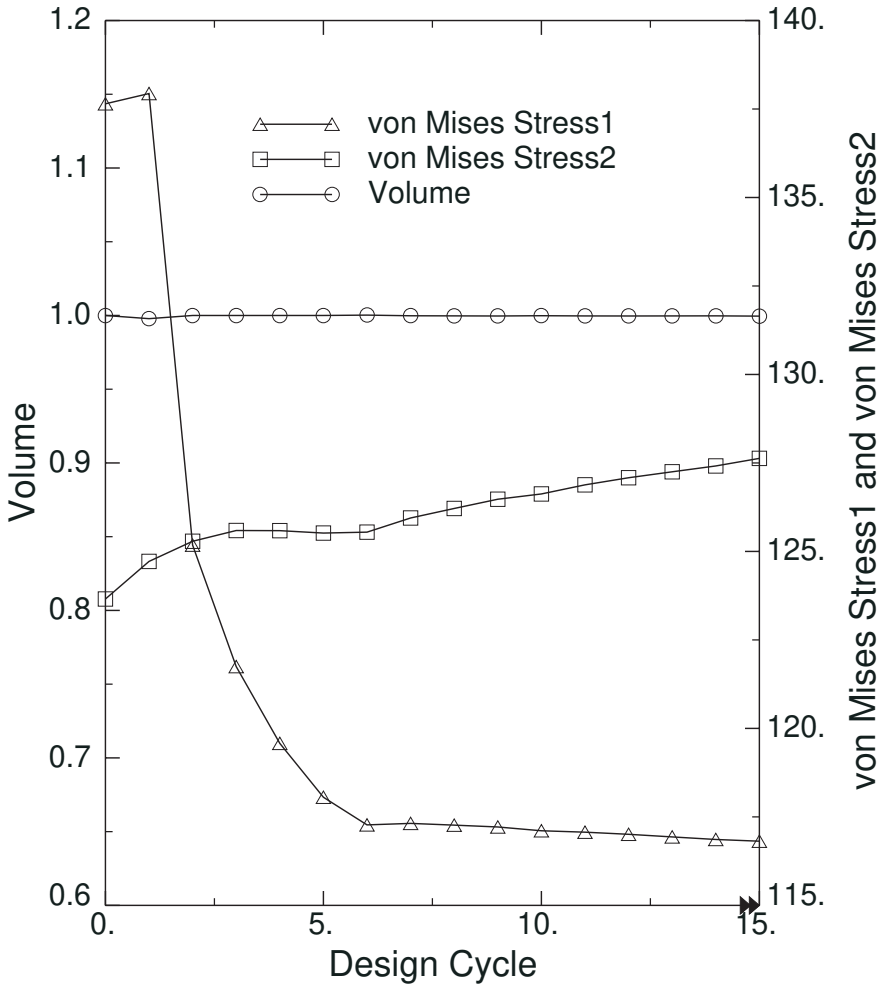


Figure 3: Design responses (von Mises stress and volume).

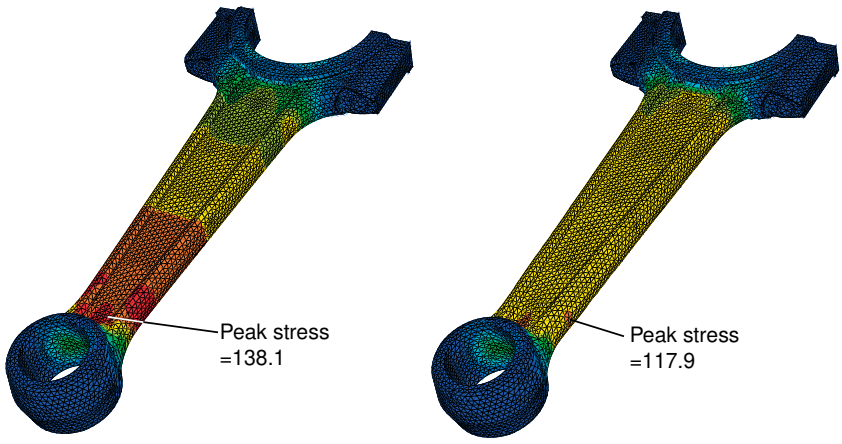


Figure 4: Distribution of von Mises stress before (left) and after optimization (right) during first load case.

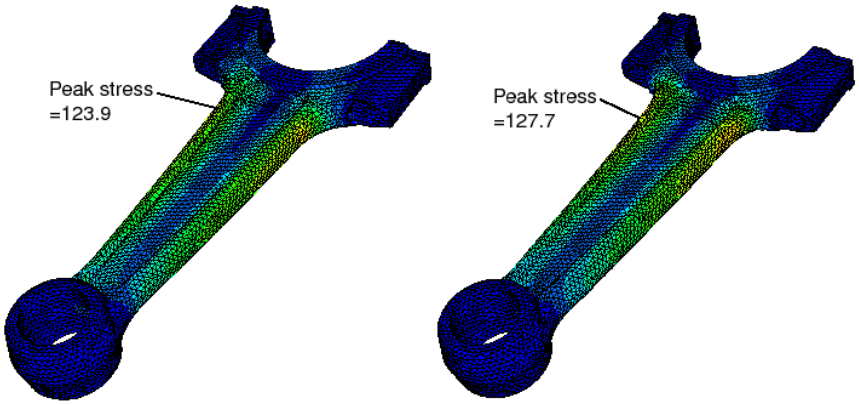


Figure 5: Distribution of von Mises stress before (left) and after optimization (right) during second load case.

Bead optimization of a plate

Products: Abaqus/Standard Abaqus/CAE

References:

- *Optimization Techniques*
- *Bead Optimization*
- *The Optimization module*
- *Understanding optimization processes*

Overview

This example uses the Optimization module to optimize the stiffness of a simple plate by introducing stiffening beads.

Application description

This example illustrates condition-based bead optimization of a simply supported plate. During a bead optimization, the nodes of the shell elements are moved in the direction of the shell normal to increase the moment of inertia, which leads to a greater stiffness or higher eigenfrequencies. The resulting beads are easy to reproduce in a sheet metal stamping process without increasing the mass or cost of the finished product. For more information, see *Bead Optimization*.

Geometry

The plate is a 600 mm × 600 mm three-dimensional shell part. A shell section is assigned to the part with a thickness of 1.5 mm. The shell section uses the default Simpson integral rule with five integration points through the shell section.

Materials

The plate is made of an elastic material with a Young's modulus of 210 GPa and a Poisson’s ratio of 0.3.

Boundary conditions and loading

The edges of the plate are fixed in all three translation degrees of freedom. A pressure of 1.0 is applied to a partitioned region at the center of the plate, as shown in *Figure 1*.

Optimization features

The bead optimization is configured as described in the following sections.

Optimization task

This example creates a bead optimization task that is governed by the condition-based optimization algorithm. The bead width is specified as 60 mm.

Design area

The entire plate is selected as the design area. By default, the Optimization module freezes the position of the nodes around the edge of the plate that were included in the boundary condition.

Design responses

A design response is created that calculates the maximum of the strain energy over all the elements in the design area. A second design response calculates the bead height.

Objective function

A single objective function attempts to minimize the maximum of the strain energy of the design area. Since compliance is defined as the sum of the strain energy, and stiffness is the reciprocal of compliance, the objective function is equivalent to maximizing the stiffness of the plate.

Constraint

A single constraint sets the bead height to 20 mm.

Abaqus modeling approaches and simulation techniques

The example creates an optimization process with the default maximum of three design cycles.

Analysis types

A static stress analysis is performed.

Run procedure

A Python script is included that reproduces the model using the Abaqus Scripting Interface in Abaqus/CAE. The Python script (*plate_bead_optimization.py*) builds the optimization model and executes the optimization job. To view the results of the optimization, you can use the Optimization Process Manager to combine the output database files that are generated and open the resulting output database file (`Beadprocess\TOSCA_POST\BeadJob_post.odb`). You can use the Visualization module to display a contour plot of the optimized nodal displacement (DISP_NORMAL_VAL) that shows the generated stiffening beads. You can return to the Optimization module to review the bead optimization model that was created in Abaqus/CAE.

The Python script can be run interactively or from the command line, and the script must be available from your working directory.

Results and discussion

The results are available in the combined output database file created from the Optimization Process Manager. The step contains three optimization iterations that correspond to the three design cycles of the optimization process. *Figure 2* shows how the bead optimization moves nodes to create stiffening beads along the bending trajectories. *Figure 3* and *Figure 4* compare the plate stresses prior to the optimization and after the introduction of the stiffening beads.

Files

plate_bead_optimization.py	Python script to create the model and the bead optimization.

References

- Emmrich, D., “Entwicklung einer FEM-basierten Methode zur Gesaltung von Sicken für biegebeanspruchte Leitstützstrukturen im Konstruktionsprozess. ” Forschungsberichte des Instituts für Produktentwicklung, 13. Karlsruhe, 2004.

Figures

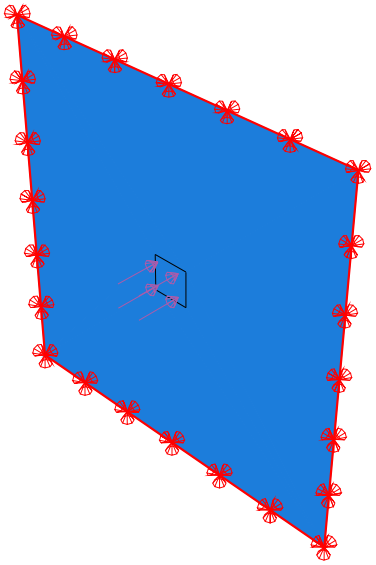


Figure 1: Pressure load and boundary conditions.

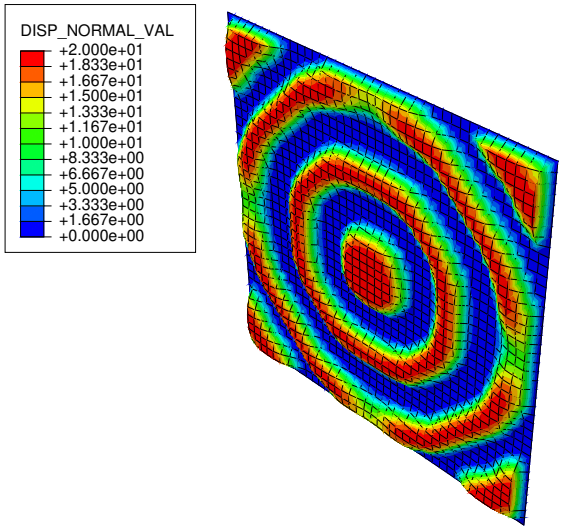


Figure 2: The optimized plate showing the displacement of the nodes.

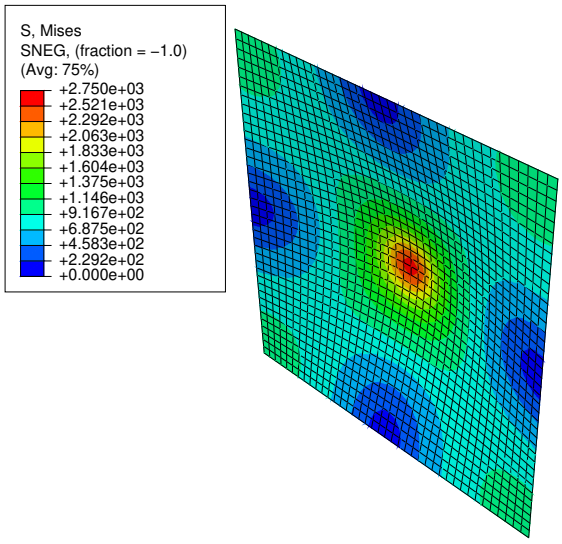


Figure 3: Stresses in the plate prior to optimization.

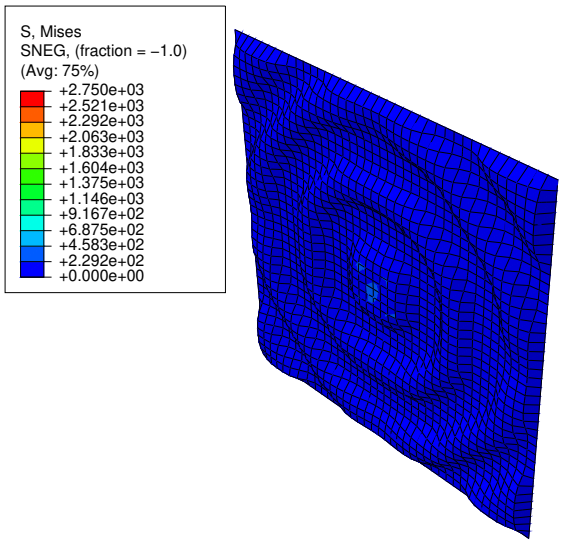


Figure 4: Stresses in the plate after optimization.

Sizing optimization analyses

In this section:

- *Sizing optimization of a gear shift control holder*
- *Sizing optimization of a car door*

Sizing optimization of a gear shift control holder

Products: Abaqus/Standard Abaqus/CAE

References:

- *Optimization Techniques*
- *Sizing Optimization*
- *The Optimization module*
- *Understanding optimization processes*

Overview

This example uses the Optimization module to minimize the sum of the strain energy (maximize the stiffness) in a pressed sheet metal holder by changing the shell thickness while maintaining the original weight.

Application description

This example illustrates sizing optimization of a pressed sheet metal holder that is used in an automotive gearbox linkage. Sizing optimization modifies the thickness of the shell elements in the design area to achieve the optimized solution. The example illustrates "free"optimization of the shell thickness, with no regard for the thickness of adjacent elements, and "clustered" optimization of the shell thickness, that forces elements in selected regions to have the same shell thickness. For more information, see *Sizing Optimization*.

Geometry

The pressed sheet metal holder is a single orphan mesh part. The part is meshed with general-purpose conventional shell (predominately S4) elements. The initial shell thickness is 1.0 mm.

Materials

The holder is made of an elastic material with a Young's modulus of 206 GPa, a Poisson's ratio of 0.3, and a density of 7840 kg/m^3.

Boundary conditions and loading

The boundary conditions and loads are shown in *Figure 1*. The four mounting holes are constrained in all directions in the initial step. During the first linear perturbation step a −2500 N load is applied to the hole in the x-direction. During the second linear perturbation step a torsional moment of 80,000 Nmm about the x-axis is applied to the hole.

Optimization features

The sizing optimization is configured as described in the following sections.

Optimization task

This example creates a sizing optimization task.

Design area

The design area of the model is the region that will be modified during the optimization, as shown in *Figure 1*.

Design responses

A design response is created for each step that determines the compliance or strain energy in the design area. A second design response calculates the volume of the entire model.

Objective function

The objective function tries to minimize the maximum compliance from the two linear perturbation steps. Compliance is the reciprocal of stiffness, and minimizing the compliance is equivalent to maximizing the global stiffness.

Constraint

The shell thickness is constrained between absolute values of 0.1 and 3.0. In addition, the volume is constrained such that the total volume of the holder is less than or equal to the original volume.

Geometry restrictions

The example introduces clustered rings of shell elements of the same thickness in the design area. In effect, clusters generate strengthening ribs or rings in the sheet metal structure you are optimizing; clustered regions can be reproduced in manufacturing by layering sheets of constant thickness.

Abaqus modeling approaches and simulation techniques

This example imports the model in the form of an orphan mesh from an input file. The input file contains the element sets that are used to define the regions of the model that are used by the optimization, such as the design area. The example creates an optimization process that you can submit for analysis.

Analysis types

The analysis includes two static, perturbation steps.

Constraints

Kinematic couplings connect the node at the center of the holes with the nodes at the edges of the hole.

Run procedure

A Python script is included that reproduces the model using the Abaqus Scripting Interface in Abaqus/CAE. The Python script (`holder_sizing_optimization.py`) imports the input file (`holder.inp`) and builds the optimization model. A second Python script (`holder_sizing_optimization_w_clustering.py`) performs the same optimization but introduces shell thickness clustering in the design area. The scripts can be run interactively or from the command line. The Python scripts and the input file must be available from your working directory.

To run the optimization, you can submit the optimization process from the **Optimization Process Manager** in the Job module. You can use the **Optimization Process Manager** to monitor the progression of the optimization. In addition, when the optimization process is complete, you can use the **Optimization Process Manager** to combine the output from the optimization into a single output database file that can be viewed in the Visualization module.

Results and discussion

The optimization process is run over 15 design cycles for the free optimization and over 13 design cycles for the optimization with clustering. During each design cycle, the optimization results are saved in optimization files. In addition, the Abaqus analysis results are saved in output database files during the initial design cycle

and during the last design cycle. To view the results of the optimization in the Visualization module, the optimization data in the optimization files and the analysis results in the output database files must be combined. The output database file created during the initial design cycle is selected as the base results output database file. During the combine operation, the optimization data from each design cycle and the analysis results from the last design cycle are appended to the base results output database file. Each frame of the combined output database file corresponds to a design cycle of the optimization. For more information about the files generated by an optimization process and how they are combined, see *Understanding the files generated by an optimization process* and *Postprocessing an optimization*.

Figure 2 shows the initial displacement magnitude at the end of the second step prior to any optimization. *Figure 3* and *Figure 4* show the value of the shell thickness and the displacement magnitude, respectively, after free sizing optimization. *Figure 5* and *Figure 6* show the value of the shell thickness and the displacement magnitude, respectively, after sizing optimization with clustering in the design area.

After the optimization, the shell thickness is increased at the end of the design area where the load and moment are applied. The shell thickness is also increased close to the mounting holes. To maintain the volume of the arm, the shell thickness is reduced in the rest of the design area. As expected, the free sizing optimization produces the best results with a 45% reduction of the maximum displacement. The optimization with circular clustering in the design area, which could be manufactured by welding sheets of metal together, still leads to a 30% reduction of the maximum displacement.

Python scripts

holder_sizing_optimization.py	Script to create the model and the optimization attributes using holder.inp.
holder_sizing_optimization_w_clustering.py	Script to create the model and the optimization attributes, including shell thickness clustering, using holder.inp.

Input files

holder.inp	Orphan mesh gear shift control holder and the node and element sets that are used by the optimization.

References

- Svanberg, K., The Method of Moving Asymptotes—A New Method for Structural Optimization International Journal for Numerical Methods in Engineering, vol. 24, pp. 359–373, 1987.

Figures

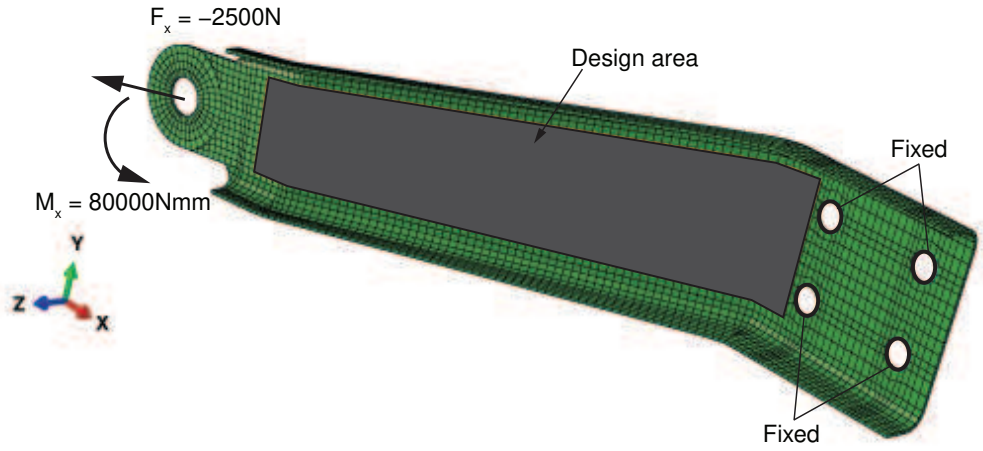


Figure 1: Design area, loads, and boundary conditions.

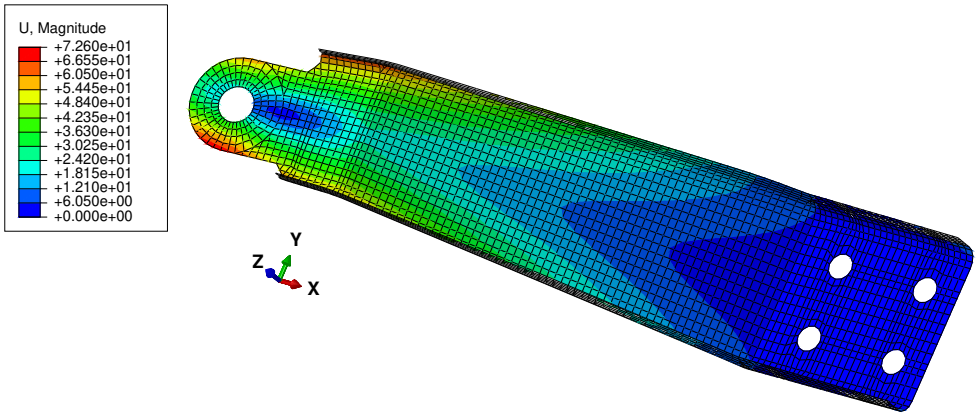


Figure 2: Displacement magnitude prior to the optimization.

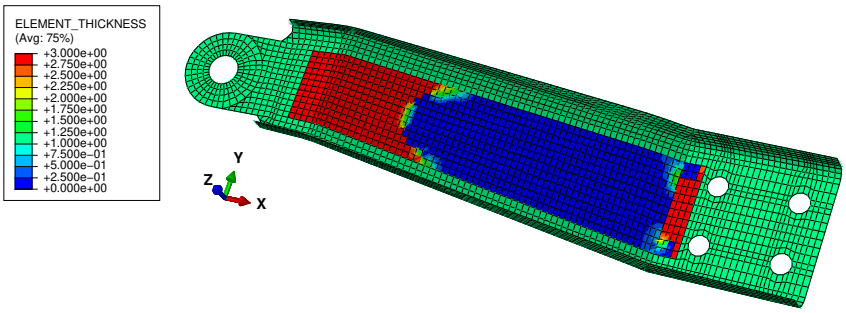


Figure 3: Absolute value of shell thickness after free optimization.

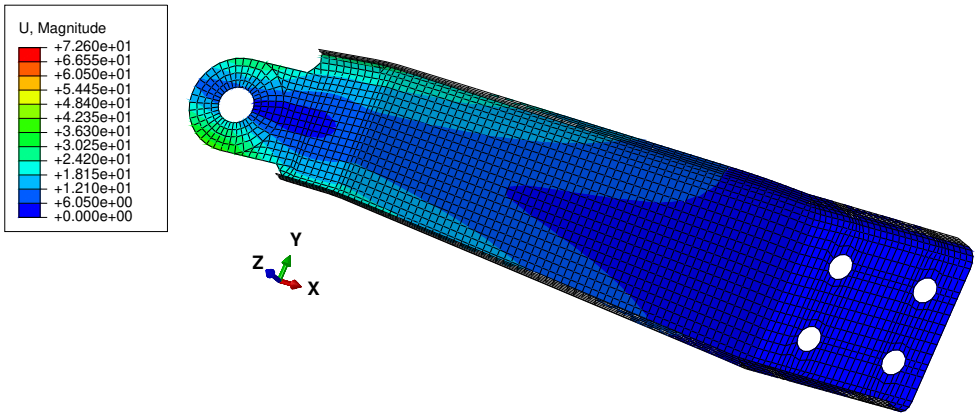


Figure 4: Displacement magnitude after free optimization.

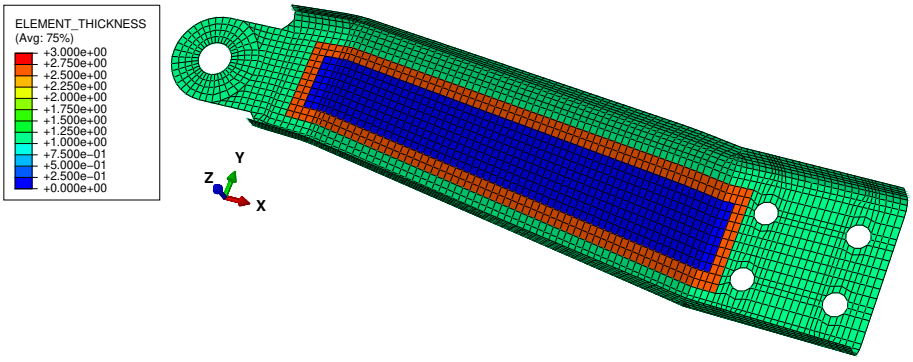


Figure 5: Absolute value of shell thickness after optimization with clustering.

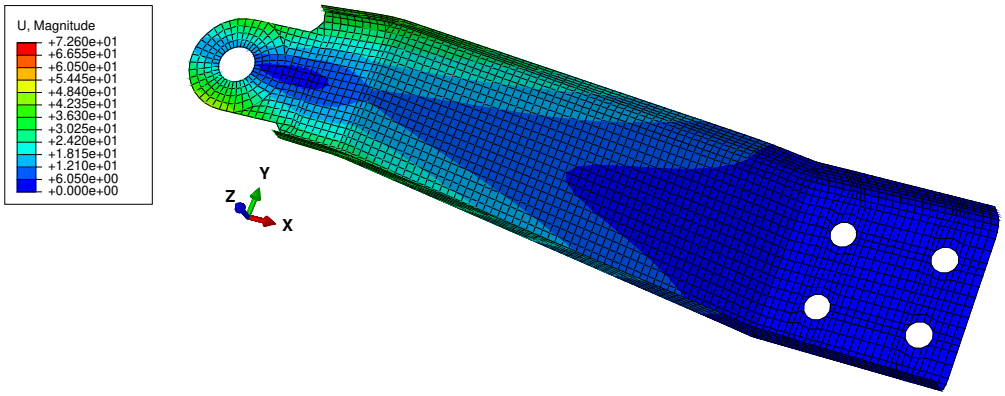


Figure 6: Displacement magnitude after optimization with clustering.

Sizing optimization of a car door

Products: Abaqus/Standard Abaqus/CAE

References:

- *Optimization Techniques*
- *Sizing Optimization*
- *The Optimization module*
- *Understanding optimization processes*

Overview

This example uses the Optimization module to minimize the weight of a car door by changing the thickness of the sheet metal components while limiting the vertical displacement (sag) of the door relative to its encastered hinges. The example also constrains the first frequency mode of the door to be higher than a specified value to avoid resonance triggered by engine vibration while maintaining its strength against the vertical load. The optimization takes into account geometric restrictions imposed by constraints in the manufacturing process and by constraints in design aesthetics. This model has been developed by The National Crash Analysis Center (NCAC) of The George Washington University under a contract with the Federal Highway Administration (FHWA) and the National Highway Traffic Safety Administration (NHTSA) of the United States Department of Transportation (US DOT).

Application description

This example illustrates sizing optimization of the sheet metal components that form a car door. Sizing optimization modifies the thickness of the shell elements in the design area to achieve the optimized solution. Clustering is applied during the sizing optimization, which forces elements in selected regions to have the same shell thickness and reproduces the real-world assembly of sheet metal components by layering parts of uniform thickness. A "free" sizing optimization of the same model generates a door that weighs less than the result of the clustered sizing optimization; however, the resulting element-by-element distribution of shell thickness would be impossible to reproduce on the production line. For more information, see *Sizing Optimization*.

Geometry

The door assembly is composed of the single orphan mesh part derived from the data supplied by the NCAC. The connections, contact definitions, and step definitions were converted to follow best practices in Abaqus. For example, regions (element sets) within the model are joined by spot welds (mesh-independent fasteners). The two hinges are modeled with hinge connectors and kinematic couplings. The door lock is modeled with a kinematic coupling.

Materials

Three materials are used to form the door—glass, plastic, and steel. The glass material properties are assigned to the element set representing the window. The plastic material properties are assigned to the element sets representing the inserts in the door trim, such as the liner. The steel material properties are assigned to the element sets representing the sheet metal door and its hinges and brackets. The material properties assumed for this analysis are shown in *Table 1*, *Table 2*, and *Table 3*. The symbols E, ν, and ρ, represent the Young's modulus, Poisson's ratio, and density, respectively.

Elastic material properties are sufficient because the small load that is applied to model vertical door sag is insufficient to cause plastic or permanent deformation.

Boundary conditions and loading

Two load cases are applied to the model:

- a Lanczos solver frequency step that determines the first five eigenvalues, and
- a static load step that determines how much the door sags in response to a vertical load on the door lock.

During the frequency step, the hinges and the lock are restrained in all six degrees of freedom. During the static load step, a load of 500 N along the z-axis is applied to the door lock. In addition, the hinges continue to be restrained in all six degrees of freedom. To prevent the door from opening, a displacement boundary condition restrains motion of the lock in the y-direction.

Optimization features

The sizing optimization is configured as described in the following sections.

Optimization task

This example creates a sizing optimization task.

Design area

The entire car door, except for the plastic trim, is selected as the design area of the model.

Design responses

The example includes three design responses:

- a weight design response that is applied to the entire assembly,
- an eigenfrequency design response that uses the default modal analysis formulation (Lanczos), and
- a displacement design response that is applied only to the door lock.

Objective function

The objective functions tries to minimize the weight design response.

Constraint

A constraint in the frequency step restricts the first eigenfrequency from falling below a value that would trigger resonance with engine vibration at idle speed (35 Hz). A constraint in the static load step restricts the absolute displacement of the door lock in the z-direction to be less than or equal to a reasonable value (1.42 mm).

Geometric restrictions

The thickness of the element sets representing the sheet metal parts is restricted to a reasonable range ($0.5 < thickness < 2.5$). Frozen area geometric restrictions force the element sets representing the window and the hinges and their brackets to remain unchanged during the optimization. In addition, the design of the exterior-facing door panel has been finalized by the styling department, and a frozen area geometric restriction prevents changes in thickness to the elements that form the outside of the door.

Clustering restrictions

Clustering is defined for each of the element sets representing the sheet metal parts, forcing each region to be of uniform thickness.

Abaqus modeling approaches and simulation techniques

This example imports the model in the form of an orphan mesh from an input file. The input file contains the element sets that are used to define the regions of the model that are used by the optimization, such as the window and the hinges and their brackets. The example creates an optimization process that you can submit for analysis.

Analysis types

The analysis includes a frequency step and a static load step.

Mesh design

Most of the door is meshed with quadrilateral regions that are assigned S4R three-dimensional shell elements, and the small number of triangular regions are assigned S3R elements. In addition, the three-dimensional solid region that represents the arm rest is assigned C3D4, C3D6, and C3D8R three-dimensional solid elements. The sizing optimization operates only on shell elements.

Run procedure

An input file (*door.inp*) defines the nodes and elements that define the door along with the element sets that represent the regions of the door. The input file also defines the kinematic couplings, multi-point constraints, and spot weld connections. A Python script (*door_sizing_optimization_w_clustering.py*) is included that imports the input file and builds the optimization model. The scripts can be run interactively or from the command line. The Python script and the input file must be available from your working directory.

To run the optimization, you can submit the optimization process from the **Optimization Process Manager** in the Job module. You can use the **Optimization Process Manager** to monitor the progression of the optimization. In addition, when the optimization process is complete, you can use the **Optimization Process Manager** to combine the output from the optimization into a single output database file that can be viewed in the Visualization module.

Results and discussion

A maximum number of 15 design cycles is specified; however, the optimization process converges over nine design cycles. The convergence criterion for a sizing optimization is based on a combination of the change of the element thickness and the change in the value of the objective function. Minimizing the weight is the objective function, and the total weight is reduced by 14% (from 30.6 kg to 26.3 kg) while satisfying the eigenfrequency and door sag constraints.

Figure 1 shows the initial element thickness at the end of the second step prior to any optimization. *Figure 2* and *Figure 3* show the results of the sizing optimization—the value of the absolute shell thickness and the change in shell thickness. Regions of the model that take no part in the optimization, such as the plastic liner, have been removed from the figures to make the output easier to understand.

Python scripts

door_sizing_optimization_w_clustering.py	Script to import the input file (door.inp) and to build the optimization model.

Input files

door.inp	Orphan mesh representation of the car door assembly; and the node and element set, material, load, boundary condition, and connector definitions that are used by the optimization.

References

- Svanberg, K., The Method of Moving Asymptotes—A New Method for Structural Optimization International Journal for Numerical Methods in Engineering, Vol. 24, pp. 359–373, 1987.
- This model has been developed by The National Crash Analysis Center (NCAC) of The George Washington University under a contract with the Federal Highway Administration (FHWA) and the National Highway Traffic Safety Administration (NHTSA) of the United States Department of Transportation (US DOT).

Tables

Table 1: Properties for the glass window.

Property	Value
E	76.0×10^9 Pa
ν	0.30
ρ	2.5×10^{-9} t/mm^3

Table 2: Properties for the plastic components.

Property	Value
E	2.8×10^9 Pa
ν	0.30
ρ	1.20×10^{-9} t/mm^3

Table 3: Properties for the steel components.

Property	Value
E	210×10^9 Pa
ν	0.30
ρ	7.89×10^{-9} t/mm^3

Figures

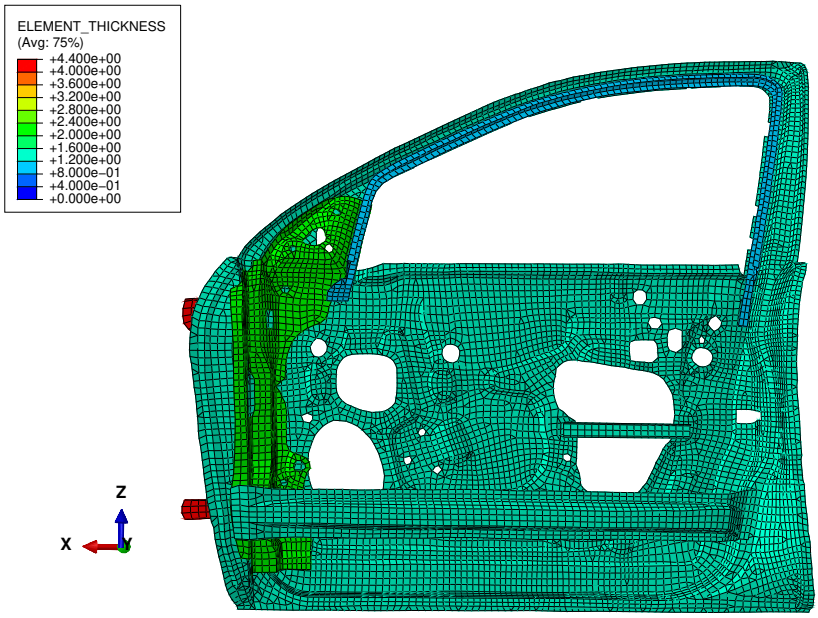


Figure 1: Element thickness prior to optimization.

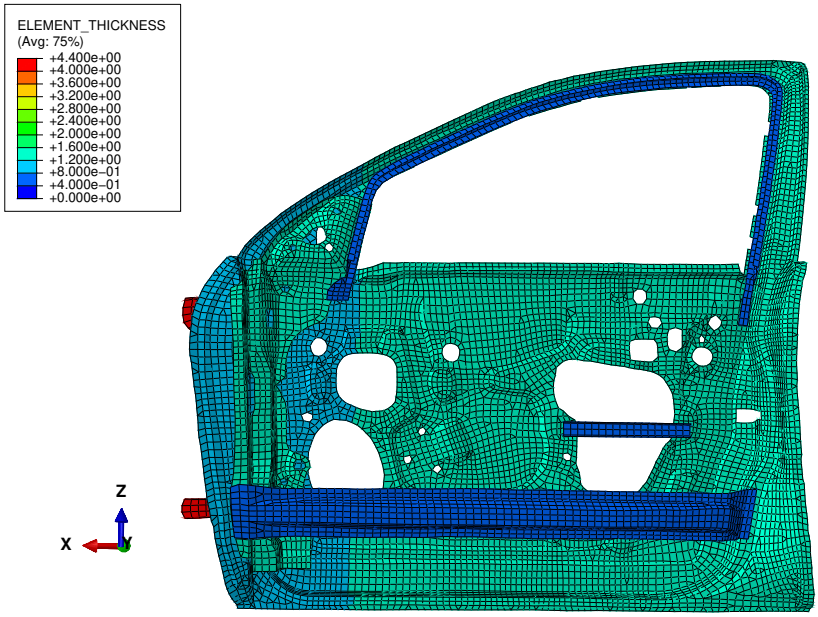


Figure 2: Element thickness after optimization.

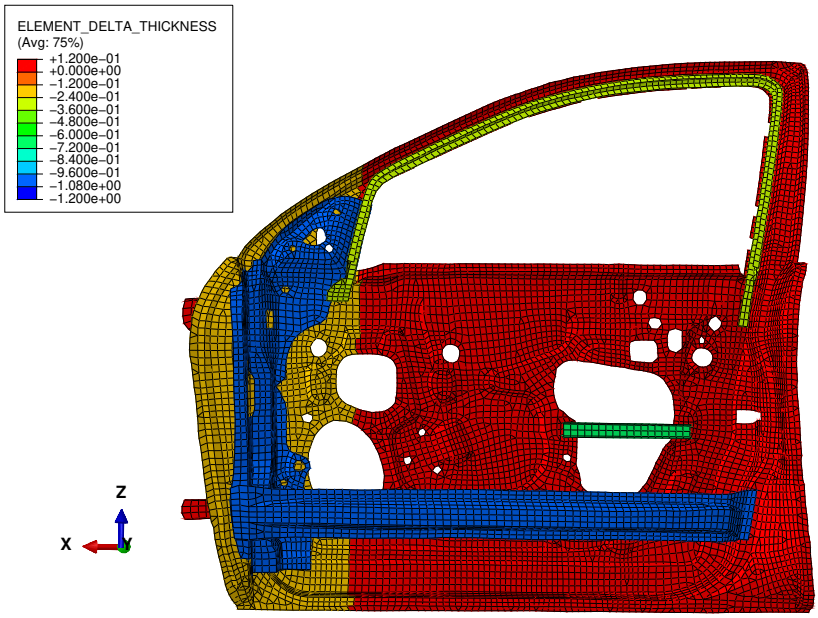


Figure 3: Change in element thickness during optimization.

Design Sensitivity Analyses

This chapter contains example problems specifically created to demonstrate the design sensitivity analysis (DSA) capability in Abaqus/Design.

In addition, *Indentation of an elastomeric foam specimen with a hemispherical punch* includes design sensitivity analysis.

Some of these examples employ a scripting command in Abaqus/CAE to aid in creating the shape variations. The usage of this command is described in *Using Abaqus/CAE to compute shape variations*.

In this section:

- *Using Abaqus/CAE to compute shape variations*
- *Design sensitivity analysis of a composite centrifuge*
- *Design sensitivities for tire inflation, footprint, and natural frequency analysis*
- *Design sensitivity analysis of a windshield wiper*
- *Design sensitivity analysis of a rubber bushing*

Using Abaqus/CAE to compute shape variations

Overview

A transitional capability for computing shape variations is provided by the internal Abaqus Scripting Interface command `_computeShapeVariations()`.

Using the command requires some familiarity with the Abaqus Scripting Interface; specifically, the user must understand the Abaqus object model (see *The Abaqus object model*) and know how to access the rootAssembly and partInstance objects.

Defining a shape variation

1. Create and mesh the model in Abaqus/CAE. Output the corresponding input file by selecting **Job->Create** and **Job->Write Input** from the main menu bar in the Job module.

   (In the discussion that follows it is assumed that the model is named `Model-1`, the part is named `Part-1`, and the part instance is named `Part-1-1`).

2. Select **Model->Copy Model** from the main menu bar to copy `Model-1` to `Model-2`, for example. Select `Model-2` from the **Model** list located in the context bar. This model will be used in subsequent steps to compute the shape variation.

3. In the Part module select the part for which a shape variation must be calculated from the **Part** list located under the context bar. Select **Feature->Edit** from the main menu bar to edit the associated sketch. Select **Add->Dimension** and **Edit->Dimension** to change the design parameter. End the edit of the sketch, and indicate that the geometry should be regenerated automatically.

   Editing the sketch will cause the mesh of `Model-2` to be deleted.

4. Use the Abaqus/CAE command line or select **File->Run Script** to execute the commands listed below. The `_computeShapeVariations()` command is accessed from the rootAssembly of either model and requires as input the original part instance, the modified part instance, and the name of the file to which the data lines of the shape variation option will be written. The `.inp` extension will be appended automatically to the specified file name.

   The following sequence of commands is appropriate for the creation of a shape variable associated with a parameter *h*:

   ```
   ra1 = mdb.models['Model-1'].rootAssembly
   ra2 = mdb.models['Model-2'].rootAssembly
   i1 = ra1.instances['Part-1-1']
   i2 = ra2.instances['Part-1-1']
   ra1._computeShapeVariations(originalInstance=i1,
                               modifiedInstance=i2,
                               fileName='shape_h')
   ```

5. To compute the shape variation, the mesh of `Model-1` is mapped to the changed geometry of `Model-2` and then smoothed. Examine the mapped mesh of `Model-2` to verify that the mesh was mapped as expected. (In general, any changes to the geometry should be small—around 1%—so as to avoid difficulties with the mapping of the mesh).

   The shape variation is calculated simply by subtracting the initial node positions from the node positions calculated after mapping and smoothing the mesh to the changed geometry.

6. To use the shape variation data, copy to the input file for your analysis the data written by the `_computeShapeVariations()` command to the `shape_h.inp` file.
7. Verify the correctness of the data by running your analysis and using the Visualization module in Abaqus/CAE to view the shape variation.

Design sensitivity analysis of a composite centrifuge

Products: Abaqus/Standard Abaqus/Design

Overview

This example uses the design sensitivity analysis capability in Abaqus/Design to examine how the key structural responses depend on design parameters such as the thicknesses of composite laminae, the layup angles, the density of the centrifuge end plates, and geometric imperfections.
Many industries use centrifuges to separate out contaminants during purification processes. Efficiency of the purification process is directly related to the speed of rotation. Consequently, the centrifuge chamber is designed to be stiff to maintain its shape and lightweight to reduce self-stressing due to centrifugal loads.

Geometry and model

The centrifuge depicted in *Figure 1* consists of a composite centrifuge chamber and aluminum end plates. The centrifuge chamber is a cylinder 970 mm long and 175 mm in diameter. It spins about its axis at 10000 rpm. The cylinder is a filament-wound composite (hybrid) laminate with a balanced layup. The laminate layup is $(0°/\pm\theta/0°/\pm\theta/0°)$, where 0° represents the layup of the fibers along the length of the cylinder and $\theta = 45°$ is the angle of the helical layers. The axial (0°) and helical layers have a thickness of 0.15 mm and 0.5 mm, respectively. Aluminum alloy end plates are bonded at both ends of the cylinder. The lower plate provides an attachment point for a magnetic bearing, while the top plate supports the magnet of the inductance drive. In the region where the cylinder meets the end plates an additional circumferential (90°) outer layer of 1 mm thickness is added to the laminate. The centrifuge is modeled using reduced-integration, 4-node shell (S4R) elements. The mesh is shown in *Figure 2*. All the displacements on the perimeter of the end plate lip are constrained at one end of the cylinder, and only the radial displacements are constrained at the other end; thus, the centrifuge is allowed to change length freely in the axial direction under loading. A static analysis with centrifugal loading is performed.

Materials

A T800 carbon fiber material is used in the 0° and 90° layers, and a HM400 material is used in the helical layers. The density of both materials is 1600 kg/m^3. The material properties for the composites are given in *Table 1*.
The aluminum alloy has a Young's modulus of E = 70 GPa, a Poisson's ratio of $\nu = 0.33$, and a density of $\rho =$ 2800 kg/m^3.

Design parameters and normalization

The design parameters are thickness of the helical layer (THM400), thickness of the axial layer (TT800), and angle of the helical layer (THETA). Because of the high speed of rotation of the centrifuge, geometric imperfections can have a significant effect on the displacements and stresses. To study this effect, an imperfection in the form of the first bending mode of the centrifuge is used; the magnitude of the imperfection, ALPHA, is chosen as a shape design parameter. An imperfection in the form of a bending mode is chosen because the maximum attainable rotation speed is known to be well-predicted by the natural frequency of the first bending mode of the centrifuge. The gradients of the nodal coordinates with respect to ALPHA required to carry out the sensitivity analysis are obtained from an eigenfrequency extraction analysis. *Figure 3* shows the first bending mode of the centrifuge. Though bending predominates in this mode, there is also a small twisting component.

All plotted sensitivity results are normalized to enable comparison between the parameters. The normalization is carried out by multiplying the response sensitivity by the value of the parameter and dividing by the maximum value of the response. For example, the sensitivity of the stress component S11 with respect to the design parameter THM400 is normalized by first multiplying the sensitivity by the value of the THM400 parameter and then dividing by the maximum S11 value found in the model. For the shape design parameter ALPHA an estimated imperfection of 0.1% of the total length of the cylinder (ALPHA = 1 mm) is used while computing the normalized sensitivities.

Results and discussion

Figure 4 shows the deformed shape of the centrifuge at 10000 rpm. The centrifuge contracts in the axial direction and bulges outward radially. Because of the lap joint where the composite cylinder meets the end plate, some bending is seen at both ends.

The normalized sensitivity of the radial displacement along the length of the composite cylinder (between the two end plates) is plotted in *Figure 5* for all the design parameters except the shape design parameter. The plot shows that the radial displacement has negative sensitivity to THETA and TT800. An increase in the layup angle of the helical layer or an increase in the thickness of the axial layers will stiffen the cylinder and reduce the radial displacement. The positive sensitivity of the radial displacement to THM400 indicates that the added self-induced centrifugal load due to the increase in the mass of the helical layers will more than negate any advantage gained in the stiffness.

Figure 6 plots the sensitivity of the radial displacement to the shape design parameter ALPHA. Since the sensitivity is not axisymmetric for this design parameter, it is plotted for every meridian in 45° increments counterclockwise around the circumference beginning at the 1–2 plane. The radial displacement sensitivities are obtained from the global Cartesian displacement sensitivities output by Abaqus via a two-dimensional vector transformation at each meridional position. The largest sensitivities in *Figure 6* are almost two orders of magnitude higher than those observed in *Figure 5*. This means that the geometric imperfections have a relatively large effect on the radial displacement and that the centrifuge manufacturing process must have a tight tolerance on axial shape imperfections.

The dominant section force in the structure is in the hoop direction of the cylinder. *Figure 7* shows the contour plot of section force in the hoop direction, and *Figure 8* shows the normalized sensitivities of section force in the hoop direction plotted as a function of the position along the length of the cylinder. As expected, only the design parameters that affect the mass (TT800 and THM400) have nonzero sensitivities with THM400 being more sensitive because of its larger thickness. As the centrifuge is allowed to contract freely in the axial direction, the net section force in the axial direction is zero. However, the stresses in the axial direction are not zero. *Figure 9* shows the contour plot of the sensitivity of the fiber stress (S11) in the helical layer with respect to the shape design parameter. To understand in detail how the stresses in the laminate are affected by the design parameters, normalized sensitivities of the fiber stress (S11) through the thickness of the cylinder wall at a point midway along the length of the cylinder are plotted in *Figure 10*. S11 through the thickness of the wall is also plotted after dividing it by the maximum value of S11 in the model. The plot shows that the axial layers are under compression and the helical layers are under tension. S11 has positive sensitivity to TT800: increasing the thickness of the axial layers will reduce the compressive stress in the axial layers and increase the tensile stress in the helical layers. S11 has negative sensitivity to THM400: increasing the thickness of the helical layers will reduce the stress in the helical layers and increase the stress in the axial layers. Increasing the angle of the helical layers will reduce the stress in the axial layers since S11 in the axial layers has a small positive sensitivity to THETA. The sensitivity of S11 to ALPHA is analogous to the sensitivity of the radial displacement to ALPHA around the circumference of the cylinder.

The sensitivities can be used to compute the change that would be required in the design parameters to achieve a particular change in response or to assess the change in the response that would result from a change in the design parameters. Consider, for example, the following objectives: (a) to reduce the compressive stress in the axial layer by 10% and (b) to determine the maximum compressive stress in the axial layer caused by a specified magnitude of the shape imperfection (ΔALPHA = 0.6).

1. The compressive fiber stress in the axial layer is 26.38 MPa. *Figure 10* indicates that the most effective way to achieve the desired reduction is to increase the thickness of the axial layer. The required increase is given by

$$\Delta TT800 = \frac{\Delta S11}{\frac{dS11}{dTT800}} = \frac{2.638}{69.69} = 0.0378 \text{ mm}$$

   An analysis of the centrifuge with TT800 = 0.15 + 0.0378 = 0.1878 mm shows the compressive stress in the axial layer to be 24.05 MPa, which is close to the desired value of 23.74 MPa.

2. The maximum compressive stress in the axial layer caused by the specified shape imperfection can be obtained from

$$S11 = S11 + \Delta ALPHA \frac{dS11}{dALPHA} = 26.38 + 0.6 \times 13.08 = 34.2 \text{ MPa}$$

   An analysis using the imperfect geometry shows the maximum compressive stress in the axial layer to be 34.3 MPa, which is close to the predicted value of 34.2 MPa.

Input files

dsacentrifuge_freq.inp	Frequency analysis of the centrifuge.
dsacentrifuge.inp	Sensitivity analysis of the centrifuge subjected to centrifugal loads.
dsacentrifuge_node_elem.inp	Node and element definition.

Tables

Table 1: Composite material properties.

Material	E_1 (MPa)	E_2 (MPa)	ν_{12}	G_{13} (MPa)	G_{12} (MPa)	G_{32} (MPa)
T800	177000.	14920.	.21	5700.	5700.	5630.
HM400	233967.	14778.	.032	5777.	10191.	5634.

Figures

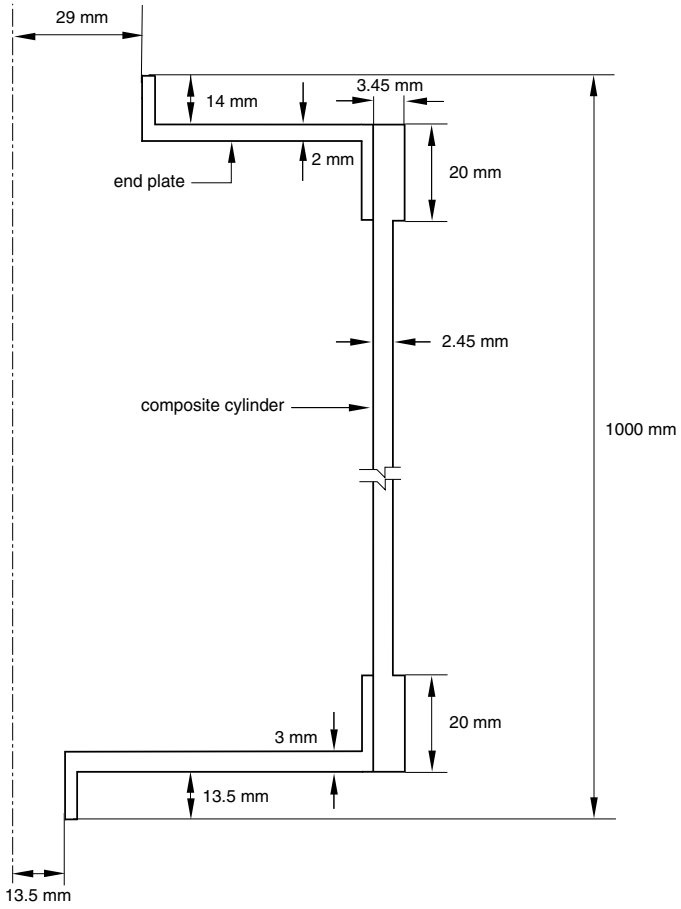


Figure 1: Centrifuge chamber.

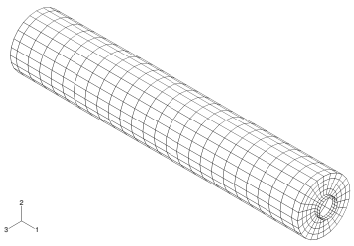


Figure 2: Centrifuge model using S4R elements.

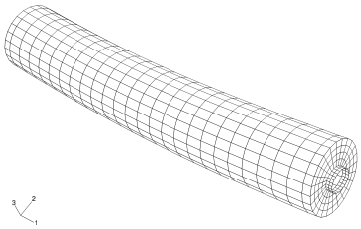


Figure 3: First bending-dominated mode of the centrifuge.

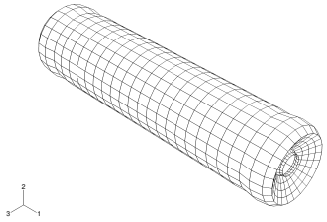


Figure 4: Deformed shape of the centrifuge; deformation scaled 400 times.

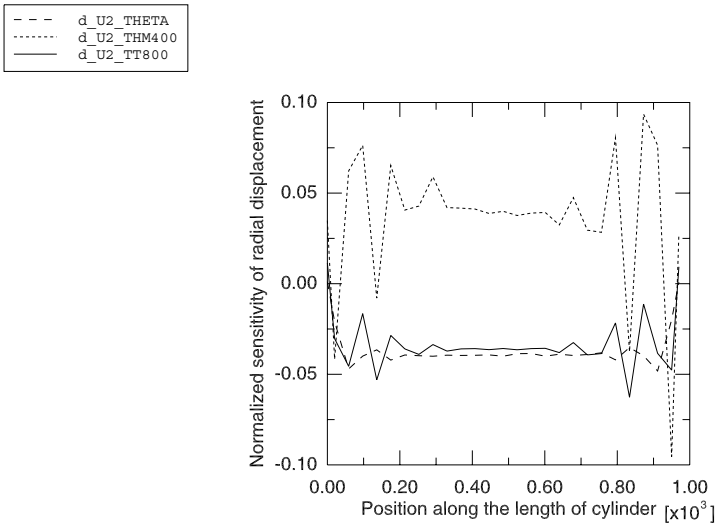


Figure 5: Normalized sensitivity of the radial displacement along the length of the cylinder.

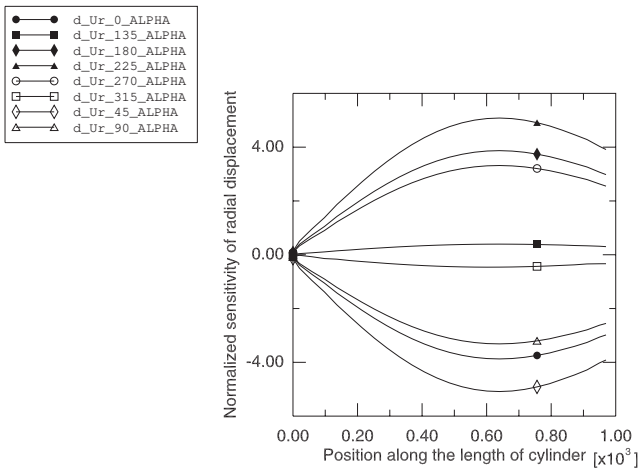


Figure 6: Normalized sensitivity of the radial displacement to the shape design parameter ALPHA. Sensitivities are plotted around the circumference in 45° increments.

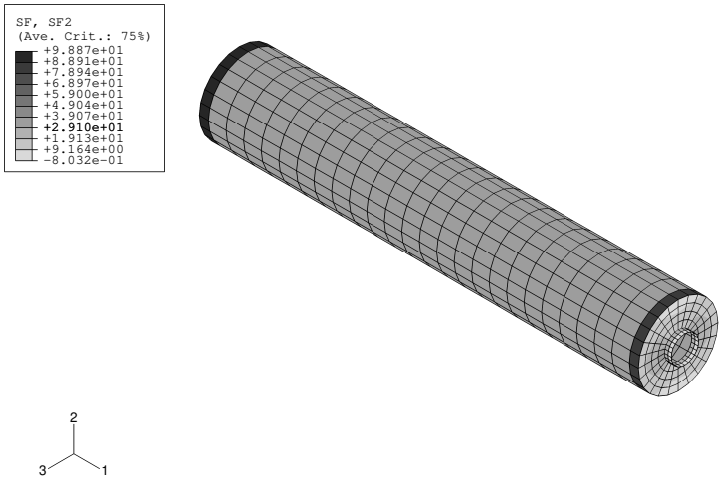


Figure 7: Section force in the hoop direction.

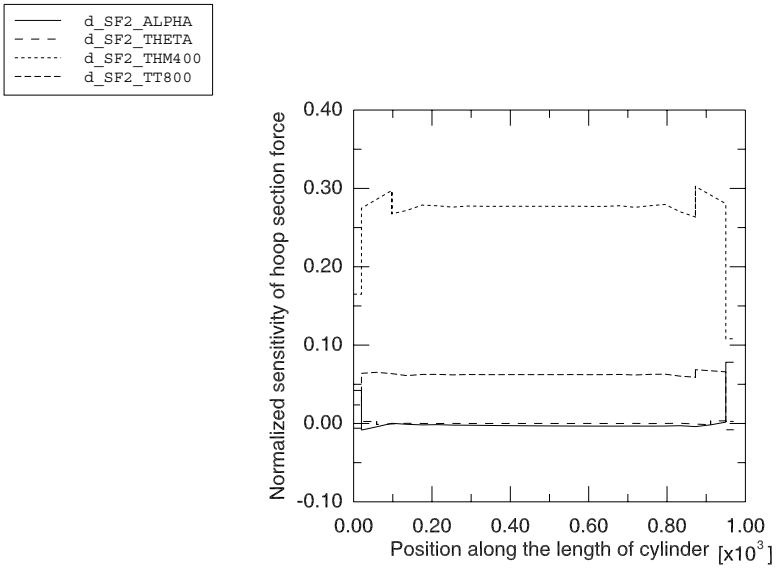


Figure 8: Normalized sensitivity of the section force in the hoop direction.

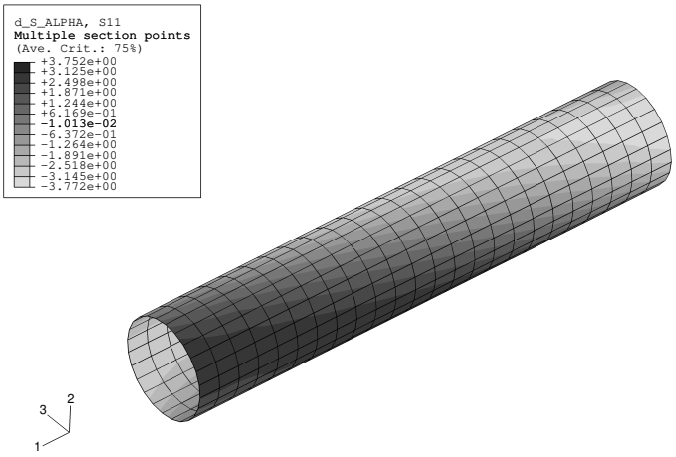


Figure 9: Sensitivity of the fiber stress, S11, in the helical layer to the shape design parameter ALPHA.

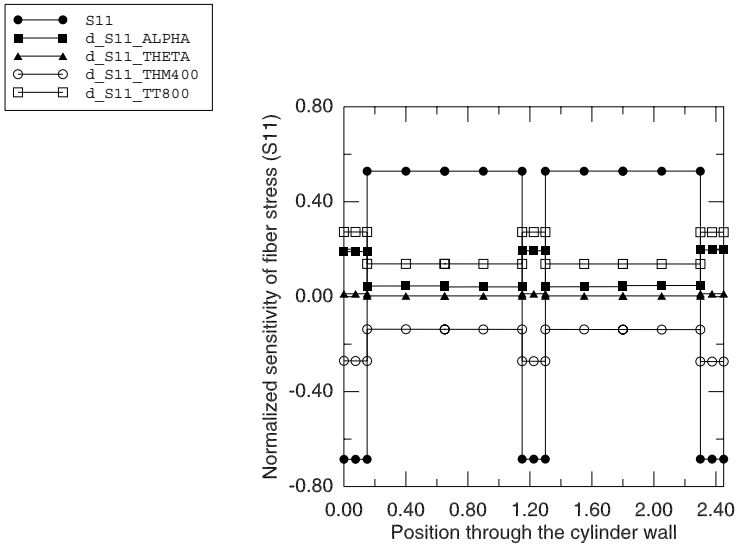


Figure 10: Normalized sensitivity of the fiber stress plotted through the thickness of the cylinder wall for element 590.

Design sensitivities for tire inflation, footprint, and natural frequency analysis

Products: Abaqus/Standard Abaqus/Design

Overview

This example demonstrates the application of design sensitivity analysis (DSA) to tire problems.
The base tire model and analysis is the same as that described in *Symmetric results transfer for a static tire analysis*.
The design sensitivity analysis shows the effects on responses such as contact pressure and natural frequencies of three important design parameters: the thickness of the side wall, the elastic modulus of the belt rebar material, and the elastic modulus of the carcass rebar material. This example demonstrates that the DSA technique in Abaqus/Design can be used effectively for highly nonlinear analyses including features such as viscoelasticity, contact, and rebar.

Geometry and model

The geometry and model information for this example is identical to that for *Symmetric results transfer for a static tire analysis*. However, because symmetric model generation and symmetric results transfer are not available in a design sensitivity analysis, the full three-dimensional model is constructed and analyzed in one model. In addition, the convergence tolerance on the residual is tightened to improve the accuracy of the tangent stiffness, thereby providing more accurate sensitivities (see *Direct Design Sensitivity Analysis*). In addition to the inflation step and the two footprint analysis steps (displacement control and load control), a frequency extraction is appended as the last step. A sensitivity analysis is performed in each step with contact pressure as the design response in the static steps and frequency as the design response in the final frequency extraction step.

Three primary design parameters are chosen for this problem. The first is the "nominal" thickness, t_0, of the tire in the region of the sidewall. The specific region affected by t_0 is shown in the symmetric portion of the tire cross-sectional view in *Figure 1*. This region consists of one layer of elements, and the thickness t_i in the discretized model is taken as the distance between an outer node and the corresponding inner node. The thickness between each of the pairs of nodes in this region is related to the nominal thickness by $t_i = f \quad t_0$, where f is defined as the ratio t_i/t_0. Thus, a change in the design parameter t_0 causes the thicknesses t_i to change proportionally (the actual value of t_0 is not important; however, for the purposes of normalization of the results, as discussed below, a value of 1 is used). A constraint on the change in the nodal coordinates due to a change in the design parameter is specified such that the outer nodes stay fixed and the inner nodes move inward along the original vectors connecting the outer nodes to the inner nodes. *Figure 1* depicts the shape of the tire cross-section that would result from a 50% change in nominal thickness. The other two design parameters chosen are the elastic modulus of the belt rebar material, E_{belt}, and the elastic modulus of the carcass rebar material, E_{carc}. The density of the rubber material, ρ_{rubber}, is also included as a design parameter for demonstration and verification purposes.

Perturbation sizes for the finite differencing operations

As documented in *Direct Design Sensitivity Analysis*, Abaqus uses the semi-analytic approach to compute sensitivities, and this approach involves finite differencing computations at the element level. By default, Abaqus automatically determines for each element the perturbation sizes of the design parameters to be used in the finite differencing computations through a heuristic perturbation sizing algorithm. The perturbation sizes for the elements with the most contribution to the sensitivities are written to the message file. Since this algorithm can be expensive, DSA controls are provided to allow the user to directly supply perturbation sizes. If appropriate

perturbation sizes are not known in advance, a smaller problem can be run with the default perturbation sizing algorithm, and the perturbation sizes determined by Abaqus can then be inserted into the larger problem.

This technique for obtaining perturbation sizes is adopted in this example. A smaller axisymmetric problem (*dsatire_axi_half.inp*) with one inflation (static) step and one frequency step is run. The perturbation sizes reported in the message file for each step of the smaller problem are subsequently specified in the full model (*dsatire.inp*) using the DSA controls.

Normalization of sensitivity results

In the subsequent plots of results, the sensitivities are normalized so that they can be compared side-by-side. Except as noted, the contour plots of contact pressures (CPRESS) and corresponding sensitivities are normalized by dividing by the maximum contact pressure at the end of the last static step, $C_{\max}$. In addition, the sensitivities of CPRESS are multiplied by the value of the design parameter. For example, the sensitivity of CPRESS with respect to $\boldsymbol{t_0}$, d_CPRESS_tNominal, is normalized as $\boldsymbol{t_0} \times$ d_CPRESS_tNominal/$C_{\max}$. The eigenfrequency (EIGFREQ) sensitivities are divided by the first eigenfrequency value and, as for CPRESS, multiplied by the value of the appropriate design parameter.

Results and discussion

The contact pressure on the tire footprint can be considered an important factor in tire handling and wear properties. As such, the results of the sensitivity analysis are discussed in terms of the contact pressure distribution on the tire footprint. *Figure 2* shows the actual values of the contact pressure on the full tire model looking from below at the end of the static footprint analysis. *Figure 3* shows the normalized contact pressure distribution on a blown-up region of the full tire. As can be seen from these plots, the contact pressure is less in the center of the footprint than in the surrounding region. The maximum contact pressure occurs at node 2645, and the center of the footprint is at node 3055. These nodes are indicated on the plots, and *Figure 4* shows the time history of CPRESS for these nodes.

The objective is to use the sensitivity results to determine how the design parameters can be modified to distribute the contact pressure more evenly so that the center of the tire picks up more of the load. *Figure 5* to *Figure 7* show the distributions of the normalized contact pressure sensitivity for each of the design parameters. Large variations in the sensitivities away from the center of the footprint are observed in the sensitivity contour plots. These variations can be attributed to the high gradients in contact pressure in the region surrounding the center of the footprint (see *Figure 3*), because even small changes in the footprint size (due to small changes in the design parameters) can lead to relatively large changes in the contact pressure. These sensitivity plots show that to increase the contact pressure at node 3055 (footprint center), the value of E_{carc} should be increased, the value of E_{belt} should be increased, and the value of $\boldsymbol{t_0}$ should be decreased. However, based on the magnitudes of the normalized sensitivities, E_{belt} has the most influence on the contact pressure at the center node. Accordingly, a new design is investigated in which the design parameter E_{belt} is increased by 5%. *Figure 8* shows the distribution of contact pressure for the new design, and *Figure 9* shows the time history of CPRESS at nodes 2645 and 3055 for the new design. These figures indicate that the contact pressure has increased in the center of the tire without appreciably affecting the surrounding contact pressure distribution; the actual increase is 3.20%. The predicted increase based on the (first-order) sensitivities is 2.32%, which is reasonably close to the actual increase considering the high degree of nonlinearity in this problem.

Figure 10 to *Figure 14* show the first five modes of the tire. The study of the frequency sensitivities provides insight into the dynamic behavior of the design. For example, we can conclude from *Figure 15* and *Figure 17* that the frequency of mode 2 is most sensitive to the sidewall shape, $\boldsymbol{t_0}$, and less sensitive to the Young's modulus of the sidewall reinforcement, E_{belt}. In retrospect, this makes good physical sense because mode 2 is primarily shear of the sidewall (see *Figure 15*), but this conclusion may have been much more difficult to formulate without the sensitivity information. *Figure 16*, *Figure 17*, and *Table 1* show the values of the eigenfrequency and corresponding normalized sensitivities for the first five modes of the tire. The eigenfrequencies for the new design (5% change in E_{belt} as discussed above) are the same (to five significant figures) as for the original model.

This behavior is accurately predicted in *Figure 17*, where it can be seen that the eigenfrequencies are essentially independent of this design parameter.

Given the highly nonlinear nature of this problem, the user is cautioned against using the sensitivities beyond their useful limit. The sensitivities are first-order derivatives; therefore, using them to predict large changes in the design parameters is not valid, since higher-order terms are not considered. In addition, using sensitivity results to predict the outcome of simultaneous changes to the design parameters assumes that superposition is valid, which again is true only for small changes in the design parameters. For example, if the design parameters E_{carc} and E_{belt} are increased by 5% and t_0 is decreased by 5%, the contact pressure at the center of the footprint increases by 4.35%, which is nearly twice the predicted change of 2.42%. This implies that a simultaneous change of 5% is too large for predicting the net effect on the contact pressure based on the sensitivities.

Input files

dsatire_axi_half.inp	Axisymmetric model with inflation and frequency analysis for obtaining perturbation sizes.
dsatire_axi_half_node.inp	Nodal coordinates for axisymmetric model.
dsatire_axi_half_psv.inp	Parameter shape variation data for axisymmetric model.
dsatire.inp	Full model including inflation, footprint, and frequency analysis.
dsatire_model.inp	Model data for full model.
dsatire_psv.inp	Parameter shape variation data for full model.

Tables

Table 1: Eigenfrequency sensitivities for the first five modes.

Mode	Eigenfrequency	Normalized eigenfrequency sensitivity with respect to:			
		t_0	ρ_{rubber}	E_{belt}	E_{carc}
1	51.6	1.56E−1	−5.00E−1	−1.84E−3	1.39E−2
2	52.9	2.18E−1	−5.13E−1	−1.81E−5	4.97E−3
3	58.7	1.45E−1	−5.69E−1	2.27E−3	1.53E−2
4	60.6	1.34E−1	−5.88E−1	1.75E−3	1.54E−2
5	84.1	1.69E−1	−8.15E−1	3.48E−3	1.86E−2

Figures

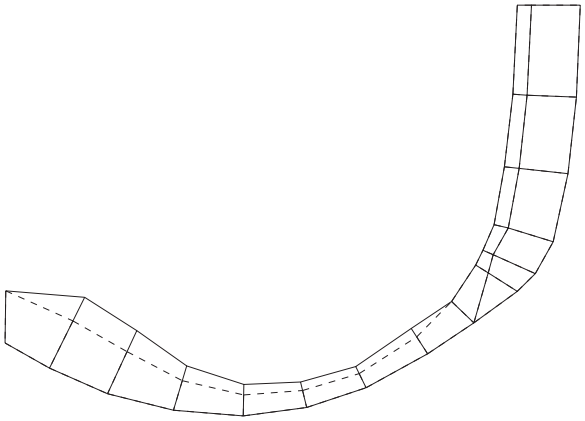


Figure 1: Effect of 50% change in design parameter t_0 on tire cross-section geometry (dashed line represents original geometry).

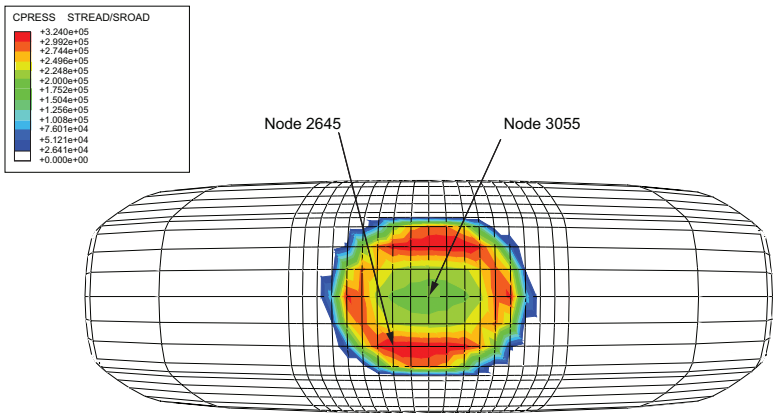


Figure 2: Contact pressure distribution on full tire at end of footprint (static) analysis.

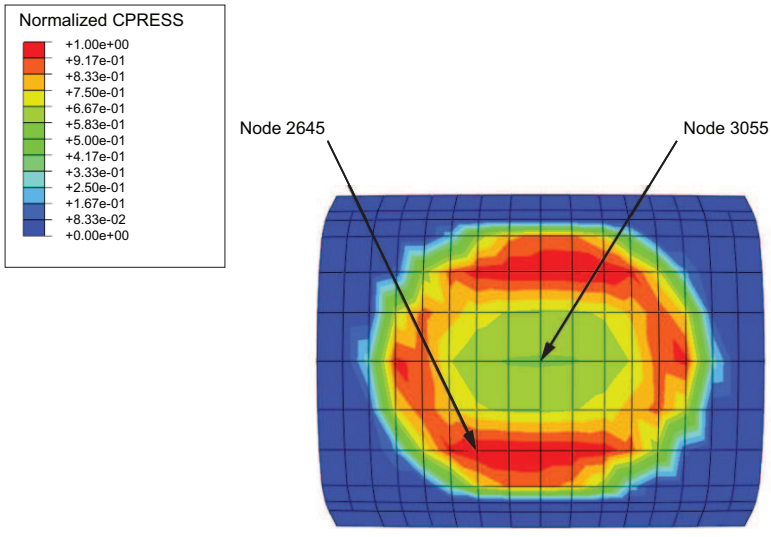


Figure 3: Normalized contact pressure distribution on blown-up view of tire footprint at end of footprint analysis.

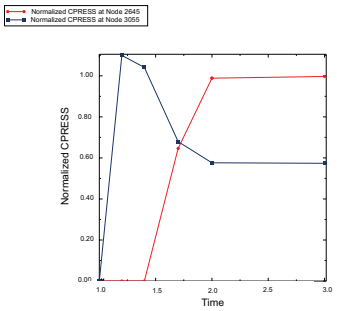


Figure 4: Normalized contact pressure history for center node (3055) and node with maximum contact pressure at end of footprint analysis (2645).

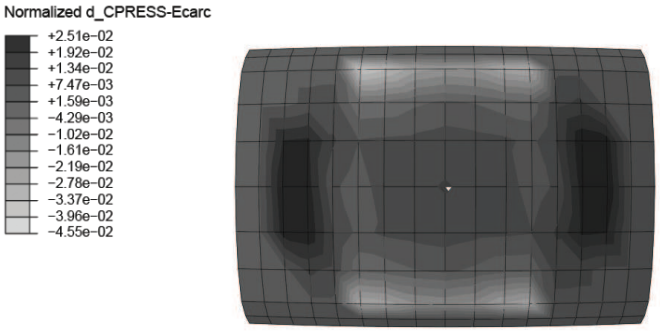


Figure 5: Distribution of normalized contact pressure sensitivity with respect to E_{carc} at end of footprint analysis.

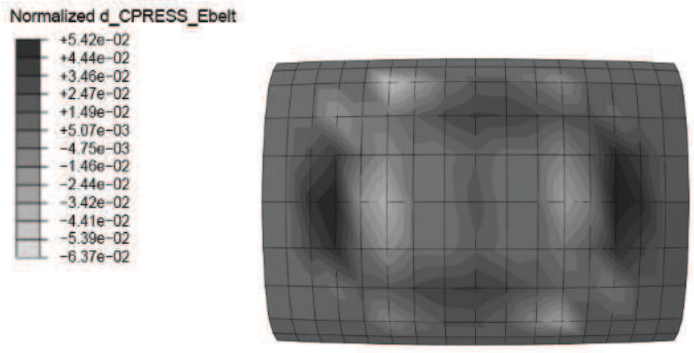


Figure 6: Distribution of normalized contact pressure sensitivity with respect to E_{belt} at end of footprint analysis.

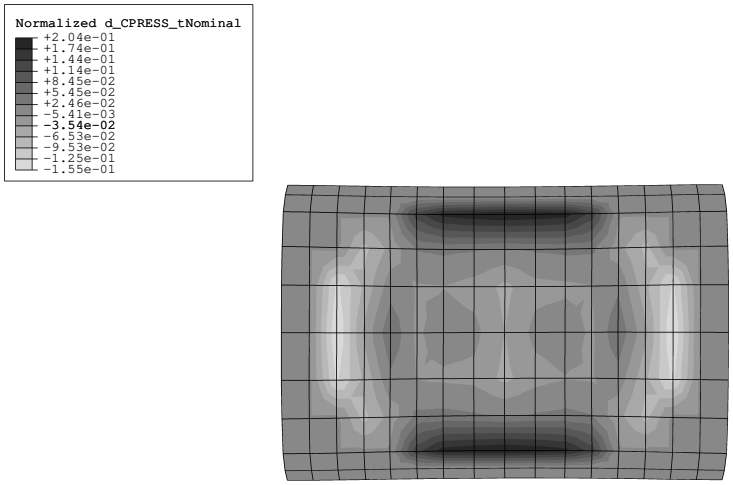


Figure 7: Distribution of normalized contact pressure sensitivity with respect to t_0 at end of footprint analysis.

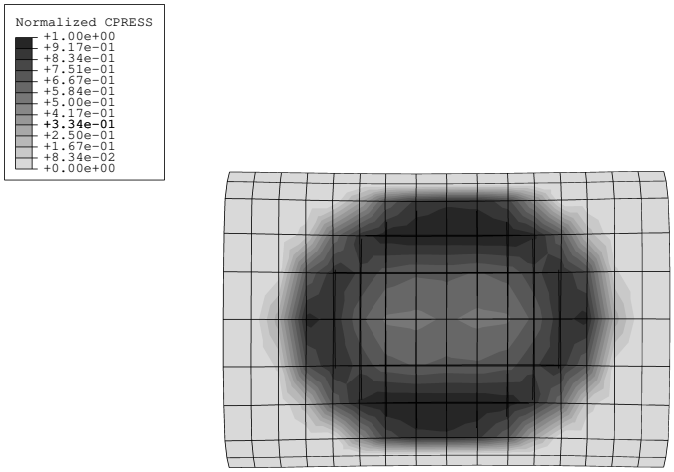


Figure 8: Normalized contact pressure distribution for new design (5% increase in E_{belt}).

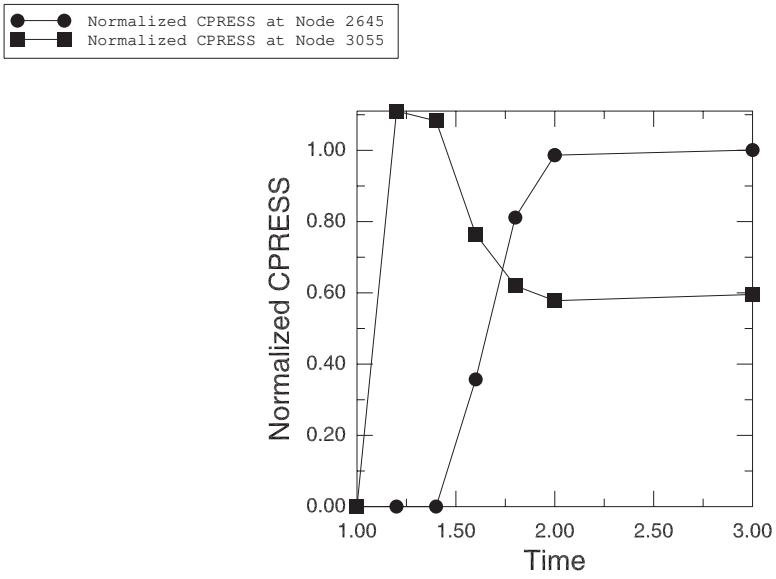


Figure 9: Normalized contact pressure history at nodes 3055 and 2645 for new design.

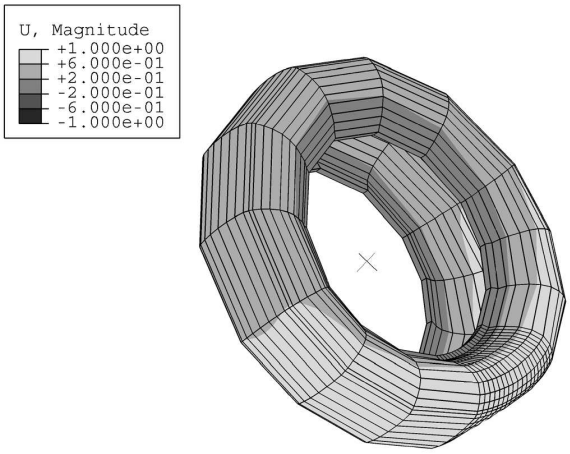


Figure 10: Mode 1.

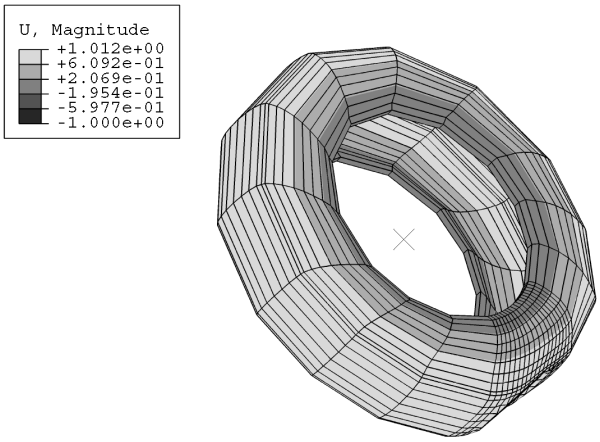


Figure 11: Mode 2.

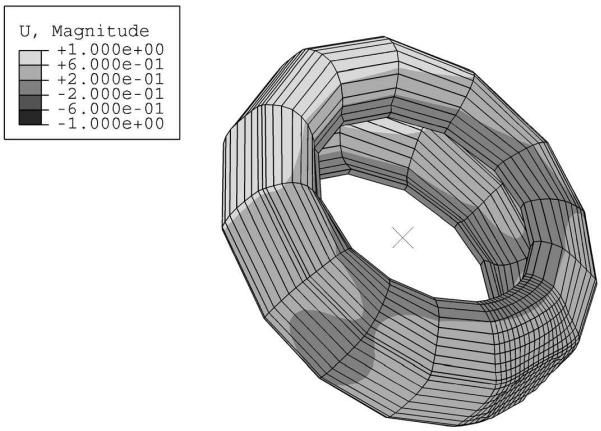


Figure 12: Mode 3.

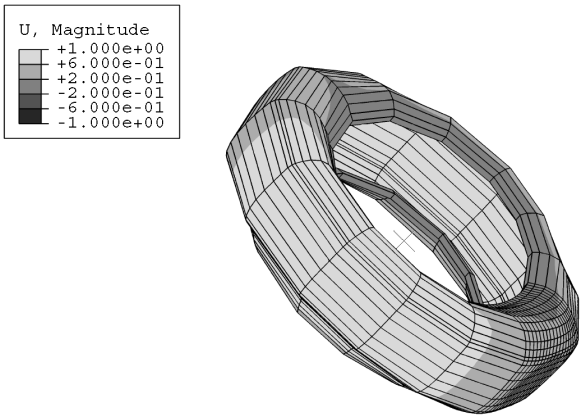


Figure 13: Mode 4.

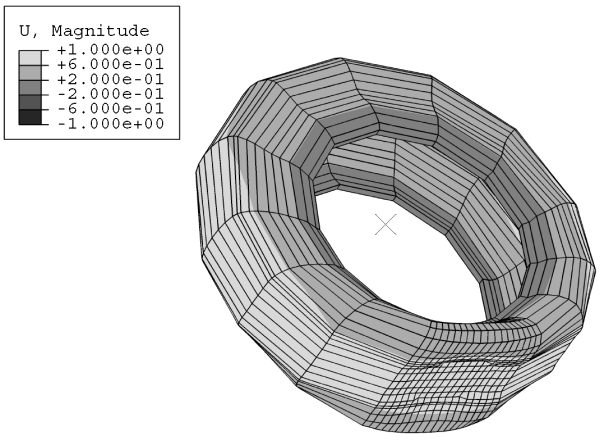


Figure 14: Mode 5.

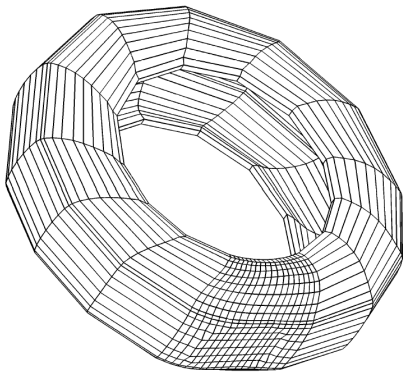


Figure 15: Mode 2 (wire frame).

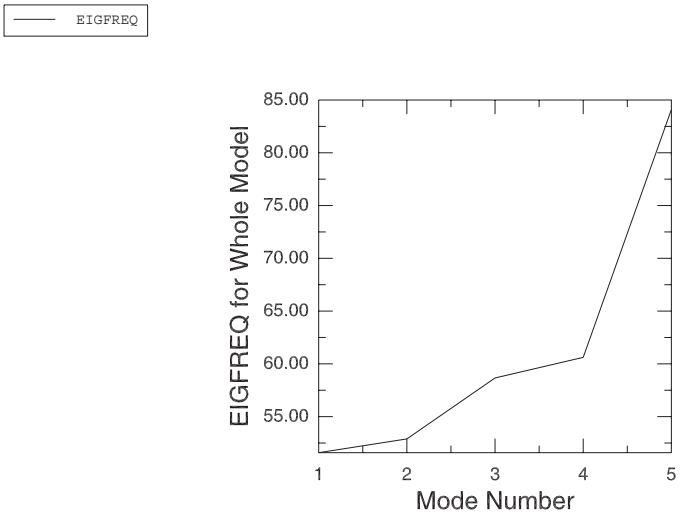


Figure 16: Eigenfrequencies for first five modes.

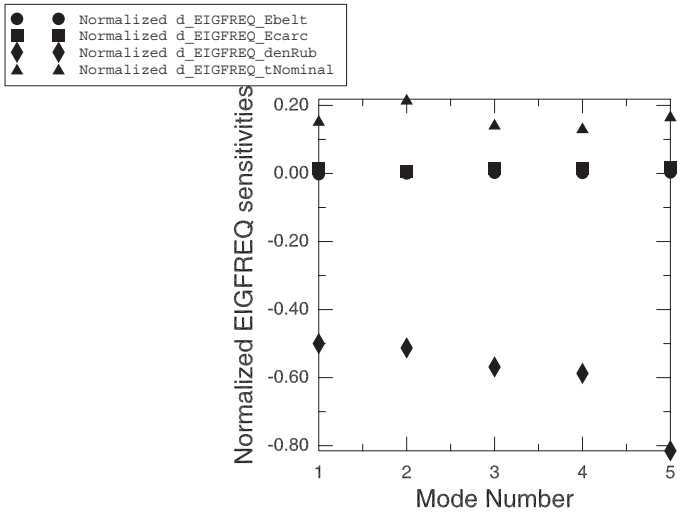


Figure 17: Normalized eigenfrequency sensitivities for first five modes.

Design sensitivity analysis of a windshield wiper

Products: Abaqus/Standard Abaqus/Design

Overview

This example shows the design analysis of a statically loaded windshield wiper blade.

The conventional wiper system is composed of three major subsystems: the blade-arm assembly, a linkage mechanism, and the electric motor. We restrict our design and analysis to the blade-arm subsystem. Wiping performance is determined by the dynamic performance of the wiper system, while safety regulations require that the blade wipe a specific area on the windshield surface. It is a major challenge to achieve systems that optimize wiping performance and satisfy the area conditions. The interaction between the force/deflection response of the blade (rubber) element and the friction at the blade-glass interface is a critical element in design. A design sensitivity analysis of the rubber-glass interaction gives insight that can be used to improve the design before undertaking the more complex dynamic stick-slip analysis. The sensitivity analysis also identifies the design parameters that are most effective in reducing the stress in highly stressed areas. Self-contact between the flanges of the rubber element is not considered in this example.

Geometry, model properties, and design parameters

The geometry of the rubber blade is shown in *Figure 1*. The arm is assumed to be stiff relative to the rubber blade and is modeled by prescribing fixed boundary conditions. The rubber blade is modeled as a plane strain model with 231 first-order hybrid elements. An incompressible hyperelastic material with a polynomial strain energy function is used to simulate the rubber material behavior. The windshield is assumed to be rigid and is modeled as an analytical rigid surface. Surface interaction between the blade and the windshield is modeled using finite-sliding contact with an isotropic Coulomb friction coefficient of 0.2. Three shape parameters and the coefficient of friction, μ, are chosen as design parameters for the design study. We choose μ as a design parameter to study the effect of the friction coefficient on the blade-glass interface forces. The shape parameters that can be modified without requiring geometry changes to other parts of the assembly are chosen as design parameters. These shape parameters will help us to study the effect of aspects of wiper geometry on wiper stiffness. The shape parameters are the thickness, $t2$, of the neck between the bottom two flanges; the thickness, $t3$, of the wiper tip; and the width, $h2$, of the lower flange. The corresponding shape gradients (the derivatives of the nodal coordinates with respect to the shape design parameters) required as input to the sensitivity analysis are shown as symbol plots in *Figure 2*, *Figure 3*, and *Figure 4*. These gradients are input using parameter shape variation and calculated using a custom Python script derived from the Abaqus/CAE replay file. This approach requires familiarity with the Abaqus Scripting Interface. Shape variation data can be generated more easily using Abaqus/CAE; details are given in *Using Abaqus/CAE to compute shape variations*.

Loading, boundary conditions, and design responses

An interference fit of 2.0 units between the blade and the windshield is used to simulate the static vertical load between them. The windshield is held fixed in the vertical direction, and the surface nodes of the upper flange are constrained in both directions. The wiping motion is simulated by prescribing a horizontal displacement of 9.0 units to the reference node of the windshield.

It is desirable to reduce the stress concentration in the rubber element since high stresses will degrade the lifecycle of the blade and affect performance. Consequently, the sensitivity of the von Mises stress is taken as the primary design response in this model. Any design change should take into account the behavior of the contact pressure between the wiper and the windshield. To this end the sensitivities of CSTRESS are also selected as design

responses. Including friction makes the stiffness matrix unsymmetric; hence, the analysis is run using the unsymmetric solver. Since we are interested in the history of the contact pressure at the tip, the incremental DSA formulation is chosen. To quantify the effect of neglecting the unsymmetric terms and of using total DSA, we also compare the maximum sensitivities for various combinations of DSA formulation and stiffness matrix symmetrization.

Results and discussion

A contour plot of the von Mises stress distribution on the deformed configuration of the rubber blade is shown in *Figure 5*. It is evident that the two necks between the flanges are the areas of interest. In a magnified contour plot of the von Mises stress distribution in the necks (*Figure 6*) we can see that the stress concentration in the lower neck is higher than in the upper one. The normalized sensitivities of the maximum von Mises stresses in the lower neck are listed in *Table 1*. Sensitivities are normalized by dividing them by the maximum Mises value (67857.7 units in element 131 of the wiper blade) and then scaling the result by the initial value of the design parameter. We notice that the shape parameters have a direct effect (increasing the parameter increases stress and vice versa) and the friction coefficient has an inverse effect on the von Mises stress. Reducing the thickness of the neck makes the wiper more flexible and reduces stress concentration. Increasing the friction coefficient increases the shear force at the tip. This increased shear force increases bending of the upper neck and decreases bending in the lower neck, thus reducing the stress concentration in the lower neck. We infer from the table that the von Mises stress is more sensitive to $t2$ and $t3$ than to the flange width and friction coefficient. To reduce stress concentration, a design with changes in $t2$ and $t3$ is considered. A 20% reduction in $t2$ and a 10% reduction in $t3$ is selected. The predicted percentage reduction in the von Mises stress is 6.76 ($0.2832 \times 0.2 + 0.1094 \times 0.1$).

The proposed design changes will affect the contact pressure at the wiper-windshield interface, and the sensitivity results are used to quantify this effect. The time history of the normalized sensitivities of contact pressure at a point at the tip is shown in *Figure 7*. Contact pressure at this point is most sensitive to $t2$. This result is somewhat unexpected since this design parameter does not affect the interface explicitly. $t2$ has a directly proportional effect, while $t3$ has an inversely proportional effect on the contact pressures. Thus, decreasing $t2$ will decrease the contact pressure while decreasing $t3$ will increase the contact pressure. The contact pressure histories for the proposed design and the current design are shown in *Figure 8*. The plots indicate that the reduced stress concentration in the proposed design comes at the cost of decreased wiper performance. (The proposed design has a lower contact pressure, which could be suboptimal.) If desired, the contact pressure can be positively influenced by considering a positive change in $t3$.

All results discussed above are obtained in an incremental DSA analysis using the unsymmetric solver. If the user is interested only in the results at particular increments (typically, the last increment), total DSA will be advantageous computationally. Similarly, neglecting the unsymmetric terms will improve computational efficiency. To quantify the error in such approximations, we compare the results for various combinations to the overall finite difference method (OFD). *Table 2* lists the relative error in the maximum Mises and contact pressure sensitivities compared to the results obtained using the OFD method. For the dominant shape parameter, $t2$, the sensitivity results of the maximum von Mises stress are in good agreement with the OFD method for all the combinations. However, neglecting the unsymmetric terms in total DSA gives inaccurate results for the contact pressure sensitivities. The relative error is large for the less dominant shape parameters, with total DSA giving poor results for both the Mises and contact pressure sensitivities. Total DSA gives inaccurate results for sensitivities with respect to the friction coefficient. Although the sensitivity results are problem dependent, we can infer that total DSA may give poor results if we neglect unsymmetric terms. Further sensitivities of interaction pressures are more sensitive to approximations than the sensitivities of maximum stresses in the structures. Less significant sensitivities are affected more by approximations than the dominant sensitivities.

The proposed design changes are made and the analysis is rerun. A 9% reduction in the stress concentration is observed. Predicted and actual contact pressure histories are shown in *Figure 9*. The difference between the predicted and the actual results is due to the nonlinear dependence of the design response on the design parameter.

However, the rerun confirms the prediction that the proposed design is more robust, with a possibly suboptimal wiping pressure.

Input files

dsawiper.inp	Primary wiper blade design sensitivity analysis.
dsawiper_pred.inp	Wiper blade analysis incorporating design changes.
dsawiper_node.inp	Node description.
dsawiper_elem.inp	Element description.
dsawiper_t2.inp	Parameter shape variation data for thickness $t2$.
dsawiper_t3.inp	Parameter shape variation data for thickness $t3$.
dsawiper_h2.inp	Parameter shape variation data for width $h2$.

Tables

Table 1: Normalized sensitivities of maximum stress values.

Design Parameter	Normalized Sensitivity
$t2$	0.2832
$t3$	0.1094
$h2$	0.0154
μ	–0.0026

Table 2: Percentage error in maximum stress and contact pressure sensitivities.

Design Parameter	Incremental DSA; Unsymmetric Solver		Incremental DSA; Symmetric Solver		Total DSA; Unsymmetric Solver		Total DSA; Symmetric Solver	
	% Error MISES	% Error CPRESS	% Error MISES	% Error CPRESS	% Error MISES	% Error CPRESS	% Error MISES	% Error CPRESS
$t2$	0.02	0.00	0.06	0.24	0.02	0.11	2.4	22.15
$t3$	0.01	0.01	0.02	0.12	1.81	61.56	1.70	53.20
$h2$	0.37	0.00	12.35	0.13	1.12	94.41	12.36	93.01
μ	1.51	0.88	1.97	0.43	51.6	0.88	28.90	21.21

Figures

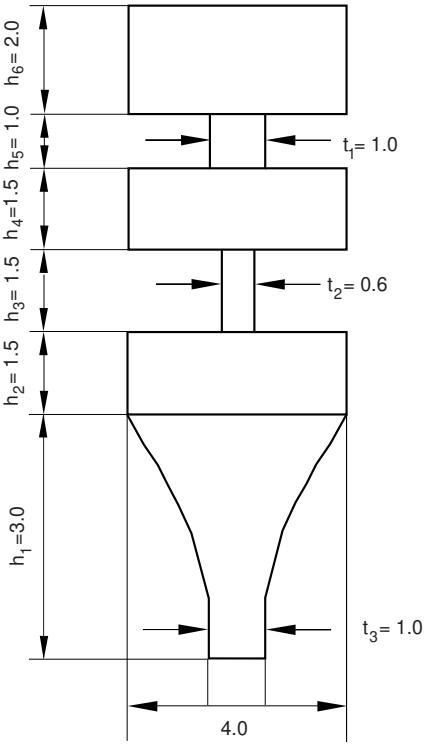


Figure 1: Model of the rubber windshield wiper.

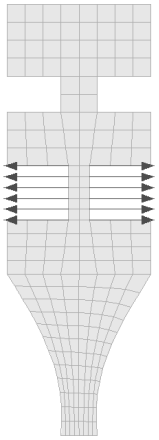


Figure 2: Symbol plot of the shape variation with respect to $t2$.

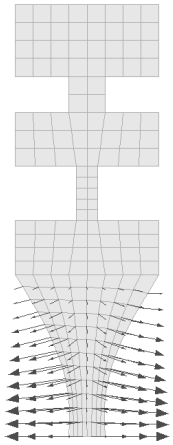


Figure 3: Symbol plot of the shape variation with respect to $t3$.

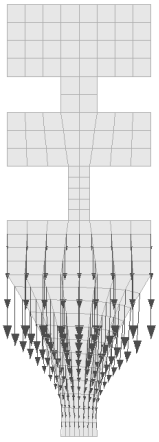


Figure 4: Symbol plot of the shape variation with respect to $h2$.

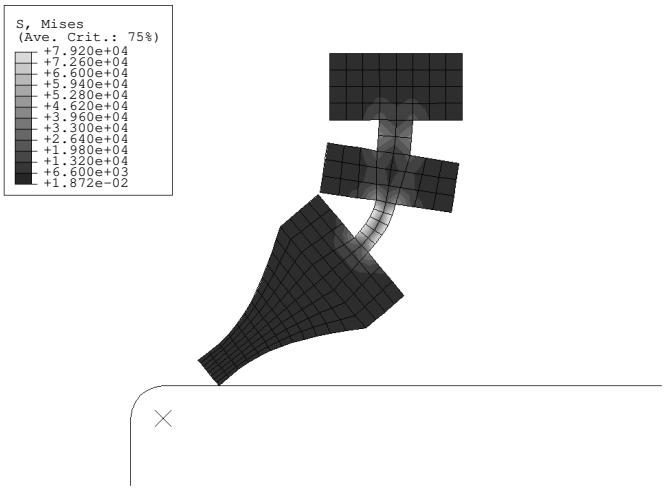


Figure 5: Contour plot of von Mises stress.

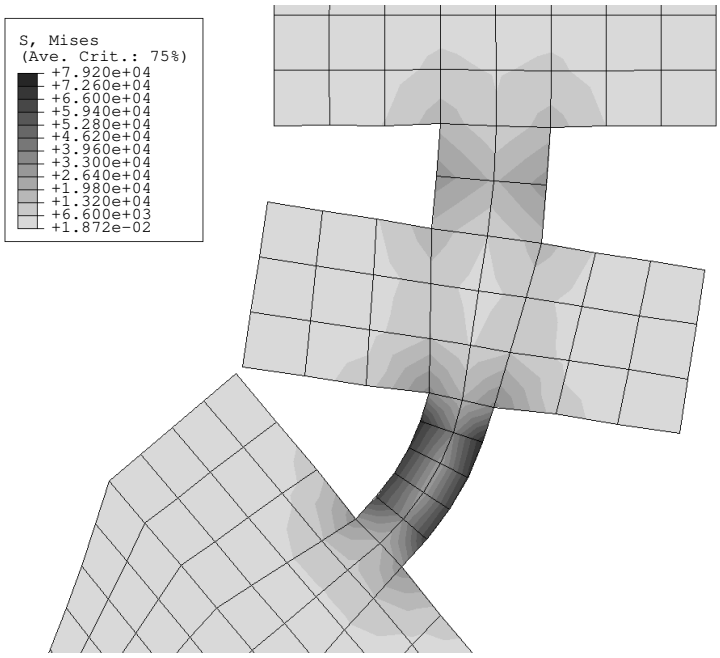


Figure 6: Magnified contour plot of von Mises stress.

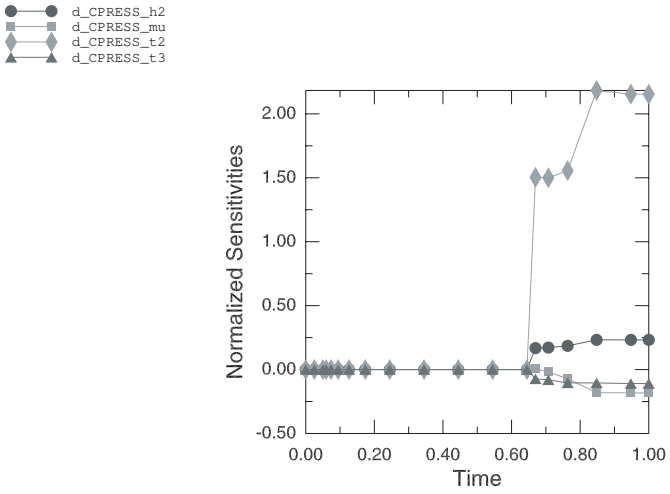


Figure 7: Time history of the normalized sensitivity of contact pressure at the wiper tip.

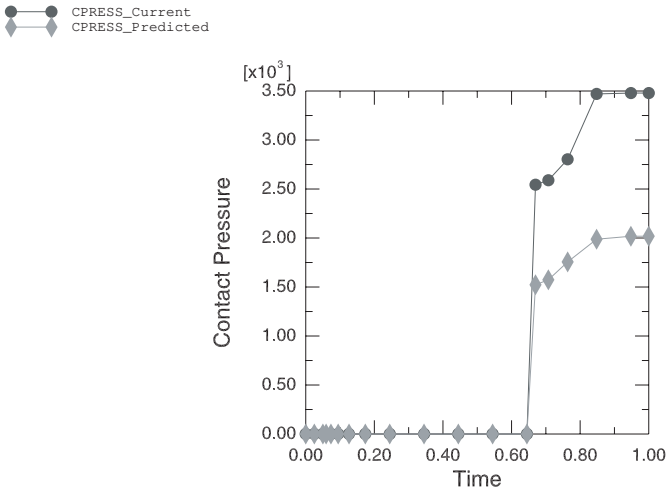


Figure 8: Current and predicted contact pressure history.

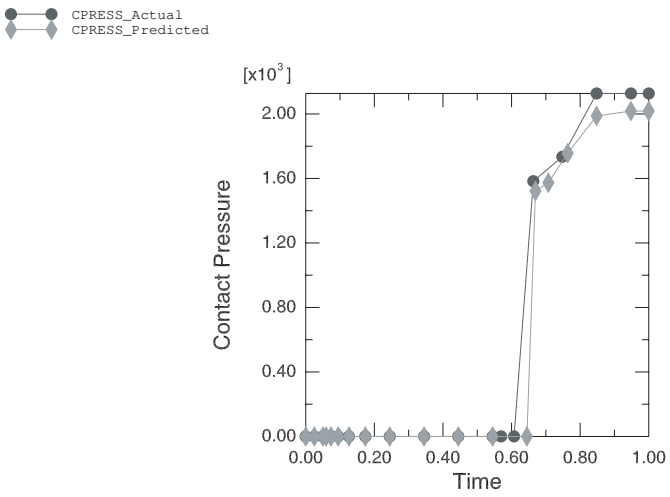


Figure 9: Actual and predicted contact pressure history.

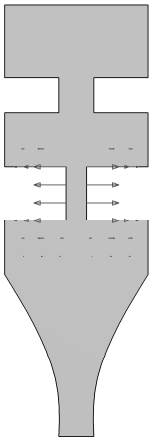


Figure 10: Symbol plot of the computed shape variation with respect to $t2$.

Design sensitivity analysis of a rubber bushing

Products: Abaqus/Standard Abaqus/Design

Overview

This example demonstrates how design sensitivity analysis can be used to improve the design of a rubber bushing. The objective is to alter the bushing geometry to lower the maximum axial stress, thus increasing the service life. Design sensitivity analysis provides a means of predicting the effect of a change in the geometry on the stress concentration, thereby aiding in identifying the important design parameters and determining an appropriate design change.

Geometry, model properties, and design parameters

The bushing consists of inner and outer steel tubes that are bonded to a central rubber cylinder (*Figure 1*). It is assumed that the outer perimeter of the bushing is fully fixed. The bushing is 457.2 mm (18.0 in) long, with an outside diameter of 508.0 mm (20.0 in) and an inside diameter of 228.6 mm (9.0 in). The steel is elastic with Young's modulus = 206.0 GPa (3.0×10^7 psi) and Poisson's ratio = 0.3. The rubber is modeled as a fully incompressible hyperelastic material that at all strain levels is relatively soft compared to the steel. The nonlinear elastic behavior of the rubber is described by a strain energy function that is a second-order polynomial in the strain invariants.

The model is discretized with standard axisymmetric elements since the axial loading results in pure axisymmetric deformation. CAX4 elements are used for the steel components, and CAX4H elements are used for the rubber component. Rigid elements (element type RAX2) are attached to the inside of the bushing in both models to represent the relatively stiff shaft. The use of these elements also simplifies the application of the loading conditions. The axisymmetric finite element mesh is shown in *Figure 1*. An axial force of magnitude 10675.0 N (2400.0 lbs) is applied to the rigid body reference node, while the outer steel tube is fully fixed.

Two design parameters are considered for the design sensitivity analysis: the thickness, t, and the fillet radius, r, of the rubber bushing at the top and bottom ends where it is bonded to the inner steel tube, which is shown in *Figure 1*. These parameters represent typical geometry properties that may be considered during design evaluation.

Design sensitivity analysis and design parameters

To carry out a design sensitivity analysis with respect to a shape design parameter, the gradients of the nodal coordinates with respect to the design parameter must be specified with parameter shape variation. One simple approach to obtaining these gradients is to perturb the shape design parameters r and t one at a time and to record the perturbed coordinates. The gradients are then found by numerically differencing the initial and perturbed nodal coordinates. In the current study the constraint is imposed that a change in the thickness causes the line of nodes connecting the thickness dimension to the fillet radius to rotate about the point of tangency of this line to the fillet radius. The Abaqus Scripting Interface command `_computeShapeVariations()` provides a semiautomated facility to compute the shape variations (see *Using Abaqus/CAE to compute shape variations*).

Results and discussion

The deformed mesh is shown in *Figure 2*. *Figure 3* shows the contours of axial stress in the rubber part of the bushing at the end of the axisymmetric analysis. The maximum stress occurs near the top fillet close to the axis. *Figure 4* and *Figure 5* show the contours of the sensitivities of the axial stress for the shape design variables r and t, respectively. *Table 1* shows the normalized sensitivities of the maximum axial stress, $\sigma_{max} =$ 0.17 MPa

(24.55 psi), with respect to the shape design variables. The normalization has been carried out by multiplying the sensitivities by a characteristic dimension (initial fillet radius $r_0 =$ 12.7 mm (0.5 in) and initial thickness $t_0 =$ 15.24 mm (0.6 in)) and dividing by the maximum stress. As can be inferred from this table, a change in the fillet radius influences the maximum stress to a larger extent than a change in the thickness of the rubber. Hence, it is desirable to change *r* to modify the stresses. To obtain approximately a 10% reduction in the maximum stress in the axial direction, the fillet radius is increased by

$$\Delta r = \frac{\Delta \sigma_{\max}}{\frac{d\sigma_{\max}}{dr}}.$$

Substituting $\Delta \sigma_{\max} = 0.1\sigma_{\max}$ and $d\sigma_{\max}/dr =$ 0.008 MPa/mm (28.75 psi/in) (see *Figure 4*) gives $\Delta r =$ 2.25 mm (0.09049 in). A reanalysis of the problem with the radius changed to $r_0 + \Delta r =$ 14.99 mm (0.59049 in) yields a reduction of 8.8% in the maximum axial stress, which is slightly less than the goal of 10%. This is expected because of the nonlinearity of the problem; to achieve the 10% reduction, this process would have to be repeated, which is essentially an optimization problem.

Input files

bushing_cax4_axi_dsa.inp	Design sensitivity analysis for the axisymmetric model.
bushing_node.inp	Node definitions.
bushing_steel.inp	Element definitions for the steel.
bushing_rubber.inp	Element definitions for the rubber.
bushing_rigid.inp	Element definitions for the rigid body.

Tables

Table 1: Normalized sensitivities of the maximum stress.

Parameter	$\frac{r_0}{\sigma_{\max}} \frac{d\sigma_{\max}}{dr}$	$\frac{t_0}{\sigma_{\max}} \frac{d\sigma_{\max}}{dt}$
r	–0.58	—
t	—	–0.11

Figures

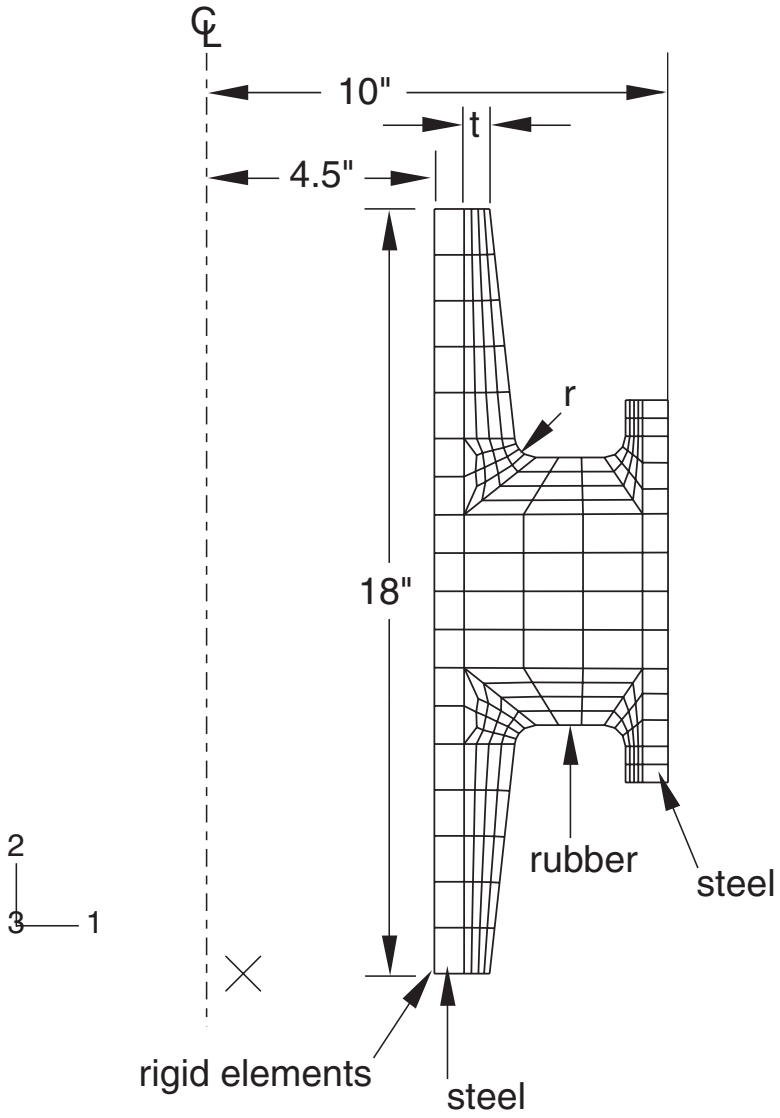


Figure 1: Axisymmetric cross-section.

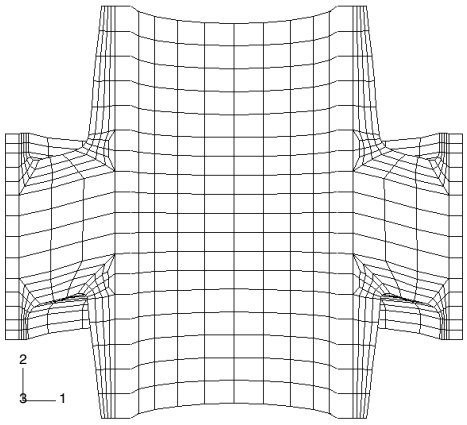


Figure 2: Deformed mesh after axial loading.

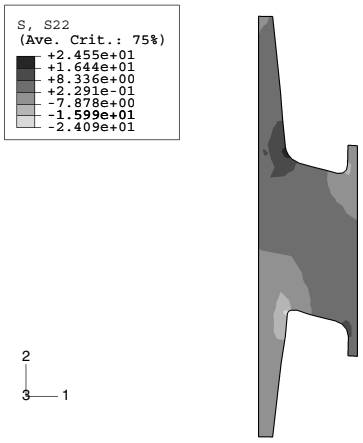


Figure 3: Variation of axial stress in the rubber after axial loading.

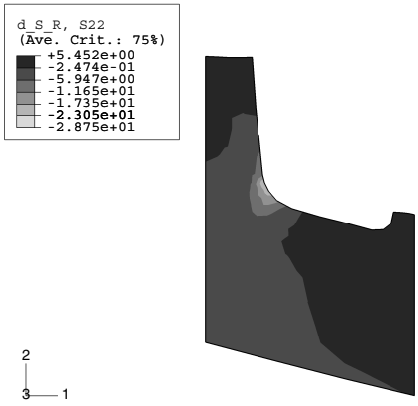


Figure 4: Variation of the sensitivity of the axial stress with respect to an increase in the radius of the fillet, *r*.

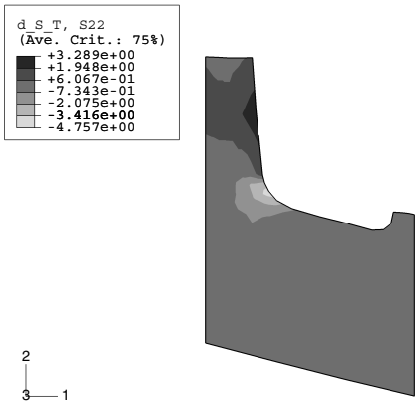


Figure 5: Variation of the sensitivity of the axial stress with respect to a decrease in the thickness of the rubber, t.

Postprocessing of Abaqus Results

In this section:

- *About user postprocessing of Abaqus results files*
- *Postprocessing examples*

About user postprocessing of Abaqus results files

Overview

You can write and utilize postprocessing programs to manipulate data stored in the Abaqus results file.

The Abaqus results file is identified by the file extension `.fil` and contains results based on user-specified output requests; see *Output to the Data and Results Files* for more information on the types of data that are written to the results file. The standard file format is binary, but it can be changed to ASCII format for each run by user request. Alternatively, it can be set to a default ASCII format during site installation. Abaqus uses Fortran unit `8` to communicate with the results file.

Sample postprocessing programs that perform commonly exercised tasks are presented in separate sections in this chapter. These include merging multiple results files and converting the resulting results file from binary format to ASCII, or vice-versa; computing principal values and directions of stress and strain; and computing a perturbed mesh for a collapse analysis by incorporating a user-specified geometric imperfection in the form of the critical buckling mode shape.

Each postprocessing program must be linked using the **make** parameter when running the Abaqus execution procedure (see *Making User-Defined Executables and Subroutines*). To link properly, the postprocessing program cannot contain a Fortran PROGRAM statement. Instead, the program must begin with a Fortran SUBROUTINE with the name ABQMAIN.

General programming concepts, Abaqus Fortran interfaces, and data processing concepts are described below. Refer to *File Output Format* for more information. The program listings in each section provide details on the program flow, how to interface with various computer platforms that use different operating systems and Fortran compilers, and how to interface with Abaqus subroutines to handle data files and records.

Refer to *Using the Abaqus Scripting Interface to access an output database* or *Using C++ to access an output database* for information on accessing data stored in the Abaqus output database.

Initialization

Details about the variables that are used in the postprocessing programs are discussed in *Accessing the Results File Information*. Abaqus uses a 512-word buffer named `ARRAY` for the reading and writing of data on the results file. This is dimensioned as `ARRAY(513)`. The integer equivalent is `JRRAY(513)` for a 64-bit computer or `JRRAY(2,513)` for a 32-bit computer. The `EQUIVALENCE` statement is used to equivalence `ARRAY` and `JRRAY` to simplify manipulation of real and integer numbers in the data record stored in the buffer.

The information concerning the Fortran unit number and format of the results file that is read is defined in `LRUNIT(2,NRU)`, where `NRU` is the number of files to be processed. The Fortran unit number for the *n*th file is stored in `LRUNIT(1,n)`. The information about the file format is stored in `LRUNIT(2,n)`, which is initialized to 1 for ASCII format and to 2 for binary format. If a new results file is to be created by the postprocessing program, the file format of the output file is defined similarly via the variable `LOUTF`, which is also initialized to 1 for ASCII format and 2 for binary format. The root file name for both input and output results files is defined through the character variable `FNAME`. The root file name case will be the same as the case in which `FNAME` is defined; Abaqus defines the file extensions to be lowercase letters. See *Accessing the Results File Information* for a discussion of the naming convention for the file extensions.

The final initialization phase is done internally by calling the Abaqus subroutines `INITPF` and `DBRNU`. The Fortran interfaces are

```
CALL INITPF(FNAME, NRU, LRUNIT, LOUTF)
CALL DBRNU (JUNIT)
```

where the arguments in the call to `INITPF` are as described above, and `JUNIT` is the Fortran unit number connecting the file.

These integer variables must be defined before the subroutines are called.

Data processing

Data manipulation requires knowledge of each data record. Details of these records are found in *Results File*.

The data organization in the results file uses a sequential format. Each record must, therefore, be retrieved in a sequential manner via a call to `DBFILE` using the interface

```
CALL DBFILE(0, ARRAY, JRCD)
```

This call can be placed inside a `DO`-loop, and the loop count should exceed the number of records stored in the file. Alternatively, `DBFILE` can be called as long as `JRCD` is equal to `0`. The first argument, `0`, indicates that a record is to be read. Each record that is read is stored in the buffer `ARRAY` and returned to the calling program for manipulation. The last argument, `JRCD`, is a return code that is set to `0` unless an end-of-file condition or an incomplete record is processed, in which case `JRCD` is set to `1`.

If it is desirable to extract or modify certain records and save them in a new results file with the same data organization as an Abaqus-generated results file, then the subroutine `DBFILW` should be called with the interface

```
CALL DBFILW(1, ARRAY, JRCD)
```

The new results file will be written with the file extension `.fin`. Refer to *Utility Routines for Accessing the Results File*.

Postprocessing examples

Sample postprocessing programs that perform commonly exercised tasks are presented in this section.

In this section:

- *Joining data from multiple results files and converting file format: FJOIN*
- *Calculation of principal stresses and strains and their directions: FPRIN*
- *Creation of a perturbed mesh from original coordinate data and eigenvectors: FPERT*
- *Output radiation view factors and facet areas: FRAD*
- *Creation of a data file to facilitate the postprocessing of elbow element results: FELBOW*
- *Visualizing beam outputs with actual beam cross-section representation for contact*
- *Translating Abaqus data to modal neutral file format for analysis in MSC.ADAMS*

Joining data from multiple results files and converting file format: FJOIN

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example illustrates how to use a Fortran program to extract specific data from different Abaqus results files and to join the data into a single results file. This program can also be used to convert the format of results files.

Postprocessing

Sometimes it is desirable to combine a number of results files into a single file or to create a new results file by retrieving selected data from different results files. The **abaqus append** utility joins two results files by stripping the header information from the second results file and appending the step information to the end of the first results file. See *Joining Results (.fil) Files* for more information on this utility.

This example postprocessing program demonstrates how a Fortran program can be used to extract specific information from results files created by separate analyses of the same model. In this example the stress and strain records in three analyses will be merged to create a new results file.

Programming details

The general discussion on programming concepts and Abaqus Fortran interfaces in *About user postprocessing of Abaqus results files* should be reviewed before running or modifying this program. Review of the results file format in *File Output Format* is also recommended.

The program `FJOIN` (named `fjoin.f` on the Abaqus release media) prompts for the values of `NRU`, `LRUNIT(1,NRU)`, `LRUNIT(2,NRU)`, and `FNAME`. Then subroutines `INITPF` and `DBNRU` are called to complete the necessary initializations and file connections. Data processing starts with a double `DO`-loop looping over all of the records to be read, one-by-one, via a call to `DBFILE`. A record can be skipped or written to the new results file with or without any modifications. Each record is identified by its record key, which is stored in the second entry of the record (see *Results File*).

Each file contains a number of header records, `1900`-series. These records contain general information about the model. Different analyses using the same model place essentially the same information in these records. Hence, when combining results files from different analyses of the same model, all `1900`-series records from the first file that is processed should be kept. Similar records in subsequent files should be skipped to avoid duplication and confusion. However, the `1910`, `1911`, `1922`, and `1980` records should be kept. They are useful for processing results within a substructure, output requests, and natural frequency extraction results.

The data for each increment of an analysis begin with the increment start record, which is identified by record key `2000`. Record `2000` is followed by the records that correspond to the data requested through file output options specified in the Abaqus input file. Record `1`, the element header record, is automatically written to the results file when element variables are requested. It is of interest when postprocessing since it contains important information about the element data, including the location of data within an element (i.e., whether data are written at the element integration points, the centroid, nodes, etc.). For this example, records `11` and `21` (the stress and strain records, respectively) are written to the results file since stress and strain were requested. The increment end record is identified by record key `2001`. When an end-of-file condition is encountered and the previously processed record is a `2001` record, a Fortran `CLOSE` is executed on the current Fortran unit number so that the processing of the next file can begin.

Program compilation and linking

The **abaqus make** utility is designed to compile and link this type of postprocessing program. It will also make the `aba_param.inc` file available during compilation. The command to compile and link the `FJOIN` program is as follows:

```
abaqus make job=fjoin
```

This command will have to be repeated if Fortran errors are discovered during the compilation or link. The commands used by the **abaqus make** utility can be changed if necessary. *System customization parameters* lists the typical compile and link commands for each computer type.

Program execution

Before program execution, the analysis jobs must be run to generate results files to be read by the program. In this example three jobs are run. The input files for these analyses are `fjoin002.inp`, `fjoin003.inp`, and `fjoin004.inp`. The results files from these analyses are output in binary format and are called `fjoin002.fil`, `fjoin003.fil`, and `fjoin004.fil`.

The `FJOIN` program will read these files via Fortran units `2`, `3`, and `4`. The name of the new file will be `fjoinxxx`. Before running the program, the results files must be renamed to `fjoinxxx.002`, `fjoinxxx.003`, and `fjoinxxx.004`. Note that the root file names are the same (defined using `FNAME`), and that the extensions are set to the Fortran unit numbers used to open the files.

When the program is executed using the command `abaqus fjoin`, the first prompt will be

```
Enter the number of files to be joined:
```

Enter `3` to set `NRU=3`. The second prompt will be

```
Enter the unit number of input file # 1:
```

Enter `2` to define `LRUNIT(1,1)=2`. At the third prompt,

```
Enter the format of input file # 1 (1-ASCII, 2-binary):
```

enter `2`. This sets `LRUNIT(2,1)=2` and means that the file being read is binary. The second and third prompts are repeated for each additional file to be processed. The program will then ask whether the new results file should be written in ASCII or binary format,

```
Enter the format of the output file (1-ASCII, 2-binary):
```

Enter `2` to set `LOUTF=2`, which specifies that binary format has been chosen for the new results file. The format of the output file may be different from the format of the input files, so this program can also be used to convert the format of results files. Finally, when the program issues the prompt

```
Enter the name of the input files (w/o extension):
```

enter `fjoinxxx` to define `FNAME` (the input files must have been given the root file name `fjoinxxx`; the output file will be created as `fjoinxxx.fin`).

As soon as the *n*th file has been processed, the message `END OF FILE #`*n* is written to the terminal. After all files have been processed, the program stops and the new results file is created. The new results file created by this program contains stress and strain records at all integration points in each element and at all nodal points.

Analysis description

The structure is a 10×10 square plate with unit thickness. The plate lies in the X–Y plane such that its bottom edge coincides with the x-axis and the left edge coincides with the y-axis. The finite element model employs a

2×2 mesh of CPS8R elements. The material is linear elastic with Young's modulus = 30×10^6 and Poisson's ratio = 0.3. Three separate analyses are performed with displacement-controlled load steps.

In the first analysis (*fjoin002.inp*), the plate is subjected to biaxial tension by prescribing a vertical displacement of 0.25 along the top edge, a horizontal displacement of 0.25 along the right edge, and symmetry boundary conditions on the left and bottom edges.

In the second analysis (*fjoin003.inp*), the structure is forced to deform in simple shear by applying a horizontal displacement of 0.25 to the top edge while holding the bottom edge fixed and allowing the horizontal displacement to vary linearly with y along the left and right edges. The vertical displacement is zero everywhere.

In the third analysis (*fjoin004.inp*), the plate is subjected to uniaxial tension by applying a displacement of 0.25 in the y-direction to the nodes along the top edge and symmetry boundary conditions to the nodes along the x- and y-axes.

Results and discussion

Since the state of stress and strain is homogeneous, the integration point and nodal averaged values of stress and strain are identical everywhere. A typical record obtained at the end of each step is included below:

Analysis	σ_{xx}	σ_{yy}	σ_{xy}
fjoin002.inp	1.07×10^6	1.07×10^6	0.0
fjoin003.inp	0.0	0.0	2.88×10^5
fjoin004.inp	0.0	7.50×10^5	0.0

Analysis	ε_{xx}	ε_{yy}	ε_{xy}
fjoin002.inp	2.50×10^{-2}	2.50×10^{-2}	0.0
fjoin003.inp	0.0	0.0	2.50×10^{-2}
fjoin004.inp	-7.50×10^{-3}	2.50×10^{-2}	0.0

Input files

fjoin002.inp	First analysis file.
fjoin003.inp	Second analysis file.
fjoin004.inp	Third analysis file.
fjoin.f	Postprocessing program.

Calculation of principal stresses and strains and their directions: FPRIN

Products: Abaqus/Standard

Overview

This example illustrates the use of a Fortran program to read stress and strain records from an Abaqus results file and to calculate principal stress and strain values and their directions.

General description

This program shows how to retrieve integration point and nodal averaged stress and strain components from an Abaqus results file and then compute principal values and directions using the Abaqus utility routine `SPRIND`. Usage of this utility routine is documented in the input file provided for this problem, and further details about the interface to this subroutine are discussed in *Obtaining Stress Invariants, Principal Stress/Strain Values and Directions, and Rotating Tensors*. The results file created by the `FJOIN` program in *Joining data from multiple results files and converting file format: FJOIN* is used here to verify that the records that have been put together are retrievable. The previously generated results file was named `fjoinxxx.fin`. To use it as an input file for postprocessing program `FPRIN`, the file extension must be changed. This program will assume that the results file has the default `.fil` extension, which corresponds to Fortran unit `8`.

Programming details

The user should first review the general discussion on programming concepts and Abaqus Fortran interfaces in *About user postprocessing of Abaqus results files* and the detailed discussion of postprocessing given in *File Output Format*.

When running program `FPRIN` (this program is named `fprin.f` on the Abaqus release media), the user will be prompted for the file name that initializes `FNAME`. Other variables, such as `LOUTF`, `NRU`, `LRUNIT(1,NRU)`, and `LRUNIT(2,NRU)`, are initialized inside the program. `INITPF` and `DBRNU` are then called to complete the neccesary initializations and file connections. Data processing starts with a double `DO`-loop over all the records to be read, one-by-one, via a call to `DBFILE`. Each record is identified by its record key, which is stored in the second entry of the record. When records `1922` and `2000` are processed by program `FPRIN`, the heading and the current step and increment numbers are written out so as to provide a way to recognize the beginning of data in each analysis. Record type `1` is then examined to determine the output location of stress and strain, the number of direct and shear stress and strain components, and either the element number or the node number for which the records are written. The stress and strain records (`11` and `21`, respectively) will be filtered out for processing by the Abaqus utility routine `SPRIND`. When a stress or strain record is passed into `SPRIND`, principal stresses or strains and the corresponding principal directions are calculated and returned in an unsorted order.

Program compilation and linking

Before program execution, the Fortran program has to be compiled and linked. Both operations, as well as the inclusion of the `aba_param.inc` file, are performed by a single execution of the **abaqus make** utility:

```
abaqus make job=fprin
```

This may have to be repeated until all Fortran errors are corrected. After successful compilation, the program's object code is automatically linked with the Abaqus object codes stored in the shared program library and interface library in order to build the executable program. Refer to *System customization parameters* to see which compile and link commands are used for a particular computer.

Program execution

Before the program is executed, a results file must have been created. In this example the results file `fjoinxxx.fin` created by the `FJOIN` program discussed in *Joining data from multiple results files and converting file format: FJOIN* is used. This file must be renamed to `fjoinxxx.fil` since Fortran unit `8` (which is associated with the `.fil` file extension) is used in the program to read the file. When the program is executed using the command `abaqus fprin`, the prompt

```
Enter the name of the input file (w/o .fil):
```

will appear. Enter `fjoinxxx` to define `FNAME`. The program processes the data and produces a file named `pvalue.dat`, which contains information about principal stresses and strains and their directions.

Results and discussion

The computed principal stress and strain values and their directions are tabulated below.

Analysis File	Principal Component	Stress × 10^5	Strain × 10^{-3}	Dir-1	Dir-2	Dir-3
fjoin002.inp	1	10.714	25.0	1.0	0.0	0.0
	2	10.714	25.0	0.0	1.0	0.0
	3	0.0	0.0	0.0	0.0	1.0
fjoin003.inp	1	–2.8846	–12.5	0.707	–0.707	0.0
	2	2.8846	12.5	0.707	0.707	0.0
	3	0.0	0.0	0.0	0.0	1.0
fjoin004.inp	1	0.0	–7.5	1.0	0.0	0.0
	2	7.5	25.0	0.0	1.0	0.0
	3	0.0	0.0	0.0	0.0	1.0

Input files

fprin.f Postprocessing program.

Creation of a perturbed mesh from original coordinate data and eigenvectors: FPERT

Products: Abaqus/Standard

Overview

This example illustrates the use of a Fortran program to create a perturbed mesh by superimposing a small imperfection in the form of the weighted sum of several buckling modes on the initial geometry.

The program retrieves the original nodal coordinates and the desired eigenvectors from an Abaqus results file, then calculates new nodal coordinates for the perturbed mesh.

General description

Collapse studies of a structure's postbuckling load-displacement (Riks) behavior are often conducted to verify that the critical buckling load and mode predicted by an eigenvalue buckling analysis are accurate. They are also done to investigate the effect of an initial geometric imperfection on the load-displacement response. A typical assumption is that an imperfection made up of a combination of the eigenmodes associated with the lowest eigenvalues will be the most critical. One method of introducing an imperfection of this type into the model is

by adding $\sum_{i=1}^{M} \alpha_i \mathbf{u}_i$ to the original mesh coordinates. In this case $\mathbf{u}_i$ is the ith eigenmode, α_i is a scaling factor of the ith eigenmode, and M is the total number of eigenmodes extracted in the buckling analysis. Since the eigenvector is typically normalized to a maximum absolute value of one, α_i is usually some fraction of a geometric parameter, such as the shell thickness. The postprocessing program described below can be used to introduce an imperfection of this type into a model.

The perturbation procedure is illustrated in *Buckling of a cylindrical shell under uniform axial pressure*. An eigenvalue buckling analysis, `fpert001`, is run first. This analysis creates the results file, `fpert001.fil`, which contains the original nodal coordinates and the eigenvectors for the buckling modes. This results file is then used to generate a perturbed mesh for the postbuckling load-displacement analysis. The postprocessing program perturbs the original mesh using the relation

$$\mathbf{X}' = \mathbf{X} + \sum_{i=1}^{M} \alpha_i \mathbf{u}_i,$$

where $\mathbf{X}'$ is the vector containing the new global coordinates; $\mathbf{X}$ is the vector of original coordinates; M is the number of buckling modes; and α_i is the imperfection factor for the ith eigenvector, $\mathbf{u}_i$. The new coordinates are written to the file `fpert002.015`, which is read by the load-displacement analysis `fpert002`.

Programming details

The general discussion on programming concepts and Abaqus Fortran interfaces in *About user postprocessing of Abaqus results files* should be reviewed before running or modifying this program. Review of the results file format in *File Output Format* is also recommended.

The `FPERT` program (this program is named `fpert.f` on the Abaqus release media) makes some assumptions concerning the type of results file it will be reading. Variables `NRU`, `LRUNIT(1,NRU)`, and `LRUNIT(2,NRU)` are initialized within the program to `1`, `8`, and `2`. These values indicate that one file will be read, the Fortran unit used will be `8`, and the file type will be binary. See *Accessing the Results File Information* for more information on opening and initializing postprocessing files.

Once the file specification parameters are set, the `INITPF` and `DBNRU` subroutines are called to open and ready the file, whose name is stored in `FNAME`, for reading. The file to which the perturbed coordinates are to be written can be directly opened using a Fortran `OPEN` statement. The Abaqus file utilities are not necessary since the file is a plain text file.

The records with the original nodal coordinates are read using the `DBFILE` routine and stored in the local array `COORDS(3,8000)`. The first index of the `COORDS` array indicates the *x*-, *y*-, and *z*-coordinate of the node. The second index indicates the node number. The second dimension should be increased if there are more than 8000 nodes in a model.

Components of the eigenvector are stored in the local array `DISP(6,8000)`. This array holds up to 6 displacement terms for each node. The second dimension should be increased if there are more than 8000 nodes in a model. Subroutine `NODEGEN`, a subroutine local to this postprocessing program, is then called to compute the new nodal coordinates. Once all the requested mode shapes are computed, the new nodal coordinates are written to the plain text file opened earlier.

Program compilation and linking

The **abaqus make** utility is designed to compile and link this type of postprocessing program. It will also make the `aba_param.inc` file available during compilation. The command to compile and link the `FPERT` program is as follows:

```
abaqus make job=fpert
```

This command will have to be repeated if Fortran errors are discovered during the compilation or link. The commands used by the **abaqus make** utility can be changed if necessary. *System customization parameters* lists the typical compile and link commands for each computer type.

Program execution

Before the program is executed, an eigenvalue buckling job must have been run with Abaqus. In this example the input file `fpert001.inp` is used to generate the results file `fpert001.fil`. When the `FPERT` program is executed using the command `abaqus fpert`, the first prompt will be

```
Enter the name of the results file (w/o .fil):
```

Enter `fpert001` to define `FNAME`. The second prompt will be

```
Enter the mode shape(s) to be used in calculating the perturbed
mesh (zero when finished):
```

Enter `1` followed by `0`, since this is the only eigenvector available in the results file for this example. At the third prompt,

```
Enter the imperfection factor to be introduced into the geometry
for this eigenmode:
```

enter `0.25`. This sets $\alpha = 0.25$, the shell thickness for this model. The program then processes the data and writes the nodal coordinates for the new mesh to `fpert002.015`.

Analysis description

For a full discussion of the analysis, refer to *Buckling of a cylindrical shell under uniform axial pressure*. The input file `fpert001.inp` (same file as `bucklecylshell_s9r5_n3.inp`) contains a 2×20 mesh of S9R5 elements and data lines for a buckling analysis. The input file `fpert002.inp` contains data lines for a Riks analysis using a perturbed mesh. The source code for the `FPERT` program is in `fpert.f`.

Results and discussion

Plots produced by these analyses are shown in *Figure 1* and *Figure 2*. *Figure 1* is obtained from the eigenvalue buckling analysis and shows the original (cylindrical) mesh and the critical buckling mode. *Figure 2* is generated when the load level has reached a local maximum (increment 8) in the Riks analysis using the perturbed mesh.

Input files

fpert001.inp	Eigenvalue buckling analysis.
fpert002.inp	Riks analysis using a perturbed mesh.
fpert.f	Postprocessing program.

Figures

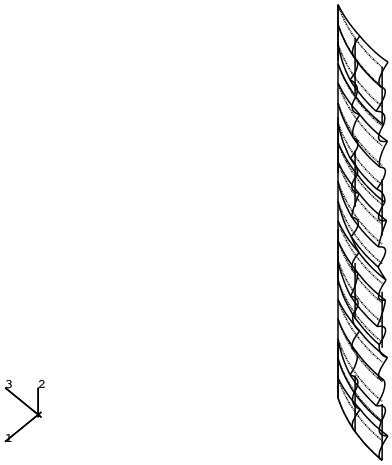


Figure 1: Undeformed shape and eigenvalue buckling mode.

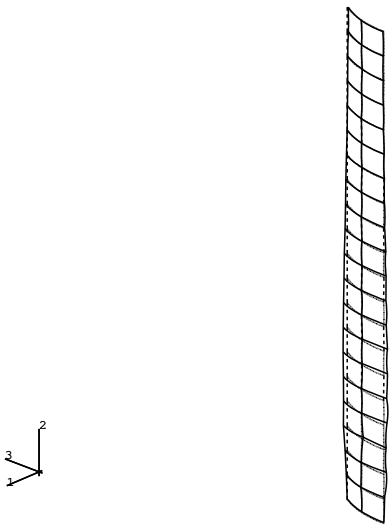


Figure 2: Deformed shape at first peak load in Riks analysis.

Output radiation view factors and facet areas: FRAD

Products: Abaqus/Standard

Overview

This example illustrates the use of a Fortran program to read the radiation view factors and the facet areas from the results file.

General description

The program shows how to retrieve the view factors and the facet areas from the results file. The results file created from the benchmark problem detailed in *Axisymmetric elemental cavity radiation view factor calculations* is used to verify that the output records have been read and output correctly. This program will assume that the results file has the default file extension, `.fil`, which corresponds to Fortran unit 8.

Programming details

Before proceeding, review the general discussion on programming concepts and Abaqus Fortran interfaces in *About user postprocessing of Abaqus results files* and the detailed discussion of postprocessing given in *File Output Format*.

When running the program `FRAD` (this program is named `frad.f` on the Abaqus release media), the user will be prompted for the file name that initializes `FNAME`. Other variables, such as `LOUTF`, `NRU`, `LRUNIT(1,NRU)`, and `LRUNIT(2,NRU)`, are initialized inside the program. `INITPF` and `DBNRU` are then called to complete the necessary initializations and file connections. By default, the results file is processed for all steps and increments in the results file. The user can restrict the output by setting `LSTEPA` and `LINCA` to the required step and increment and uncommenting the simple `IF - END IF` block. Data processing starts with a `DO`-loop over all the records to be read, one-by-one, by means of a call to `DBFILE`. Each record is identified by its record key, which is stored in the second entry of the record. When records `1922` and `2000` are processed by `FRAD`, the heading and the current step and increment numbers are written out so as to provide a way to recognize the beginning of data in each analysis. Record types `1605`, `1606`, `1607`, and `1609` are then read; and the desired output is written to the output file `vfout`.

Program compilation and linking

Before it can be executed, the Fortran program must be compiled and linked. Both operations, as well as the inclusion of the `aba_param.inc` file, are performed by a single execution of the **abaqus make** utility:

```
abaqus make job=frad
```

This procedure may have to be repeated until all Fortran errors are corrected. After successful compilation, the program's object code is linked automatically with the Abaqus object codes stored in the shared program library and the interface library to build the executable program. Refer to *System customization parameters* for information about the compile and link commands for a particular computer.

Program execution

Before the program is executed, a results file must have been created with the desired output being written to that file. In this example the results file `xrvda4n1.fil` created by running the input file `xrvda4n1.inp`

discussed in *Axisymmetric elemental cavity radiation view factor calculations* is used. When the program is executed using the command `abaqus frad`, the prompt

```
Enter the name of the input file (w/o .fil):
```

will appear. Enter `xrvda4n1` to define `FNAME`. The program processes the data and produces a file named `vfout`, which contains the required information.

Results and discussion

The radiation view factors and facet areas are read and output to `vfout`. The output agrees with the expected results.

Input files

frad.f	Postprocessing program.

Creation of a data file to facilitate the postprocessing of elbow element results: FELBOW

Products: Abaqus/Standard

Overview

This example illustrates the use of a Fortran program to read selected element integration point records from an Abaqus results file to facilitate the postprocessing of elbow element results. *X–Y* data are created that are suitable for use with the *X–Y* plotting capability in Abaqus/CAE.

General description

This program shows how to retrieve integration point data for elbow elements from an Abaqus results file to visualize one of the following:

1. Variation of a variable along a line of elbow elements,
2. Variation of a variable around the circumference of a given elbow element, or
3. Ovalization of a given elbow element.

An ASCII file containing *X–Y* data is created that can be read into Abaqus/CAE for visualization purposes.

To execute option 1, the elbow elements must be numbered such that they increase monotonically within the range of elements considered; all elements in the desired range must be elbow elements. *X–Y* data will be created with the *X*-data being the distance along the line of elbow elements, measured along the elbow centerline and the *Y*-data being the variable value. The user must ensure that the integration point coordinates (COORD) are written to the results file if either option 2 or 3 is needed. For option 2 *X*-data are the distance around the circumference of the elbow element, measured along the middle surface, and *Y*-data are the variable value. For option 3 the *X–Y* data are the current coordinates of the middle-surface integration points around the circumference of the elbow element, projected to a local coordinate system in the plane of the deformed cross-section. The origin of the local system coincides with the center of the cross-section; the plane of the deformed cross-section is defined as the plane that contains the center of the cross-section and integration points 1 and 2.

Programming details

The user is prompted for the name of the results file (assumed to be binary) and the postprocessing option (1, 2, or 3). The user is then prompted for additional information depending on the option that was chosen; this information includes

- The range of element numbers (options 2 and 3 require only a single element number),
- The section point number (options 1 and 2 only),
- The integration point number (option 1 only),
- The element variable (options 1 and 2 only),
- The component of the variable (as defined in *Results File*, options 1 and 2 only),
- The step number, and
- The increment number.

The data are processed in a double `DO`-loop over all records, via a call to `DBFILE`. The desired data are stored in variable `VAR`; the integration point coordinates are stored in `COORDS`. The program checks to make sure the

requested data are available in the results file. An error is issued if the user tries to process data that are not found in the results file.

Program compilation and linking

Before program execution, compile and link the Fortran program by using the **abaqus make** utility:

```
abaqus make job=felbow
```

Repeat this command until all Fortran errors are corrected. After successful compilation, the program's object code is linked automatically with the Abaqus object codes stored in the shared program library and interface library to build the executable program. Refer to *System customization parameters* to see which compile and link commands are used for a particular computer.

Program execution

Before executing the program, run an analysis that creates a results file containing the appropriate output. This analysis includes, for example, output for the elements in a given range and the integration point coordinates of the elements. When the program is executed using the command `abaqus felbow`, the prompt

```
Enter the name of the input file (w/o .fil):
```

will appear. Enter the name of the results file to define `FNAME`. The user is then prompted for other information, such as the desired postprocessing option, element number, etc. The program processes the data and produces a file named `output.dat` that contains the information required to visualize the elbow element results.

Results and discussion

Elastic-plastic collapse of a thin-walled elbow under in-plane bending and internal pressure contains several figures created with the aid of this program. The output agrees with the expected results.

Input files

felbow.f	Postprocessing program.

Visualizing beam outputs with actual beam cross-section representation for contact

Products: Abaqus/Standard

References:

- *About General Contact in Abaqus/Standard*
- *Contact Surface Representation for Beams*

Overview

This example provides perspective on visualizing different types of Abaqus/Standard solution results on beams.

Application description

By default, in Abaqus/Standard contact edges representing beam elements have a circular cross-section, regardless of the actual cross-section of the element. An alternative contact surface representation is available for beam elements with a noncircular cross-section (see *Contact Surface Representation for Beams*).

This example demonstrates the visualization of different output variables when actual beam cross-section representation for noncircular beams is enabled in Abaqus/Standard general contact analyses.

Abaqus modeling approaches and simulation techniques

Mesh design

The model is meshed with second-order shear deformable beam elements of type B32.

Interactions

The simulation involves contact between two beams with trapezoidal and hexagonal cross-sections, respectively. *Figure 1* shows the initial and final configurations for the simulation from two view angles. The top-right portion of *Figure 1* shows that the beams originally lie in parallel planes. The top-left portion of *Figure 1* shows that one of the beams is originally L-shaped and the other beam is originally curved. *Figure 2* shows an oblique, translucent view of the original configuration, with nodes represented as circles.

A method to represent the beam surface geometry for contact calculations accurately (see *Contact Surface Representation for Beams*) is used for this simulation. With this method, Abaqus/Standard automatically generates meshes of quadrilateral surface elements to represent the beam contact surfaces. Motion of the internally generated nodes associated with the beam surface meshes is driven by the motion of the user-defined beam nodes. Different visualization options for these beams are discussed below.

Analysis steps

Symbols representing the boundary conditions in *Figure 1* and *Figure 2* show that one end of the L-shaped beam is clamped and the other end is constrained to move only along the beam axis. The continuously curved beam segment moves downward driven by a displacement boundary condition at one of its ends. The boundary conditions are expected to cause contact between the beams. The simulation uses a general static procedure.

Discussion of results and comparison of cases

Figure 3 shows four options for representing the beam geometry in the vicinity of contact for this simulation:

- The surface-element representation shown in the upper-left corner of *Figure 3* will likely appear by default for most postprocessors. Postprocessors typically do not render beam element profiles by default but will represent the automatically generated surface elements by default.
- The upper-right corner of *Figure 3* has beam element rendering active during postprocessing, and the surface-element representation remains active in the display group. In this case, some interference occurs between the two types of representations, with the rendered beam representation primarily showing through.
- The bottom-right corner of *Figure 3* has beam element rendering active during postprocessing, with surface elements removed from the display group. In this case, only the beam element rendering appears.
- The bottom-left corner of *Figure 3* has beam element rendering inactive and surface elements removed from the display group. In this case, only references lines of the beam elements are displayed the beam cross-section is not apparent in this view.

The first and third options listed above are best for viewing the beam configurations. *Figure 4* compares the surface element representation (first option) and the beam element rendering (third option) near the corner of the L-shaped beam. The surface element representation provides more detail at the corners, whereas the beam element rendering is approximate. Surface element representations are less detailed at junctions of three or more beam elements.

Figure 5 and *Figure 6* show contour plots of the displacement magnitude and rotation magnitude, respectively, for the same four options of representing beams as discussed above. Three of these options provide similar plots, but the bottom-left plots in the respective figures do not show the beam cross-section. The maximum displacement magnitude in *Figure 6* is greater in the top two plots with surface elements in the display group. These plots consider displacement values for nodes on the exterior surface of the beam, which can be larger than displacements along the reference line of the beam, due to rotational effects.

Figure 7 shows Mises stress contour plots for the same four options of representing beams. Element results associated with beam elements are not known to surface elements; therefore, the Mises results are not apparent in the upper-left corner of *Figure 7* associated with surface elements. Stress, strain, and other elements results fundamentally associated with beam elements should be viewed on a beam element representation.

Figure 8 shows contact pressure contour plots for the same four options of representing beams. Contact results for this simulation are best viewed on the surface element representation without beam element rendering (top-left plot of *Figure 8*). Beam element rendering in the top-right plot mostly obscures CPRESS plotted on the surface elements in the top-right of *Figure 8*. The two bottom cases in *Figure 8* without surface elements in the display group lead to diagnostic messages regarding the selected variable not being available for any elements in the current display group.

Input files

std_gc_beamRealCross_beamBeam.inp	Model with actual beam cross-section representation in an Abaqus/Standard general contact analysis.

Figures

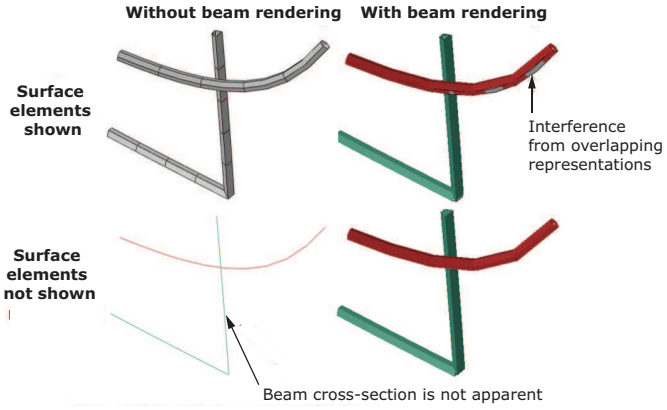


Figure 1: Simple contact model involving two beams.

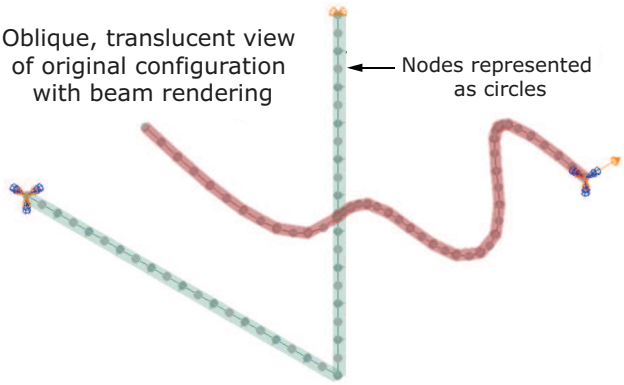


Figure 2: Original configuration showing nodal positions for a contact model involving two beams.

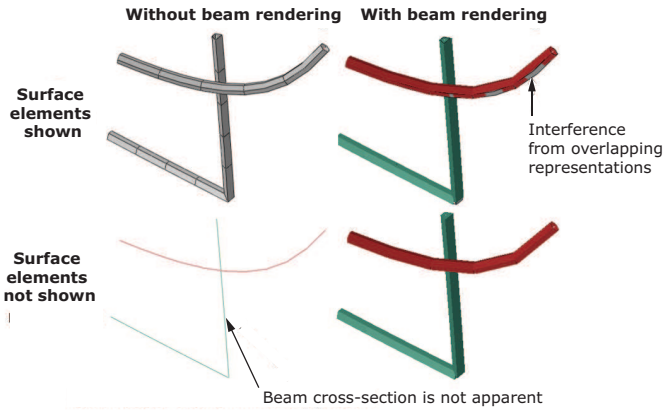


Figure 3: Comparison of four representations of the deformed configuration for the same simulation involving two beams in contact.

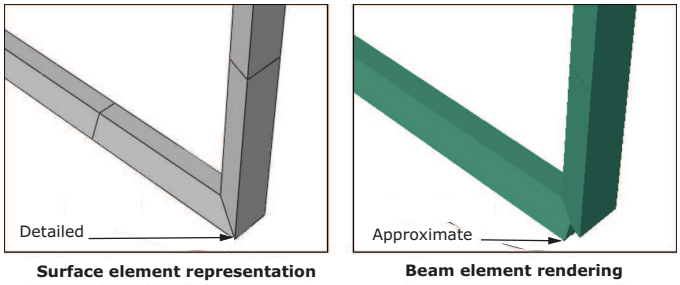


Figure 4: Close-up views of two representations of the deformed configuration for the same simulation involving two beams in contact.

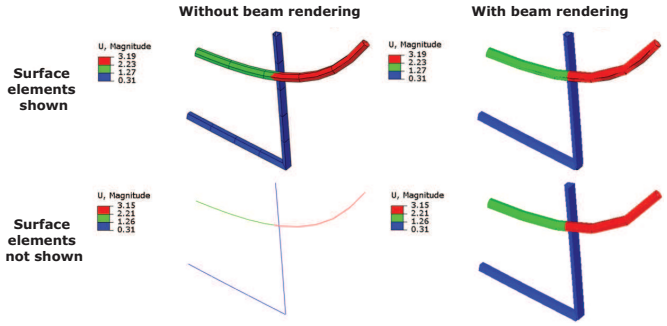


Figure 5: Comparison of four representations of displacement magnitude contour plots for the same simulation involving two beams in contact.

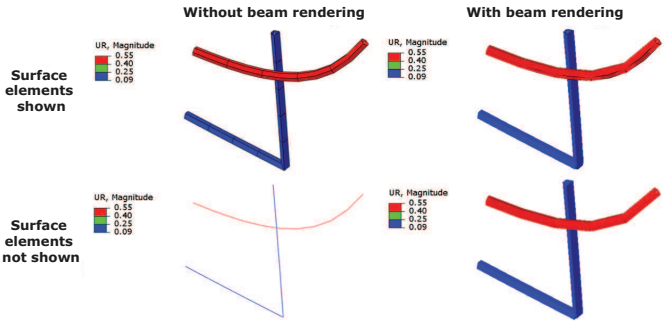


Figure 6: Comparison of four representations of rotation magnitude contour plots for the same simulation involving two beams in contact.

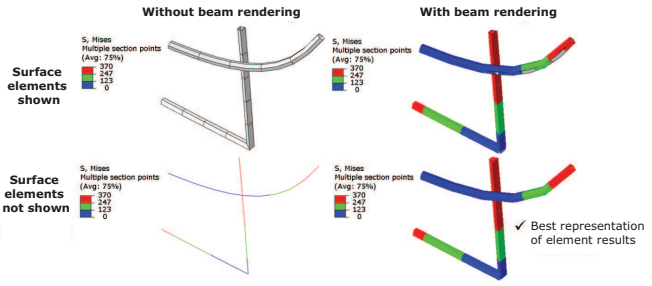


Figure 7: Comparison of four representations of Mises stress contour plots for the same simulation involving two beams in contact.

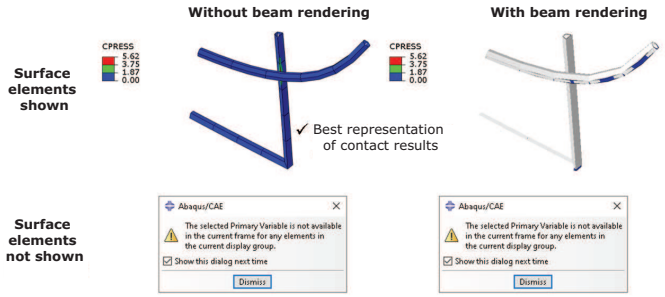


Figure 8: Comparison of four representations of contact pressure contour plots for the same simulation involving two beams in contact.

Translating Abaqus data to modal neutral file format for analysis in MSC.ADAMS

Products: Abaqus/Standard

References:

- *Translating an Abaqus Substructure to an MSC.Adams Flexible Body*

Overview

This example shows how you can investigate the flexibility of a substructure in your model during a dynamic analysis.

The following **abaqus adams** translator features are demonstrated:

- creating Abaqus models of MSC.ADAMS components,
- converting the Abaqus results into an MSC.ADAMS modal neutral (`.mnf`) file, the format required by ADAMS/Flex, and
- displaying results of an ADAMS/Flex flexibility analysis.

Application description

The following cases are illustrated:

- A three-dimensional steel link substructure meshed with solid elements and with displacement degrees of freedom at its nodes, as shown in *Figure 1*. Multi-point constraints connect it to the other components in the model. The analysis includes two steps: an eigenfrequency extraction step and a substructure generation step defined for a flexible body.
- The same steel link meshed with beam elements. Because the beam elements have both displacement and rotational degrees of freedom at their nodes, no multi-point constraints are required to connect the substructure to the rest of the model. The analysis includes two steps: an eigenfrequency extraction step and a substructure generation step defined for a flexible body.

Abaqus modeling approaches and simulation techniques

Both cases described in this section share the same general approach:

1. Create an Abaqus model for each flexible component of the MSC.ADAMS model. Each component is modeled as an Abaqus substructure.
2. Run the Abaqus analysis.
3. Run the **abaqus adams** translator to read the Abaqus results from the SIM database produced by the analysis and to create the modal neutral (`.mnf`) file for MSC.ADAMS.
4. Read the modal neutral file into MSC.ADAMS. A separate modal neutral file must be created for each flexible part in MSC.ADAMS.

Summary of analysis cases

Case 1	Link modeled with solid elements.
Case 2	Link modeled with beam elements.

Case 1 Link modeled with solid elements

This example models a simple flexible link component using three-dimensional continuum elements.

Analysis types

The example includes an eigenfrequency extraction and a substructure generation analysis.

Mesh design

The link is modeled with 642 C3D10 tetrahedral solid elements (1368 nodes).

Materials

The steel used in this case has a Young's modulus of 2.07×10^{11} N/m^2 (3.0×10^7 lbf/in^2) and a Poisson's ratio of 0.29. The density of the model is 7.8×10^3.

Boundary conditions

The `RETNODES` node set is fixed in the 1- and 6-directions.

Constraints

This analysis includes two multi-point constraints: one applied to the `LEFTCYL` node set and the other applied to the `RIGHTCYL` node set.

Analysis steps

The analysis includes two steps: an eigenfrequency extraction step and a substructure generation analysis step.

Output requests

Element stiffness matrices and mass matrices are written to the SIM database for the element set `PROP1` as part of the substructure generation analysis step.

Run procedure

You can perform the analysis of the link with solid elements using the procedure shown below.

1. Enter the following commands to extract the input files from the compressed archive files provided with the Abaqus release:

```
abaqus fetch job=adams_ex1
abaqus fetch job=adams_ex1_nodes
abaqus fetch job=adams_ex1_elements
```

2. Enter the following command to execute the Abaqus analysis:

```
abaqus job=adams_ex1
```

3. Enter the following command to execute the **abaqus adams** translator and translate the results in a SIM database generated in the Abaqus analysis to a modal neutral file for use with ADAMS/Flex:

```
abaqus adams job=adams_ex1 substructure_sim=adams_ex1_Z1
```

Results and discussion

Because the solid elements have only displacement degrees of freedom at their nodes, multi-point constraints are used to provide a connection to the other components in the MSC.ADAMS model. Two nodes are added along the centerline of the beam at the centers of the hinge holes. The C3D10 nodes that lie on the faces of the hinge holes are connected to the extra nodes with BEAM-type multi-point constraints, allowing the nodes to transmit both forces and moments between the link and other MSC.ADAMS components.

Twenty fixed-interface vibration modes are computed to represent the dynamic behavior of the link.

MSC.ADAMS uses the fixed-interface vibration modes and the constraint modes to characterize the flexibility of the link. The eight lowest fixed-interface vibration frequencies computed by Abaqus are shown in *Table 1*. These frequencies are reported in the `adams_ex1.dat` file. The **abaqus adams** translator combines these fixed-interface modes with the static constraint modes to compute an equivalent modal basis to be used by ADAMS/Flex. The first six frequencies of this equivalent basis are approximately zero. The next eight frequencies for the unconstrained model are shown in *Table 2*. These frequencies are written to the screen when executing the **abaqus adams** translator.

Case 2 Link modeled with beam elements

This example models a simple flexible link component using three-dimensional beam elements.

Analysis types

As in Case 1, this example includes an eigenfrequency extraction and a substructure generation analysis.

Mesh design

The mesh for the beam model uses 10 B31 elements (11 nodes).

Materials

The steel material definition is the same as in Case 1.

Boundary conditions

The beam elements have both displacement and rotational degrees of freedom at their nodes.

Analysis steps

The analysis includes two steps: an eigenfrequency extraction step and a substructure generation analysis step.

Run procedure

You can perform the analysis of the link with beam elements using the procedure shown below.

1. Enter the following command to extract the input files from the compressed archive files provided with the Abaqus release:

   ```
   abaqus fetch job=adams_ex2
   ```

2. Enter the following command to execute the Abaqus analysis:

   ```
   abaqus job=adams_ex2
   ```

3. Enter the following command to execute the **abaqus adams** translator and translate the results in a SIM database generated in the Abaqus analysis to a modal neutral file for use with ADAMS/Flex:

```
abaqus adams job=adams_ex2 substructure_sim=adams_ex2_Z1
```

Results and discussion

The primary difference between the beam model and the solid model is that the beam model uses only 10 B31 elements (11 nodes). Because the beam elements have both displacement and rotational degrees of freedom at their nodes, no multi-point constraints are needed to connect the link to other MSC.ADAMS components. The rest of the model is essentially identical to the solid model of the link.

The first eight nonzero frequencies for the unconstrained model are shown in *Table 3*. These frequencies are close to those of the solid model of the link. Although the computational cost in Abaqus is much less for this model than for the solid model, the computational costs in MSC.ADAMS for the two models are very similar because both models have 32 modes (12 constraint modes and 20 fixed-interface vibration modes).

Input files

Case 1 Link modeled with solid elements

adams_ex1.inp	Input file to analyze a link model subjected to a gravity load.
adams_ex1_nodes.inp	Node definitions for Case 1.
adams_ex1_elements.inp	Element definitions for Case 1.

Case 2 Link modeled with beam elements

adams_ex2.inp	Input file to analyze a link model subjected to a gravity load.

Tables

Table 1: Fixed-interface vibration frequencies for the solid link model.

Frequency, Hz
206
391
570
1124
1228
1817
1879
2541

Table 2: Nonzero frequencies for the solid link model that are used by ADAMS/Flex.

Frequency, Hz
194

Frequency, Hz
535
574
1055
1551
1762
1801
2653

Table 3: Nonzero frequencies for the beam link model that are used by ADAMS/Flex.

Frequency, Hz
205
555
610
1070
1618
1742
1775
2568

Figures

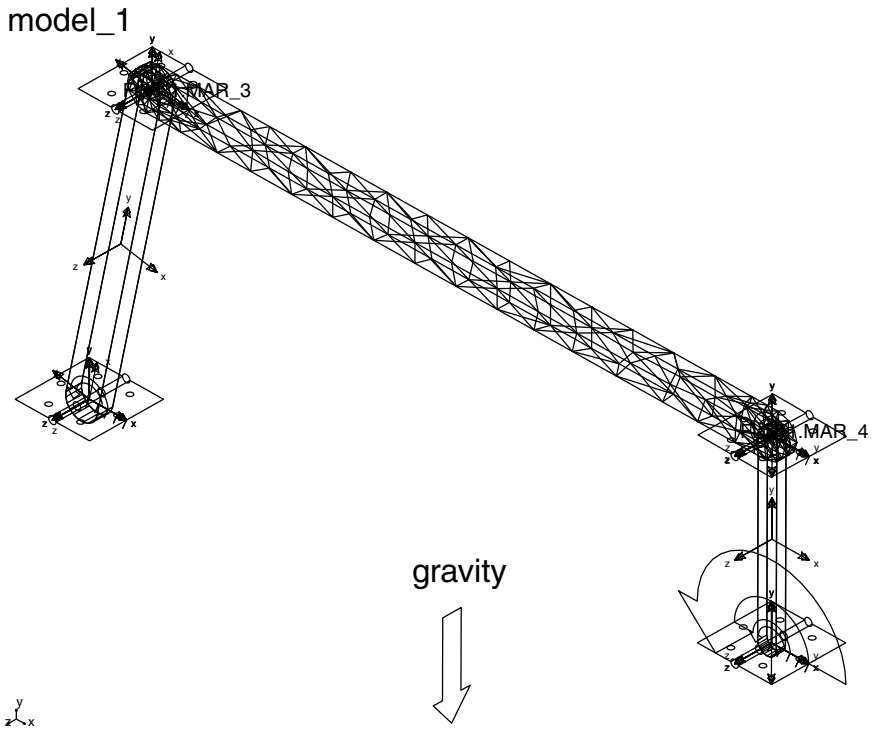


Figure 1: Solid link model.

Product Index

This index lists the sections in this guide that apply to a particular Abaqus product.

In this section:

- *Abaqus/Standard*
- *Abaqus/Explicit*
- *Abaqus/CAE*
- *Abaqus/Viewer*
- *Abaqus/AMS*
- *Abaqus/Aqua*
- *Abaqus/Design*

Abaqus/Standard

This section lists the topics in this guide that apply to Abaqus/Standard.

Abaqus/Aqua analyses

Jack-up foundation analyses

Riser dynamics

Acoustic and shock analyses

Coupled acoustic-structural analysis of a pick-up truck

Coupled acoustic-structural analysis of a speaker

Fully and sequentially coupled acoustic-structural analysis of a muffler

Long-duration response of a submerged cylinder to an underwater explosion

Application of bushing connectors in the analysis of a three-point linkage

Application of bushing connectors in the analysis of a three-point linkage

Buckling and collapse analyses

Buckling of a column with spot welds

Buckling of an imperfection-sensitive cylindrical shell

Elastic-plastic K-frame structure

Laminated composite shells: buckling of a cylindrical panel with a circular hole

Snap-through buckling analysis of circular arches

Unstable static problem: reinforced plate under compressive loads

Crank mechanism

Crank mechanism

Design sensitivities for tire inflation, footprint, and natural frequency analysis

Design sensitivities for tire inflation, footprint, and natural frequency analysis

Design sensitivity analysis of a composite centrifuge

Design sensitivity analysis of a composite centrifuge

Design sensitivity analysis of a rubber bushing

Design sensitivity analysis of a rubber bushing

Design sensitivity analysis of a windshield wiper

Design sensitivity analysis of a windshield wiper

Driveshaft mechanism

Driveshaft mechanism

Dynamic stress analyses

Detroit Edison pipe whip experiment

Impact analysis of a pawl-ratchet device

Modeling of an automobile suspension

Nonlinear dynamic analysis of a structure with local inelastic collapse

Pressurized fuel tank with variable shell thickness

Seismic analysis of a concrete gravity dam

Electromagnetic analyses

Eigenvalue analysis of a piezoelectric transducer

Modeling of a silicon piezoresistive pressure sensor

Thermal-electrical modeling of an automotive fuse

Transient dynamic nonlinear response of a piezoelectric transducer

Eulerian and co-simulation analyses

Dynamic impact of a scooter with a bump

Flap mechanism

Flap mechanism

Forming analyses

Axisymmetric forming of a circular cup

Deep drawing of a cylindrical cup

Extrusion of a cylindrical metal bar with frictional heat generation

Inertia welding simulation using Abaqus/Standard and Abaqus/CAE

Moldflow translation examples

Stretching of a thin sheet with a hemispherical punch

Superplastic forming of a rectangular box

Unstable static problem: thermal forming of a metal sheet

Upsetting of a cylindrical billet: coupled temperature-displacement and adiabatic analysis

Upsetting of a cylindrical billet: quasi-static analysis with mesh-to-mesh solution mapping (Abaqus/Standard) and adaptive meshing (Abaqus/Explicit)

Fracture and damage

A plate with a part-through crack: elastic line spring modeling

Analysis of skin-stiffener debonding under tension

Contour integrals for a conical crack in a linear elastic infinite half space

Crack growth in a three-point bend specimen

Debonding behavior of a double cantilever beam

Debonding behavior of a single leg bending specimen

Elastic-plastic line spring modeling of a finite length cylinder with a part-through axial flaw

Failure of blunt notched fiber metal laminates

Postbuckling and growth of delaminations in composite panels

Fully Coupled Thermal-Electrochemical-Structural Analysis in Batteries

Fully coupled thermal-electrochemical-structural analysis in batteries

Geneva mechanism

Geneva mechanism

Heat transfer and thermal-stress analyses

A sequentially coupled thermomechanical analysis of a disc brake with an Eulerian approach

Conductive, convective, and radiative heat transfer in an exhaust manifold

Coolant manifold cover gasketed joint

Coupled thermomechanical analysis of viscoelastic dampers

Exhaust manifold assemblage

Thermal-stress analysis of a disc brake

Thermal-stress analysis of a reactor pressure vessel bolted closure

Import analyses

Deep drawing of a square box

Springback of two-dimensional draw bending

Mass diffusion analyses

Diffusion toward an elastic crack tip

Hydrogen diffusion in a vessel wall section

Mode-based dynamic analyses

Analysis of a rotating fan using substructures and cyclic symmetry

Brake squeal analysis

Dynamic analysis of antenna structure utilizing residual modes

Linear analysis of the Indian Point reactor feedwater line

Response spectra of a three-dimensional frame building

Steady-state dynamic analysis of a vehicle body-in-white model

Modeling the Watts test of adhesives

Modeling the Watts test of adhesives

Postprocessing examples

Calculation of principal stresses and strains and their directions: FPRIN

Creation of a data file to facilitate the postprocessing of elbow element results: FELBOW

Creation of a perturbed mesh from original coordinate data and eigenvectors: FPERT

Joining data from multiple results files and converting file format: FJOIN

Output radiation view factors and facet areas: FRAD

Translating Abaqus data to modal neutral file format for analysis in MSC.ADAMS

Visualizing beam outputs with actual beam cross-section representation for contact

Prediction of cure-induced spring-in of an angle bracket

Prediction of cure-induced spring-in of an angle bracket

Resolving overconstraints in a multi-body mechanism model

Resolving overconstraints in a multi-body mechanism model

Sequential thermomechanical analysis of a directed energy deposition build

Sequential thermomechanical analysis of a directed energy deposition build

Sequential thermomechanical analysis of a laser powder bed fusion build

Sequential thermomechanical analysis of a laser powder bed fusion build

Sizing optimization analyses

Sizing optimization of a car door

Sizing optimization of a gear shift control holder

Snubber-arm mechanism

Snubber-arm mechanism

Soils analyses

Analysis of a pipeline buried in soil

Axisymmetric simulation of an oil well

Calculation of phreatic surface in an earth dam

Hydraulically induced fracture in a wellbore

Permafrost thawing–pipeline interaction

Plane strain consolidation

Static and quasi-static stress analyses

Analysis of an automotive boot seal

Axisymmetric analysis of a threaded connection

Axisymmetric analysis of bolted pipe flange connections

Bending of a beam in a sleeve

Collapse of a concrete slab

Damage and failure of a laminated composite plate

Direct cyclic analysis of a cylinder head under cyclic thermomechanical loadings

Elastic-plastic collapse of a thin-walled elbow under in-plane bending and internal pressure

Energy computations in a contact analysis

Erosion of material (sand production) in an oil wellbore

Fluid pressure penetration analysis of an O-ring seal of a pipe connection

Hydrostatic fluid elements: modeling an airspring

Indentation of an elastomeric foam specimen with a hemispherical punch

Jointed rock slope stability

Notched beam under cyclic loading

Parametric study of a linear elastic pipeline under in-plane bending

Pressure penetration analysis of an air duct kiss seal

Self-contact in rubber/foam components: jounce bumper

Self-contact in rubber/foam components: rubber gasket

Shell-to-solid submodeling and shell-to-solid coupling of a pipe joint

Stress-free element reactivation

Submodel stress analysis of pressure vessel closure hardware

Submodeling of a stacked sheet metal assembly

Transient loading of a viscoelastic bushing

Uniaxial ratcheting under tension and compression

Using a composite layup to model a yacht hull

Substructure analysis of a one-piston engine model

Substructure analysis of a one-piston engine model

Tail-skid mechanism

Tail-skid mechanism

Tire analyses

Acoustics in a circular duct with flow

Analysis of a solid disc with Mullins effect and permanent set

Coupled acoustic-structural analysis of a tire filled with air

Dynamic analysis of an air-filled tire with rolling transport effects

Steady-state dynamic analysis of a tire substructure

Steady-state rolling analysis of a tire

Subspace-based steady-state dynamic tire analysis

Symmetric results transfer for a static tire analysis

Tread wear simulation in Abaqus/Standard

Topology, shape, and bead optimization analyses

Bead optimization of a plate

Shape optimization of a connecting rod

Topology optimization of an automotive control arm

Trailing edge flap mechanism

Trailing edge flap mechanism

Vehicle analyses

Display body analysis of a pick-up truck model

Inertia relief in a pick-up truck

Substructure analysis of a pick-up truck model

Abaqus/Explicit

This section lists the topics in this guide that apply to Abaqus/Explicit.

Acoustic and shock analyses

Analysis of a speaker using Abaqus-Dymola co-simulation

Convergence studies for shock analyses using shell elements

Deformation of a sandwich plate under CONWEP blast loading

Fully and sequentially coupled acoustic-structural analysis of a muffler

Long-duration response of a submerged cylinder to an underwater explosion

Response of a submerged cylinder to an underwater explosion shock wave

UNDEX analysis of a detailed submarine model

Application of bushing connectors in the analysis of a three-point linkage

Application of bushing connectors in the analysis of a three-point linkage

Buckling and collapse analyses

Buckling of a column with spot welds

Crank mechanism

Crank mechanism

Cylinder-cam mechanism

Cylinder-cam mechanism

Driveshaft mechanism

Driveshaft mechanism

Dynamic stress analyses

Cask drop with foam impact limiter

Collapse of a stack of blocks with general contact

Confined compression of multiple two-dimensional cylinders with different material properties

Crimp forming with general contact

Eroding projectile impacting eroding plate

Explosive pipe closure

High-velocity impact of a ceramic target

Knee bolster impact with general contact

Oblique impact of a copper rod

Progressive failure analysis of thin-wall aluminum extrusion under quasi-static and dynamic loads

Rigid projectile impacting eroding plate

Seismic analysis of a concrete gravity dam

Tennis racket and ball

Water sloshing in a baffled tank

Eulerian and co-simulation analyses

Dynamic impact of a scooter with a bump

Impact of a water-filled bottle

Rivet forming

Flap mechanism

Flap mechanism

Forming analyses

Axisymmetric extrusion: transient and steady-state

Axisymmetric forming of a circular cup

Cup/trough forming

Extrusion of a cylindrical metal bar with frictional heat generation

Flat rolling: transient and steady-state

Forging with multiple complex dies

Forging with sinusoidal dies

Ring rolling

Rolling of thick plates

Section rolling

Stretching of a thin sheet with a hemispherical punch

Two-step forming simulation

Upsetting of a cylindrical billet: coupled temperature-displacement and adiabatic analysis

Upsetting of a cylindrical billet: quasi-static analysis with mesh-to-mesh solution mapping (Abaqus/Standard) and adaptive meshing (Abaqus/Explicit)

Fracture and damage

Debonding behavior of a double cantilever beam

Debonding behavior of a single leg bending specimen

Failure of blunt notched fiber metal laminates

Postbuckling and growth of delaminations in composite panels

Gear assemblies

Gear assemblies

Heat transfer and thermal-stress analyses

Thermal-stress analysis of a disc brake

Import analyses

Deep drawing of a square box

Springback of two-dimensional draw bending

Occupant safety analyses

Seat belt analysis of a simplified crash dummy

Side curtain airbag analysis using the lumped kinetic molecular method

Side curtain airbag impactor test

Particle methods and fluid dynamics

Mixing of granular media in a drum mixer

Particle cluster drop

Postprocessing examples

Joining data from multiple results files and converting file format: FJOIN

Static and quasi-static stress analyses

Collapse of a concrete slab

Damage and failure of a laminated composite plate

Fluid pressure penetration analysis of an O-ring seal of a pipe connection

Hydrostatic fluid elements: modeling an airspring

Indentation of a thick plate

Indentation of an elastomeric foam specimen with a hemispherical punch

Parametric study of a linear elastic pipeline under in-plane bending

Self-contact in rubber/foam components: jounce bumper

Self-contact in rubber/foam components: rubber gasket

Shell-to-solid submodeling and shell-to-solid coupling of a pipe joint

Tail-skid mechanism

Tail-skid mechanism

Tire analyses

Analysis of a solid disc with Mullins effect and permanent set

Import of a steady-state rolling tire

Vehicle analyses

Continuum modeling of automotive spot welds

Abaqus/CAE

This section lists the topics in this guide that apply to Abaqus/CAE.

Acoustic and shock analyses

Coupled acoustic-structural analysis of a pick-up truck

Buckling and collapse analyses

Buckling of a column with spot welds

Eulerian and co-simulation analyses

Rivet forming

Forming analyses

Inertia welding simulation using Abaqus/Standard and Abaqus/CAE

Upsetting of a cylindrical billet: quasi-static analysis with mesh-to-mesh solution mapping (Abaqus/Standard) and adaptive meshing (Abaqus/Explicit)

Fracture and damage

Contour integrals for a conical crack in a linear elastic infinite half space

Heat transfer and thermal-stress analyses

Thermal-stress analysis of a reactor pressure vessel bolted closure

Sizing optimization analyses

Sizing optimization of a car door

Sizing optimization of a gear shift control holder

Static and quasi-static stress analyses

Shell-to-solid submodeling and shell-to-solid coupling of a pipe joint

Using a composite layup to model a yacht hull

Topology, shape, and bead optimization analyses

Bead optimization of a plate

Shape optimization of a connecting rod

Topology optimization of an automotive control arm

Abaqus/Viewer

This section lists the topics in this guide that apply to Abaqus/Viewer.

Static and quasi-static stress analyses

Using a composite layup to model a yacht hull

Abaqus/AMS

This section lists the topics in this guide that apply to Abaqus/AMS.

Mode-based dynamic analyses

Steady-state dynamic analysis of a vehicle body-in-white model

Abaqus/Aqua

This section lists the topics in this guide that apply to Abaqus/Aqua.

Abaqus/Aqua analyses

Jack-up foundation analyses

Riser dynamics

Abaqus/Design

This section lists the topics in this guide that apply to Abaqus/Design.

Design sensitivities for tire inflation, footprint, and natural frequency analysis

Design sensitivities for tire inflation, footprint, and natural frequency analysis

Design sensitivity analysis of a composite centrifuge

Design sensitivity analysis of a composite centrifuge

Design sensitivity analysis of a rubber bushing

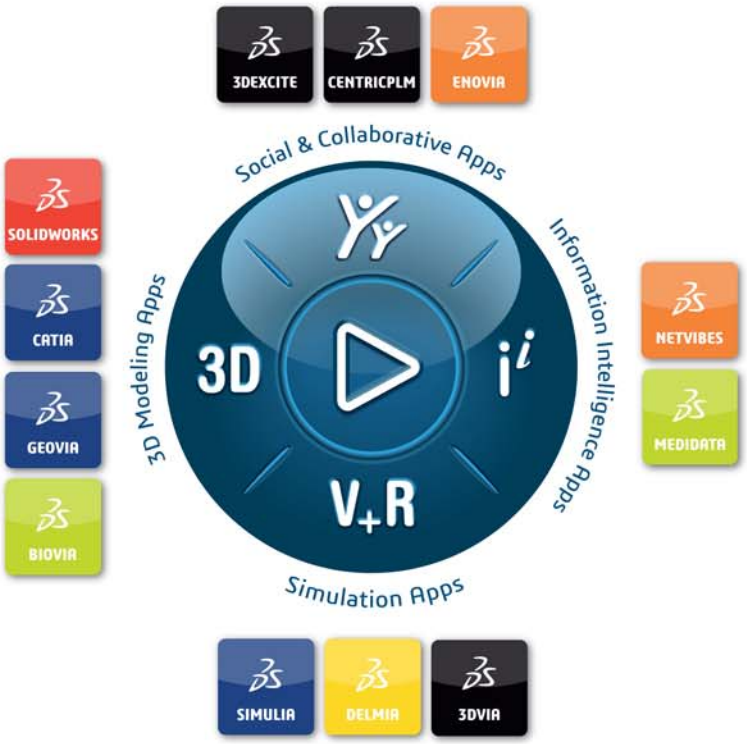
Design sensitivity analysis of a rubber bushing

Design sensitivity analysis of a windshield wiper

Design sensitivity analysis of a windshield wiper

Static and quasi-static stress analyses

Indentation of an elastomeric foam specimen with a hemispherical punch



Our **3D**EXPERIENCE® platform powers our brand applications, serving 12 industries, and provides a rich portfolio of industry solution experiences.

Dassault Systèmes is a catalyst for human progress. We provide business and people with collaborative virtual environments to imagine sustainable innovations. By creating virtual twin experiences of the real world with our **3DEXPERIENCE** platform and applications, our customers can redefine the creation, production and life-cycle-management processes of their offer and thus have a meaningful impact to make the world more sustainable. The beauty of the Experience Economy is that it is a human-centered economy for the benefit of all – consumers, patients and citizens.

Dassault Systèmes brings value to more than 300,000 customers of all sizes, in all industries, in more than 150 countries. For more information, visit **www.3ds.com**.

Europe/Middle East/Africa
Dassault Systèmes
10, rue Marcel Dassault
CS 40501
78946 Vélizy-Villacoublay Cedex
France

Asia-Pacific
Dassault Systèmes
17F, Foxconn Building,
No. 1366, Lujiazui Ring Road
Pilot Free Trade Zone, Shanghai 200120
China

Americas
Dassault Systèmes
175 Wyman Street
Waltham, Massachusetts
02451-1223
USA

DASSAULT SYSTEMES

©2025 Dassault Systèmes. All rights reserved. **3DEXPERIENCE**, the 3DS logo, the Compass icon, IFWE, 3DEXCITE, 3DVIA, BIOVIA, CATIA, CENTRIC PLM, DELMIA, ENOVIA, GEOVIA, MEDIDATA, NETVIBES, OUTSCALE, SIMULIA and SOLIDWORKS are commercial trademarks or registered trademarks of Dassault Systèmes, a European company (Societas Europaea) incorporated under French law, and registered with the Versailles trade and companies registry under number 322 306 440, or its subsidiaries in the United States and/or other countries.