SIMULIA

3DEXPERIENCE®

ABAQUS 2025 FD04
BENCHMARKS GUIDE

DASSAULT SYSTEMES

Contents

Trademarks and Legal Notices....8
Abaqus Benchmarks Guide....9
What's New....10
Analysis Tests....11
Static stress/displacement analysis....12
Beam/gap example....13
Analysis of an anisotropic layered plate....15
Composite shells in cylindrical bending....20
Thick composite cylinder subjected to internal pressure....35
Uniform collapse of straight and curved pipe segments....42
Snap-through of a shallow, cylindrical roof under a point load....54
Pressurized rubber disc....59
Uniaxial stretching of an elastic sheet with a circular hole....69
Necking of a round tensile bar....74
Concrete slump test....82
The Hertz contact problem....89
Crushing of a pipe....102
Radial stretching of a cylinder....108
Buckling analysis....111
Buckling analysis of beams....112
Buckling of a ring in a plane under external pressure....119
Buckling of a cylindrical shell under uniform axial pressure....125
Buckling of a simply supported square plate....140
Lateral buckling of an L-bracket....146
Buckling of a column with general contact....150
Dynamic stress/displacement analysis....156
Subspace dynamic analysis of a cantilever beam....157
Double cantilever elastic beam under point load....161
Explosively loaded cylindrical panel....166
Free ring under initial velocity: comparison of rate-independent and rate-dependent plasticity....171
Large rotation of a one degree of freedom system....177
Motion of a rigid body in Abaqus/Standard....182
Rigid body dynamics with Abaqus/Explicit....196
Revolute MPC verification: rotation of a crank....201
Pipe whip simulation....203
Impact of a copper rod....207
Frictional braking of a rotating rigid body....219
Compression of cylindrical shells with general contact....225
Steady-state slip of a belt drive....228
Crash simulation of a motor vehicle....230
Truss impact on a rigid wall....236
Plate penetration by a projectile....244
Oblique shock reflections....251

Mode-based dynamic analysis....256
Free vibrations of a spherical shell....257
Eigenvalue analysis of a beam under various end constraints and loadings....263
Vibration of a cable under tension....266
Free and forced vibrations with damping....268
Verification of Rayleigh damping options with direct integration and modal superposition....277
Eigenvalue analysis of a cantilever plate....281
Vibration of a rotating cantilever plate....286
Response spectrum analysis of a simply supported beam....291
Linear analysis of a rod under dynamic loading....294
Random response to jet noise excitation....307
Random response of a cantilever subjected to base motion....311
Double cantilever subjected to multiple base motions....314
Analysis of a cantilever subject to earthquake motion....318
Residual modes for modal response analysis....332
Steady-state transport analysis....335
Steady-state transport analysis....336
Steady-state spinning of a disk in contact with a foundation....346
Heat transfer and thermal-stress analysis....352
Convection and diffusion of a temperature pulse....353
Freezing of a square solid: the two-dimensional Stefan problem....361
Coupled temperature-displacement analysis: one-dimensional gap conductance and radiation....367
Quenching of an infinite plate....376
Two-dimensional elemental cavity radiation view factor calculations....382
Axisymmetric elemental cavity radiation view factor calculations....386
Three-dimensional elemental cavity radiation view factor calculations....391
Radiation analysis of a plane finned surface....401
Eulerian analysis....406
Eulerian analysis of a collapsing water column....407
Deflection of an elastic dam under water pressure....411
Electromagnetic analysis....415
Eigenvalue analysis of a piezoelectric cube with various electrode configurations....416
Modal dynamic analysis for piezoelectric materials....420
Steady-state dynamic analysis for piezoelectric materials....423
TEAM 2: Eddy current simulations of long cylindrical conductors in an oscillating magnetic field....426
TEAM 4: Eddy current simulation of a conducting brick in a decaying magnetic field....430
TEAM 6: Eddy current simulations for spherical conductors in an oscillating magnetic field....433
TEAM 13: Three-dimensional nonlinear magnetostatic analysis....438
Induction heating of a cylindrical rod by an encircling coil carrying time-harmonic current....441
Coupled pore fluid flow and stress analysis....445
Partially saturated flow in a porous medium....446
Demand wettability of a porous medium: coupled analysis....456
Wicking in a partially saturated porous medium....463
Desaturation in a column of porous material....469
Substructures and mass diffusion analysis....476
Analysis of a frame using substructures....477
Thermomechanical diffusion of hydrogen in a bending beam....481
Acoustic analysis....485

A simple coupled acoustic-structural analysis....486
Analysis of a point-loaded, fluid-filled, spherical shell....491
Acoustic radiation impedance of a sphere in breathing mode....494
Acoustic-structural interaction in an infinite acoustic medium....500
Acoustic-acoustic tie constraint in two dimensions....508
Acoustic-acoustic tie constraint in three dimensions....511
A simple steady-state dynamic acoustic analysis....514
Acoustic analysis of a duct with mean flow....517
Real exterior acoustic eigenanalysis....519
Coupled exterior acoustic eigenanalysis....521
Acoustic scattering from a rigid sphere....523
Acoustic scattering from an elastic spherical shell....526
Adaptivity analysis....530
Indentation with different materials....531
Wave propagation with different materials....534
Adaptivity patch test with different materials....540
Wave propagation in a shock tube....545
Propagation of a compaction wave in a shock tube....549
Advection in a rotating frame....552
Water sloshing in a pitching tank....559
Abaqus/Aqua analysis....563
Pull-in of a pipeline lying directly on the seafloor....564
Near bottom pipeline pull-in and tow....567
Slender pipe subject to drag: the “reed in the wind”....572
Underwater shock analysis....574
One-dimensional underwater shock analysis....575
The submerged sphere problem....580
The submerged infinite cylinder problem....583
The one-dimensional cavitation problem....588
Plate response to a planar exponentially decaying shock wave....602
Cylindrical shell response to a planar step shock wave....604
Cylindrical shell response to a planar exponentially decaying shock wave....607
Spherical shell response to a planar step wave....610
Spherical shell response to a planar exponentially decaying wave....613
Spherical shell response to a spherical exponentially decaying wave....616
Air-backed coupled plate response to a planar exponentially decaying wave....619
Water-backed coupled plate response to a planar exponentially decaying wave....622
Coupled cylindrical shell response to a planar step wave....625
Coupled spherical shell response to a planar step wave....628
Fluid-filled spherical shell response to a planar step wave....631
Response of beam elements to a planar wave....634
Soils analysis....636
The Terzaghi consolidation problem....637
Consolidation of a triaxial test specimen....644
Finite-strain consolidation of a two-dimensional solid....650
Limit load calculations with granular materials....654
Finite deformation of an elastic-plastic granular material....659
The one-dimensional thermal consolidation problem....662

Consolidation around a cylindrical heat source....665
Fracture mechanics....672
Contour integral evaluation: two-dimensional case....673
Contour integral evaluation: three-dimensional case....691
Center slant cracked plate under tension....702
A penny-shaped crack under concentrated forces....705
Fully plastic J-integral evaluation....709
Ct-integral evaluation....716
Nonuniform crack-face loading and J-integrals....722
Single-edged notched specimen under a thermal load....724
Design sensitivity analysis....729
Design sensitivity analysis for cantilever beam....730
Sensitivity of the stress concentration factor around a circular hole in a plate under uniaxial tension....736
Sensitivity analysis of modified NAFEMS problem 3DNLG-1: Large deflection of Z-shaped cantilever under an end load....739
Modeling discontinuities using XFEM....747
Crack propagation of a single-edge notch simulated using XFEM....748
Crack propagation in a plate with a hole simulated using XFEM....755
Crack propagation in a beam under impact loading simulated using XFEM....760
Dynamic shear failure of a single-edge notch simulated using XFEM....762
Propagation of hydraulically driven fracture using XFEM....764
Element Tests....768
Continuum elements....769
Torsion of a hollow cylinder....770
Geometrically nonlinear analysis of a cantilever beam....773
Cantilever beam analyzed with CAXA and SAXA elements....780
Two-point bending of a pipe due to self weight: CAXA and SAXA elements....784
Cook's membrane problem....787
Infinite elements....791
Wave propagation in an infinite medium....792
Infinite elements: the Boussinesq and Flamant problems....802
Infinite elements: circular load on half-space....806
Spherical cavity in an infinite medium....811
Structural elements....816
The barrel vault roof problem....817
The pinched cylinder problem....826
The pinched sphere problem....842
Skew sensitivity of shell elements....846
Performance of continuum and shell elements for linear analysis of bending problems....864
Tip in-plane shear load on a cantilevered hook....887
Analysis of a twisted beam....890
Twisted ribbon test for shells....901
Ribbon test for shells with applied moments....903
Triangular plate-bending on three point supports....905
Shell elements subjected to uniform thermal loading....909
Shell bending under a tip load....911
Variable thickness shells and membranes....913
Transient response of a shallow spherical cap....917
Simulation of propeller rotation....922

Connector elements....926
Dynamic response of a two degree of freedom system....927
Linear behavior of spring and dashpot elements....931
Special-purpose elements....936
Acoustic modes of an enclosed cavity....937
Fluid filled rubber bladders....941
Delamination analysis of laminated composites....944
Material Tests....952
Elasticity....953
Viscoelastic rod subjected to constant axial load....954
Transient thermal loading of a viscoelastic slab....959
Uniform strain, viscoplastic truss....965
Fitting of rubber test data....969
Fitting of elastomeric foam test data....992
Rubber under uniaxial tension....999
Anisotropic hyperelastic modeling of arterial layers....1002
Plasticity and creep....1012
Uniformly loaded, elastic-plastic plate....1013
Test of ORNL plasticity theory under biaxial loading....1019
One-way reinforced concrete slab....1024
Triaxial tests on a saturated clay....1033
Uniaxial tests on jointed material....1039
Verification of creep integration....1042
Simple tests on a crushable foam specimen....1050
Simple proportional and nonproportional cyclic tests....1055
Biaxial tests on gray cast iron....1060
Indentation of a crushable foam plate....1069
Notched unreinforced concrete beam under 3-point bending....1078
Mixed-mode failure of a notched unreinforced concrete beam....1088
Slider mechanism with slip-rate-dependent friction....1099
Cylinder under internal pressure....1105
Creep of a thick cylinder under internal pressure....1112
Pressurization of a thick-walled cylinder....1117
Stretching of a plate with a hole....1120
Pressure on infinite geostatic medium....1123
NAFEMS Benchmarks....1130
Standard benchmarks: linear elastic tests....1131
LE1: Plane stress elements—elliptic membrane....1132
LE2: Cylindrical shell bending patch test....1135
LE3: Hemispherical shell with point loads....1139
LE4: Axisymmetric hyperbolic shell under uniform internal pressure....1142
LE5: Z-section cantilever....1144
LE6: Skew plate under normal pressure....1147
LE7: Axisymmetric cylinder/sphere under pressure....1150
LE8: Axisymmetric shell under pressure....1152
LE9: Axisymmetric branched shell under pressure....1154
LE10: Thick plate under pressure....1156
LE11: Solid cylinder/taper/sphere—temperature loading....1159

Standard benchmarks: linear thermoelastic tests....1161
T1: Plane stress elements—membrane with hot-spot....1162
T2: One-dimensional heat transfer with radiation....1164
T3: One-dimensional transient heat transfer....1167
T4: Two-dimensional heat transfer with convection....1172
Standard benchmarks: free vibration tests....1176
FV2: Pin-ended double cross: in-plane vibration....1177
FV4: Cantilever with off-center point masses....1180
FV12: Free thin square plate....1182
FV15: Clamped thin rhombic plate....1185
FV16: Cantilevered thin square plate....1187
FV22: Clamped thick rhombic plate....1189
FV32: Cantilevered tapered membrane....1191
FV41: Free cylinder: axisymmetric vibration....1193
FV42: Thick hollow sphere: uniform radial vibration....1195
FV52: Simply supported “solid” square plate....1197
Proposed forced vibration benchmarks....1199
Test 5: Deep simply supported beam: frequency extraction....1200
Test 5H: Deep simply supported beam: harmonic forced vibration....1202
Test 5T: Deep simply supported beam: transient forced vibration....1206
Test 5R: Deep simply supported beam: random forced vibration....1210
Test 13: Simply supported thin square plate: frequency extraction....1211
Test 13H: Simply supported thin square plate: harmonic forced vibration....1213
Test 13T: Simply supported thin square plate: transient forced vibration....1216
Test 13R: Simply supported thin square plate: random forced vibration....1221
Test 21: Simply supported thick square plate: frequency extraction....1223
Test 21H: Simply supported thick square plate: harmonic forced vibration....1226
Test 21T: Simply supported thick square plate: transient forced vibration....1229
Test 21R: Simply supported thick square plate: random forced vibration....1235
Proposed nonlinear benchmarks....1237
NL1: Prescribed biaxial strain history, plane strain....1238
NL2: Axisymmetric thick cylinder....1241
NL3: Hardening with two variables under load control....1243
NL4: Snap-back under displacement control....1245
NL5: Straight cantilever with end moment....1247
NL6: Straight cantilever with axial end point load....1249
NL7: Lee's frame buckling problem....1251
Two-dimensional test cases in linear elastic fracture mechanics....1253
Test 1.1: Center cracked plate in tension....1254
Test 1.2: Center cracked plate with thermal load....1256
Test 2.1: Single edge cracked plate in tension....1258
Test 3: Angle crack embedded in a plate....1260
Test 4: Cracks at a hole in a plate....1262
Test 5: Axisymmetric crack in a bar....1264
Test 6: Compact tension specimen....1266
Test 7.1: T-joint weld attachment....1268
Test 8.1: V-notch specimen in tension....1270
Fundamental tests of creep behavior....1272

Test 1A: 2D plane stress – uniaxial load, secondary creep....1273
Test 1B: 2D plane stress – uniaxial displacement, secondary creep....1275
Test 2A: 2D plane stress – biaxial load, secondary creep....1277
Test 2B: 2D plane stress – biaxial displacement, secondary creep....1279
Test 3A: 2D plane stress – biaxial (negative) load, secondary creep....1281
Test 3B: 2D plane stress – biaxial (negative) displacement, secondary creep....1283
Test 4A: 2D plane stress – biaxial (double) load, secondary creep....1285
Test 4B: 2D plane stress – biaxial (double) displacement, secondary creep....1287
Test 4C: 2D plane stress – shear loading, secondary creep....1289
Test 5A: 2D plane strain – biaxial load, secondary creep....1291
Test 5B: 2D plane strain – biaxial displacement, secondary creep....1293
Test 6A: 3D – triaxial load, secondary creep....1295
Test 6B: 3D – triaxial displacement, secondary creep....1297
Test 7: Axisymmetric – pressurized cylinder, secondary creep....1299
Test 8A: 2D plane stress – uniaxial load, primary creep....1302
Test 8B: 2D plane stress – uniaxial displacement, primary creep....1304
Test 8C: 2D plane stress – stepped load, primary creep....1306
Test 9A: 2D plane stress – biaxial load, primary creep....1308
Test 9B: 2D plane stress – biaxial displacement, primary creep....1310
Test 9C: 2D plane stress – biaxial stepped load, primary creep....1312
Test 10A: 2D plane stress – biaxial (negative) load, primary creep....1314
Test 10B: 2D plane stress – biaxial (negative) displacement, primary creep....1316
Test 10C: 2D plane stress – biaxial (negative) stepped load, primary creep....1318
Test 11: 3D – triaxial load, primary creep....1320
Test 12A: 2D plane stress – uniaxial load, primary-secondary creep....1322
Test 12B: 2D plane stress – uniaxial displacement, primary-secondary creep....1324
Test 12C: 2D plane stress – stepped load, primary-secondary creep....1326
Composite tests....1328
R0031(1): Laminated strip under three-point bending....1329
R0031(2): Wrapped thick cylinder under pressure and thermal loading....1333
R0031(3): Three-layer sandwich shell under normal pressure loading....1336
Geometric nonlinear tests....1338
3DNLG-1: Elastic large deflection response of a Z-shaped cantilever under an end load....1339
3DNLG-2: Elastic large deflection response of a pear-shaped cylinder under end shortening....1342
3DNLG-3: Elastic lateral buckling of a right angle frame under in-plane end moments....1345
3DNLG-4: Lateral torsional buckling of an elastic cantilever subjected to a transverse end load....1348
3DNLG-5: Large deflection of a curved elastic cantilever under transverse end load....1351
3DNLG-6: Buckling of a flat plate when subjected to in-plane shear....1354
3DNLG-7: Elastic large deflection response of a hinged spherical shell under pressure loading....1356
3DNLG-8: Collapse of a straight pipe segment under pure bending....1359
3DNLG-9: Large elastic deflection of a pinched hemispherical shell....1363
3DNLG-10: Elastic-plastic behavior of a stiffened cylindrical panel under compressive end load....1366
Product Index....1369
Abaqus/Standard....1370
Abaqus/Explicit....1379
Abaqus/CAE....1383
Abaqus/Aqua....1384
Abaqus/Design....1385

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 Benchmarks Guide

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.

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.

In addition to this guide, there are two other guides that contain worked problems. The *Abaqus Example Problems Guide* contains many solved examples from which users can learn how to run simulations involving nontrivial physics. Many of these problems are quite difficult and test a combination of capabilities in the code. 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 Example Problems Guide*, and the *Abaqus Verification Guide*.

Input files referred to in the guides 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 *barrelvault_s8r5_reg22.inp*, type

```
abaqus fetch job=barrelvault_s8r5_reg22.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*.

What's New

This page describes recent changes in Abaqus Benchmarks.

2025 FD01

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 Benchmarks Guide*.

2024 FD03

Fatigue Crack Growth Analysis Using Field Import–Based Submodeling Approach

New tests verify fatigue crack growth analysis using both the traditional node-based submodeling approach and the field import–based submodeling approach.

Benefits: The tests verify the capability in Abaqus/Standard.
A single edge notch subjected to pure mode II cyclic fatigue loading is carried out using both the traditional node-based submodeling approach and the field import–based submodeling approach. Similar results are obtained when comparing the crack profiles obtained using both approaches.

For more information, see *Crack propagation of a single-edge notch simulated using XFEM*.

Using Low-Density Foam with Eulerian Elements and ALE Analysis

New tests verify the use of low-density foam with Eulerian elements and ALE adaptive meshing.

Benefits: The tests verify the capability in Abaqus/Explicit.

For more information, see *Indentation with different materials*, *Wave propagation with different materials*, and *Adaptivity patch test with different materials*.

Analysis Tests

In this section:

- *Static stress/displacement analysis*
- *Buckling analysis*
- *Dynamic stress/displacement analysis*
- *Mode-based dynamic analysis*
- *Steady-state transport analysis*
- *Heat transfer and thermal-stress analysis*
- *Eulerian analysis*
- *Electromagnetic analysis*
- *Coupled pore fluid flow and stress analysis*
- *Substructures and mass diffusion analysis*
- *Acoustic analysis*
- *Adaptivity analysis*
- *Abaqus/Aqua analysis*
- *Underwater shock analysis*
- *Soils analysis*
- *Fracture mechanics*
- *Design sensitivity analysis*
- *Modeling discontinuities using XFEM*

Static stress/displacement analysis

In this section:

- *Beam/gap example*
- *Analysis of an anisotropic layered plate*
- *Composite shells in cylindrical bending*
- *Thick composite cylinder subjected to internal pressure*
- *Uniform collapse of straight and curved pipe segments*
- *Snap-through of a shallow, cylindrical roof under a point load*
- *Pressurized rubber disc*
- *Uniaxial stretching of an elastic sheet with a circular hole*
- *Necking of a round tensile bar*
- *Concrete slump test*
- *The Hertz contact problem*
- *Crushing of a pipe*
- *Radial stretching of a cylinder*

Beam/gap example

Products: Abaqus/Standard

Overview

This example verifies the performance of a gap element in a simple case.

Three parallel cantilever beams are initially separate but have possible contact points in five locations, as shown in *Figure 1*. A pair of pinching loads is applied, as shown. Only small displacements are considered, so each beam responds in pure bending. The problem is entirely linear, except for the switching contact conditions.

The sequence of events is readily imagined:

1. The top and bottom beams bend as the pinching forces are applied, and the first contact occurs when the tip of the top beam hits the tip of the middle beam (gap 3 closes). Up to this point the problem is symmetric about the middle beam, but it now loses that symmetry.
2. Subsequent to this initial contact, the top and middle beams bend down and the bottom beam continues to bend up until contact occurs at gap 5.
3. As the load continues to increase, gap 2 closes.
4. Next, gap 3 opens as the support provided to the top beam by gap 2 causes the outboard part of the top beam to reverse its direction of rotation. At this point (when gap 3 opens), the solution becomes symmetric about the middle beam once again.
5. Finally, as the pinching loads increase further, gaps 1 and 4 also close. From this point on the contact conditions do not switch, no matter how much more load is applied.

Problem description

Each cantilever is modeled using five cubic beam elements of type B23. Initially all gaps are open, with an initial gap clearance of 0.01. The pinching loads are increased monotonically from 0 to 200. The beam lengths, modulus, and cross-section are shown in *Figure 1*. (The units of dimension and force are consistent but not physical.)

The loads are applied in 10 equal increments, with the increment size given directly in the static analysis.

Results and discussion

The solution is summarized in *Table 1*.

Input files

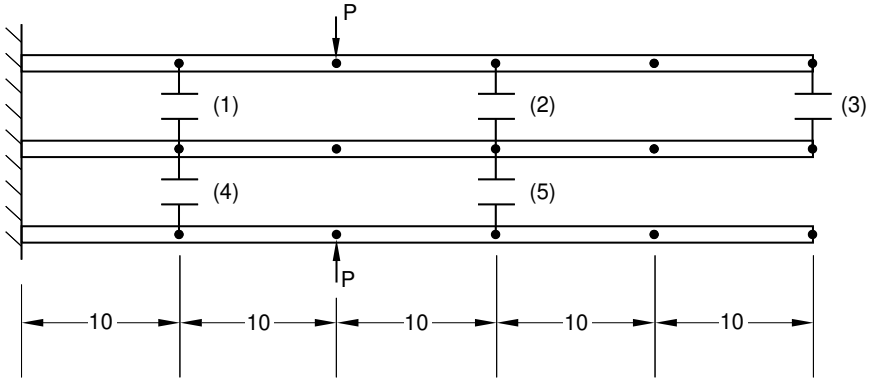
beamgap.inp	Input data for this problem.

Tables

Table 1: Beam/gap example: solution summary.

Increment	Pinching	Force in gap				
	force, P	1	2	3	4	5
1	20	Open	6.5	0.732	Open	7.97
2	40	Open	18.3	Open	Open	18.3
3	60	Open	28.7	Open	Open	28.7
4	80	Open	39.1	Open	Open	39.1
5	100	Open	49.5	Open	Open	49.5
6	120	Open	59.8	Open	Open	59.8
7	140	10.7	68.6	Open	10.7	68.6
8	160	31.6	75.9	Open	31.6	75.9
9	180	52.5	83.2	Open	52.5	83.2
10	200	73.4	90.4	Open	73.4	90.4

Figures



Material properties:

Young's modulus = 10^8 force/length^2

Beam section data:

hexagonal, circumscribing radius = 0.5
wall thickness = 0.1

Figure 1: Beam/gap example.

Analysis of an anisotropic layered plate

Products: Abaqus/Standard

References:

- *Orientations*

Overview

This example illustrates the use of a user-defined coordinate system in the analysis of multilayered, laminated, composite shells.

The problem considered in this example is the linear analysis of a flat plate made from two layers oriented at ±45°, subjected to a uniform pressure loading. The example verifies simple laminated composite plate analysis. The Abaqus results are compared with the analytical solution given in Spilker et al. (1976). The cross-section is not balanced, so the response includes membrane-bending coupling. Composite failure measures are defined for the plane stress orthotropic material.

Problem description

The structure is a two-layer, composite, orthotropic, square plate that is simply supported on its edges. The layers are oriented at ±45° with respect to the plate edges. *Figure 1* shows the loading and the plate dimensions. Each layer has the following material properties:

E_{11}	276 GPa (40×10^6 lb/in^2)
E_{22}	6.9 GPa (10^6 lb/in^2)
$G_{12} = G_{13} = G_{23}$	3.4 GPa (0.5×10^6 lb/in^2)
ν_{12}	0.25

These properties define linear elastic behavior for a lamina under plane stress conditions (*Linear Elastic Behavior*). More general orthotropic properties (for solid continuum elements) can be specified using the elastic stiffness matrix.

In this example the plate is considered to be at an arbitrary angle to the global axis system to illustrate the use of a local coordinate system. The plate is shown in *Figure 2*.

The boundary conditions require that displacements that are transverse and normal to the shell edges are fixed, but motions that are parallel to the edges are permitted. A convenient set of local displacement degrees of freedom is defined so that the boundary conditions and the output of nodal variables can be interpreted more easily (*Transformed Coordinate Systems*).

A local coordinate system is used to define the direction of the layers. The rotation of the material axes of the layers with respect to the standard directions used by Abaqus for stress and strain components in shells is defined in four of the models used and, again for illustration purposes, by means of user subroutine *ORIENT* in four other models. The section is not balanced since it has only two layers in different orientations, which results in membrane-bending coupling. The motion does not exhibit symmetry for the same reason, and the entire shell must be modeled.

An alternative means of defining the layer orientation is to use a local coordinate to define the orientation of the section and then to define the in-plane angle of rotation relative to the section orientation directly with the layer

data in the shell section or general shell section definition. In this case the section force and section strain are calculated in the section orientation directions (rather than the default shell directions).

Three types of models are used. One is an 8 × 8 mesh of S9R5 elements, which are shell elements that allow transverse shear along lines in the element. However, the analytical solution of Spilker et al. uses thin shell theory, which neglects transverse shear effects. We have, therefore, introduced an artificially high transverse shear stiffness in this model.

The second type of model is a 16 × 16 mesh of triangular shells; models for both S3R and SC6R elements are provided. These elements are general-purpose shell elements that allow transverse shear deformation. An artificially high transverse shear stiffness is introduced. No mesh convergence studies have been performed, but finer meshes should improve accuracy since these elements use a constant bending strain approximation.

The third type of model is made up of STRI65 shell elements, which are also based on the discrete Kirchhoff theory. An 8 × 8 mesh is used.

Failure measures

To demonstrate the use of composite failure measures (*Plane Stress Orthotropic Failure Measures*), limit stresses are defined. The stress-based failure criteria are defined as follows:

X_t (Psi)	X_c (Psi)	Y_t (Psi)	Y_c (Psi)	S (Psi)	$\overset{*}{f}$
60.0×10^4	-24.0×10^4	1.0×10^4	-3.0×10^4	2.0×10^4	0.0

Printed failure indices are requested for maximum stress theory (MSTRS) and Tsai-Hill theory (TSAIH). All failure measures are written to the results file (CFAILURE).

Results and discussion

Table 1 summarizes the results by comparing displacement and moment values to the analytical solution. It is clear by the results presented in the table that all models give good results, with the second-order models providing higher accuracy than the first-order S3R model, as would be expected.

Figure 3 shows the failure surface for Tsai-Hill theory (i.e., those stress values $(\sigma_{11}, \sigma_{22})$ that, for a given σ_{12}, yield a failure index $R = 1.0$), along with the stress state at each section point in the center of the plate. Only section point 6 has a stress state outside the failure surface ($R > 1.0$).

Input files

anisoplate_s3r_orient.inp	S3R element model with the orientation for the material defined with **ORIENTATION*.
anisoplate_s3r_usr_orient.inp	S3R element model with the orientation for the material defined in user subroutine *ORIENT*.
anisoplate_s3r_usr_orient.f	User subroutine *ORIENT* used in anisoplate_s3r_usr_orient.inp.
anisoplate_sc6r_orient.inp	SC6R element model with the orientation for the material defined with **ORIENTATION*.
anisoplate_sc6r_usr_orient.inp	SC6R element model with the orientation for the material defined in user subroutine *ORIENT*.
anisoplate_sc6r_orient_gensect.inp	SC6R model with the orientation for the shell section defined with **ORIENTATION* and the orientation for the material defined by an angle on the data lines for **SHELL GENERAL SECTION*.
anisoplate_sc6r_usr_orient.f	User subroutine *ORIENT* used in anisoplate_sc6r_usr_orient.inp.

anisoplate_s9r5_orient.inp	S9R5 model with the orientation for the material defined with **ORIENTATION*.
anisoplate_s9r5_usr_orient.inp	S9R5 model with the orientation for the material defined in user subroutine *ORIENT*.
anisoplate_s9r5_usr_orient.f	User subroutine *ORIENT* used in anisoplate_s9r5_usr_orient.inp.
anisoplate_s9r5_orient_sect.inp	S9R5 model with the orientation for the shell section defined with **ORIENTATION* and the orientation for the material defined by an angle on the data lines for **SHELL SECTION*.
anisoplate_s9r5_orient_gensect.inp	S9R5 model with the orientation for the shell section defined with **ORIENTATION* and the orientation for the material defined by an angle on the data lines for **SHELL GENERAL SECTION*.
anisoplate_stri65_orient.inp	STRI65 element model with the orientation for the material defined with **ORIENTATION*.
anisoplate_stri65_usr_orient.inp	STRI65 element model with the orientation for the material defined in user subroutine *ORIENT*.
anisoplate_stri65_usr_orient.f	User subroutine *ORIENT* used in anisoplate_stri65_usr_orient.inp.

References

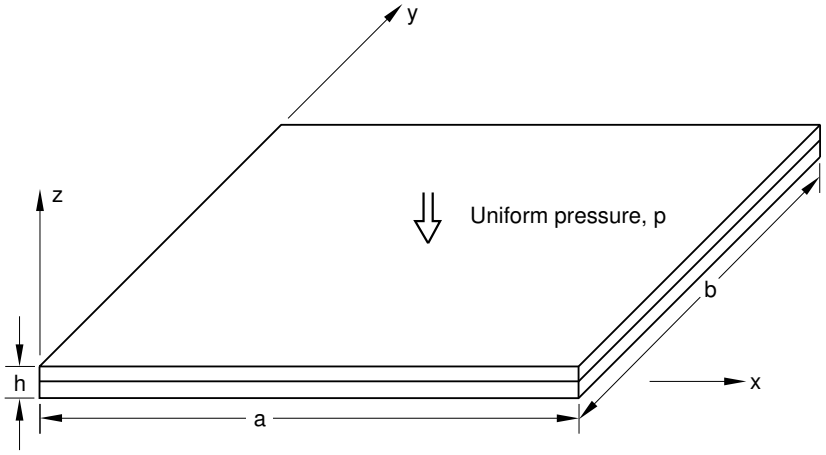
- Spilker, R. L., S. Verbiese, O. Orringer, S. E. French, E. A. Witmer, and A. Harris, "Use of the Hybrid-Stress Finite-Element Model for the Static and Dynamic Analysis of Multilayer Composite Plates and Shells," *Report for the Army Materials and Mechanics Research Center, Watertown, MA*, 1976.

Tables

Table 1: Results for pressure loading of anisotropic plate.

Element type	In-plane disp. at $x = 0, y = b/2$ (mm)	Normal disp. at center of plate (mm)	Moment, M_x or M_y at center of plate (N-m)
Analytical	0.3762	23.25	42.05
S3R	0.3724	22.86	40.54
SC6R	0.3724	22.84	40.54
STRI65	0.3760	23.24	42.28
S9R5	0.3752	23.25	42.23

Figures



Geometric properties:

a = b = 254 mm (10 in)
h = 5.08 mm (0.2 in)

Loading:

p = 689.4 kPa (100 lb/in^2)

Figure 1: Geometry and loading for flat plate.

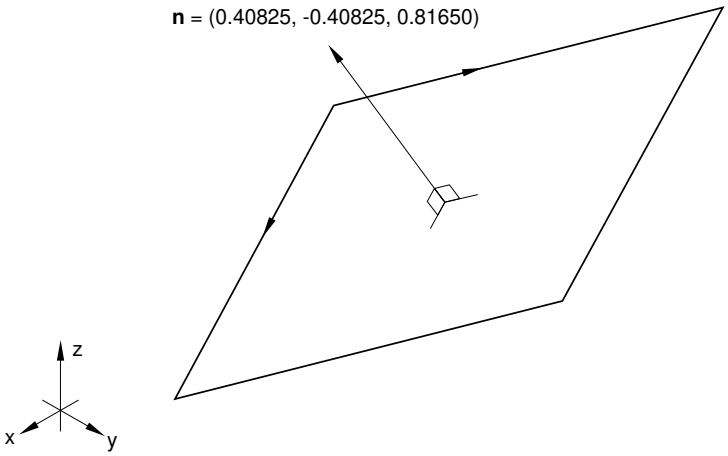


Figure 2: Orientation of plate in space.

LINE	VARIABLE	SCALE FACTOR
1	section pt. 1	+1.00E+00
2	section pt. 2	+1.00E+00
3	section pt. 3	+1.00E+00
4	section pt. 4	+1.00E+00
5	section pt. 5	+1.00E+00
6	section pt. 6	+1.00E+00

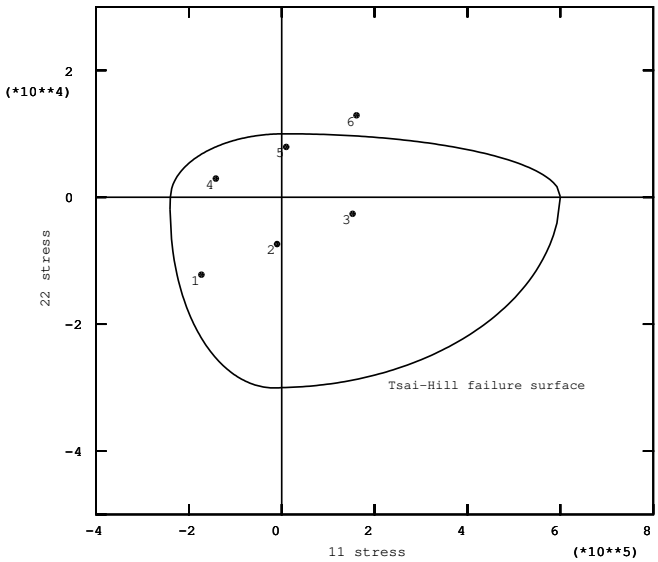


Figure 3: The stress state at each section point in the center of the plate, plotted with the Tsai-Hill failure surface. Note that section point 6 has failed.

Composite shells in cylindrical bending

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example verifies the transverse shear stress calculations in Abaqus for multilayer composite shells and demonstrates the use of the plane stress orthotropic failure measures.
A discussion of the transverse shear stresses obtained by composite solids in Abaqus/Standard is included in this example. The problem consists of a two- or three-layer plate subjected to a sinusoidal distributed load, as described by Pagano (1969). The resulting transverse shear and axial stresses through the thickness of the plate are compared to two existing analytical solutions by Pagano (1969). The first solution is derived from classical laminated plate theory (CPT), while the second is an exact solution from linear elasticity theory.

Problem description

A schematic of the model is shown in *Figure 1*. The structure is a composite plate composed of orthotropic layers of equal thickness. It is simply supported at its ends and bounded along its edges to impose plane strain conditions in the *y*-direction. Each layer models a fiber/matrix composite with the following properties:

E_L	172.4 GPa (25×10^6 lb/in^2)
E_T	6.90 GPa (1.0×10^6 lb/in^2)
G_{LT}	3.45 GPa (0.5×10^6 lb/in^2)
G_{TT}	1.38 GPa (0.2×10^6 lb/in^2)
$\nu_{LT} = \nu_{TT}$	0.25

where *L* signifies the direction parallel to the fibers and *T* signifies the transverse direction. In Abaqus/Standard two methods are used to specify the lay-up definition for the conventional shell element model. First, a composite shell section is defined to specify the thickness, number of integration points, material name, and orientation of each layer. Second, a composite general shell section is defined to specify the thickness, material name, and angle of orientation relative to the section orientation (the default shell directions in this case) for each layer. In Abaqus/Explicit only the former method is used. The material properties are specified using the orthotropic elastic in plane stress definition. The orientation of the fibers in each layer is defined by an in-plane rotation angle measured relative to the local shell directions or relative to an orientation definition given for the general shell section.

In addition to the methods outlined above, a third method of stacking continuum shell elements is used to specify the lay-up definition for a composite model. This method can be used effectively to study localized behavior, since continuum shell elements handle high aspect ratios between the in-plane dimension and the thickness dimension well.

The lay-up definition for the continuum (solid) element model in Abaqus/Standard is specified using a composite solid section definition. The thickness, material name, and orientation definition are specified for each layer.

A distributed load with a sinusoidal distribution in space, $p = p_0 \sin(\pi x/l)$, is applied to the top of the composite plate. In Abaqus/Standard the load is applied using user subroutine *DLOAD* in a static linear analysis step. In addition, an Abaqus/Standard input file is included that demonstrates the use of the DCOUP3D element to apply this distributed load. In Abaqus/Explicit the load is applied instantaneously at time $t = 0$.

Two composite plates are analyzed in this example. The first is a two-layer plate with the fibers oriented parallel and orthogonal to the x-axis in the bottom and top layer, respectively. In the second plate, which has three layers of equal thickness, the fibers in the outer layers are oriented parallel to the x-axis, while the fibers in the middle layer are orthogonal to the x-axis. The span-to-thickness ratio of the plates, $s = l/h$, is varied from 4 to 30 in the Abaqus/Standard analysis; in Abaqus/Explicit this ratio is 4 throughout the analysis.

A 1 × 10 mesh of second-order S8R shell elements is used to model the plates in Abaqus/Standard. A 2 × 10 mesh of first-order S4R shell elements is used to model the plates in Abaqus/Explicit. The S4R, S8R, and S8RT shell elements are well-suited for modeling thick composite shells since they account for transverse shear flexibility. Five integration points are specified through the thickness of each layer with the models that use the shell section. This provides sufficient data to describe the stress distributions through the thickness of each layer. For the models that use the general shell section, only three points are available for output. (Since the analysis is linear elastic, three points are sufficient to determine all fields through the thickness.) The plate with the lowest span-to-thickness ratio is also analyzed with Abaqus/Standard using a 1 × 10 mesh of second-order C3D20R composite solid elements.

To illustrate the stacking capability of continuum shell elements, several meshes are provided for the two- and three-layer plates with a span-to-thickness ratio of 4. The two-layer plate is modeled with a 2 × 10 mesh of SC8R elements, each element representing a single layer of the 90/0 composite plate. One model of the three-layer plate uses a 1 × 10 mesh of SC8R elements using a single element through the thickness with a composite section definition. Another model of the three-layer plate uses a 3 × 10 mesh of SC8R elements, each element representing a single layer of the 0/90/0 composite plate. Additional models of the three-layer plate with 6, 12, and 24 elements through the thickness are provided. In these models each composite layer is modeled with 2, 4, and 8 elements through the thickness, respectively.

Additional input files using SC8R elements are included to illustrate defining the stacking and thickness direction independent of the element nodal connectivity.

Failure measures

The plane stress orthotropic failure measures are defined in *Plane Stress Orthotropic Failure Measures*. To demonstrate their use, let the limit stresses and limit strains be given as follows:

Stress Values:	X_t	X_c	Y_t	Y_c	S
(GPa)	2.07×10^{-4}	-8.28×10^{-5}	3.45×10^{-6}	-1.03×10^{-5}	6.89×10^{-6}
(lb/in^2)	30.0	−12.0	0.5	−1.5	1.0
Strain Values:	$X_{\epsilon t}$	$X_{\epsilon c}$	$Y_{\epsilon t}$	$Y_{\epsilon c}$	S_ϵ
	$17. \times 10^{-2}$	$-7. \times 10^{-2}$	$5. \times 10^{-2}$	-1.3×10^{-2}	$11. \times 10^{-2}$

The scaling factor for the Tsai-Wu coefficient F_{12} is $\overset{*}{f} =$ 0.0. These values are chosen such that failure occurs under the stress-based failure criteria for the given loading in the two-layer case with $s =$ 4.

Results and discussion

The results for each of the analyses are discussed in the following sections.

Abaqus/Standard results

Figure 2 shows the maximum z-displacement as a function of the span-to-thickness ratio of the two- and three-layer plates in a normalized form as

$$w = \frac{100 E_T h^3 u_z(l/2)}{l^4}.$$

As seen in the figure, the finite element displacements for both the two- and three-layer plates agree well with the prediction from elasticity theory for a wide range of s values. The CPT results are stiff at low values of s since shear flexibility is neglected.

For $s = 4$, *Figure 3* and *Figure 4* show the transverse shear stress (TSHR13) and the axial stress (S11) distributions through the plate thickness for the two-layer plate normalized as

$$\overline{\tau}_{xz} = \frac{\tau_{xz}(0,z)}{p_o}$$

and

$$\overline{\sigma}_x = \frac{\sigma_x(l/2,z)}{p_o}.$$

Figure 5 and *Figure 6* show the corresponding results for the three-layer plate. It is seen that the shell element results are much closer to the predictions of CPT than to elasticity theory because of the assumption of linear stress variation through the thickness in the first-order shear flexible theory used for elements such as S8R and S4R.

Figure 7 compares the elasticity solution of the transverse shear distribution for the three-layer plate to an approximate solution using the output variable SSAVG4. SSAVG4 is the average transverse shear stress in the local 1-direction. Since SSAVG4 is constant over an element, mesh refinement (in this case 24 continuum shell elements through the thickness) is typically required to capture the variation of shear stress through the thickness of the plate.

The output variables CTSHR13 and CTSHR23 offer a more economical alternative to SSAVG4 and SSAVG5 for estimating shear stress in stacked continuum shells. *Figure 8* and *Figure 9* show very good agreement between the elasticity solution of the transverse shear distribution for the three- and two-layer plates to the solution using the output variable CTSHR13 for a 3×10 and 2×10 mesh of continuum shell elements, respectively. The shear stress computed using CTSHR13 is continuous across the continuum shell element interfaces. In addition, while the estimates of the transverse shear distributions using SSAVG4 and CTSHR13 (shown in *Figure 7* and *Figure 8*) are both good, using CTSHR13 requires a mesh of only 3 continuum shell elements through the thickness, as compared to 24 elements for SSAVG4.

Figure 10 compares the transverse shear stress distribution obtained with the solid element model with the shell element result. The figure shows that the transverse shear stresses predicted by solid elements do not vanish at the free surfaces of the structure. It also shows that the stress is discontinuous at layer interfaces. The reason for this is that in the composite solid element, the transverse shear stresses are obtained directly from the displacement field in contrast to the shell element, where the transverse shear stresses are obtained from an equilibrium calculation. These deficiencies decrease if the number of solid elements used in the discretization through the section thickness is increased. Although the transverse shear stresses are inaccurate, the displacement field and components of stress in the plane of the layer (not shown here) are in much better agreement with the analytical result. In fact, these results are somewhat better than the results obtained with the S8R elements. The composite solid elements were not used to analyze the thinner plates since the solid elements would not have any advantage over plate elements in that case.

For $s = 10$, *Figure 11* and *Figure 12* show that the transverse shear and axial stress distributions of the finite element results—along with the CPT predictions—agree with elasticity theory. The stress distributions become more accurate with increasing span-to-thickness ratio (as the plate becomes thinner in comparison to the span).

In *Figure 13* and *Figure 14* the maximum stress theory and Tsai-Wu theory failure indices are plotted as a function of the normalized distance from the midsurface for the two- and three-layer cases, respectively. The indices are calculated at the center of the plate for S8R elements with $s = 4$. Values of the failure index greater than or equal to 1.0 indicate failure. Discontinuous jumps in the failure index occur at layer boundaries as a result of the orientation of the material. The strain levels are well below those required for failure, so no strain-based failure indices are plotted.

Abaqus/Explicit results

The explicit dynamic analysis is run for a sufficiently long time so that a quasi-static state is reached—that is, the plates are in steady-state vibration. Since step loadings are applied, static solutions of stresses can be obtained as half of their vibration amplitudes.

Figure 15 and *Figure 16* show the transverse shear stress (TSHR13) and the axial stress (S11) distributions through the plate thickness for the two-layer S4R model normalized as:

$$\overline{\tau}_{xz} = \frac{\tau_{xz}(0,z)}{p_o}$$

and

$$\overline{\sigma}_x = \frac{\sigma_x(l/2,z)}{p_o},$$

compared with classical plate theory (CPT) and linear elasticity theory.

Figure 17 and *Figure 18* show the corresponding results for the three-layer plate. In *Figure 19* and *Figure 20*, the maximum stress theory and Tsai-Wu theory failure indices are plotted as a function of the normalized distance from the midsurface for the two- and three-layer cases, respectively. The indices are calculated at the center of the plate. Values of the failure index greater than or equal to 1.0 indicate failure. Discontinuous jumps in the failure index occur at layer boundaries due to the orientation of the material. The strain levels are well below those required for failure, so no strain-based failure indices are plotted.

Input files

Abaqus/Standard input files

compositeshells_s8r.inp	Three-layer plate with $s = 4$ using S8R elements.
compositeshells_s8r.f	User subroutine defining nonuniform distributed load for use with compositeshells_s8r.inp.
compositeshells_s8r_gensect.inp	Three-layer plate with $s = 4$ using S8R elements and **SHELL GENERAL SECTION*.
compositeshells_s8r_gensect.f	User subroutine *DLOAD* used in compositeshells_s8r_gensect.inp.
compositeshells_s4.inp	S4 element model.
compositeshells_s4.f	User subroutine *DLOAD* used in compositeshells_s4.inp.
compositeshells_s4_gensect.inp	S4 element model with **SHELL GENERAL SECTION*.
compositeshells_s4_gensect.f	User subroutine *DLOAD* used in compositeshells_s4_gensect.inp.
compositeshells_s4_dcoup3d.inp	S4 element model loaded using a DCOUP3D element.
compositeshells_s4r.inp	S4R element model.
compositeshells_s4r.f	User subroutine *DLOAD* used in compositeshells_s4r.inp.
compositeshells_s4r_gensect.inp	S4R element model with **SHELL GENERAL SECTION*.
compositeshells_s4r_gensect.f	User subroutine *DLOAD* used in compositeshells_s4r_gensect.inp.
compositeshells_c3d20r.inp	C3D20R composite solid element model.
compositeshells_c3d20r.f	User subroutine *DLOAD* used in compositeshells_c3d20r.inp.
compositeshells_sc8r_stackdir_1.inp	SC8R model using STACK DIRECTION=1.
compositeshells_sc8r_stackdir_2.inp	SC8R model using STACK DIRECTION=2.
compositeshells_sc8r_stackdir_3.inp	SC8R model using STACK DIRECTION=3.
compositeshells_sc8r_gensect.inp	SC8R model using **SHELL GENERAL SECTION*.

compshell2_std_sc8r_stack_2.inp	Two-layer plate with SC8R elements, two elements stacked through the thickness.
compshell3_std_sc8r_stack_1.inp	Three-layer plate with SC8R elements, single element through the thickness.
compshell3_std_sc8r_stack_3.inp	Three-layer plate with SC8R elements, three elements stacked through the thickness.
compshell3gs_std_sc8r_stack_3.inp	Three-layer plate with SC8R elements, three elements stacked through the thickness using a general shell section definition.
compshell3_std_sc8r_stack_6.inp	Three-layer plate with SC8R elements, six elements stacked through the thickness.
compshell3_std_sc8r_stack_12.inp	Three-layer plate with SC8R elements, 12 elements stacked through the thickness.
compshell3_std_sc8r_stack_24.inp	Three-layer plate with SC8R elements, 24 elements stacked through the thickness.
compositeshells_sc8r.f	User subroutine *DLOAD* used with the SC8R models.

Abaqus/Explicit input files

compshell3_1.inp	Three-layer plate modeled with S4R elements.
compshell3_1_sc8r.inp	Three-layer plate modeled with SC8R elements.
compshell3_1_sc8r_stackdir_1.inp	Three-layer plate modeled with SC8R elements using STACK DIRECTION=1.
compshell3_1_sc8r_stackdir_2.inp	Three-layer plate modeled with SC8R elements using STACK DIRECTION=2.
compshell3_1_sc8r_stackdir_3.inp	Three-layer plate modeled with SC8R elements using STACK DIRECTION=3.
compshell3_2.inp	Three-layer plate with a different thickness and modeled with S4R elements.
compshell2_1.inp	Two-layer plate modeled with S4R elements.
compshell2_2.inp	Two-layer plate modeled with S4R elements.
compshell2_1_sc8r.inp	Two-layer plate modeled with SC8R elements.

References

- Pagano, N. J., “Exact Solutions for Composite Laminates in Cylindrical Bending,” *Journal of Composite Materials*, vol. 3, pp. 398–411, 1969.

Figures

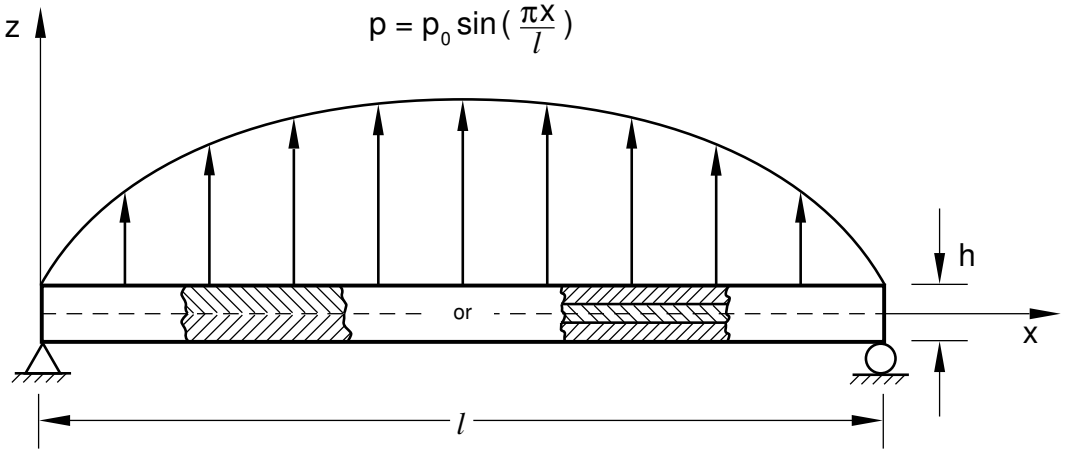


Figure 1: Composite plate subject to distributed loading.

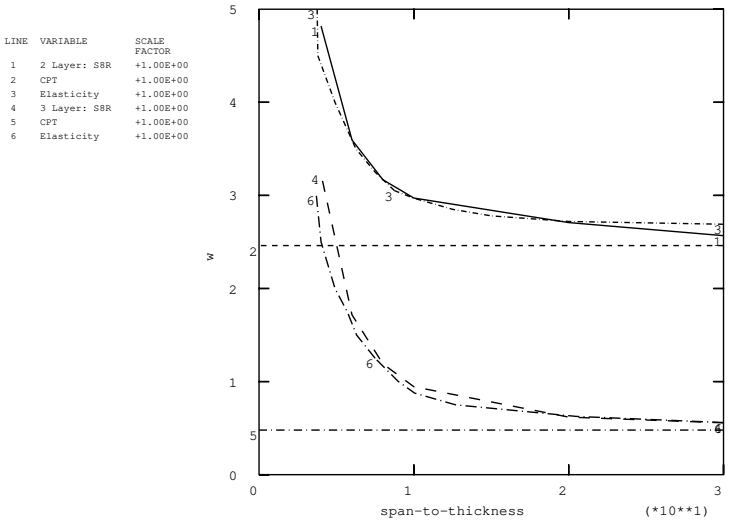


Figure 2: Maximum deflection of two- and three-layer plates with various span-to-thickness ratios; Abaqus/Standard analysis.

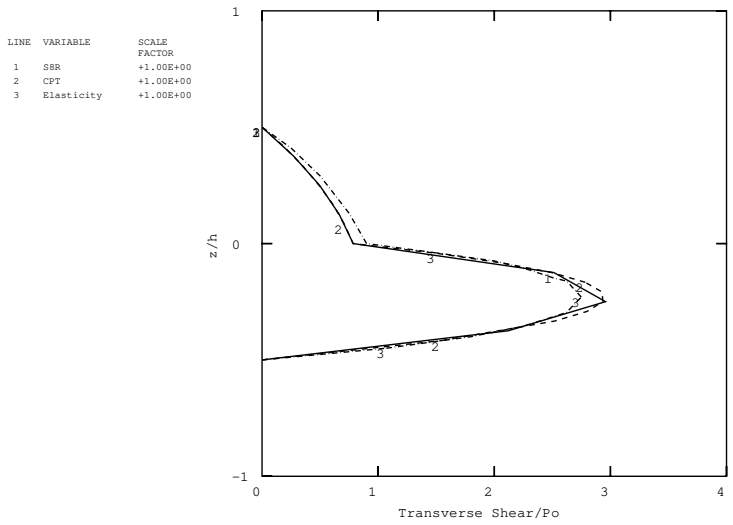


Figure 3: Transverse shear stress distribution through the thickness of a two-layer plate ($s = 4$); Abaqus/Standard analysis.

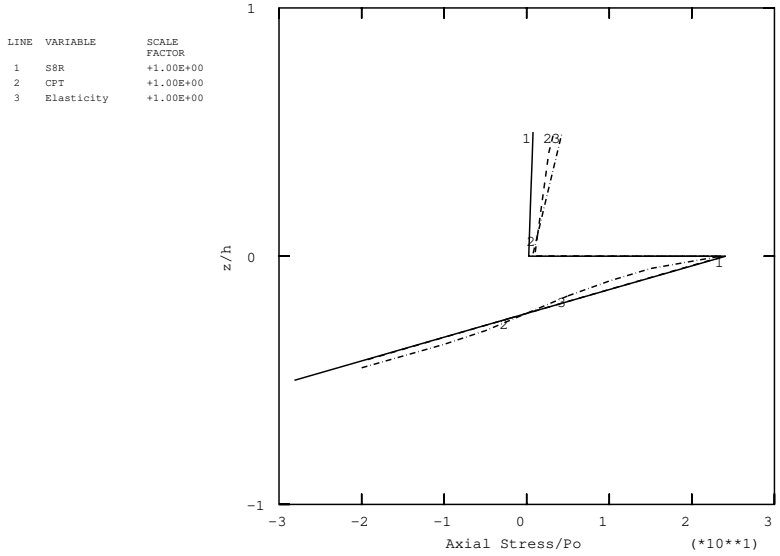


Figure 4: Axial stress distribution through the thickness of a two-layer plate ($s = 4$); Abaqus/Standard analysis.

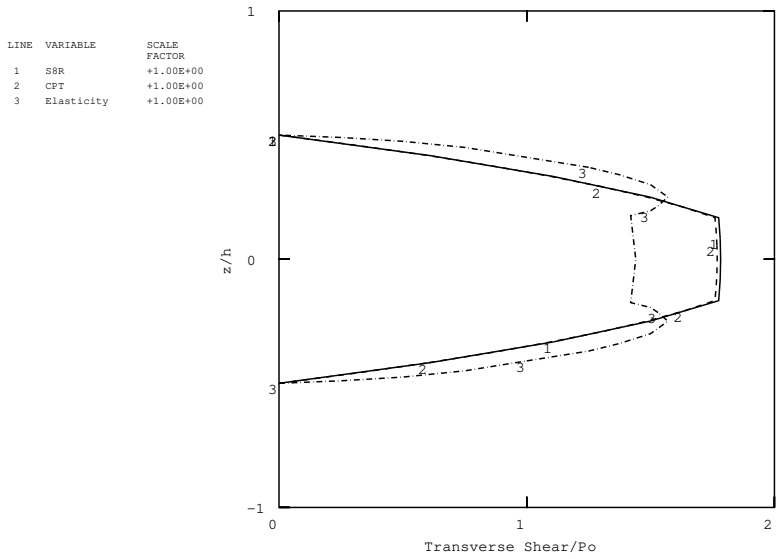


Figure 5: Transverse shear stress distribution through the thickness of a three-layer plate ($s = 4$); Abaqus/Standard analysis.

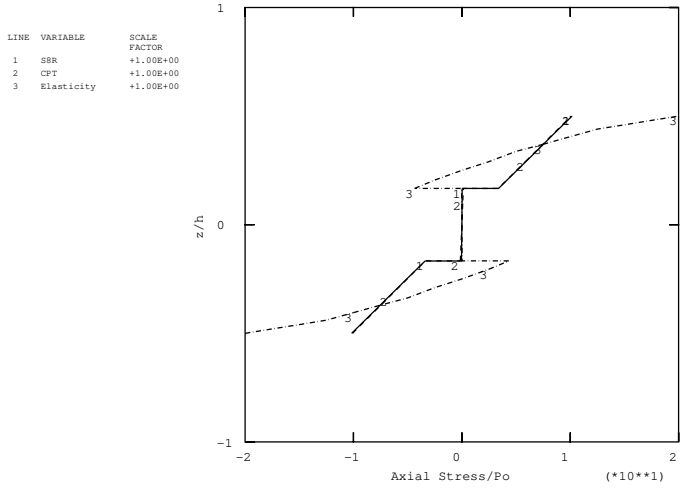


Figure 6: Axial stress distribution through the thickness of a three-layer plate ($s = 4$); Abaqus/Standard analysis.

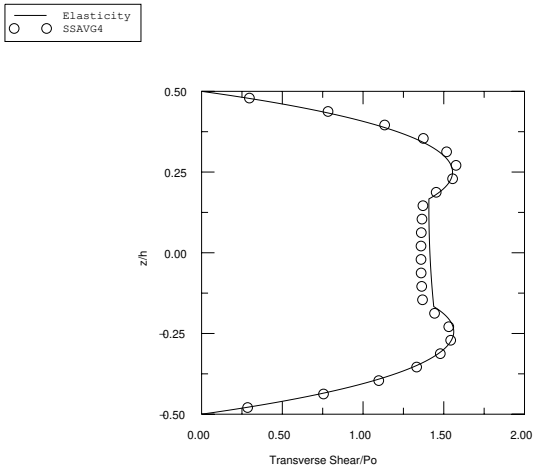


Figure 7: Comparison of the elasticity solution of the transverse shear stress distribution in a three-layer plate to the output variable SSAVG4 with 24 SC8R elements stacked through the thickness; Abaqus/Standard analysis.

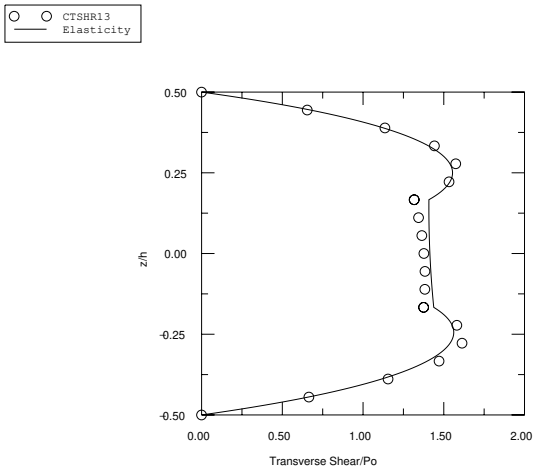


Figure 8: Comparison of the elasticity solution of the transverse shear stress distribution in a three-layer plate to the output variable CTSHR13 with 3 SC8R elements stacked through the thickness; Abaqus/Standard analysis.

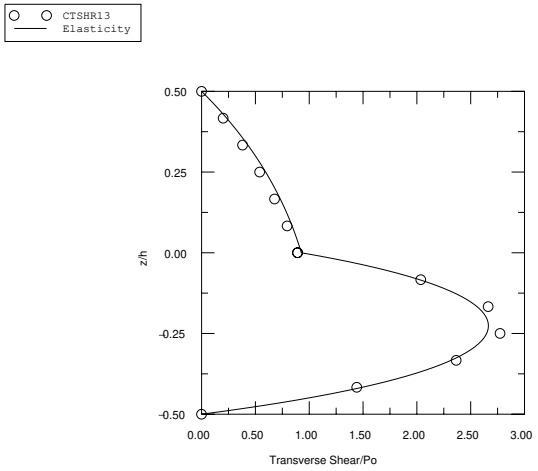


Figure 9: Comparison of the elasticity solution of the transverse shear stress distribution in a two-layer plate to the output variable CTSHR13 with 2 SC8R elements stacked through the thickness; Abaqus/Standard analysis.

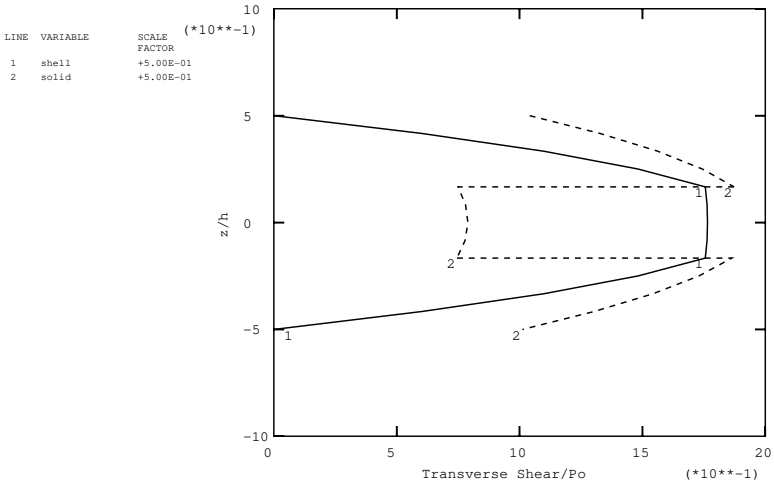


Figure 10: Transverse shear stress distribution through the thickness of a three-layer plate ($s = 4$): shells versus solid elements; Abaqus/Standard analysis.

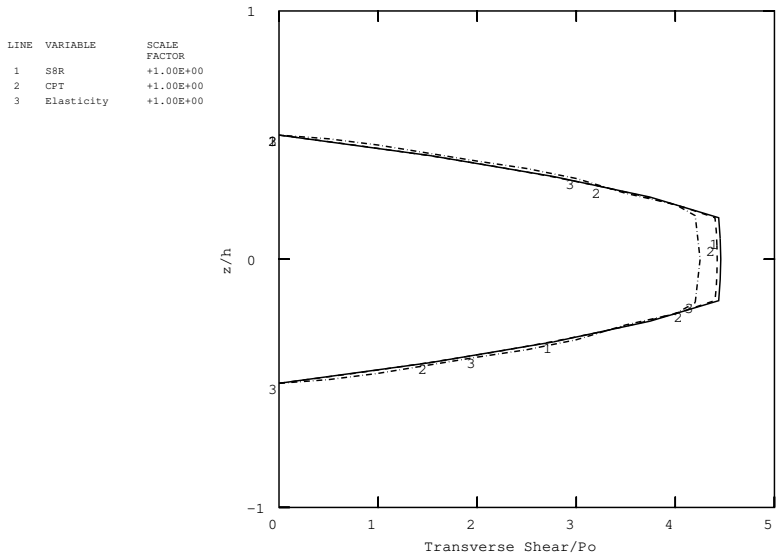


Figure 11: Transverse shear stress distribution through the thickness of a three-layer plate ($s = 10$); Abaqus/Standard analysis.

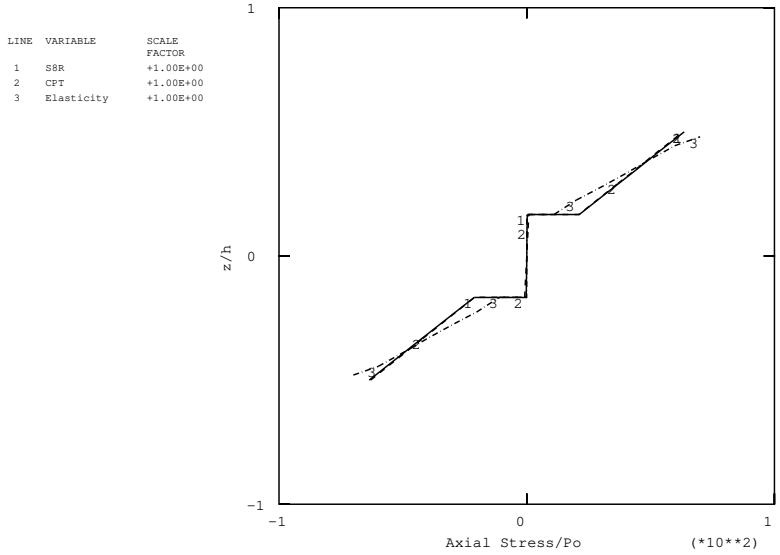


Figure 12: Axial stress distribution through the thickness of a three-layer plate ($s = 10$); Abaqus/Standard analysis.

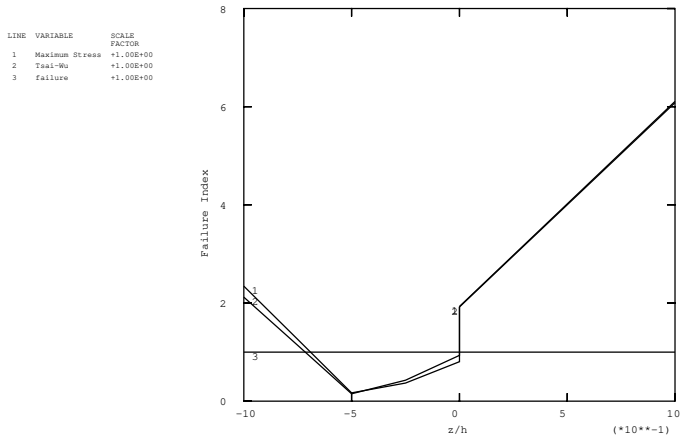


Figure 13: Maximum stress theory and Tsai-Wu theory ($F_{12} = 0.0$) failure indices as a function of normalized distance from the midsurface. Two-layer plate, $s = 4$; Abaqus/Standard analysis.

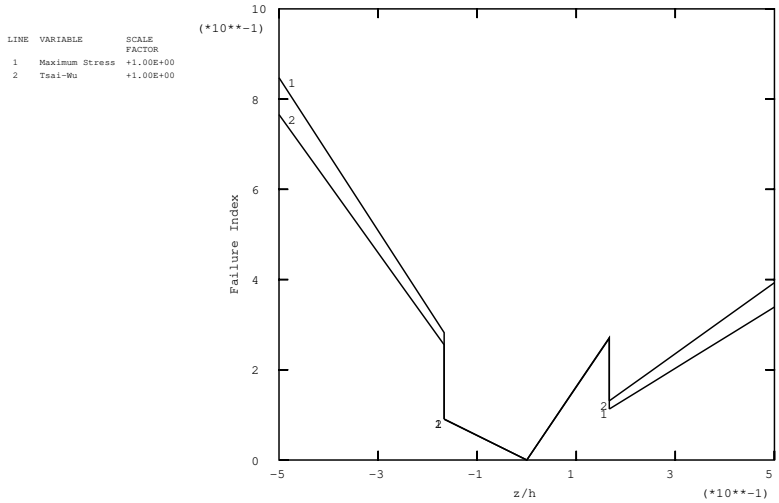


Figure 14: Maximum stress theory and Tsai-Wu theory ($F_{12} = 0.0$) failure indices as a function of normalized distance from the midsurface. Three-layer plate, $s = 4$; Abaqus/Standard analysis.

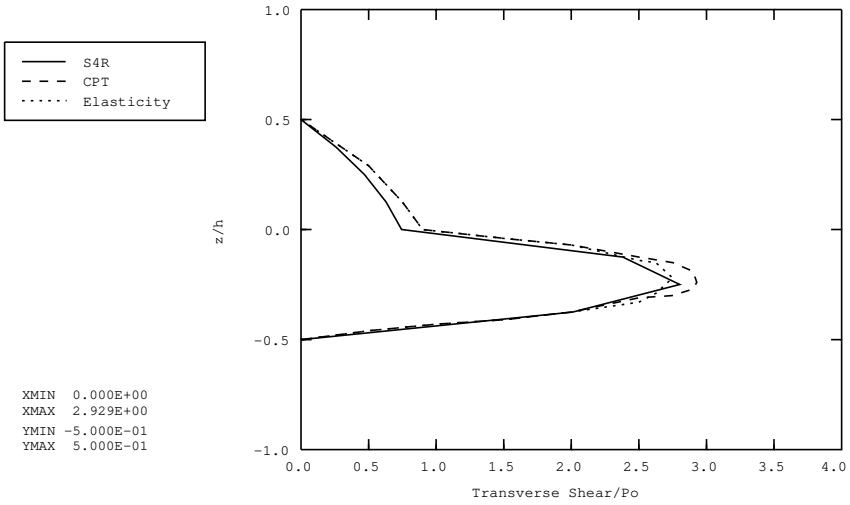


Figure 15: Transverse shear stress distribution through the thickness of a two-layer plate; Abaqus/Explicit analysis.

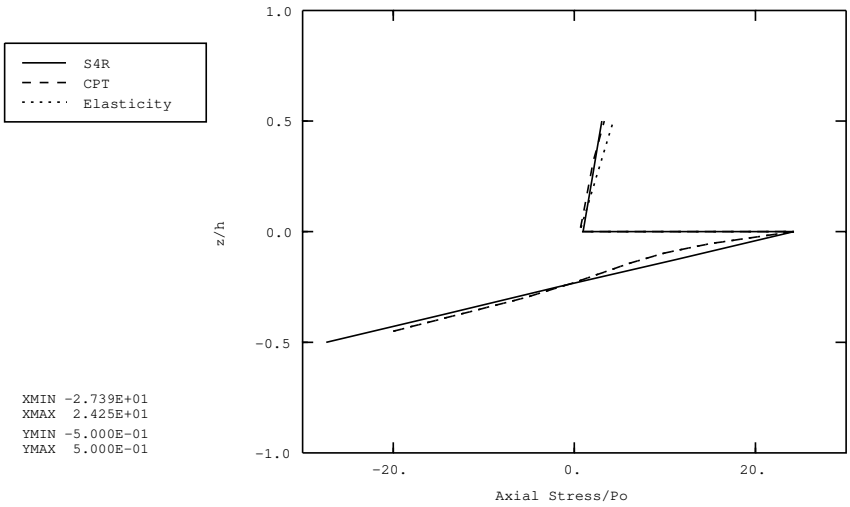


Figure 16: Axial stress distribution through the thickness of a two-layer plate; Abaqus/Explicit analysis.

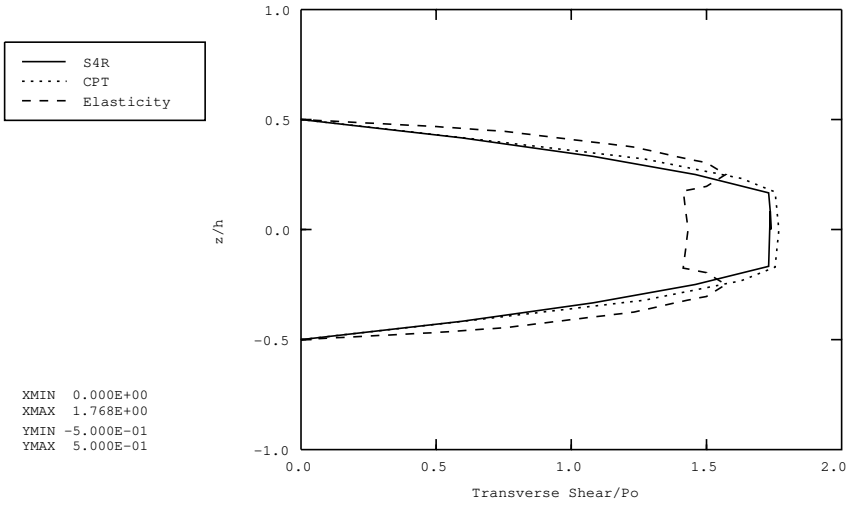


Figure 17: Transverse shear stress distribution through the thickness of a three-layer plate; Abaqus/Explicit analysis.

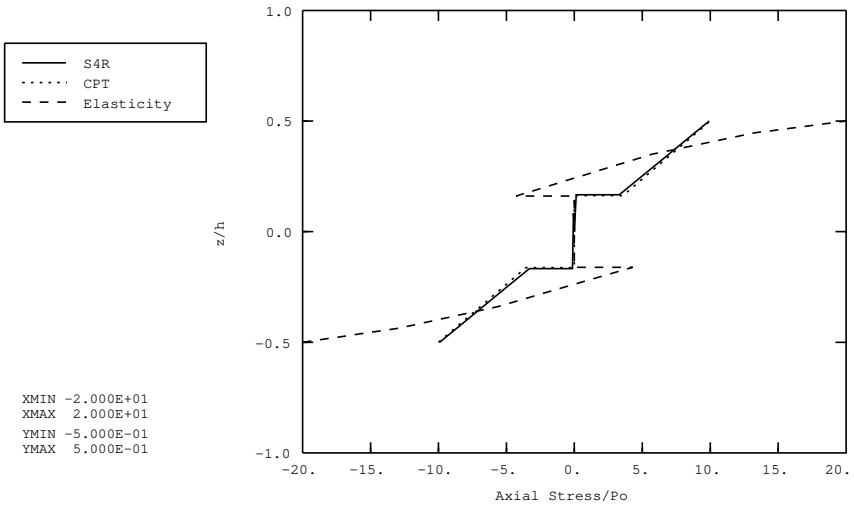


Figure 18: Axial stress distribution through the thickness of a three-layer plate; Abaqus/Explicit analysis.

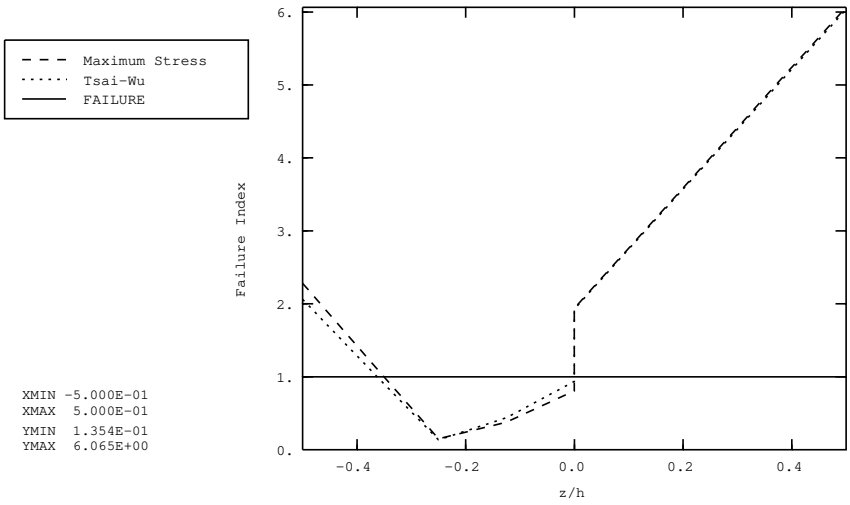


Figure 19: Maximum stress theory and Tsai-Wu theory failure indices as a function of normalized distance from the midsurface. Two-layer plate; Abaqus/Explicit analysis.

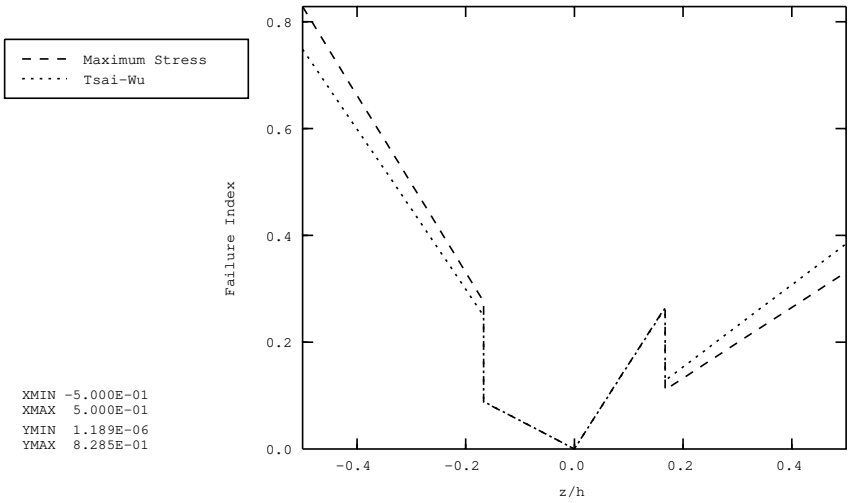


Figure 20: Maximum stress theory and Tsai-Wu theory failure indices as a function of normalized distance from the midsurface. Three-layer plate; Abaqus/Explicit analysis.

Thick composite cylinder subjected to internal pressure

Products: Abaqus/Standard

Overview

This example provides verification of the composite solid (continuum) elements in Abaqus.
The problem consists of an infinitely long composite cylinder, subjected to internal pressure, under plane strain conditions. The solution is compared with the analytical solution of Lekhnitskii (1968) and with a finite element model where each layer is discretized with one element through the thickness. A finite element analysis of this problem also appears in Karan and Sorem (1990).

Most composites are used as structural components. Shell elements are generally recommended to model such components. Illustrations of composite shell elements in bending can be found in *Analysis of an anisotropic layered plate*, *Composite shells in cylindrical bending*, and *Axisymmetric analysis of bolted pipe flange connections*. In some cases, however, the analyst cannot avoid the use of continuum elements to model structural components. In these problems careful selection of the element type is usually essential to obtain an accurate solution. The performance of continuum elements for the analysis of bending problems is discussed in *Performance of continuum and shell elements for linear analysis of bending problems*. The discussion considers only the behavior of structures composed of homogeneous materials, but the same considerations apply when modeling composite structures with continuum elements. In other cases the deformation through the thickness of the composite may be nonlinear—for example, when material nonlinearities are present—and several elements may be required through the thickness for an accurate analysis. Such a discretization can be accomplished only with continuum elements. Other problems where the use of continuum elements may be preferred include thick composites where transverse shear effects are predominant, composites where the normal strain cannot be ignored, and when accurate interlaminar stresses are required; i.e., near localized regions of complex loading or geometry. In these problems the solutions obtained by solid elements are generally more accurate than those obtained by shell elements. An exception is the distribution of transverse shear stress through the thickness. The transverse shear stresses in solid elements usually do not vanish at the free surfaces of the structure and are usually discontinuous at layer interfaces. A discussion of the transverse shear stress calculations for solid and shell elements can be found in *Composite shells in cylindrical bending*.

In this problem the normal strain cannot be ignored since the displacement field due to the internal pressure is nonlinear through the cylinder thickness. At least two quadratic elements through the thickness are required to obtain accurate results. The example, therefore, demonstrates the use of composite solid elements for a problem where a shell element analysis would be inadequate.

Problem description

The cylinder configuration and material details are shown in *Figure 1*. The inside radius, d_i, is 60 mm, and the outside radius, d_o, is 140 mm. The structure consists of eight orthotropic layers of equal thickness, arranged in a stacking sequence of $[0°, 90°]_4$. The laminae are stacked in the radial direction, with the material fibers oriented along the circumferential and axial directions. In other words, the fibers are rotated 0° or 90° about the radial direction, where a 0° rotation implies primary fibers oriented along the circumferential direction. For this purpose we define a local coordinate system where the 1, 2, and 3 directions refer to the radial, circumferential, and axial directions, respectively. The fiber composite with the primary fibers along the circumferential direction has the following orthotropic elastic properties in this coordinate system:

$E_1 = 10.0$ GPa, $E_2 = 250.0$ GPa, $E_3 = 10.0$ GPa,

$G_{12} = G_{23} = 5.0$ GPa, $G_{13} = 2.0$ GPa,

$\nu_{12} = 0.01$, $\nu_{23} = \nu_{13} = 0.25$.

We also define the composite with the primary fibers along the axial direction of this local coordinate system.

Recognizing that the Poisson's ratios, ν_{ij}, must obey the relations $\frac{\nu_{ij}}{E_i} = \frac{\nu_{ji}}{E_j}$ for an orthotropic material with engineering constants, the rotated material properties are

$E_1 = 10.0$ GPa, $E_2 = 10.0$ GPa, $E_3 = 250.0$ GPa,

$G_{12} = 2.0$ GPa, $G_{13} = G_{23} = 5.0$ GPa,

$\nu_{12} = 0.25$, $\nu_{23} = \nu_{13} = 0.01$.

Each of these sets of elastic material properties is specified by giving the engineering constants. The name of each material is referred to in the composite solid section definition. This material definition ensures that the output components in the different layers are provided in the same coordinate system.

There is another method in Abaqus that can be used to define the ply orientation of the composite material. In this method only one definition of the material properties is used, but a separate orientation definition is given for each layer. This layer orientation is specified, together with the material name, in the solid section definition. The orientation can be specified by referring to a local coordinate system or by specifying an angle relative to the section orientation definition. The section orientation is specified in the solid section definition. Since the material properties of each layer in this case are specified in a different local coordinate system, the output variables are provided in different coordinate systems. Input files illustrating both methods are provided.

In addition to the material description for each layer, the stacking direction, the thickness of each layer, and the number of section points through the layer thickness required for the numerical integration of the element matrices to complete the description of the composite arrangement are defined. Three section integration points are specified in each layer. Since the analysis is linear elastic, this is sufficient to describe the stress distributions through the section. The layers can be stacked in any of the three isoparametric element coordinate directions, which—in turn—are defined by the order in which the nodes are given on the element data line. In this example the element connectivity is specified so that the first isoparametric direction lies along the radial direction.

Geometry and model

Because of symmetry, only a segment of the body needs to be analyzed. For simplicity of boundary condition application a quarter segment is chosen and is discretized with four elements in the circumferential direction and one element in the axial direction. One, two, four, or eight elements are used in the radial direction. *Figure 2* shows the finite element discretization for the case where two elements are used in the radial direction. A nonuniform mesh, with two material layers in the inside element and six layers in the outside element, is used to capture the variation of the radial displacement through the section.

The model is bounded in the axial direction to impose plane strain conditions.

The load is a constant internal pressure of $P =$ 50 MPa applied in a linear perturbation step.

Results and discussion

All displacements and stresses reported here are normalized with respect to pressure, using

$$\overline{u} = \frac{u}{p} \times 10^3 \; ; \qquad \overline{\sigma} = \frac{\sigma}{p}.$$

The predicted displacements and stresses at the inside and outside surfaces of the cylinder are compared with the analytical results in *Table 1* and *Table 2*. Results are shown for different element types and for different mesh densities. The tables show that a model discretized with one solid element (linear or quadratic) in the radial direction is inadequate to model the nonlinear variation of the displacement field. A substantial improvement is

obtained with two elements through the thickness. The tables further show that the convergence of the finite element results onto the analytical solution is slow with mesh refinement. A mesh with two nonuniform quadratic elements through the thickness predicts remarkably accurate results, with the exception of the circumferential stress at the outside surface of the cylinder. The outside stress is, however, more than two orders of magnitude smaller than the inside stress and is, therefore, not a good measure of the accuracy of the solution.

The displacement and stress fields through the thickness are shown in *Figure 3* through *Figure 5*. The figures compare the normalized radial displacement, the circumferential stress, and the radial stress with the analytical solution for the case where the cylinder is discretized with two C3D20R elements (of different sizes) in the radial direction. The figures show that the radial displacement and circumferential stress are in good agreement with the analytical solution. The radial stress, especially near the inside of the cylinder, is not quite as accurate. For example, the analytical solution at the inside surface is $\overline{\sigma}_r = -1.0$ ($\sigma_r = -p$). The finite element result for this mesh is $\overline{\sigma}_r = -0.741$ (25.9% error). This result must be seen in light of mesh refinement; no improvement in the radial stress at the inside surface is obtained with four elements through the thickness, and it only improves to $\overline{\sigma}_r = -0.926$ (7.4% error) when eight elements are used through the thickness (the results for the four-element and eight-element meshes are not shown in the figures). It is clear from these figures why quadratic elements and a refined mesh are required for an accurate analysis.

Input files

thickcompcyl_2el_nonuniform.inp	Model discretized with two nonuniform elements in the radial direction.
thickcompcyl_1el_sectorient.inp	Model in which the ply orientation is specified with a rotation relative to the section orientation. This model is discretized with one element in the radial direction.
thickcompcyl_4el_orient.inp	Model in which the ply orientation is specified with an orientation reference. This model is discretized with four elements in the radial direction.
thickcompcyl_8el.inp	Model in which each layer is discretized with one homogeneous element through the thickness.

References

- Karan, S. S., and R. M. Sorem, “Curved Shell Elements Based on Hierarchical *p*-Approximation in the Thickness Direction for Linear Static Analysis of Laminated Composites,” International Journal for Numerical Methods in Engineering, vol. 29, pp. 1391–1420, 1990.
- Lekhnitskii, S. G., *Anisotropic Plates*, translated from second Russian edition by S. W. Tsai and T. Cheron, Gordon and Breach, New York, 1968.

Tables

Table 1: Normalized radial displacement at inside and outside of cylinder. Analytical solution: $\overline{u}_r^{(i)} = 1.4410$; $\overline{u}_r^{(o)} = 0.1476$.

Element type	Elements in radial direction	Inside		Outside	
		u_r	% error	u_r	% error
C3D8	1	1.1825	17.9	–0.2407	263.0
C3DI	1	1.2227	15.2	0.1004	32.0
C3DI[(1)]	2	1.4231	12.4	0.1876	27.1

Element type	Elements in radial direction	Inside		Outside	
		u_r	% error	u_r	% error
C3DI[2]	2	1.5526	7.74	0.1828	23.8
C3D20R	1	1.2581	12.7	0.1646	11.5
C3D20R[1]	2	1.3609	5.56	0.1448	1.90
C3D20R[2]	2	1.3869	3.75	0.1481	0.34
C3D20R	4	1.3922	3.39	0.1447	1.95
C3D20R	8	1.4161	1.73	0.1496	1.35
1 - Uniform mesh					
2 - Nonuniform mesh					

Table 2: Normalized circumferential stress at inside and outside of cylinder. Analytical solution: $\overline{\sigma}_\theta^{(i)} = 5.7060; \overline{\sigma}_\theta^{(o)} = 0.0103$.

Element type	Elements in radial direction	Inside		Outside	
		σ_θ	% error	σ_θ	% error
C3D8	1	3.608	36.8	–0.0307	397.0
C3DI	1	3.912	31.4	0.0362	251.1
C3DI[1]	2	4.686	17.9	0.004	60.8
C3DI[2]	2	4.838	15.2	–0.0081	179.1
C3D20R	1	5.132	10.1	0.0414	300.0
C3D20R[1]	2	5.496	3.68	0.0134	30.0
C3D20R[2]	2	5.548	2.77	0.0192	85.6
C3D20R	4	5.574	2.31	0.0119	15.1
C3D20R	8	5.606	1.75	0.0107	3.90
1 - Uniform mesh					
2 - Nonuniform mesh					

Figures

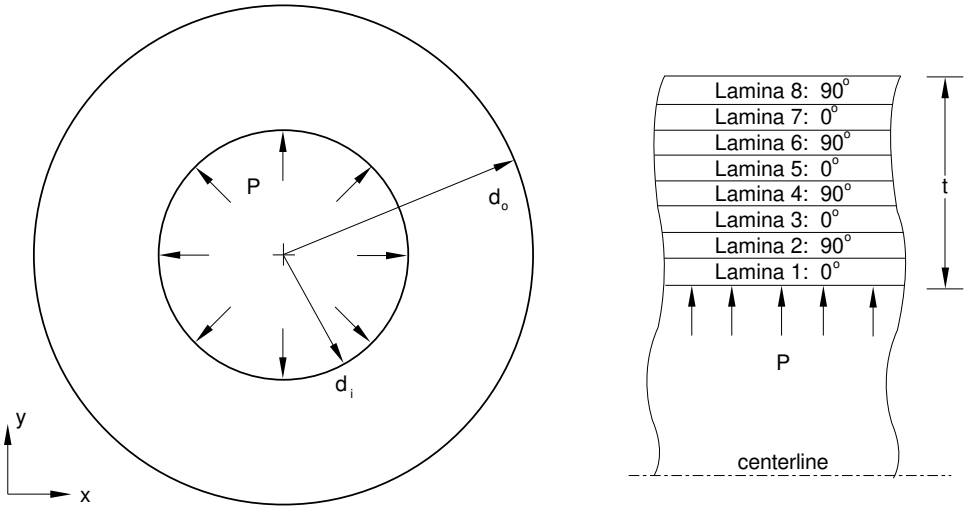


Figure 1: Geometry of laminated cylinder.

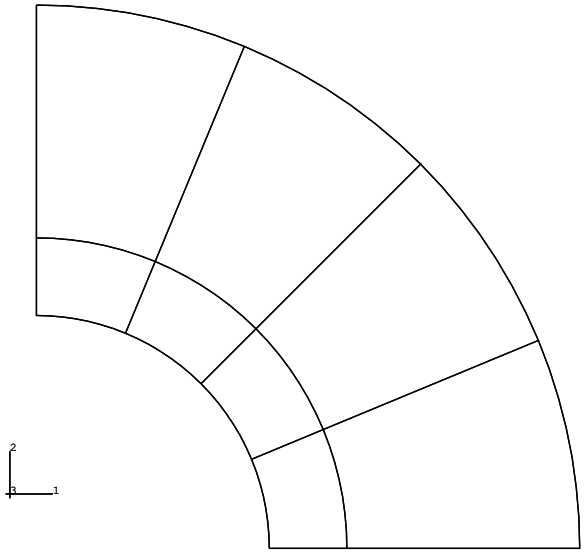


Figure 2: Finite element discretization with two elements in the radial direction.

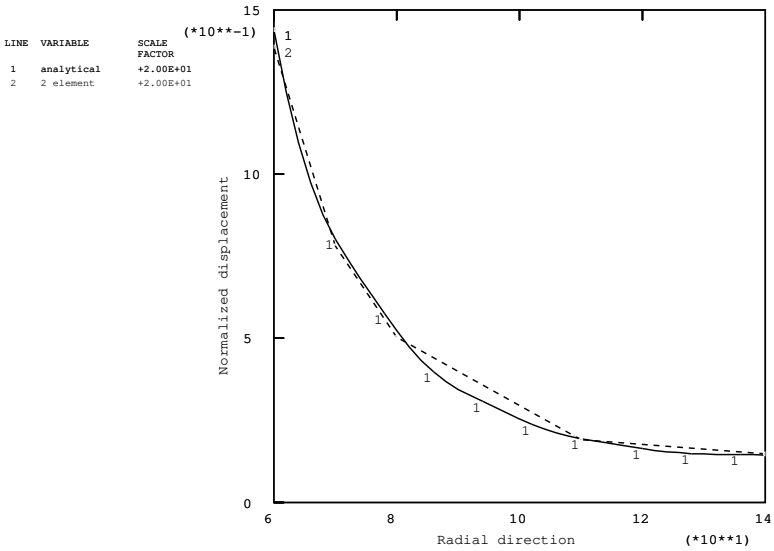


Figure 3: Radial displacement versus cylinder radius.

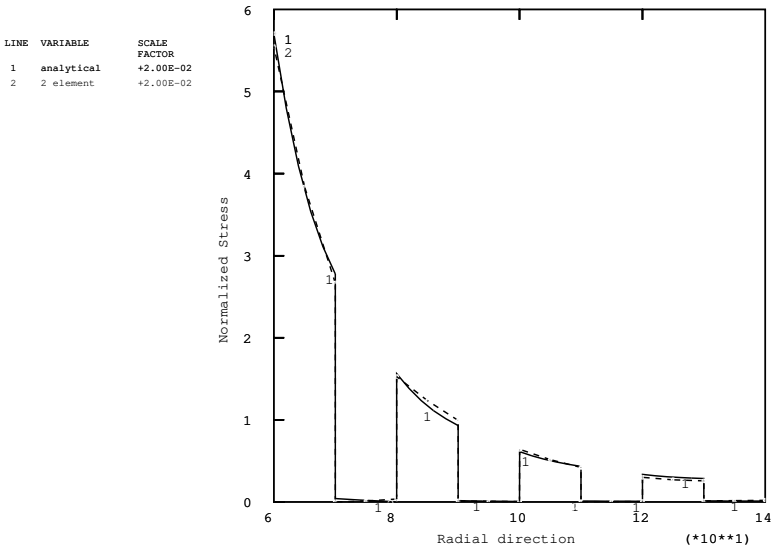


Figure 4: Circumferential stress versus cylinder radius.

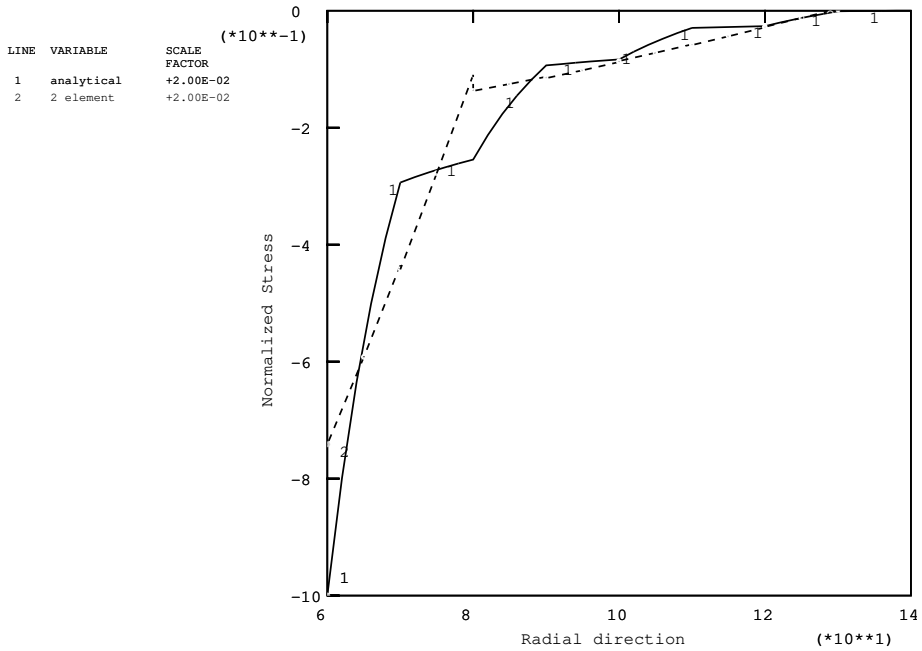


Figure 5: Radial stress versus cylinder radius.

Uniform collapse of straight and curved pipe segments

Products: Abaqus/Standard

Overview

This example provides verification of the uniform collapse of initially straight and initially curved pipe segments.

The failure of pipe segments under conditions of pure bending is an interesting problem of nonlinear structural response. In the case of straight, thin-walled, metal cylinders, the failure usually occurs by the cylinder buckling into a pattern of small, diamond-shaped waves, in the same fashion as a cylinder failing under axial compression (see *Buckling of a cylindrical shell under uniform axial pressure*). The use of peak axial stress as a buckling criterion, taking the same critical value for any combination of axial load and bending moment, is a useful design approach—see Chapter 11 of Timoshenko and Gere (1961). However, for thicker walled cases, when the material modulus is low (such as rubber or a metal tube that shows significant yield before it collapses), it is possible to observe uniform collapse of the cylinder, in the sense that the pipe gradually ovalizes out of round and, thus, loses its bending stiffness. This one-dimensional deformation pattern in initially straight pipes was originally investigated by Brazier (1927). The collapse of initially curved pipes under bending moments is a rather different case because the response of the pipe will depend on whether the moment causes in-plane or out-of-plane response. In this example we look at in-plane loading only. For both cases the mode of deformation being studied is uniform collapse of the section—that is, it is assumed that all cross-sections deform in the same way. Since shell theory is used, this effectively reduces the problems to one dimension, thus making them attractive introductory studies to the investigation of structural collapse. It should be emphasized that, for the actual structure, the possibility of diamond-pattern buckling remains and should be investigated (by using appropriately detailed shell models) before using the results obtained in these examples for design—see *Buckling of a cylindrical shell under uniform axial pressure*. *Elastic-plastic collapse of a thin-walled elbow under in-plane bending and internal pressure* investigates collapse of curved and straight pipe segments of the same material and dimensions, but put together into an actual 90° piping elbow with adjacent straight pipe runs, thus describing a more realistic case.

The one-dimensional cross-sectional ovalization pattern expected allows very simple modeling to be adopted. Element type ELBOW31B is a pipe with uniformly deforming cross-section (using Fourier interpolation around the pipe) and, thus, is ideal for these cases: a single element suffices. As a companion, the problems are also modeled with a single axial segment of general 8-node shell elements (type S8R5). This case is somewhat more complicated because the ends of the segment modeled must be constrained to allow ovalization but no warping. Such conditions can be implemented using surface-based kinematic and distributing coupling constraints, as demonstrated in this example problem.

Problem description

The pipes chosen for the study are relatively thin-walled, large radius pipes and are shown in *Figure 1* and *Figure 2*. The dimensions of the pipes are taken from the more complex elbow collapse study. A unit length of pipe is considered. The material is the same and is the measured response of type 304 stainless steel specimens at room temperature, as reported by Sobel and Newman (1979). The stress-strain curve is shown in *Figure 3*. Results are also obtained for elastic response only, which is the case discussed by Brazier for collapse of an initially straight pipe.

Loading

The load on the pipe has two components—a “dead” load, consisting of internal pressure (with a closed-end condition), and a “live” load consisting of pure bending. The pressure is applied to the model in an initial step and then held constant as the bending moment is increased. Four different pressure values are used, ranging from no pressure to 5.17 MPa (750 lb/in^2). This range seems to cover all practical values; the highest pressure gives

a membrane hoop stress value of about 97 MPa (14000 lb/in^2). For the shell models the equivalent end force caused by the closed-end condition is applied as a follower force because it rotates with the motion of the end plane.

Models

In all of the cases involving elastic-plastic response, seven integration points are used through the pipe wall. 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. In problems involving significant strain reversals (such as ratcheting or low-cycle fatigue studies), nine or more points are generally recommended.

Elbow element

The elbow element model consists of one element of type ELBOW31B. One node is restrained in all six degrees of freedom; the other is free, except for the prescribed rotation. A rotation is prescribed rather than a moment, since it is anticipated that the collapse will be unstable.

For comparison two levels of Fourier interpolation are used in the element: four modes, with 12 integration points around the pipe, and six modes, with 18 integration points around the pipe.

Typical elbow element input data for this problem are shown in *unifcollapspipe_str_elbelem.inp* and *unifcollapspipe_curv_elbelem.inp*.

Shell element

The shell element model has six elements of type S8R5 around the half-pipe. Mesh convergence studies, not included in this example, have demonstrated that such a mesh gives accurate predictions of strains and displacements in this case.

Constraints and boundary conditions for the shell element model

For the shell model the main problem is to prescribe appropriate boundary conditions. The plane $z = 0$ is a plane of symmetry, and so for nodes on that plane we must have

$$u_z = \phi_x = \phi_y = 0.$$

The motion is also symmetric about any rotated cross-sectional plane. To remove the rigid body rotation mode about the z-axis, we can choose one cross-sectional plane that does not rotate. This is taken to be the plane $x = 0$. For all nodes on that plane the symmetry constraints are

$$u_x = \phi_y = \phi_z = 0.$$

At the other end of the piece of pipe being modeled we need the same conditions, but with respect to the rotated axis system, the rotation being about the z-axis only. To impose these conditions we introduce a “beam” node, labeled b, to represent the motion of the end plane. This node is defined to have global displacement components u_x^b, u_y^b, and rotation ϕ_z^b, as its degrees of freedom. Pure bending of the shell model is modeled by prescribing the rotation

$$\boldsymbol{\phi}^b = \phi_z^b \mathbf{e}_z$$

for the “beam” node. A rotation is prescribed rather than a moment, since it is anticipated that the collapse of the pipe will be unstable.

Surface-based kinematic and distributing couplings are applied to impose the necessary symmetry constraint on the nodes at the end of the pipe section, and a surface-based distributing coupling element is used to remove the translational rigid body mode of the pipe.

A kinematic coupling can be applied to constrain the nodes on the end plane of the shell model to impose the symmetry constraint while permitting ovalization of the cross-section. These nodes have to remain coplanar with respect to the end cross-sectional plane, with the orientation of this end plane determined by the rotation of the reference node, which is referred to as the “beam” node.

Such a condition can be implemented by constraining the end plane nodes to follow the motion of the beam node in the direction normal to the end plane. Since the constraint directions in a kinematic coupling corotate with the motion of the reference node, which in this particular model would be the beam node, the plane determined by the constraint direction would rotate along with the beam node. The initial normal to the end plane would be in the x-direction, with the end plane nodes free to translate in the y- and z-directions. However, these directions would be determined subsequently by the rotated axis system, following the motion of the beam node.

The translational rigid body mode in the y-direction can be removed by constraining the average y-direction motion of the nodes on the rotating end plane. A distributing coupling is used to constrain the average motion of the end nodes to the motion of its reference node. This reference node is then constrained in the y-direction, which constrains the motion of the end nodes only in an average sense. This can be expressed as

$$\frac{1}{S}\int_S u_y dS = u_y^b.$$

The elements in the shell model (S8R5) use quadratic interpolation functions; hence, the weighting factors for the nodal displacements work out to 1/6 for the corner nodes and 4/6 for the midside nodes. However, since most of the corner nodes are connected to two elements, the weights used for the distributing coupling for such nodes are 2/6, considering the contribution to both the neighboring elements. Since the only purpose of the distributing coupling is to prevent rigid body motions, the choice of weight factors is not critical.

Results and discussion

The results for the two models are discussed below.

Initially straight pipe

The results based on the elastic material assumption are summarized in *Figure 4* and *Figure 5*. These plots are based on the analyses with shell elements. *Figure 4* shows the variation of moment with curvature of the pipe. The unstable behavior of the collapse is evident from this plot in that the moment reaches a peak and then decreases with increasing curvature. Brazier's (1927) solution is also shown in this plot. Brazier's analysis is a first-order correction only to the usual bending theory and does not consider any pressure effect. It agrees well with the present zero pressure results up to peak load. The stiffening effect of internal pressure P is evident in this plot: the peak moment at the highest pressure (5.17 MPa, 750 lb/in^2) is about 28% above the peak moment with zero pressure. The magnitude of the deformation is shown in *Figure 5*, where the outside dimension of the pipe section in the x–y plane is shown as a function of curvature.

The results with the elastic-plastic material behavior are rather different and are shown in *Figure 6* and *Figure 7*. As we would expect, the moments are much lower. In addition, the severe instability in the behavior is now reduced by the internal pressure—so much so that the highest pressure solution always shows positive stiffness, even at quite large curvatures. There is also far less ovalization of the cross-section in this elastic-plastic case: the pipe is losing bending stiffness by yielding and, thus, reduces distortion of the cross-section.

The elbow and shell element models are compared in *Figure 8* (elastic, no pressure) and *Figure 9* (elastic-plastic, no pressure). The elbow element models agree well with the shell element solutions, up to well beyond the collapse point, using either four or six modes, which illustrates the relative efficiency of the elbow elements for such a case.

Initially curved pipe

For the initially curved pipe an appropriate orientation must be used to impose the kinematic coupling correctly since the constraint directions on the end plane are not aligned initially with the global coordinate system. The

results for an initially curved pipe, based on the elastic material assumption, are shown in *Figure 10* and *Figure 11*. The response is quite different from the straight pipe results, in that opening and closing moments give distinctly different responses. With an opening moment, the ovalization of the section tends to increase the pipe's resistance to further bending, thus giving stiffening response. Under a closing moment, the pipe becomes progressively weaker in bending and never attains more than 20–25% of the moment possible in the straight pipe. The effect of internal pressure is now very much less than in the corresponding straight pipe, and the change in pipe dimensions (as shown in *Figure 11*) is not as severe.

The elastic-plastic results for the same case are summarized in *Figure 12* and *Figure 13*. In contrast to the corresponding straight pipe solutions (*Figure 6* and *Figure 7*), the closing moment solutions show collapse (negative stiffness) at all values of internal pressure tested.

The effect of internal pressure is quite significant. The opening moment cases with lower pressures show an interesting behavior: the initial weakening of the section caused by yielding is to some extent offset later in the loading by the stiffening associated with large-displacement effects.

The elbow and shell element models are compared in *Figure 14* (elastic, no pressure) and *Figure 15* (elastic-plastic, no pressure).

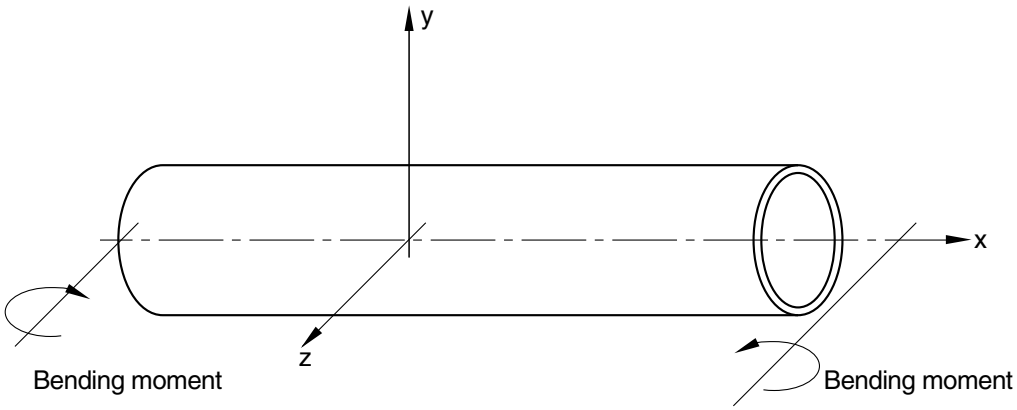
Input files

unifcollapspipe_str_elbelem.inp	Straight pipe, elastic analysis (4 Fourier mode elbow element model).
unifcollapspipe_str_shellkcdc.inp	Straight pipe, no pressurization, elastic analysis (shell element model).
unifcollapspipe_curv_elbelem.inp	Initially curved pipe, opening mode, elastic analysis (4 Fourier mode elbow element model).
unifcollapspipe_curv_shellkcdc.inp	Initially curved pipe, opening mode, no pressurization, elastic analysis (shell element model).

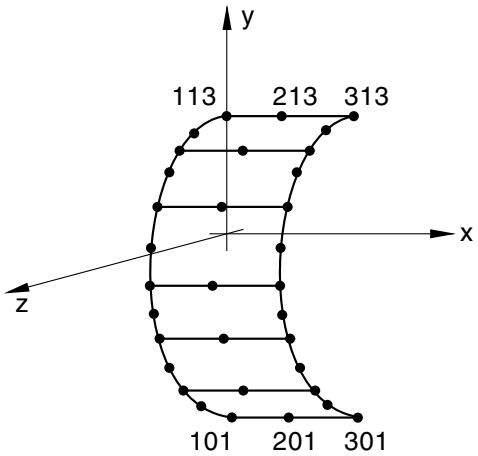
References

- Brazier, L. G., “On the Flexure of Thin Cylindrical Shells and Other 'Thin' Sections,” *Proceedings of the Royal Society, London, Series A*, vol. 116, pp. 104–114, 1927.
- Sobel, L. H., and S. Z. Newman, “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.
- Timoshenko, S. P., and J. M. Gere, Theory of Elastic Stability, McGraw-Hill, New York, 1961.

Figures



Pipe outside diameter	406.9 mm (16.02 in)
Pipe wall thickness	10.4 mm (0.41 in)



Shell mesh

Figure 1: Brazier problem: pure bending collapse of an initially straight pipe.

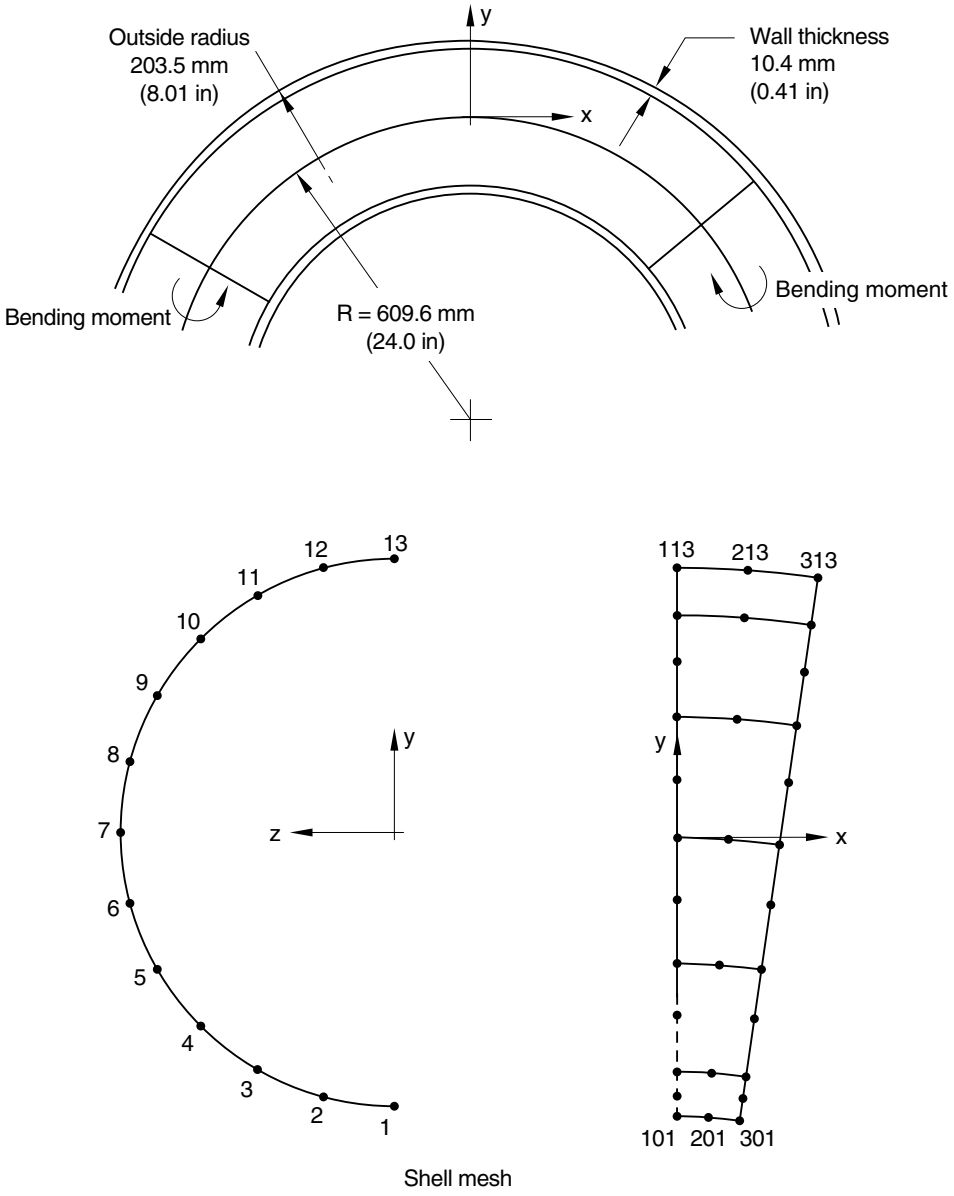
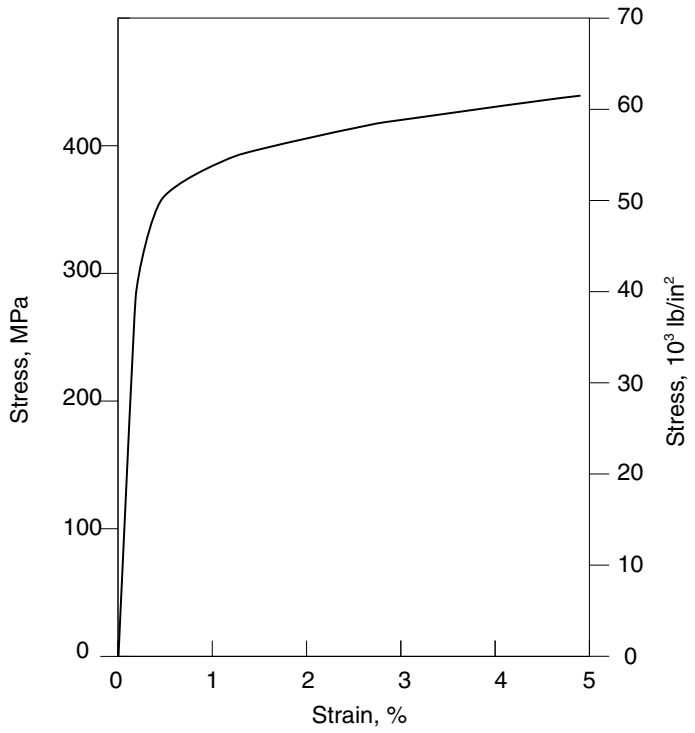


Figure 2: Curved pipe bending problem.



Young's modulus: 193 GPa (28 x 10^6 lb/in^2)
Poisson's ratio: 0.2642

Plastic strain	Stress, MPa	Stress, lb/in^2
0.0	271.93	39440
0.00473	345.91	50170
0.01264	378.87	54950
0.02836	403.62	58540
0.04910	424.17	61520

Figure 3: Assumed stress-strain behavior for pipe material.

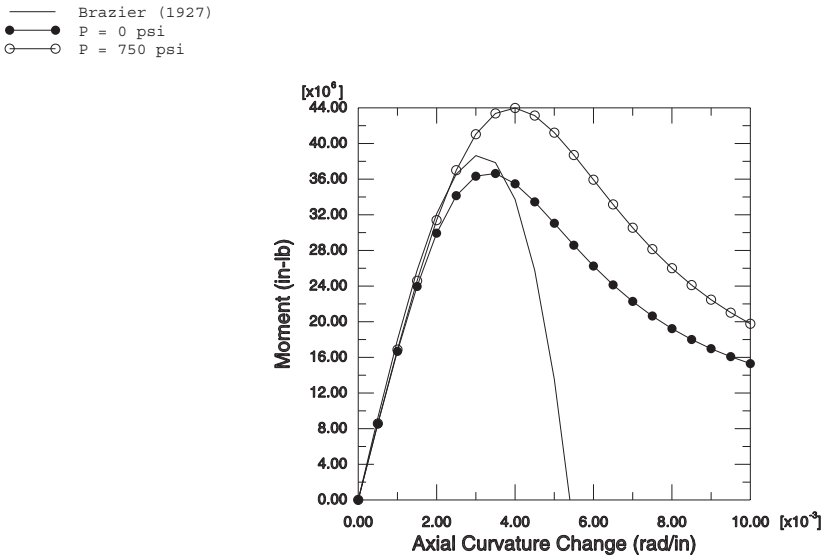


Figure 4: Moment-curvature—initially straight, elastic pipe (shell model).

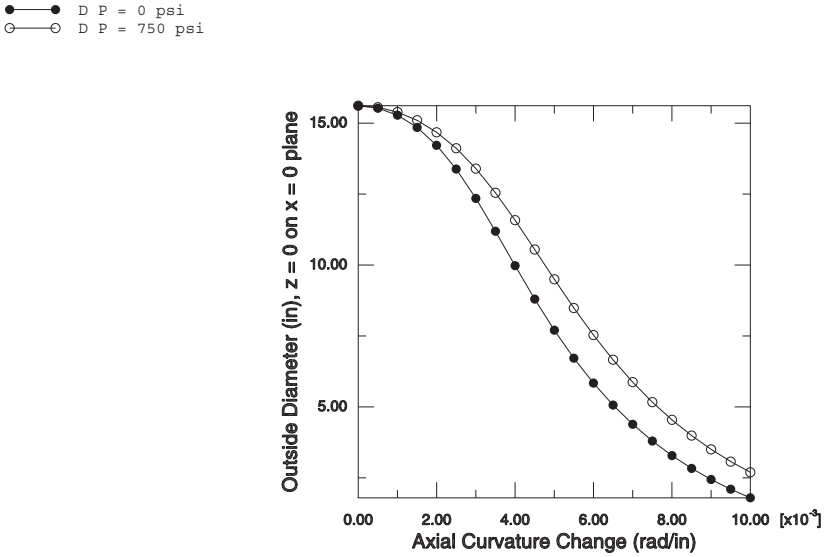


Figure 5: Deformation of section—initially straight, elastic pipe (shell model).

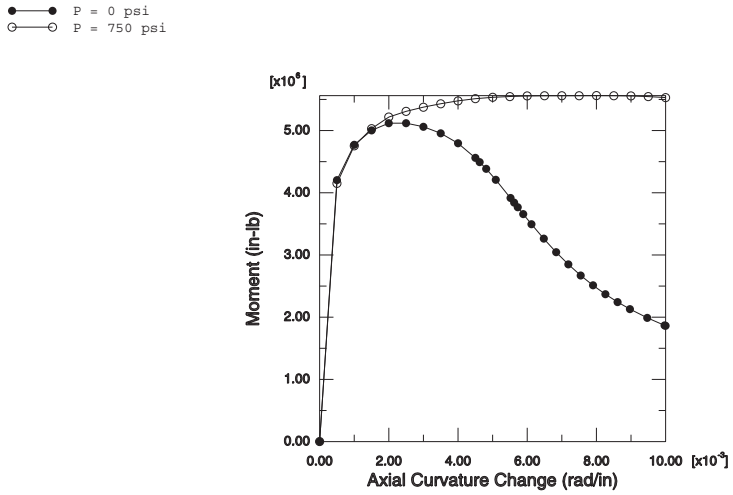


Figure 6: Moment-curvature—initially straight, elastic-plastic pipe (shell model).

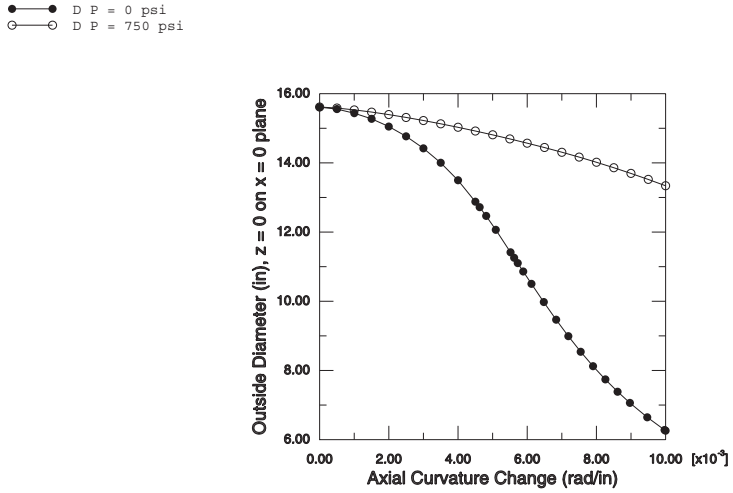


Figure 7: Deformation of section—initially straight, elastic-plastic pipe (shell model).

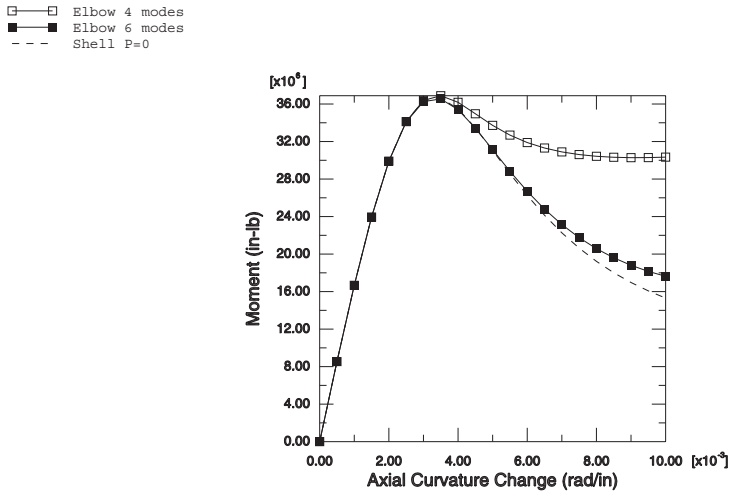


Figure 8: Moment-curvature—comparison of shell and elbow models, initially straight, elastic pipe.

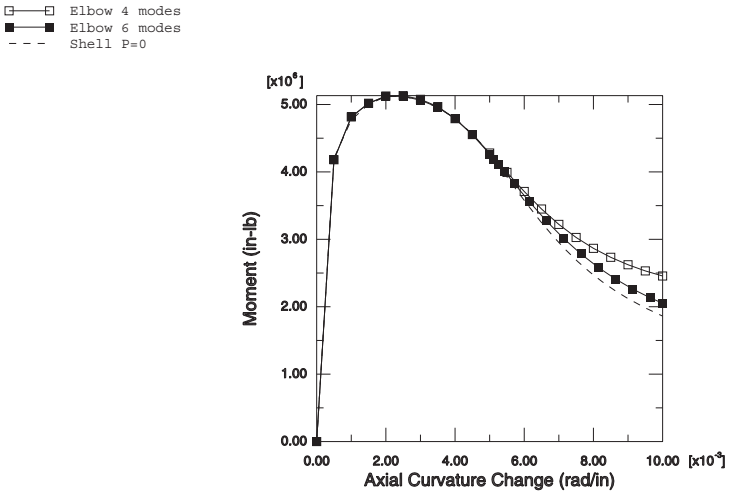


Figure 9: Moment-curvature—comparison of shell and elbow models, initially straight, elastic-plastic pipe.

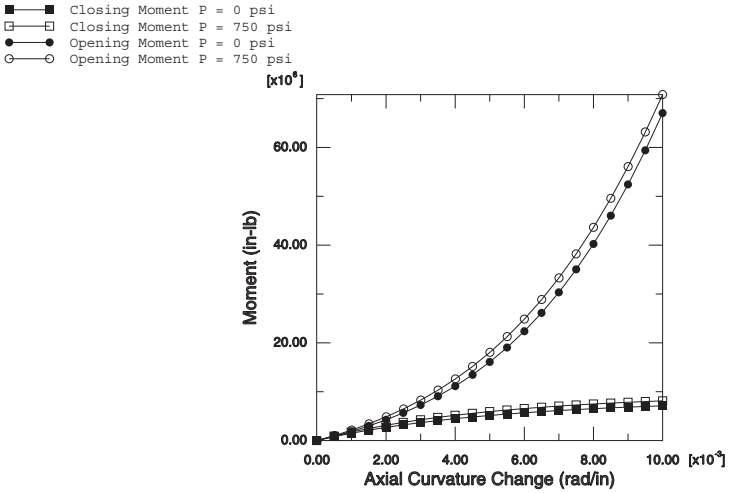


Figure 10: Moment-curvature—initially curved, elastic pipe (shell model).

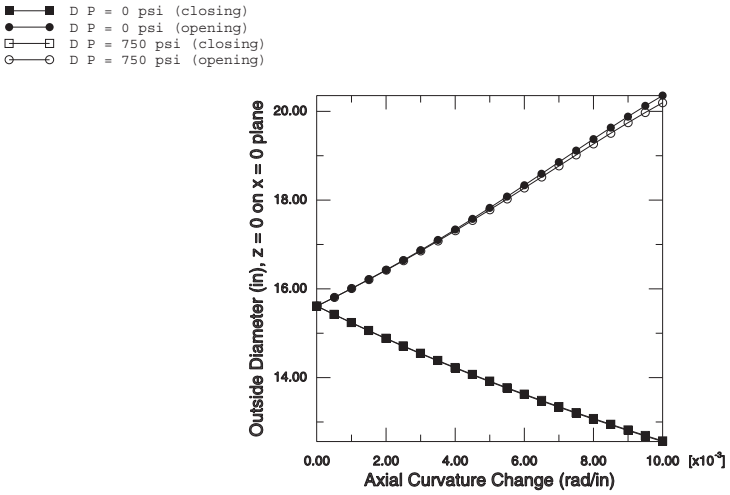


Figure 11: Deformation of section—initially curved, elastic pipe (shell model).

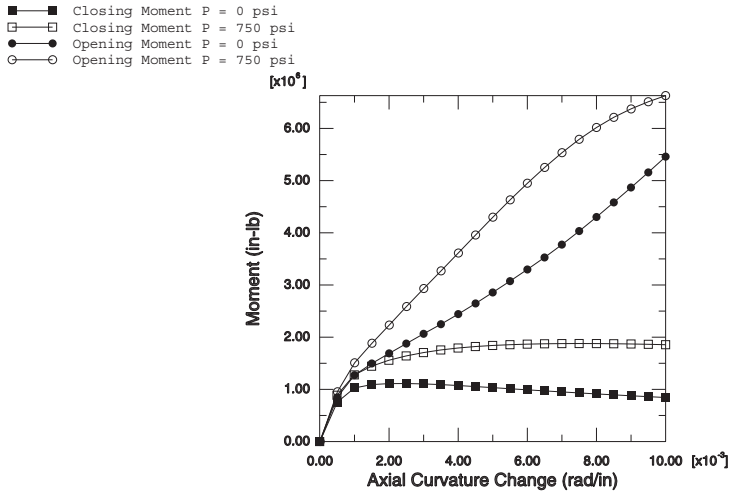


Figure 12: Moment-curvature—initially curved, elastic-plastic pipe (shell model).

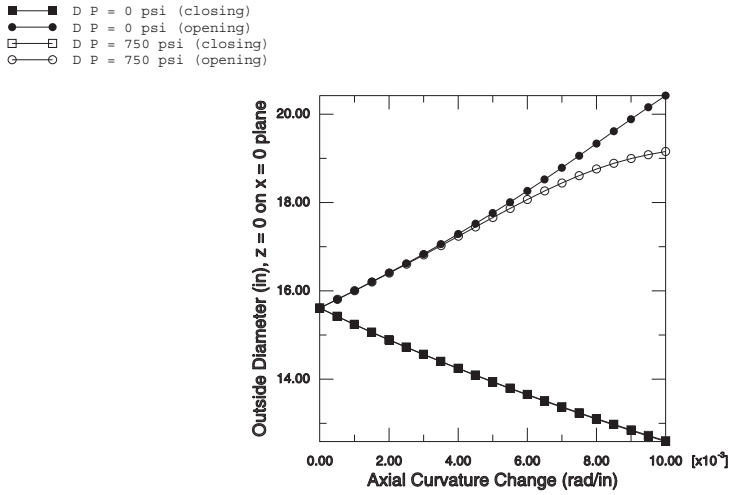


Figure 13: Deformation of section—initially curved, elastic-plastic pipe (shell model).

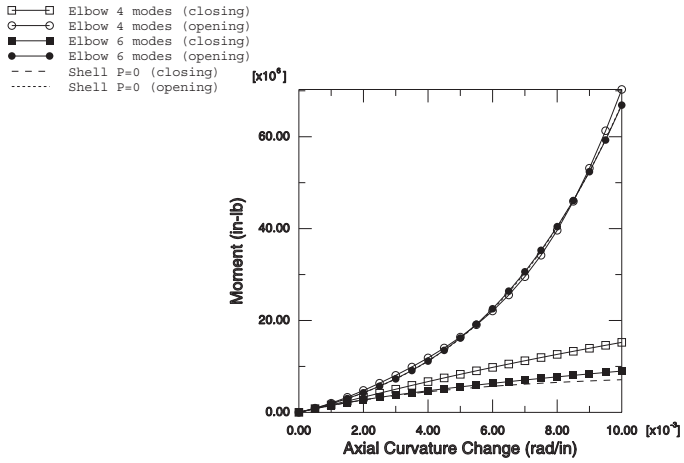


Figure 14: Moment-curvature—comparison of shell and elbow models, initially curved, elastic pipe.

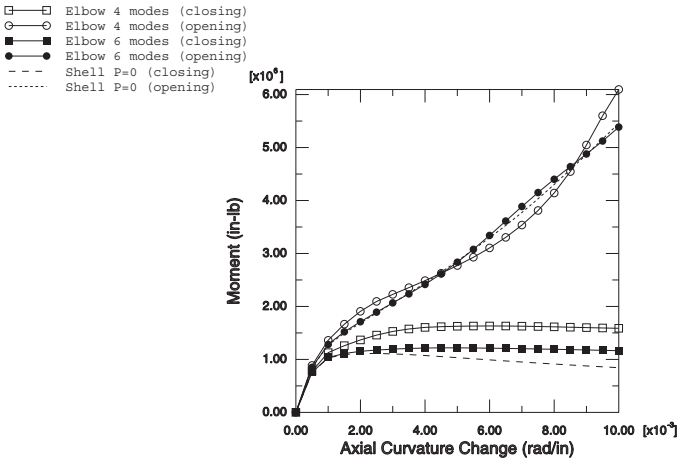


Figure 15: Moment-curvature—comparison of shell and elbow models, initially curved, elastic-plastic pipe.

Snap-through of a shallow, cylindrical roof under a point load

Products: Abaqus/Standard

Overview

This example illustrates the use of the modified Riks method to obtain the unstable static equilibrium response of an elastic shell structure that exhibits snap-through behavior.
The shell in this example is a shallow, cylindrical roof, pinned along its straight edges and loaded by a point load at its midpoint. Since the example has been studied by several authors, comparison with those published results provides verification of this type of analysis. An illustration of the volume proportional damping stabilization capability is also shown as an alternative to the Riks method.

Problem description

The dimensions of the roof are shown in *Figure 1*. The material is linear elastic, with a Young's modulus of 3.103 GPa and a Poisson's ratio of 0.3.

Modeling and solution control

The roof is assumed to deform in a symmetric manner, so one quadrant is discretized, as shown in *Figure 1*. We use two regular 6 × 6 meshes of shell elements, one of type S4R5 (4-node elements with one integration point) and one of type S4R (finite membrane strain shell element), and an 8 × 8 mesh of triangular shell elements of type S3R. In addition, two regular 6 × 6 meshes of continuum shells are provided, one of type SC6R (finite membrane strains, in-plane continuum shell wedge) and one of type SC8R (finite membrane strains, hexahedron continuum shell). No mesh convergence studies have been performed, but the comparison of the results given by these meshes with published numerical solutions suggests that, at least with respect to load-deflection behavior, these meshes give reasonably accurate results.

When using the modified Riks method, the load magnitude and suggested initial increment size should provide a reasonable estimate for the sense and magnitude of the first increment in load. It is known that the critical load for this case will not exceed 750 N. With an initial time step of .025 for a time period of 1.0, we give a load of 3000 N. This implies an initial load increment of about 75 N on the entire roof. Furthermore, we are not interested in postsnap behavior much beyond the magnitude of the critical load, so we terminate the analysis when a load proportionality factor of 0.06 has been reached. This corresponds to a total load on the entire roof of 720 N. In this problem the static equilibrium load actually reverses direction as the roof goes through an unstable snap. The modified Riks algorithm is able to track such load reversals. Gauss integration is used for the shell cross-section.

When using the volume proportional damping capability, a total load of 1332 N is applied, which is roughly equivalent to the load at which the Riks method analysis stops. The initial load increment is 10 percent of the total load. This algorithm does not capture load reversals; when such reversals would occur, the structure accelerates and the increased velocity produces enough viscous forces to balance the externally applied load. As a result, the external load stays almost constant during the unstable part of the deformation.

Results and discussion

Figure 2 shows the downward vertical displacement of the point under the load (the middle of the roof) and of the midpoint of the free edge of the shell as functions of the applied load on the entire roof. The roof collapses unstably at a load of about 600 N, with the equilibrium load falling rapidly to a value of about −380 N as the snap-through occurs. During the latter part of the snap-through the middle point of the roof moves upward slightly

(snaps back) from a displacement of about 16.8 mm to 14.1 mm just before the end of the snap-through. Following snap-through, the shell stiffens rapidly as the load increases, as would be expected. In the original, unloaded configuration, the centerline of the roof rises about 12.7 mm above the pinned edges. From *Figure 2* it can be seen that the instability occurs when the point being loaded has a downward displacement of about 14.4 mm, when it is just below the horizontal plane defined by the pinned straight edges. However, at this point of instability, the point in the middle of the free edge has only displaced downward by about 3 mm. At the end of the snap-through the point under the load has displaced about 16.3 mm, while the middle of the free edge has displaced about 26.3 mm. Thus, during the snap the point under the load moves a total distance of only about 2 mm, while the middle of the free edge moves 23.3 mm.

Several authors have investigated this same problem (see the references at the end of this example) and have obtained results that agree fairly closely with those obtained here. *Figure 3* shows a comparison of these various solutions for the variation of load with displacement of the point under the load.

Figure 4 shows a comparison of the Riks method and the automatic stabilization method (volume proportional damping) in terms of the downward vertical displacements of the point under the load as functions of the applied load. While the deformation is stable (that is, during the initial loading) and after the snap-through takes place, both curves are very similar, which means that the damping introduces negligible dissipation. However, during the snap-through the strain energy that the structure wants to relieve in going from one stable configuration to the next is dissipated through damping instead of through decreasing the load. The disadvantage of this method is that it produces an almost constant loading without giving information on how far from a static equilibrium state it is (that is, how severe the snap-through is). On the other hand, this method still works when instabilities are local, in which case the Riks method may fail.

Input files

roofsnapthrough_s3r.inp	S3R element model.
roofsnapthrough_s4.inp	S4 element model.
roofsnapthrough_s4r.inp	S4R element model.
roofsnapthrough_s4r5.inp	S4R5 element model.
roofsnapthrough_stri65.inp	STRI65 element model.
roofsnapthrough_sc6r.inp	SC6R element model.
roofsnapthrough_sc8r.inp	SC8R element model.
roofsnapthrough_stabilize.inp	Same model with automatic stabilization (volume proportional damping) instead of the Riks method, default damping.
roofsnapthrough_stabilizefactor.inp	Same model with automatic stabilization (volume proportional damping) instead of the Riks method, user-defined damping.
roofsnapthrough_stabilize_adap.inp	Same model with adaptive automatic stabilization (volume proportional damping) instead of the Riks method, default damping.
roofsnapthrough_postoutput.inp	Tests the **POST OUTPUT* capability for the model in roofsnapthrough_stabilize.inp.

References

- Crisfield, M. A., “A Fast Incremental/Iterative Solution Procedure that Handles 'Snap-Through',” *Computers and Structures*, vol. 13, pp. 55–62, 1981.
- 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.

- Sabir, A. B., and A. C. Lock, "The Application of Finite Elements to the Large Deflection, Geometrically Nonlinear Behavior of Cylindrical Shells," in *Variational Methods in Engineering*, edited by C. A. Brebbia and H. Tottenbam, Southampton U. Press, 1982.

Figures

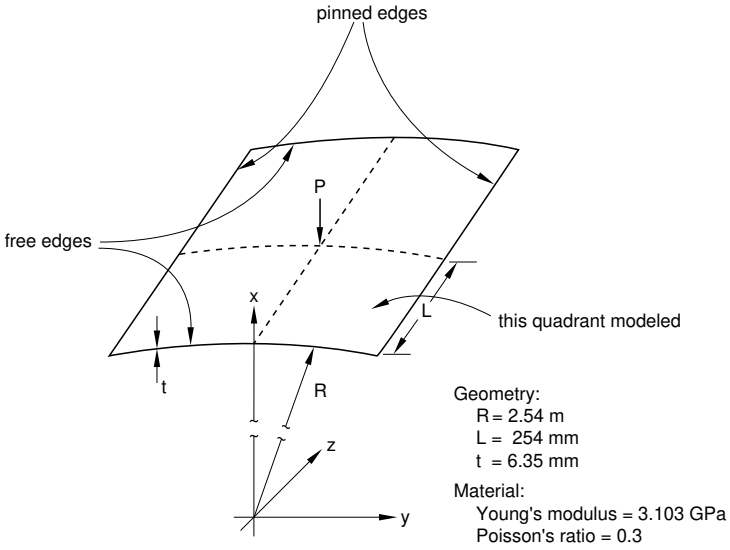


Figure 1: Shallow cylindrical roof under point load.

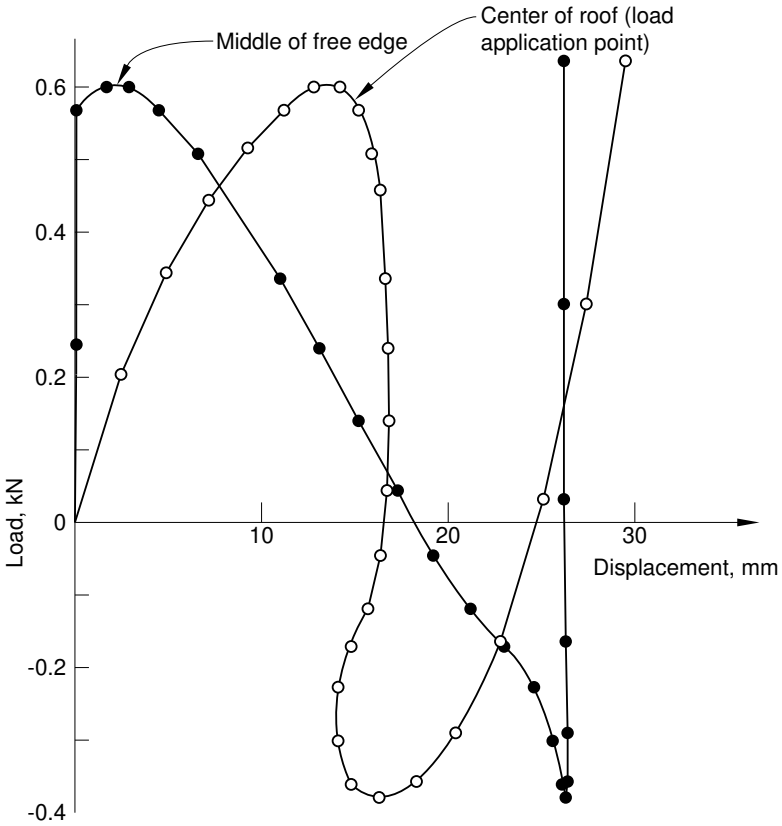


Figure 2: Load-displacement response for shallow cylindrical shell.

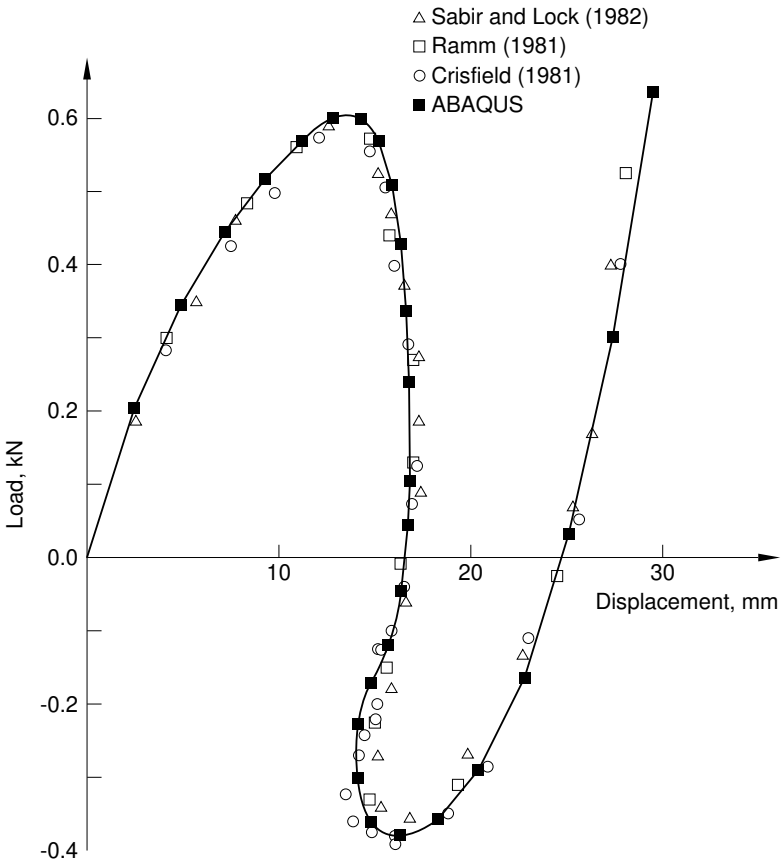


Figure 3: Comparison of solutions for shallow cylindrical roof.

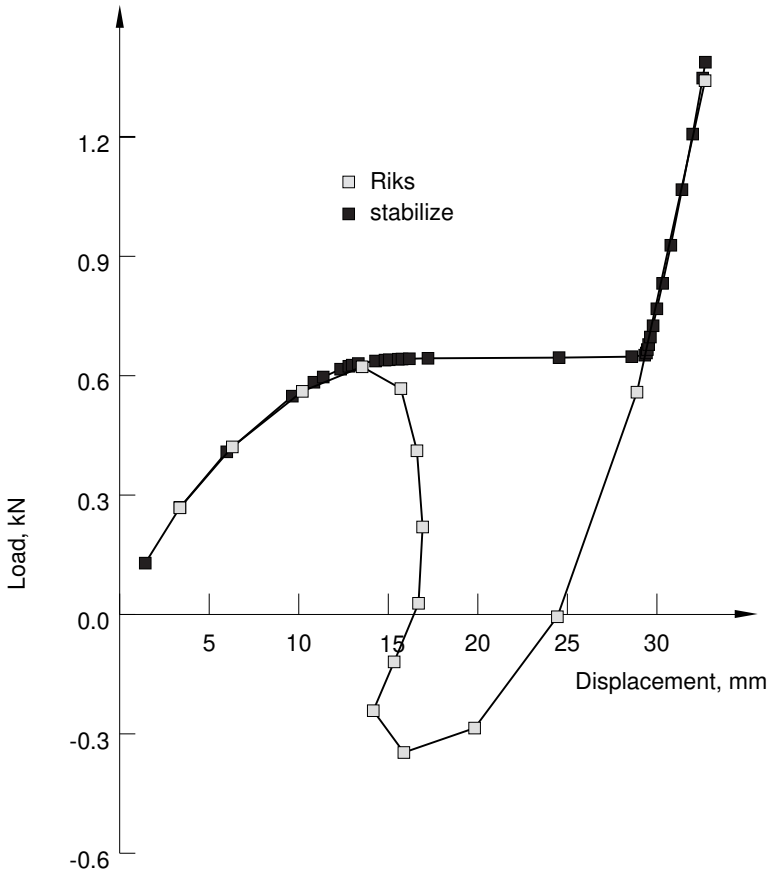


Figure 4: Comparison of Riks and stabilized solutions for shallow cylindrical roof.

Pressurized rubber disc

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example is an illustration of a rubber elasticity problem involving finite strains on a membrane-like structure. In this example a rubber disc, pinned around its outside edge, is subjected to pressure so that it bulges into a spherical shape. The published results of Oden (1972) and Hughes and Carnoy (1981) are used to verify the Abaqus quasi-static solution.

The example shows that Abaqus can solve this type of problem. The Abaqus/Standard results also demonstrate that, because of the treatment of the pinned-edge condition, the load stiffness matrix associated with the pressure loading is not symmetric at the outer edge of the pressurized face of the disc. It is found that, after a small amount of straining, these nonsymmetric terms must be included in the stiffness matrix for the solution to be numerically efficient.

Both a thick and a thin disc are tested. The solutions obtained using Abaqus/Explicit show dynamic effects when compared to the quasi-static solution found by Abaqus/Standard. The thin disc model in Abaqus/Explicit demonstrates the ability of Abaqus/Explicit to handle volume expansion of membrane-like structures; the application of fluid cavity elements in Abaqus/Explicit is also demonstrated.

Problem description

The radius of the thick disc analyzed in both Abaqus/Standard and Abaqus/Explicit is 190.5 mm (7.5 in), and its thickness is 12.7 mm (0.5 in). The thin disc analyzed in Abaqus/Explicit has the same radius and a thickness of 1.270 mm.

The mesh used for the Abaqus/Standard analysis is shown in *Figure 1*. The mesh uses 5 axisymmetric continuum elements (type CAX8H) along the radial direction and one element through the thickness. These are 8-node, second-order, mixed formulation elements. Other elements are also used in the Abaqus/Standard analysis, particularly the lower-order incompatible mode elements, which perform comparatively as well as the second-order elements. When the modified elements, CAX6MH and C3D10MH, are used for this problem, a greater refinement of the mesh is required to ensure good performance. These elements are not used in the validation against published results. Since the maximum extension is expected to be at the center of the disc, the length of the elements in the radial direction decreases from the circumference to the center so that the element that is adjacent to the centerline is nearly square. This element size gradient is obtained by defining a parabola between the two end nodes and placing the third point (which defines the parabola) at a position between one-quarter and one-half of the distance from the centerline to the other end of the line of nodes, thus weighting the nodal generation toward the centerline of the disc.

The problem is analyzed in two and three dimensions in Abaqus/Explicit, using different element types: continuum, shell, and membrane elements for the thick disc and shell and membrane elements for the thin disc. All cases use 10 elements in the radial direction and two elements through the thickness, twice as many as in the Abaqus/Standard analysis; hence, roughly the same number of degrees of freedom are used in both the dynamic solution and the quasi-static solution with a similar element grading in the radial direction.

No attempt has been made at a mesh convergence study. The agreement with published results (Oden, 1972, and Hughes and Carnoy, 1981) for the quasi-static case suggests that the mesh used is adequate to predict the overall response accurately.

The material is modeled as a Mooney-Rivlin material, with the constants (for the polynomial strain energy function) $C_{10} =$ 0.55 MPa (80 lb/in^2) and $C_{01} =$ 0.138 MPa (20 lb/in^2): these are the values used by Oden (1972) and Hughes and Carnoy (1981). In the Abaqus/Standard analysis it is an incompressible material. For

the Ogden strain energy function, the equivalent material constants used are $\mu_1 = 2C_{10}$, $\alpha_1 = 2$, $\mu_2 = 2C_{01}$, and $\alpha_2 = -2$. Abaqus/Explicit requires some compressibility for hyperelastic materials. In the input files used here, D_1 is not given. Hence, a default value of D_1 is chosen. This gives an initial bulk modulus ($K_0 = \frac{2}{D_1}$) that is 20 times higher than the initial shear modulus $2(C_{10} + C_{01})$. This ratio is much lower than the ratio exhibited by most rubberlike materials, but the results are not particularly sensitive to this value because the material is unconfined. Decreasing D_1 by an order of magnitude 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.

For the continuum element cases the pinned condition at the outside of the disc requires special treatment. In the axisymmetric cases the central node on that edge (node 31) is fixed in both directions. The edge is constrained to remain straight, while still being able to change length, and is free to rotate about the pinned node. For simplicity these constraints are imposed by requiring that the displacement of the node at the top of the outer edge (node 51) be equal and opposite to that of the node at the bottom of the edge (node 11). Two equations are required:

$$u_x^{51} + u_x^{11} = 0 \quad \text{and} \quad u_y^{51} + u_y^{11} = 0.$$

These constraints are imposed by using an equation constraint. The three-dimensional continuum case in Abaqus/Explicit (C3D8R) is treated in a similar manner by adding two more equations. Since only a wedge is actually modeled for the Abaqus/Standard three-dimensional analyses, the CYCLSYMMPC (*General Multi-Point Constraints*) is used to impose the appropriate constraints. No constraints are required for the shell element cases.

Loading and solution method

The loading consists of a uniform pressure applied to the bottom surface of the disc. The modified Riks method is used in Abaqus/Standard since the loading is proportional and because the solution may exhibit instability. A pressure magnitude of 1.38 MPa (200 lb/in^2) is specified: this magnitude is somewhat arbitrary since the Riks method is chosen. From other studies we expect that an initial pressure of about 0.014 MPa (2 lb/in^2) should take the disc a reasonable way into the nonlinear regime. Hence, an initial increment of 0.01 and a period of 1 are specified in the static analysis to achieve this level of pressure in the initial increment. (Since the Riks algorithm is used, the actual pressure magnitude at the end of the first increment will differ somewhat from the initial value of 0.014 MPa, depending on the extent of nonlinearity in that increment. See the descriptions of the Riks option in *Unstable Collapse and Postbuckling Analysis* and *Modified Riks algorithm* for more details.)

Since the surface to which the pressure is applied rotates and stretches, there is a stiffness contribution associated with the pressure (a “load stiffness matrix”). Because of the treatment of the pinned outer edge, the perimeter of the surface to which the pressure is applied is not fully constrained and, hence, gives rise to a nonsymmetric contribution in the local stiffness matrix (see Hibbitt, 1979). During that part of the solution where strains and rotations are not very large, it makes little difference to the number of iterations needed to solve the equilibrium equations if this nonsymmetric contribution is ignored. However, to continue the analysis beyond a pressure of about 0.07 MPa (10 lb/in^2)—when the displacement at the center of the disc is about half the radius—it is essential that these terms are included. This requires that the Abaqus/Standard analysis use the unsymmetric equation solver. In practical cases, if the unsymmetric equation solver is not used in the initial run, it can be introduced on a restarted run if necessary. An example using S4R elements with enhanced hourglass control is also included.

The effect of uniform tensile prestress in Abaqus/Standard is also investigated. The prestress is applied as equal radial and circumferential stresses. Prestress values of 0.35, 0.7, and 1.4 MPa (50, 100, and 200 lb/in^2) are investigated.

In the explicit dynamic analysis the pressure is ramped up over the duration of the step. The maximum applied pressure for the thick disc case is 0.317 MPa (46 psi) and is applied by using a distributed load or by prescribing the pressure directly to a fluid cavity reference node. In the fluid-driven case the fluid cavity is modeled using the surface-based fluid cavity capability (see *Fluid Cavity Definition*). The fluid cavity surface is defined underneath the disc so that the initial volume of the fluid cavity is zero. For both load cases the 0.317 MPa

pressure value was chosen based on the final value obtained in the quasi-static simulation via Abaqus/Standard utilizing the Riks method for incrementation control. The maximum pressure for the thin disc is 0.036 MPa (4.5 psi) and is prescribed at a fluid cavity reference node as in the thick disc case. The rate of loading was observed to affect the simulation for all cases in Abaqus/Explicit.

A thick disc example for the two-dimensional axisymmetric continuum case in Abaqus/Explicit illustrates controlling the duration of the analysis and forcing output when an extreme value criterion is reached. When nodal variables are monitored, the end of the analysis is specified to occur when the center of the plate has bulged out to twice its initial radius. Thickness strain is monitored in the bottom row of elements, and an output state is written when the strain falls below the specified value. Additional examples using S4R and M3D4R elements with enhanced hourglass control are included.

Results and discussion

Plots of the deformed shape of the disc at various stages in the Abaqus/Standard analysis are shown in *Figure 2*. A plot of the deformed shape of the thick disc at the end of the step for the two-dimensional Abaqus/Explicit axisymmetric continuum case is shown in *Figure 3*. This result was obtained using a load duration of 0.01 sec. In both analyses, at the end of the loading the center of the plate has bulged out to a position approximately twice the initial radius. At this point element 1 has deformed so much that it would be difficult to continue the analysis without rezoning, and the solution beyond this point is of little practical interest.

The thickness of the disc at the centerline is plotted against the z-displacement of the center of the disc for the Abaqus/Standard analysis in *Figure 4*. To produce a smoother curve, a slightly modified input file with smaller and more time increments was used. The slight bump at the right end of the curve suggests some localization in the plate slightly away from the center.

Figure 5 shows a plot of thinning strain at the center of the disc versus the normalized displacement of the centerline node of the disc for the Abaqus/Explicit analysis. The results in *Figure 5* are purely kinematic (the near incompressibility of the hyperelastic constitutive model dictates the thinning as a function of the membrane stretching) and agree with the results obtained with Abaqus/Standard.

A comparison between the Abaqus/Standard results and those obtained by Oden (1972) and Hughes and Carnoy (1981) is shown in *Figure 6*, where the applied pressure is plotted against the z-displacement at the center of the disc. All three solutions agree quite closely. Abaqus/Standard gives identical results for the Mooney-Rivlin and Ogden models with corresponding parameters.

Figure 7 shows a plot of pressure versus displacement of the centerline node of the disc for all the Abaqus/Explicit element cases considered here for a step duration of 0.01 sec. These results show significant dynamic effects compared to the quasi-static results obtained with Abaqus/Standard at the initial times. The early time response is dictated by the inertia of the disc—it simply takes some time to get the disc moving. This is manifested by the steep initial slope of the pressure versus displacement curves in *Figure 7*. During the early part of the response, the center part of the disc is moving as a rigid body until the effect of the pinned boundary condition causes the disc to begin to bulge. As the deformed shape evolves, the Abaqus/Explicit results in *Figure 7* are closer to the quasi-static results. The membrane and shell models using the ENHANCED hourglass control option produce the same solutions as the ones using the default hourglass control option.

Abaqus/Standard pressure-displacement curves for different values of initial tensile prestress in the rubber plate are also shown in *Figure 6*. As expected, the stiffening effect of the tensile prestress requires a higher pressure for the disc to displace a certain amount. Models using the hybrid CAXA elements produce the same axisymmetric solutions when axisymmetric boundary conditions are imposed.

The pressure-displacement curves for loading using the fluid cavity elements in Abaqus/Explicit are shown in *Figure 8*. The results approximately match those obtained using the distributed load curves shown in *Figure 7*. The pressure-displacement curve for the thin disc (load applied using fluid cavity elements) is shown in *Figure 9*. The results approximately match those obtained with an implicit dynamic analysis of these models in Abaqus/Standard.

The axisymmetric continuum case is reanalyzed in Abaqus/Explicit by increasing the duration of the load to 0.10 sec. This case demonstrates some of the inherent difficulties of trying to solve static problems with a dynamic simulation. Increasing the duration of the step by an order of magnitude should decrease the dynamic effects and give results that are closer to the quasi-static results obtained with Abaqus/Standard. *Figure 10*, which is a plot of pressure versus centerline displacement for this slower case, shows that there are still significant dynamic effects in the solution. Some of the early inertia-dominated "lag" in the solution has been eliminated, at the expense of exciting the response of the structure in the lowest structural mode. In the faster case (step duration of 0.01 sec) the loading rate was at a higher frequency than the frequency of the structural mode, and the disc is driven into the bulged shape faster than it can respond by vibration in a structural mode. In the slower case the loading is at a low enough frequency that the structure has time to respond and is vibrating about the static equilibrium configuration. The pressure versus displacement curve of *Figure 10* is oscillating about the curve defined by the quasi-static results.

Input files

Abaqus/Standard input files

Polynomial energy function:

rubberdisk_c3d8ih_poly.inp	C3D8IH elements.
rubberdisk_c3d10mh_poly.inp	C3D10MH elements.
rubberdisk_cax4ih_poly.inp	CAX4IH elements.
rubberdisk_cax6h_poly.inp	CAX6H elements.
rubberdisk_cax6mh_poly.inp	CAX6MH elements.
rubberdisk_cax8h_poly.inp	CAX8H elements.
rubberdisk_caxa8h1_poly.inp	CAXA8H1 elements.
rubberdisk_postoutput.inp	Data for postprocessing the restart file.
rubberdisk_max1_poly.inp	MAX1 elements.
rubberdisk_max2_poly.inp	MAX2 elements.
rubberdisk_mgax1_poly.inp	MGAX1 elements.
rubberdisk_s4r_poly.inp	S4R elements.
rubberdisk_sax1_poly.inp	SAX1 elements.
rubberdisk_saxa11_poly.inp	SAXA11 elements.

Ogden strain energy function:

rubberdisk_c3d8ih_ogden.inp	C3D8IH elements.
rubberdisk_c3d10mh_ogden.inp	C3D10MH elements.
rubberdisk_cax4ih_ogden.inp	CAX4IH elements.
rubberdisk_cax6h_ogden.inp	CAX6H elements.
rubberdisk_cax6mh_ogden.inp	CAX6MH elements.
rubberdisk_cax8h_ogden.inp	CAX8H elements.
rubberdisk_caxa8h1_ogden.inp	CAXA8H1 elements.

Tensile prestress:

rubberdisk_c3d8ih_prestress.inp	C3D8IH elements.
rubberdisk_c3d10mh_prestress.inp	C3D10MH elements.
rubberdisk_cax4ih_prestress.inp	CAX4IH elements.
rubberdisk_cax6h_prestress.inp	CAX6H elements.

rubberdisk_cax6mh_prestress.inp	CAX6MH elements.
rubberdisk_cax8h_prestress.inp	CAX8H elements.
rubberdisk_caxa8h1_prestress.inp	CAXA8H1 elements.
rubberdisk_s4r_ogden.inp	S4R elements.
rubberdisk_s4r_ogden_eh.inp	S4R elements with enhanced hourglass control.
rubberdisk_sax1_ogden.inp	SAX1 elements.
rubberdisk_saxa11_ogden.inp	SAXA11 elements.

The DIRECTIONS=YES parameter is used with the **EL FILE* option in the input file *rubberdisk_c3d8ih_poly.inp*.

Abaqus/Explicit input files

disccax4r.inp	Thick disc, CAX4R elements, with **DLOAD* loading.
discc3d8r.inp	Thick disc, C3D8R elements, with **DLOAD* loading.
discs4r.inp	Thick disc, S4R elements, with **DLOAD* loading.
discs4r_enh.inp	Thick disc, S4R elements, with **DLOAD* loading and enhanced hourglass control.
discsax1.inp	Thick disc, SAX1 elements, with **DLOAD* loading.
discm3d4r.inp	Thick disc, M3D4R elements, with **DLOAD* loading.
discm3d4r_enh.inp	Thick disc, M3D4R elements, with **DLOAD* loading and enhanced hourglass control.
discflcax4r_surfcav.inp	Thick disc, CAX4R elements, with fluid pressure loading. The surface-based fluid cavity capability is used to model the fluid cavity.
discflc3d8r_surfcav.inp	Thick disc, C3D8R elements, with fluid pressure loading. The surface-based fluid cavity capability is used to model the fluid cavity.
discflsax1_surfcav.inp	Thick disc, SAX1 elements, with fluid pressure loading. The surface-based fluid cavity capability is used to model the fluid cavity.
discfls4r_surfcav.inp	Thick disc, S4R elements, with fluid pressure loading. The surface-based fluid cavity capability is used to model the fluid cavity.
discthinflsax1_surfcav.inp	Thin disc, SAX1 elements, with fluid pressure loading. The surface-based fluid cavity capability is used to model the fluid cavity.
discthinfls4r_surfcav.inp	Thin disc, S4R elements, with fluid pressure loading. The surface-based fluid cavity capability is used to model the fluid cavity.
discflm3d4r_surfcav.inp	Thick disc, M3D4R elements, with fluid pressure loading. The surface-based fluid cavity capability is used to model the fluid cavity.
disccax4r_extreme.inp	Thick disc, CAX4R elements, with **DLOAD* loading and **EXTREME VALUE* criterion.
disccax4r_mr.inp	Thick disc, CAX4R elements, with Mooney-Rivlin strain energy potential.

References

- Hibbitt, H. D., “Some Follower Forces and Load Stiffness,” *International Journal for Numerical Methods in Engineering*, pp. 937–941, 1979.
- Hughes, T. J. R., and E. Carnoy, “Nonlinear Finite Element Shell Formulation Accounting for Large Membrane Strains,” *Nonlinear Finite Element Analysis of Plates and Shells, AMD*, vol. 48, pp. 193–208, 1981.
- Oden, J. T., Finite Elements of Nonlinear Continua, McGraw-Hill, 1972.

Figures

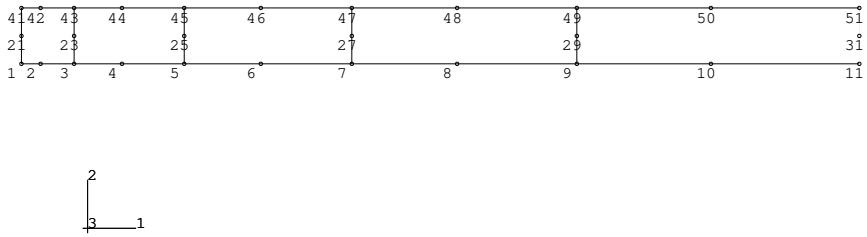


Figure 1: Mesh for pressurized rubber disk.

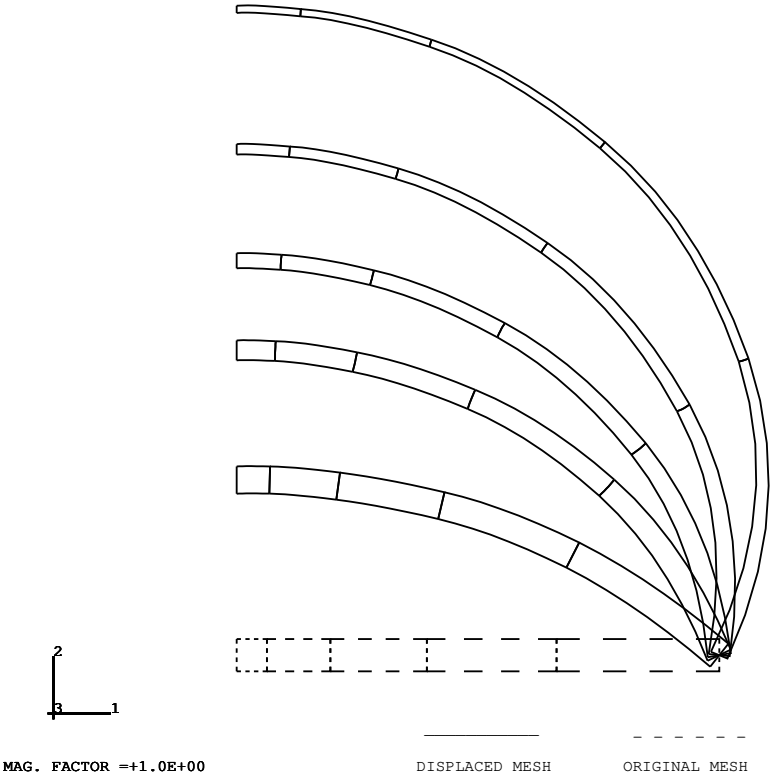


Figure 2: Displaced shapes of pressurized rubber disk, Abaqus/Standard analysis.

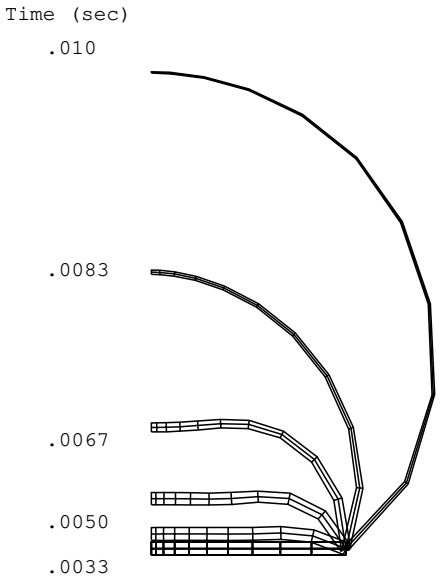


Figure 3: Displaced shapes for the axisymmetric continuum mesh, thick disc model, Abaqus/Explicit analysis.

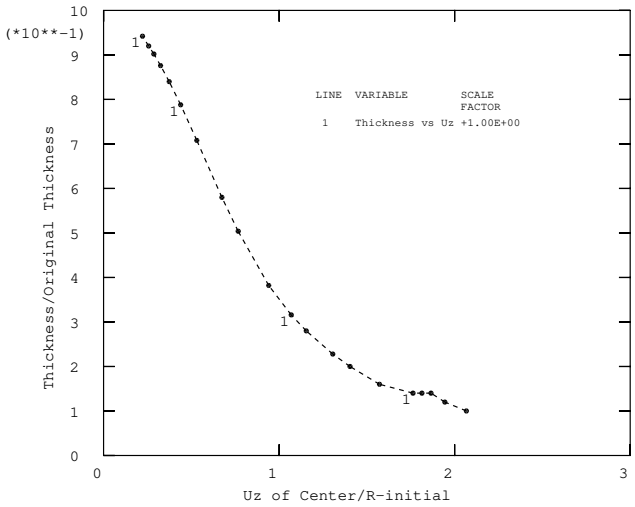


Figure 4: Central thickness versus central displacement, Abaqus/Standard analysis.

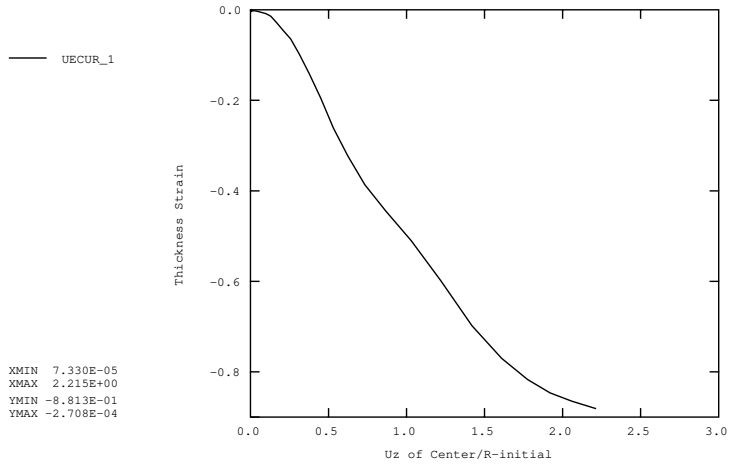


Figure 5: Thickness strain versus central displacement for the axisymmetric continuum mesh, thick disc model, Abaqus/Explicit analysis.

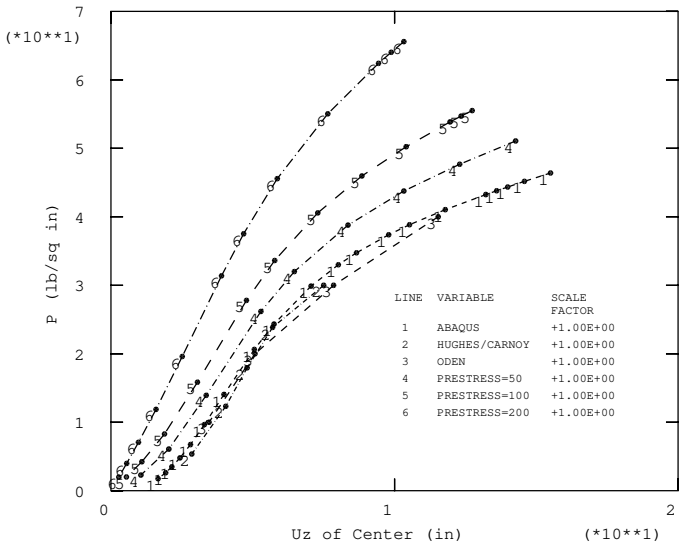


Figure 6: Comparison of pressure-deflection results, Abaqus/Standard analysis.

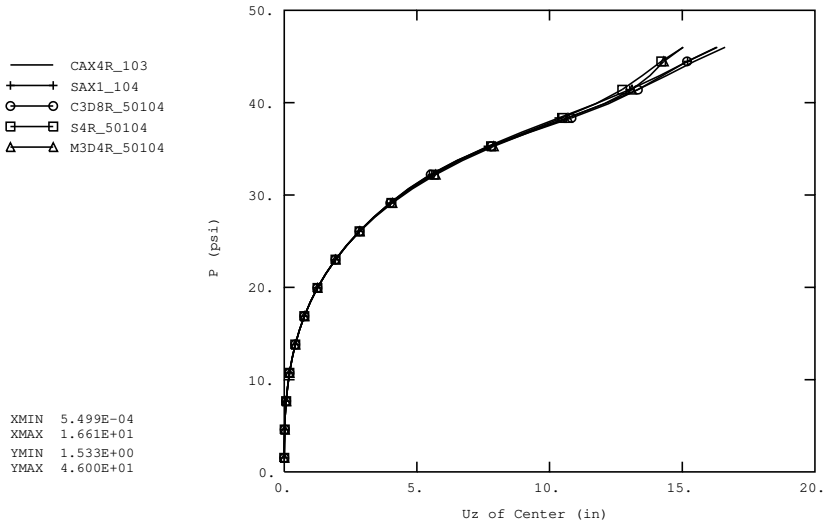


Figure 7: Pressure versus deflection results for load ramp duration of 0.01 sec, thick disc model with distributed loading, Abaqus/Explicit analysis.

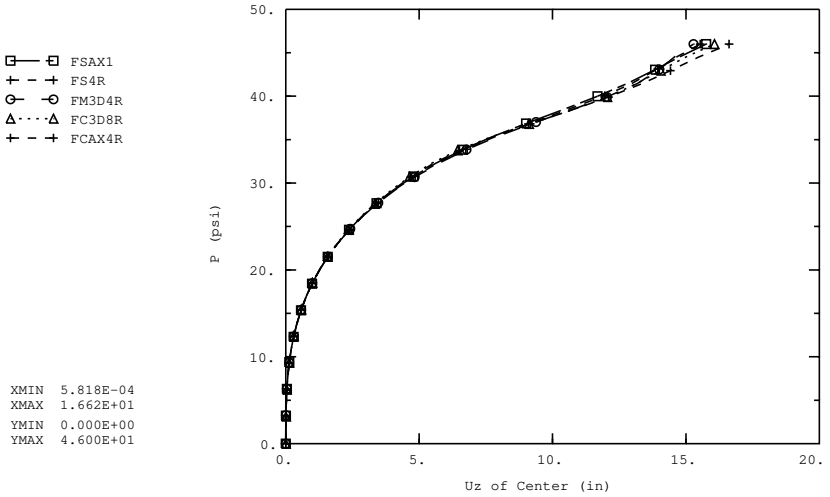


Figure 8: Pressure versus deflection results for load ramp duration of 0.01 sec, thick disc model with fluid pressure loading, Abaqus/Explicit analysis.

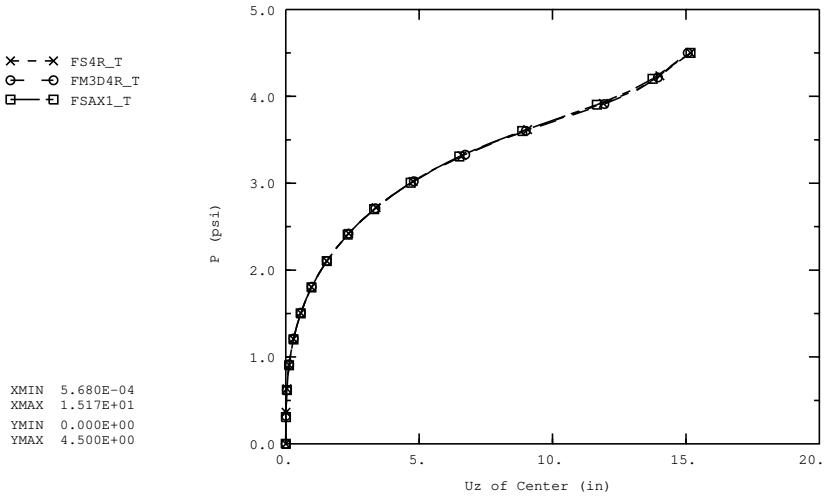


Figure 9: Pressure versus deflection results for load ramp duration of 0.01 sec, thin disc model with fluid pressure loading, Abaqus/Explicit analysis.

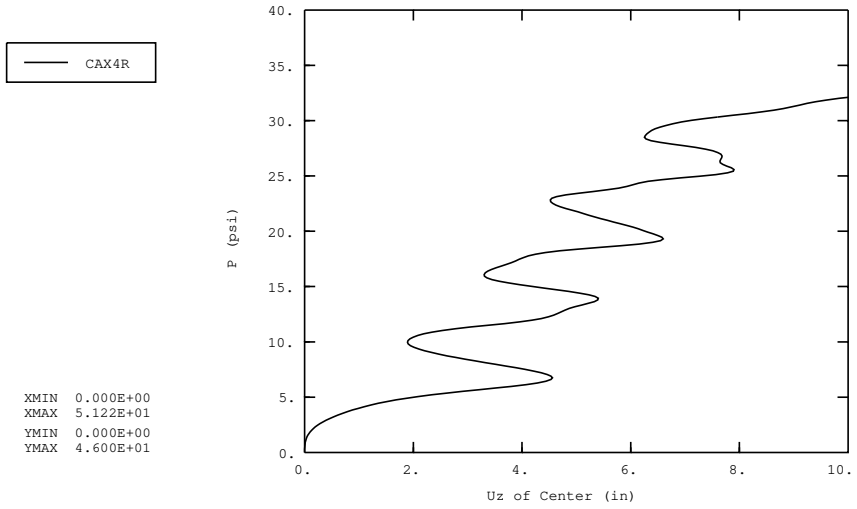


Figure 10: Pressure versus deflection results for load ramp duration of 0.10 sec, thick disc model with distributed loading, Abaqus/Explicit analysis.

Uniaxial stretching of an elastic sheet with a circular hole

Products: Abaqus/Standard

Overview

This example demonstrates the use and verifies the results of hyperelastic and viscoelastic materials in plane stress. This example considers the uniform large stretching of a thin, initially square sheet containing a centrally located circular hole. Plane stress conditions are assumed, and the results are compared with those provided in Oden (1972) for four different forms of the strain energy function using the experimental results of Treloar (1944).

Problem description

The geometry and the mesh for a quarter-sheet are shown in *Figure 1*. The undeformed square sheet is 2 mm (0.079 in) thick and is 165 mm (6.5 in) on each side. It has a centrally located internal hole of radius 6.35 mm (0.25 in). The body is modeled with 32 second-order plane stress reduced-integration elements (element type CPS8R). The incompressibility of the material requires the use of the "hybrid" elements for plane strain, axisymmetric, or three-dimensional cases; but in plane stress the thickness change is available as a free variable that can be used to enforce the constraint of constant volume (incompressibility), so this standard displacement formulation element (CPS8R) is appropriate. No mesh convergence studies have been performed, but the good agreement with the results given by Oden (1972) suggests that the model chosen has comparable accuracy with the model used by Oden.

Four different material models are used. The experimental data of Treloar (1944) composed of uniaxial, biaxial, and planar tension data are applied to these models. Two of the four models are forms of the standard polynomial hyperelasticity model in Abaqus. One is the classical Mooney-Rivlin strain energy function:

$$U = C_{10}\left(I_1 - 3\right) + C_{01}\left(I_2 - 3\right).$$

The other is due to Biderman:

$$U = C_{10}\left(I_1 - 3\right) + C_{01}\left(I_2 - 3\right) + C_{20}\left(I_1 - 3\right)^2 + C_{30}\left(I_1 - 3\right)^3.$$

In both cases the material is assumed to be incompressible. The constants used by Oden (1972) are C_{10}= 0.1863 MPa (27.02 psi); C_{01}= 0.00979 MPa (1.42 psi); and, for the Biderman model, C_{20}= –0.00186 MPa (–0.27 psi), and C_{30}= 0.0000451 MPa (0.00654 psi), with all other C_{IJ}= 0. For the Mooney-Rivlin material $N = 1$ is specified in the hyperelastic material definition (*Hyperelastic Behavior of Rubberlike Materials*), and only C_{10} and C_{01} are given. For the Biderman material $N = 3$, and nine constants must be given. Since the material is incompressible the constants D_i are set to zero.

The third material model is the Ogden hyperelasticity model in Abaqus:

$$U = \sum_{i=1}^{N} \frac{2\mu_i}{\alpha_i^2}\left(\overline{\lambda}_1^{\alpha_i} + \overline{\lambda}_2^{\alpha_i} + \overline{\lambda}_3^{\alpha_i} - 3\right) + \sum_{i=1}^{N} \frac{1}{D_i}\left(J_{el} - 1\right)^{2i}.$$

The Ogden hyperelastic parameters are obtained using test data in the hyperelastic material definition to fit the experimental data of Treloar. Three pairs of parameters μ_i and α_i are derived for $N = 3$.

The fourth material model is the Marlow hyperelasticity model in Abaqus. In this model the deviatoric part of the response is derived from one set of test data (uniaxial, biaxial, or planar) such that the material's behavior is represented exactly in the deformation mode for which test data are available. Three examples are provided in which the model is based on uniaxial, biaxial, or planar test data, respectively.

In addition, the Biderman model and the Marlow model are used in conjunction with the viscoelastic material model. The shear relaxation is defined by time-dependent moduli expanded in a Prony series with two terms:

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

with $\bar{g}_1^P = 0.25$, $\tau_1 = 5.0$ sec and $\bar{g}_2^P = 0.25$, $\tau_2 = 10$ sec. The bulk behavior is assumed to remain incompressible.

Loading and controls

The sheet is stretched to a width of 1181 mm (46.5 in)—over seven times its initial width—in the *x*-direction, while the edges parallel to the *x*-axis are restrained from stretching in the *y*-direction. The *y*-direction restraints are imposed directly with a boundary condition. The stretch in the *x*-direction is prescribed by imposing uniform normal displacement on the right-hand edge of the mesh. All the nodes on that edge are constrained to have the same *x*-displacement by using an equation constraint. The displacement of the retained node (node 1601) is then prescribed to stretch the sheet. This technique allows the total stretching force to be obtained directly as the reaction force at this node. The symmetry conditions at $x = 0$ and at $y = 0$ are also imposed with a boundary condition.

An initial increment of 5% of the final displacement is suggested. The size of subsequent increments is chosen by the automatic incrementation scheme.

In the viscoelastic case a second step is added, driven by the quasi-static procedure. The deformation is kept the same, and the stresses relax. The time period is 100 sec, which is much larger than the time constants of the material. As a result, the long-term behavior of the material should be obtained. Setting $t = \infty$ in the expression for the time-dependent moduli provides $G_R(\infty) = 0.5G_0$ and $K_R(\infty) = K_0$. Since the deformation is almost completely constrained during the relaxation step, we expect the stresses to be halved in this process. The maximum difference in the creep strain increment over a time increment is specified using a value of 0.1, which enables automatic incrementation. This value controls the error in the integration of the viscoelastic model by limiting the difference in the strain increments defined by forward Euler and backward Euler integrations. The value of 10% strain error per increment used here is very large and suggests that no attempt is being made to limit this source of error: rather, we are allowing the automatic time incrementation to reach the long-term (steady-state) solution as quickly as possible.

Results and discussion

The final displaced configuration for the case with the Biderman material model is shown in *Figure 2*; and the load responses are shown in *Figure 3*, where the load is plotted as a function of the overall nominal strain of the sheet in the *x*-direction. The results of the first three hyperelastic models are seen to agree quite closely with Oden's. The results of the Marlow hyperelastic model also agree well with Oden's, although they are not shown in *Figure 3*. The Mooney-Rivlin strain energy function (with C_{10} and C_{01} as the only nonzero terms) cannot predict the "locking" of the response at higher strains that is predicted by the Biderman and Ogden strain energy functions. *Figure 4* shows the load-time response for the case including the viscoelastic relaxation step.

Input files

CPS8R elements:

elasticsheet_cps8r_biderman.inp	Biderman material model. The Mooney-Rivlin model is obtained by modifying the **HYPERELASTIC* option to give $N = 1$ and providing only the first two constants on the data line.
elasticsheet_cps8r_ogdendata.inp	Ogden hyperelasticity formulation with the TEST DATA INPUT option.

elasticsheet_cps8r_bidervisco.inp	Viscoelastic Biderman material model including the relaxation step.
elasticsheet_bidervisco_stabil.inp	Viscoelastic Biderman material model including the relaxation step and automatic stabilization.
elasticsheet_bidervisco_stabil_adap.inp	Viscoelastic Biderman material model including the relaxation step and adaptive automatic stabilization.
elasticsheet_postoutput.inp	Data used to postprocess the results file from elasticsheet_cps8r_biderman.inp.
elasticsheet_cps8r_marlowu.inp	Marlow material model using uniaxial test data.
elasticsheet_cps8r_marlowb.inp	Marlow material model using biaxial test data.
elasticsheet_cps8r_marlowp.inp	Marlow material model using planar test data.
elasticsheet_cps8r_marlowuvisco.inp	Viscoelastic Marlow material model using uniaxial test data and including the relaxation step.
elasticsheet_cps8r_marlowbvisco.inp	Viscoelastic Marlow material model using biaxial test data and including the relaxation step.
elasticsheet_cps8r_marlowpvisco.inp	Viscoelastic Marlow material model using planar test data and including the relaxation step.

CPS4 elements:

elasticsheet_cps4_biderman.inp	Biderman material model.
elasticsheet_cps4_ogdendata.inp	Ogden hyperelasticity formulation with the TEST DATA INPUT option.
elasticsheet_cps4_bidervisco.inp	Viscoelastic Biderman material model including the relaxation step.
elasticsheet_cps4_marlowu.inp	Marlow material model using uniaxial test data.
elasticsheet_cps4_marlowuvisco.inp	Viscoelastic Marlow material model using uniaxial test data and including the relaxation step.

References

- Oden, J. T., *Finite Elements of Nonlinear Continua,* McGraw-Hill, New York, 1972.
- Treloar, L. R. G., “Stress-Strain Data for Vulcanised Rubber Under Various Types of Deformation,” Trans. Faraday Soc., 40, pp. 59–70, 1944.

Figures

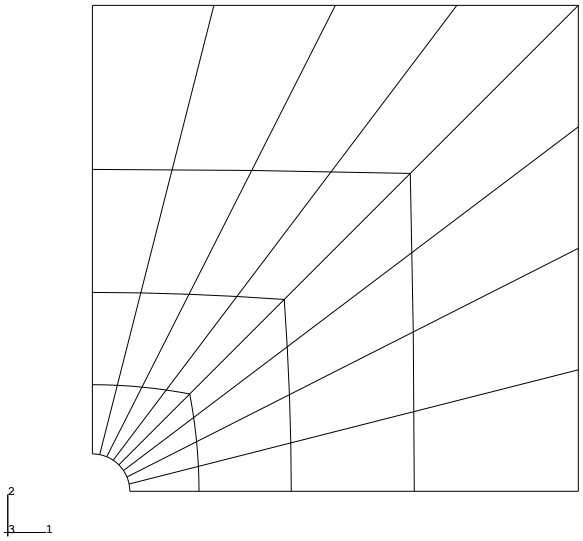


Figure 1: Rubber sheet and mesh.

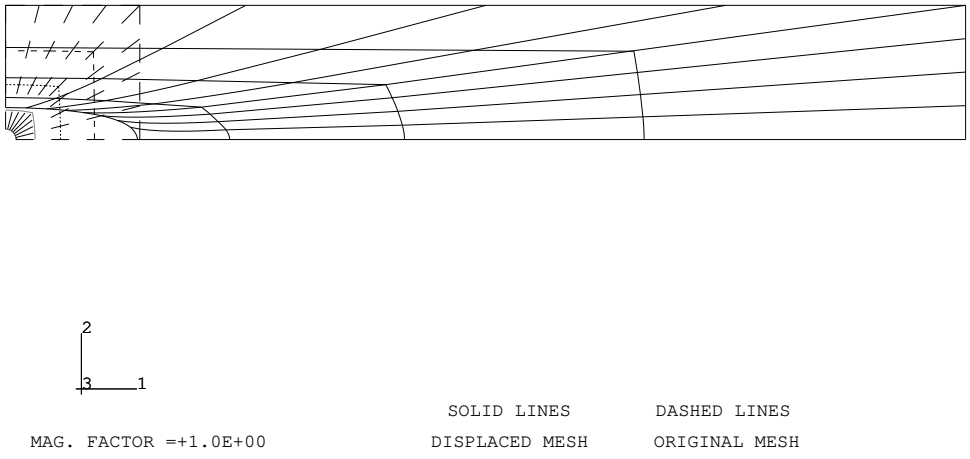


Figure 2: Final displaced configuration, Biderman model.

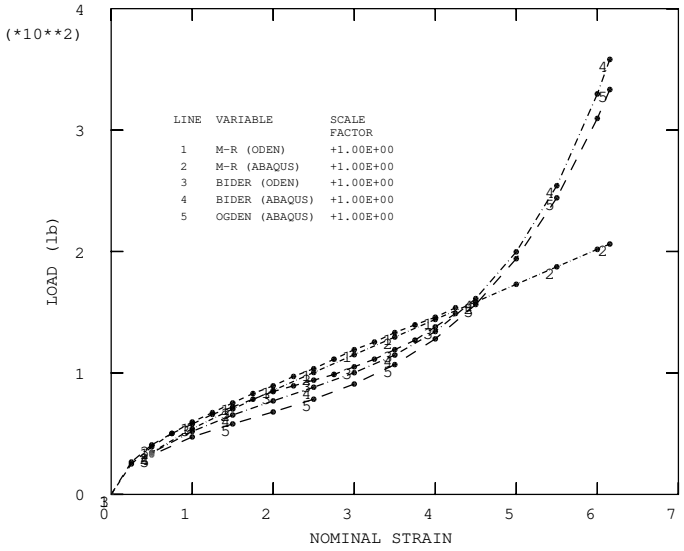


Figure 3: Applied force versus overall nominal strain.

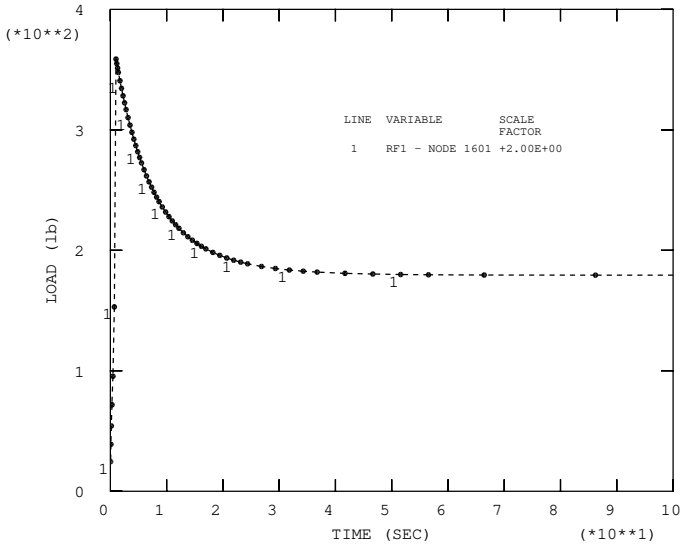


Figure 4: Load versus time, Biderman model, with a relaxation period of 100 secs.

Necking of a round tensile bar

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example illustrates necking and softening of a round tensile bar.
This example has been studied by Needleman (1972), Tvergaard and Needleman (1984), Needleman and Tvergaard (1985), and Aravas (1987). The material is assumed to be a rate-independent metal in which triaxial tension stress can allow voids to nucleate and grow. The example illustrates the use of the porous metal plasticity model with void nucleation. In Abaqus/Explicit a porous failure criteria is used to model failure of the material after a critical void volume fraction is reached.

Problem description

We consider a long specimen with a circular cross-section. The specimen has an initial length of $2l_0$ and a radius of R_0, with $l_0/R_0 = 4$. R_0 is assumed to be equal to 1 unit. Only a quarter of the specimen needs to be analyzed because of the symmetry about the $r = 0$ and $z = 0$ axes. *Figure 1* shows the mesh used in the analysis. Both the geometry and the deformation are assumed to be axisymmetric. Axisymmetric elements are used, and the mesh is refined near the center of the specimen because of the expected softening and intense deformation in that region. An initial geometric imperfection is used to induce necking in the specimen analyzed with Abaqus/Standard. In Abaqus/Explicit the imperfection is not needed because stress wave effects induce necking at the center of the bar.

Material

The material properties used in the computation are:

Young's modulus, E:	300
Poisson's ratio, ν:	0.3
Porous material parameters:	q_1 = 1.5, q_2 = 1.0, and q_3 = 2.25
Initial relative density:	1.0 (f_0 = 0.0)
Void nucleation parameters:	ε_N = 0.3, s_N = 0.1, and f_N = 0.04
Porous failure criteria:	f_F = 0.6, f_c = 0.59 (for Abaqus/Explicit only)

The work hardening behavior (yield stress, σ_y, versus equivalent plastic strain, $\bar{\varepsilon}_m^{pl}$) given for the metal plasticity is of the form

$$\frac{\sigma_y}{\sigma_0} = \left(\frac{\sigma_y}{\sigma_0} + \frac{3G}{\sigma_0}\bar{\varepsilon}_m^{pl}\right)^N,$$

where σ_0 = 1 is the initial yield stress, N = 0.1 is the hardening parameter, and G is the elastic shear modulus. Necking is expected to start when the yield stress approaches the work hardening rate, which occurs at a strain of about 10 to 12%. Hence, the work hardening behavior is described more accurately for $0.08 \leq \bar{\varepsilon}_m^{pl} \leq 0.3$ than for the rest of the curve.

The parameters q_1, q_2, and q_3 were introduced by Tvergaard (1981) to make the predictions of the Gurson model agree with numerical studies of an elastic-plastic medium containing a periodic array of voids. The parameter values used in this analysis are those chosen by Tvergaard.

The void nucleation parameters used in the material description are the same as those given by Tvergaard and Needleman (1984) and Needleman and Tvergaard (1985). These parameters describe the normal distribution of the nucleation strain (see *Porous Metal Plasticity*). The area under the normal distribution curve represents the total volume fraction of the nucleated voids and is approximately equal to f_N. With the normal distribution, the amount of voids nucleated between $\overline{\varepsilon_m}^{pl} = \varepsilon_N - s_N = 0.2$ and $\overline{\varepsilon_m}^{pl} = \varepsilon_N + s_N = 0.4$ is about 68% of f_N.

Boundary conditions and loading

The kinematic boundary conditions are symmetry about $r = 0$ (all nodes along $r = 0$ have $u_r = 0$ prescribed) and symmetry about $z = 0$ (all nodes along $z = 0$ have $u_z = 0$ prescribed). All the nodes on the top of the specimen along $z = 4.0$, in the node set `TOPSIDE`, are pulled in the z-direction while being held fixed in the r-direction. In the Abaqus/Explicit analysis the nodes in node set `TOPSIDE` are pulled with a prescribed velocity that increases linearly from 0 to 30 at 0.025 s and then decreases linearly from 30 to 0 at 0.05 s; in the Abaqus/Standard analysis the displacement is applied directly to obtain the deformations desired in the two analysis steps described below.

In the Abaqus/Standard analysis the accuracy of the implicit integration of the void nucleation and growth equation is controlled by prescribing a maximum allowable time increment in the automatic time incrementation scheme.

Results and discussion

The example problem focuses on the neck development, which is a precursor to failure in the form of "cup-cone fracture." The formation of the neck results in a triaxial state of stress at the center of the specimen, which accelerates the growth of the nucleated voids. A detailed analysis of the cup-cone fracture can be found in Tvergaard and Needleman (1984), which predicts that void nucleation is followed by the formation of a planar crack at the center of the neck as a result of the coalescence of voids. The planar crack propagates along a zig-zag path closer to the traction-free surface, eventually leading to the formation of the well-known cup-cone fracture.

Abaqus/Standard results

The calculations in the first step are terminated at an overall nominal strain $\varepsilon = u/l_0 = 19\%$, thereby making it possible to compare the results with those of Aravas (1987). In the second step the calculations are carried on further to an overall nominal strain of 19.75%.

The results of the analysis are illustrated in *Figure 2* to *Figure 6*. *Figure 2* shows the computed force as a function of the overall nominal strain. The maximum load is reached at an overall nominal strain of about 10.2%. The nominal stress–nominal strain curve, as well as the contour plots of void volume fraction (*Figure 3*) and hydrostatic pressure (*Figure 4*) at an overall nominal strain of 19%, match well with the results obtained by Aravas (1987).

Figure 5 and *Figure 6* show the contour plots of void volume fraction and hydrostatic pressure at an overall nominal strain of 19.75%; a comparison with *Figure 3* and *Figure 4* reveals a significant growth of voids and a corresponding decrease of the hydrostatic tension in the neck region, indicative of the material softening that has taken place.

Abaqus/Explicit results

The results of the analysis are illustrated in *Figure 2* and *Figure 7* through *Figure 10*. *Figure 2* shows the computed nominal stress as a function of the nominal strain. The maximum load is reached at a nominal strain of about 9%, after which the specimen softens due to coalescence of voids and eventually fractures across the

neck region. Due to the relatively high speed of the loading in the Abaqus/Explicit analysis, the void growth and coalescence and the failure propagation are coupled with dynamic effects. These dynamic effects are the source of the small differences observed in the results obtained with Abaqus/Explicit and Abaqus/Standard. *Figure 7* and *Figure 8* show total void volume fraction and pressure stress contours at $\varepsilon = 22.88\%$ ($t = 0.0317$ s). *Figure 9* shows the broken tensile specimen (at $t = 0.05$ s), where only the elements whose void volume fraction is still below the ultimate failure ratio are shown. The deformed mesh is shown next to the initial mesh. *Figure 10* shows contours of pressure in the broken bar. As was mentioned earlier, the tensile bar typically fails in a "cup-cone fracture"; because a symmetric solution was assumed in this model, a proper cup-cone fracture cannot develop in this case.

Input files

Abaqus/Standard input files

neckingtensilebar_cax4r.inp	CAX4R elements.
neckingtensilebar_cax6.inp	CAX6 elements.
neckingtensilebar_cax6m.inp	CAX6M elements.
neckingtensilebar_cax8r.inp	CAX8R elements.

Abaqus/Explicit input files

neck.inp	Abaqus/Explicit analysis.
neck_ale.inp	Model using the **ADAPTIVE MESH* option.
neck_ef1.inp	External file referenced in the adaptive mesh input file.
neck_ef2.inp	External file referenced in the adaptive mesh input file.

References

- Aravas, N., "On the Numerical Integration of a Class of Pressure-Dependent Plasticity Models," *International Journal for Numerical Methods in Engineering*, vol. 24, pp. 1395–1416, 1987.
- Needleman, A., "A Numerical Study of Necking in Circular Cylindrical Bars," *Journal of the Mechanics and Physics of Solids*, vol. 20, pp. 111–127, 1972.
- Needleman, A., and V. Tvergaard, "Material Strain-Rate Sensitivity in the Round Tensile Bar," *Brown University Report, Division of Engineering*, 1985.
- Tvergaard, V., "Influence of Voids on Shear Band Instabilities under Plane Strain Conditions," *International Journal of Fracture*, vol. 17, pp. 389–406, 1981.
- Tvergaard, V., and A. Needleman, "Analysis of the Cup-Cone Fracture in a Round Tensile Bar," *Acta Metallurgica*, vol. 32, pp. 157–169, 1984.

Figures

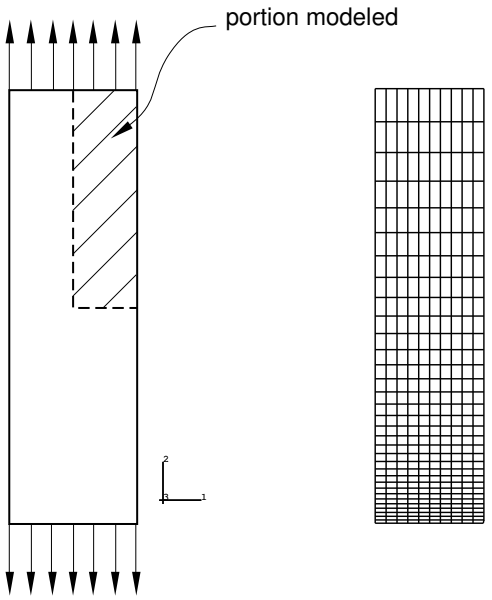


Figure 1: Geometry and mesh for the round tensile bar.

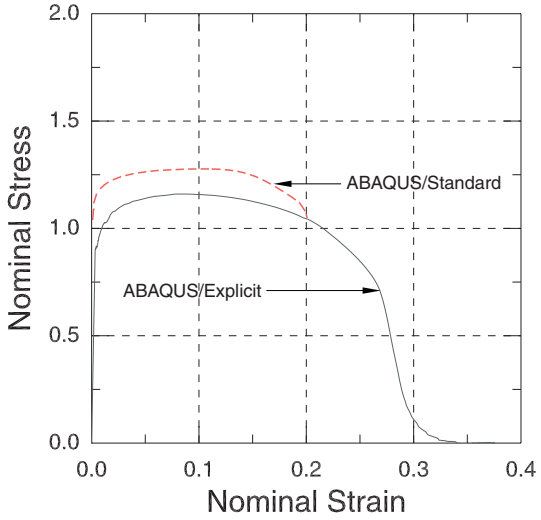


Figure 2: Overall nominal stress, $\sigma_N = F/\left(\pi R_0^2\right)$, vs. overall nominal strain, $\varepsilon = u/l_0$.

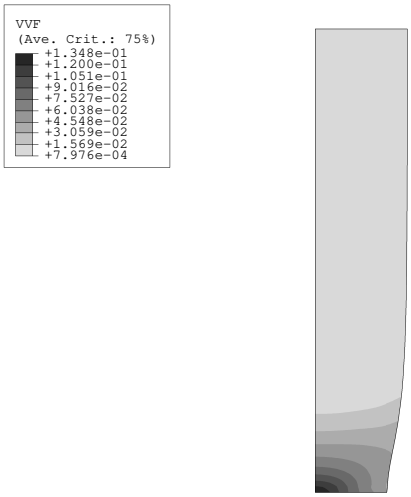


Figure 3: Void volume fraction at $\varepsilon =$ 19% (Abaqus/Standard).

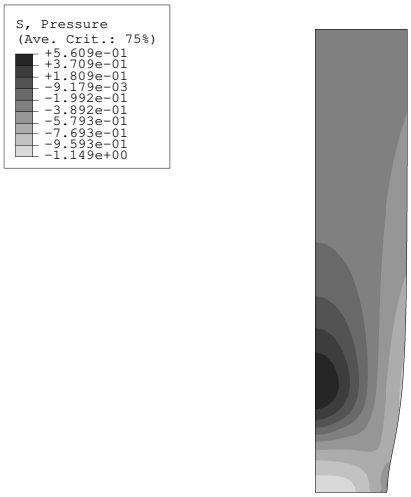


Figure 4: Hydrostatic pressure at $\varepsilon =$ 19% (Abaqus/Standard).

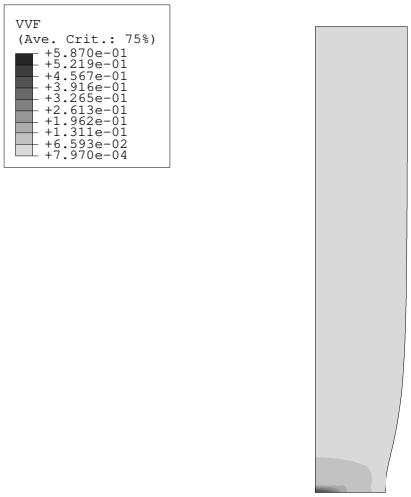


Figure 5: Void volume fraction at $\varepsilon =$ 19.75% (Abaqus/Standard).

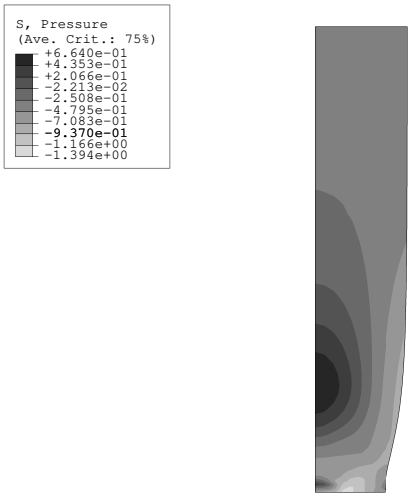


Figure 6: Hydrostatic pressure at $\varepsilon =$ 19.75% (Abaqus/Standard).

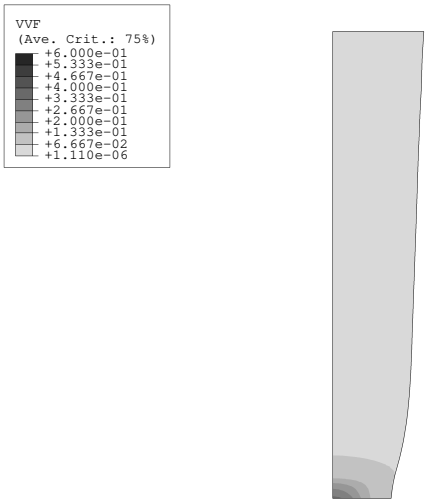


Figure 7: Total void volume fraction at $\varepsilon =$ 22.88% (Abaqus/Explicit).

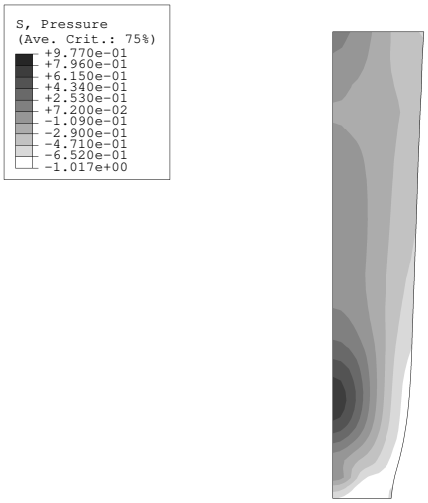


Figure 8: Hydrostatic pressure at $\varepsilon =$ 22.88% (Abaqus/Explicit).

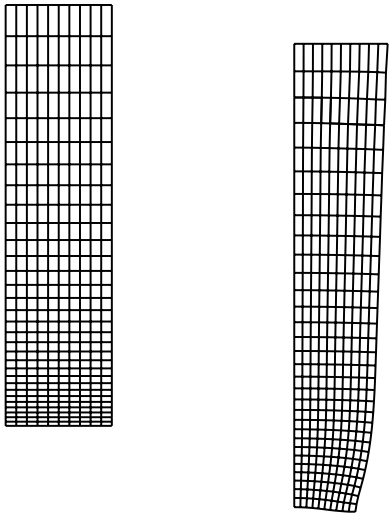


Figure 9: Final broken bar and its initial configuration (Abaqus/Explicit).

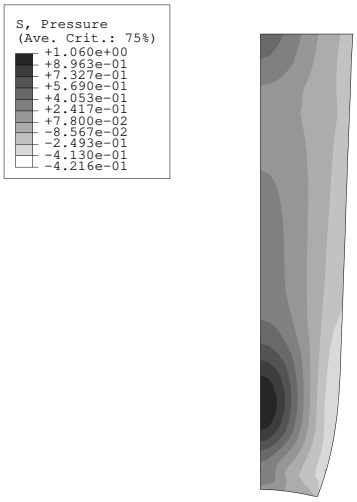


Figure 10: Hydrostatic pressure in the broken bar (Abaqus/Explicit).

Concrete slump test

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example illustrates the use of the extended Drucker-Prager plasticity model in Abaqus for a problem involving finite deformation.
Abaqus provides three different yield criteria of the Drucker-Prager class. In all three the yield function is dependent on both the confining pressure and the deviatoric stress in the material. The simplest is a straight line in the meridional (p–q) plane. The other yield criteria are a hyperbolic surface and a general exponential surface in the meridional plane. *Extended Drucker-Prager Models* describes these yield criteria in detail.

In this example the effects of different material parameters for the linear Drucker-Prager model are examined by simulating a concrete slump test. The other two Drucker-Prager yield criteria are verified by using parameters that reduce them to equivalent linear forms.

The slump test is a standardized procedure performed on fresh, wet concrete to determine its consistency and ability to flow. The test consists of filling a conical mold with concrete to a specified height. The mold is then removed, and the concrete is allowed to deform under its own weight. The reduction in height of the concrete cone, referred to as the "slump," is an indication of the consistency and strength of the concrete. This example is a simulation of such a test. A finite element analysis of this problem has been published by Famiglietti and Prevost (1994).

Problem description

No specific system of units is used in this example for the dimensions, the material parameters, or the loads. The units are assumed to be consistent. A standard, conical mold is used when performing a slump test on concrete. The cone is 0.3 units high. The radius at the base of the cone is 0.1, and the radius at the top is 0.05. An axisymmetric model is used to analyze the response of the concrete. The mesh used in the example is shown in *Figure 1*. First-order CAX4 elements are used for the Abaqus/Standard models, and first-order CAX4R elements are used for the Abaqus/Explicit models. We also include a three-dimensional model in Abaqus/Standard using two cylindrical elements spanning a 180° segment. No mesh convergence studies have been performed.

Material parameters

The material properties reported by Famiglietti and Prevost are used in this example.

A Young's modulus of 2.25 and a Poisson's ratio of 0.125 define the elastic response of the concrete. A density of $\rho =$ 0.1 is used.

It is assumed that the inelastic behavior is governed by the cohesion or shear strength and by the friction angle of the material. A cohesion of $c =$ 0.0011547 is used, and the responses at four different friction angles ($\phi =$ 0°, 5°, 20°, and 35°) are compared. Perfect plasticity is assumed. Since these parameters are provided for a Mohr-Coulomb plasticity model, they must be converted to linear Drucker-Prager parameters. *Extended Drucker-Prager Models* describes a method for converting Mohr-Coulomb parameters to equivalent linear Drucker-Prager parameters. Plane strain deformation and an associated plastic flow rule, where the dilation angle ψ is equal to the material friction angle β, are assumed for the purpose of this conversion. The corresponding linear Drucker-Prager parameters, β and d, are given in *Table 1*. The values are obtained using the expressions given in *Extended Drucker-Prager Models*.

Reducing the hyperbolic yield function into a linear form requires that $p_t|_0 = d/\tan\beta$. Reducing the exponent yield function into a linear form requires that $b = 1.0$ and that $a = (\tan\beta)^{-1}$. The material parameters for the exponential and hyperbolic yield criteria that create equivalent linear models are given in *Table 1*. Neither the hyperbolic nor the exponential yield criteria can be reduced to a linear model where $\beta = 0°$ (Mises yield surface).

The hyperbolic and exponential yield criteria both use a hyperbolic flow potential in the meridional stress plane. This flow potential, which is continuous and smooth, ensures that the flow direction is well-defined. The function asymptotically approaches the straight-line Drucker-Prager flow potential at high confining pressure stress but intersects the hydrostatic pressure axis at an angle of 90°. This function is, therefore, preferred as a flow potential for the Drucker-Prager model over the straight-line potential, which has a vertex on the hydrostatic pressure axis.

To match the hyperbolic flow potential as closely as possible to the straight-line Drucker-Prager flow potential, the parameter ϵ must be set to a small value. The default value for the exponent model, $\epsilon = 0.1$, is assumed in this example. This value ensures that the results obtained with this model will not deviate substantially from an equivalent straight-line flow potential, except for a small region in the meridional plane around the triaxial extension point. The size of this region diminishes as ϵ decreases. This parameter rarely needs to be modified for problems where a linear flow potential is desired for modeling the inelastic deformation. Reducing ϵ to a smaller value may cause convergence problems.

The inelastic material properties are specified using the extended Drucker-Prager plasticity model with hardening.

Loading

The loading is a gravity load, $g = 0.666$, applied to the entire model. In Abaqus/Standard the load is increased linearly from zero at the beginning of the step to its maximum value at the end of the step. In Abaqus/Explicit the load is ramped up using a smooth step amplitude definition. This amplitude definition provides a smooth loading rate, which is desirable in quasi-static or steady-state simulations.

The base of the concrete cone is held fixed in the vertical (2) direction but is free to move in the radial (1) direction. Thus, friction between the concrete and the support is not considered in this example.

This step accounts for finite strains and large displacements.

Solution controls in Abaqus/Standard

The models with the hyperbolic and exponential yield criteria use the default values for solution controls. However, for the linear Drucker-Prager model field equation tolerances are used to override the automatic calculation of the average forces to decrease the computational time required for the analysis. The convergence criteria is set to 1%, and the average force is set to 5.0×10^{-5}. The convergence check for the maximum allowable correction in displacement during an increment is also disabled. In addition, the time incrementation parameters are set automatically for this model to avoid premature cutbacks of the automatic time incrementation scheme. This is done because the linear flow potential used with this model creates a discontinuity in the solution when a material point reaches the vertex of the yield surface on the hydrostatic pressure axis. The error introduced in the solution by these relaxed tolerances is not large but results in a substantial reduction in computational time.

The maximum time increment is limited in the models such that no more than 2.0% of the total load is applied in any given increment. This is done so that the point of initial yield and the shape of the inelastic response are captured accurately during the analyses (see *Figure 4* and *Figure 5*).

The unsymmetric solver is activated for the exponential and hyperbolic yield models. This is needed because the hyperbolic flow potential used with the linear yield criteria causes nonassociated inelastic flow that results in an unsymmetric system of equations.

Results and discussion

Figure 2 shows the deformed shape and contours of the plastic strain in the vertical direction, PE22, for the linear Drucker-Prager model with $\beta = 0°$. *Figure 3* shows a similar plot for the linear Drucker-Prager model with $\beta = 30.16°$. The difference in the inelastic response seen in these figures can be attributed to two effects. First, the self-weight of the structure causes hydrostatic pressure stresses throughout most of the specimen, except for a thin layer at the outside surface of the cone where there are hydrostatic tensile stresses. The equivalent Mises stress, q, at which inelastic deformation occurs (the elastic extent) increases with increasing friction angle and pressure stress. This mechanism is illustrated in *Figure 4* for the two limit cases ($\beta = 0°$ and $\beta = 43.32°$) considered in this example. The figure shows the stress history in the meridional stress plane (equivalent pressure stress versus equivalent shear stress) for a material point located in the center of the cone near the base. Second, associated flow is assumed, so shearing is accompanied by dilation. Because of the confined nature of the geometry, an increase in volume strain is accompanied by an increase in pressure stress, further adding to the strength of the material. The second mechanism can easily be verified by performing nondilatant, $\psi = 0°$, tests that will show larger slumps.

The response at different friction angles is also illustrated in *Figure 5*. The dimensionless slump parameter is the displacement of the center of the top surface of the concrete divided by the initial height, h_0. The yield fraction is the ratio of the Drucker-Prager cohesion parameter, d, to the portion of applied load, $\rho g h_0$. Typical dimensionless slumps for actual concrete, as reported by Christensen (1991), can range from 0.2 to 0.8. *Figure 6* compares the results of slump tests on two different concrete mixtures, normal and light, to computational results obtained with friction angles of 0° and 30.16°. The experimental data are generally within the range bounded by these two computational models.

The results obtained with the linear versions of the exponent and hyperbolic yield criteria are identical to those obtained with the linear Drucker-Prager criterion. In Abaqus/Standard the analyses with the exponential and hyperbolic criteria generally require fewer iterations to achieve a converged solution compared to analyses with the linear criterion. This is attributed to the smooth, continuous hyperbolic flow potential used with the exponential and hyperbolic yield criteria.

The results discussed in the previous paragraphs correspond to the Abaqus/Standard analyses using CAX4 elements. The solutions obtained with the Abaqus/Explicit simulations using CAX4R elements are in close agreement. Similarly, the three-dimensional solution obtained with cylindrical elements also agrees closely with the corresponding axisymmetric solution. The results of these simulations are not reported here.

Input files

Abaqus/Standard input files

concreteslump_castiron.inp	Cast iron plasticity model.
concreteslump_beta30.inp	Linear Drucker-Prager model with $\beta = 30.16°$.
concreteslump_beta0.inp	Model with $\beta = 0°$. Note that Mises plasticity, rather than Drucker-Prager plasticity, is used.
concreteslump_beta8.inp	Exponential Drucker-Prager model with $\beta = 8.574°$.
concreteslump_beta43.inp	Hyperbolic Drucker-Prager model with $\beta = 43.32°$.
concreteslump_3dcyl.inp	Cylindrical element model with $\beta = 0°$.

The differences in the following data files are only in the Drucker-Prager parameters:

concreteslump_phi0.inp	Mohr-Coulomb model with $\phi = 0°$.
concreteslump_phi5.inp	Mohr-Coulomb model with $\phi = 5°$.
concreteslump_phi20.inp	Mohr-Coulomb model with $\phi = 20°$.
concreteslump_phi35.inp	Mohr-Coulomb model with $\phi = 35°$.

The differences in the following data files are only in the Mohr-Coulomb parameters:

Abaqus/Explicit input files

concreteslump_castiron_xpl.inp	Cast iron plasticity model.
concreteslump_beta30_xpl.inp	Linear Drucker-Prager model with $\beta = 30.16°$.
concreteslump_beta0_xpl.inp	Model with $\beta = 0°$. Note that Mises plasticity, rather than Drucker-Prager plasticity, is used.
concreteslump_beta8_xpl.inp	Exponential Drucker-Prager model with $\beta = 8.574°$.
concreteslump_beta43_xpl.inp	Hyperbolic Drucker-Prager model with $\beta = 43.32°$.

The differences in the following data files are only in the Drucker-Prager parameters:

concreteslump_phi0_xpl.inp	Mohr-Coulomb model with $\phi = 0°$.
concreteslump_phi5_xpl.inp	Mohr-Coulomb model with $\phi = 5°$.
concreteslump_phi20_xpl.inp	Mohr-Coulomb model with $\phi = 20°$.
concreteslump_phi35_xpl.inp	Mohr-Coulomb model with $\phi = 35°$.

The differences in the following data files are only in the Mohr-Coulomb parameters:

References

- Christensen, G., Modeling the Flow of Fresh Concrete: The Slump Test, Ph.D. dissertation, Princeton University, 1991.
- Famiglietti, C. M., and J. H. Prevost, "Solution of the Slump Test Using a Finite Deformation Elasto-Plastic Drucker-Prager Model," *International Journal for Numerical Methods in Engineering*, vol. 37, pp. 3869–3903, 1994.

Tables

Table 1: Drucker-Prager material parameters. For all models it is assumed that $\beta = \psi$.

Mohr-Coulomb		Linear		Exponential		Hyperbolic
c	ϕ	β	d	a	b	p_t
1.1547×10^{-3}	0°	0.000°	2.00×10^{-3}	N/A	N/A	N/A
1.1547×10^{-3}	5°	8.574°	1.989×10^{-3}	6.632	1.0	1.319×10^{-2}
1.1547×10^{-3}	20°	30.164°	1.844×10^{-3}	1.721	1.0	3.173×10^{-3}
1.1547×10^{-3}	35°	43.322°	1.555×10^{-3}	1.060	1.0	1.649×10^{-3}

Figures

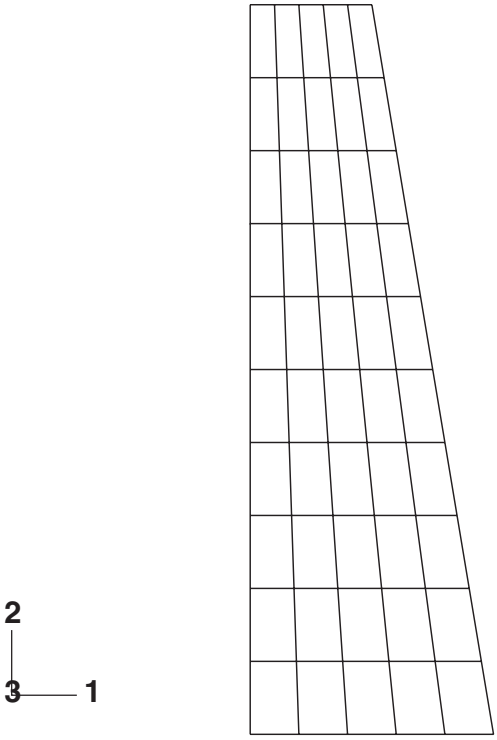


Figure 1: Undeformed mesh (CAX4 elements).

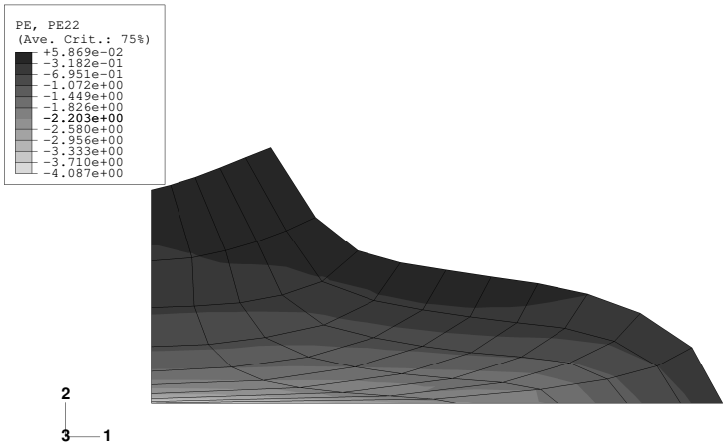


Figure 2: Contours of PE22 for model with $\beta =$ 0°.

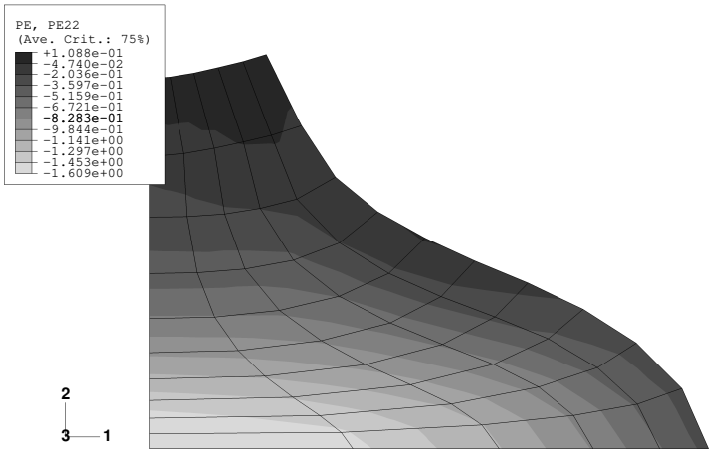


Figure 3: Contours of PE22 for model with $\beta =$ 30.16°.

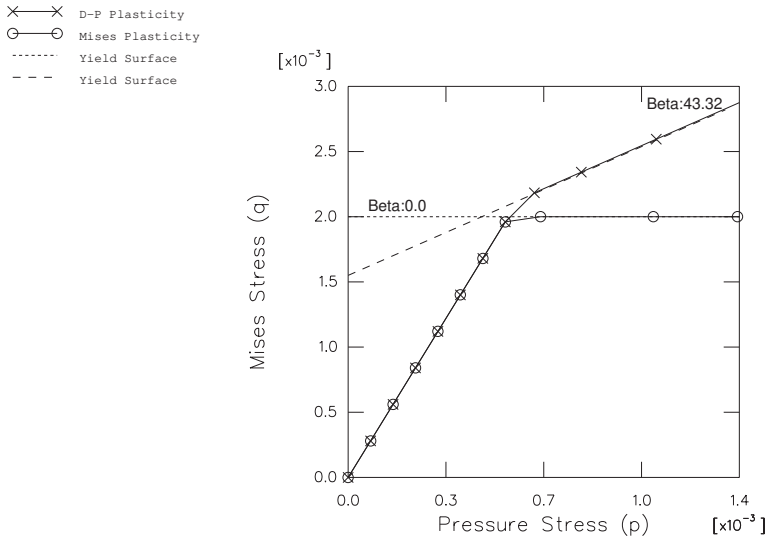


Figure 4: Material point trajectory in meridional stress plane for $\beta =$ 0° and $\beta =$ 43.32°.

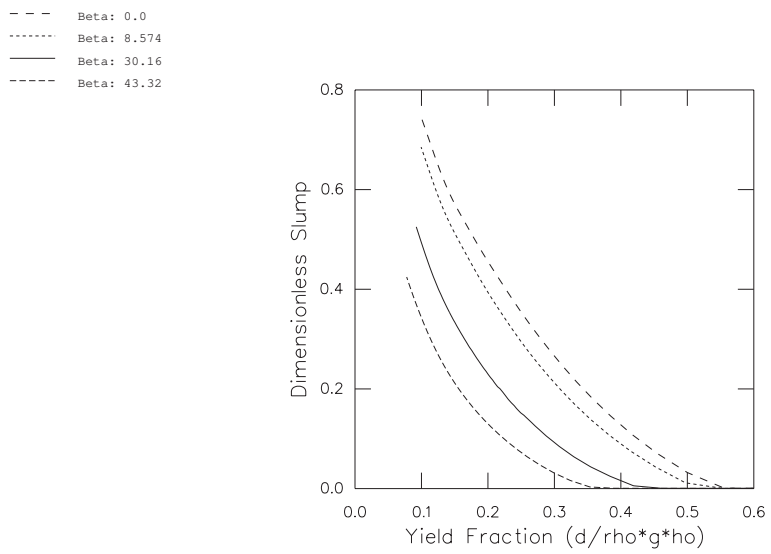


Figure 5: Dimensionless slump vs. yield fraction.

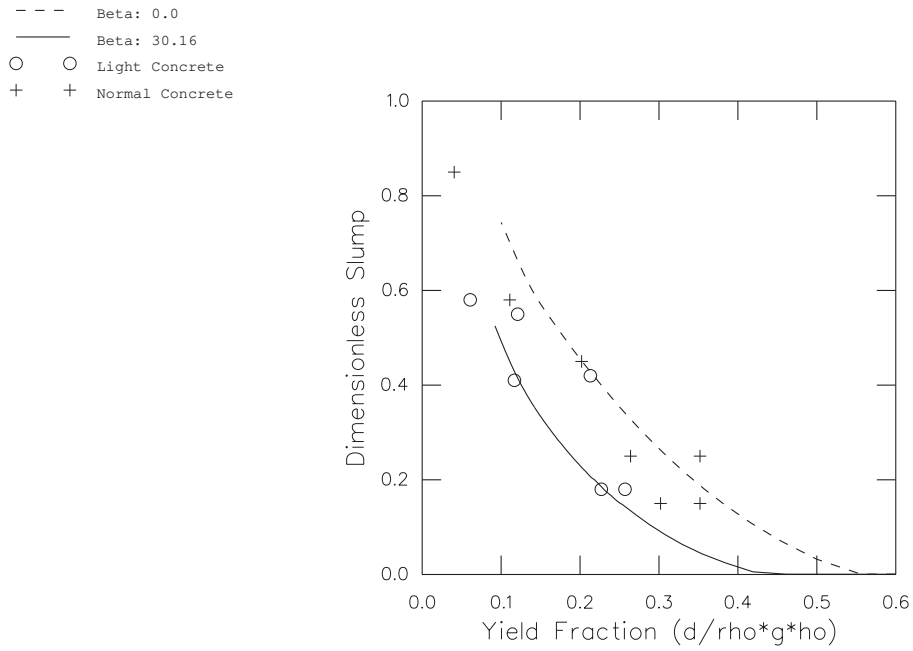


Figure 6: Comparison of experimental slump test results (from Christensen) with computational results.

The Hertz contact problem

Products: Abaqus/Standard Abaqus/Explicit

Overview

The Hertz contact problem (see Timoshenko and Goodier, 1951) provides a classic example for verifying the contact capabilities in Abaqus.
It also serves as an excellent illustration of the use of substructuring in Abaqus/Standard for locally nonlinear cases (local surface contact). In addition, the problem is analyzed under dynamic conditions in Abaqus/Standard to illustrate the use of contact surfaces in such cases.

Problem description

The Hertz contact problem studied consists of two identical, infinitely long cylinders pressed into each other. The solution quantities of most interest are the pressure distribution on the contacting area, the size of the contact area, and the stresses near the contact area. The material behavior is assumed to be linear elastic, and geometric nonlinearities are ignored. Therefore, the only nonlinearity in the problem is the contact constraint.

The cylinders in this example have a radius of 254 mm (10 in) and are elastic, with Young's modulus of 206 GPa (30×10^6 lb/in^2) and Poisson's ratio of 0.3. Smooth contact (no friction) is assumed.

The contact area remains small compared to the radius of the cylinders, so the vertical displacements along the diametric chord of the cylinder that is parallel to the contact plane are almost uniform. This, together with the symmetry of the problem, requires only one-quarter of one cylinder to be modeled. Displacements are prescribed on the diametric cut parallel to the rigid plane to load the problem. For this example, the nodes along the diametric cut are displaced vertically down by 10.16 mm (0.4 in). The total load per unit length of the cylinder can be obtained by summing the corresponding reaction forces on the cylinder or equivalently as the reaction force on the rigid body reference node.

For illustration, the problem is modeled in both two and three dimensions.

In the two-dimensional Abaqus/Standard case the quarter-cylinder is modeled with 20 8-node plane strain elements (see *Figure 1*). In the two-dimensional Abaqus/Explicit case the quarter-cylinder is modeled with either 171 4-node plane strain (CPE4R) elements (see *Figure 5*) or 130 6-node plane strain (CPE6M) elements (see *Figure 6*). In the three-dimensional cases a cylinder of unit thickness is modeled, with the out-of-plane displacements fixed on the two exterior faces of the model to impose the plane strain condition. The bulk of the cylinder is modeled in Abaqus/Standard with 16 20-node bricks; the remaining four elements that abut the surface where contact may occur are modeled with element type C3D27, which is a brick element that allows a variable number of nodes. This element is intended particularly for three-dimensional contact analysis. Element type C3D27 always has at least 21 nodes: the corner nodes, the midedge nodes, and one node at the element's centroid. The midface nodes may be omitted at the user's discretion. In this case the midface nodes on the surfaces where contact may occur are retained. The other midface nodes (on the element faces that are interior to the cylinder) are omitted, making those faces compatible with the 20-node bricks used in the remainder of the model. This use of 27-node brick elements is strongly recommended for three-dimensional contact problems in which second-order elements are used: it is almost essential for cases where partial contact may occur over element surfaces, as is the case in this example. The reason is that the interpolation on the surface of a quadratic element without a midface node is based on the four corner nodes and the four midedge nodes only and is, therefore, rather incomplete (it is not a product of Lagrange interpolations). Therefore, if a quadratic element is specified as part of the secondary surface definition and there is no midface node on the contacting face, Abaqus/Standard will generate the midface node automatically and modify the element definition appropriately. In Abaqus/Explicit meshes with either C3D8R elements or C3D10M elements are used.

It is clearly advisable to refine the portion of the mesh near the expected contact region to predict the contact pressure and contact area accurately. This refinement is accomplished in Abaqus/Standard by using one of the default multi-point constraints provided for this purpose (*General Multi-Point Constraints*). In Abaqus/Explicit a more refined mesh with mesh gradation is used.

To be consistent with the Hertz solution, geometric nonlinearities are neglected for all Abaqus/Explicit cases.

Contact modeling

Because of symmetry, the contact problem can be modeled as a deformable cylinder being pressed against a flat, rigid surface. Therefore, two contact surfaces are required: one (the secondary surface in Abaqus/Standard) on the deformable cylinder and the other (the main surface in Abaqus/Standard) on the rigid body.

For illustrative purposes several different techniques are used to define the contacting surface pairs. The secondary surface is defined by (1) grouping the free faces of elements in an element set that includes all elements in the region that potentially will come into contact (Abaqus defines the faces automatically), (2) specifying the faces of the elements (or the element sets) in the contact region, or (3) identifying the nodes on the deformable body in the contact region that may come into contact. The main surface is defined by (1) specifying the faces of the rigid elements (or element sets) used to define the rigid body or (2) defining the rigid surface with a surface definition and rigid body constraint. Any combination of these techniques can be used together.

By default, Abaqus uses a finite-sliding contact formulation for modeling the interaction between contact pairs. The contacting surfaces undergo negligible sliding relative to each other, which makes this problem a candidate for the small-sliding contact formulation. For a discussion of small- versus finite-sliding contact, see *Contact Formulations in Abaqus/Standard* or *Contact Formulations for Contact Pairs in Abaqus/Explicit*.

The surface contact formulation in Abaqus/Standard gives an accurate solution for the contact area and pressure distribution between the surfaces because of the choice of integration scheme used. Irons and Ahmad (1980) suggest a Gaussian integration rule for calculating self-consistent areas for surface boundary condition problems, which for second-order elements can lead to oscillating results for the pressure distribution on the surface. Oden and Kikuchi explain why this behavior occurs (1980) and present the remedy of using Simpson's integration rule instead. This technique is used in Abaqus/Standard, and no oscillations in the pressure distribution are found.

The default contact pair formulation in the normal direction in Abaqus/Standard is hard contact, which gives strict enforcement of contact constraints. Some standard analyses of this problem 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 hard and augmented Lagrangian contact algorithms are described in *Contact Constraint Enforcement Methods in Abaqus/Standard*.

The default contact pair formulation in Abaqus/Explicit is kinematic contact, which gives strict enforcement of contact constraints. (Note: the small-sliding contact option mentioned previously is available only with kinematic contact.) The explicit dynamic analyses of this problem are conducted with both kinematic and penalty contact to demonstrate that the penetration characteristic of the penalty method can affect stress results significantly in problems with displacement-controlled loading and purely elastic response. The kinematic and penalty contact algorithms are described in *Contact Constraint Enforcement Methods in Abaqus/Explicit*.

Substructure Abaqus/Standard model

This type of contact problem is very suitable for analysis using the substructuring technique in Abaqus/Standard, since the only nonlinearity in the problem is the contact condition, which is quite local. The cylinder can be defined as a substructure and, thus, reduced to a small number of retained degrees of freedom on the surface where contact may occur or where boundary conditions may be changed. During the iterative solution for contact only these external degrees of freedom on the substructure appear in the equations, thus substantially reducing the cost per iteration. Once the local nonlinearity has been resolved, the solution in the cylinder is recovered as a purely linear response to the known displacements at these retained degrees of freedom. This technique is

particularly effective in this case because the rigid surface is flat and there is no friction on the surface; therefore, only the displacement component normal to the surface needs to be retained in the nonlinear iterations.

All information that is relevant to the substructure generation must be given within the substructure generation step, including the degrees of freedom that will be retained. The substructure creation and usage cannot be included in the same input file. Only one substructure can be generated per input file. Any number of unit load cases can be defined for the substructure by using a substructure load case. Although this feature is not necessary in this example, it is used in one of the input files for verification purposes.

Substructures are introduced into an analysis model using elements, where the element number and nodes are defined for each usage of each substructure. Node and element numbers within a substructure and at the usage level are independent—the same node and element numbers can be reused in different substructures and on the usage level. It is also possible to refer to a substructure several times if the structure has identical sections. Thus, once a substructure has been created, it is used just as a standard element type.

Results and discussion

Results for the Abaqus/Standard and Abaqus/Explicit analyses are discussed in the following paragraphs.

Abaqus/Standard results

Despite the rather coarse mesh, *Figure 2* shows that the contact pressure between the cylinders predicted by the two-dimensional Abaqus/Standard model is in good agreement with the analytical distribution. The numerical solution is less accurate at the boundary of the contact patch where the contact pressure is characterized by a strong gradient. This aspect is also captured by the contact pressure error indicator. The only realistic way to improve the numerical solution would be to use a more detailed discretization. Almost identical results are obtained from the three-dimensional Abaqus/Standard model.

Figure 3 shows contours of Mises equivalent stress. This plot verifies that the highest stress intensity (where the material will yield first) occurs inside the body and not on the surface. *Figure 4* shows the deformed configuration. In that figure the contacting surface of the cylinder appears to be curved downward because of the magnification factor used to exaggerate the displacements to show the results more clearly.

In this example substructuring reduces the computer time required for the job substantially because it allows the nonlinear contact problem to be resolved among a small number of active degrees of freedom. Substructuring involves considerable computational “overhead” because of the complex data management required. The reduced stiffness matrix coupling the retained degrees of freedom on a substructure is a full matrix. Thus, the method is not always as advantageous as this example would suggest. The use of substructures usually increases the analysis time in a purely linear analysis, unless a substructure can be used several times. In such cases the advantage of the method is that it allows a large analysis to be divided into several smaller analysis jobs, in each of which a substructure is created or substructures are used to build the next level of the analysis model.

Abaqus/Explicit results

The prescribed displacements on the diametric cut are ramped up over a relatively long time (.01 s) to minimize inertial effects. The displacements are then fixed for a short time (.001 s) to verify that the explicit dynamic results are truly quasi-static. Throughout the analysis the total kinetic energy is less than .1% of the total internal energy. In addition, the sum of the vertical reaction forces along the diametric cut closely matches the sum from the nodes in contact with the rigid body. These results indicate that the analysis can be accepted as quasi-static.

Figure 7 and *Figure 8* show the contact pressures between the cylinders for the two-dimensional models using kinematic and penalty contact, respectively. The contact pressure distribution shows the classical elliptic distribution. The maximum pressure occurs at the symmetry plane and, for the kinematic contact analysis, is within 1% of the classical solution. However, the contact pressure is significantly lower when penalty contact is used because of the contact penetration. Almost identical results are obtained from the three-dimensional Abaqus/Explicit models.

Figure 9 and *Figure 10* show contours of Mises equivalent stress for kinematic and penalty contact, respectively. Again, the stress is significantly less with penalty contact than with kinematic contact. These plots verify that the highest stress intensity (where the material will yield first) occurs inside the body and not on the surface. *Figure 11* and *Figure 12* show the deformed configurations for the two types of contact enforcement; note the contact penetration in *Figure 12*.

In most cases kinematic contact and penalty contact will produce very similar results. However, there are exceptions, as this problem demonstrates. The user should be aware of the characteristics of both contact constraint methods, which are discussed in *Contact Constraint Enforcement Methods in Abaqus/Explicit*. The kinematic contact method is better suited for this analysis because the penetrations associated with the penalty method influence the solution significantly. It is uncommon for these penetrations to be significant. Factors that tend to increase the significance of contact penetrations are: 1) displacement-controlled loading, 2) highly confined regions, 3) coarse meshes, and 4) purely elastic response. The penetrations can be reduced by increasing the penalty stiffness. However, increasing the penalty stiffness will tend to decrease the stable time increment and, thus, increase the analysis cost. This time reduction can be avoided allowing mass scaling to account for the contact stiffness.

Figure 13 shows the contact pressure between the cylinders for a model meshed with CPE6M elements that uses kinematic contact enforcement. *Figure 14* and *Figure 15* show contours of Mises equivalent stress and the deformed configuration, respectively, for this analysis. The maximum contact pressure is again within 1% of the classical solution, and the distribution of Mises equivalent stress is very similar to that obtained with CPE4R elements and kinematic contact enforcement. Similar results are obtained using C3D10M elements.

Dynamic analysis in Abaqus/Standard

A simple dynamic example is created in Abaqus/Standard by giving the cylinder a uniform initial velocity with the contact conditions all open. This represents the experiment of dropping the cylinder onto a rigid, flat floor under a gravity field.

The impact algorithm used in Abaqus/Standard for dynamic contact is based on the assumption that, when any contact occurs, the total momentum of the bodies remains unchanged while the points that are contacting will acquire the same velocity instantaneously. In this example the cylinder contacts a rigid surface, which implies that each contacting point will suddenly have zero vertical velocity. This means that a compressive stress wave will emanate from the contacting point and will travel back into the cylinder. After some time this will cause the cylinder to rebound.

It is important to understand that the Abaqus/Standard dynamic contact algorithm is a “locally perfectly plastic impact” algorithm, as described above, which gives excellent results when it is used correctly. However, it is readily seen that, if the cylinder were modeled as a concentrated mass, with one vertical degree of freedom, the algorithm would imply that the cylinder stops instantaneously when it hits the rigid surface. In reality neither the cylinder nor the surface it hits are rigid: stress waves are started in each. Enough of this detail must be modeled for the results to be meaningful. In this example, the cylinder itself is modeled in reasonable detail to capture at least the overall dynamic behavior. If the physical problem from which the example has been developed is that of two cylinders with equal and opposite velocities, this solution is probably useful. If the physical problem is that of a single cylinder hitting a flat surface, it may be necessary to include some elements to model the material below the surface (and the propagation of energy into that domain), unless that material is very dense so that this propagation can be neglected.

Input files

Abaqus/Standard input files

hertzcontact_2d_relem.inp	Two-dimensional model with rigid elements.
hertzcontact_2d_relem_auglagr.inp	Two-dimensional model with rigid elements and augmented Lagrangian contact.

hertzcontact_2d_rsurf.inp	Two-dimensional model with a rigid surface.
hertzcontact_2d_substr.inp	Analysis using substructuring.
hertzcontact_2d_gen1.inp	Substructure generation referenced by the analysis hertzcontact_2d_substr.inp.
hertzcontact_3d.inp	Three-dimensional problem.
hertzcontact_3d_surf.inp	Three-dimensional problem, surface-to-surface approach.
hertzcontact_3d_auglagr.inp	Three-dimensional problem with augmented Lagrangian contact.
hertzcontact_3d_auglagr_surf.inp	Three-dimensional problem with augmented Lagrangian contact, surface-to-surface approach.
hertzcontact_2d_dynamic.inp	Dynamic analysis.
hertzcontact_2d_5inc.inp	Two-dimensional analysis with the step divided into five increments and the restart file saved.
hertzcontact_2d_res.inp	A restart analysis from increment 2 of the previous job. These files are included to verify the restart capability with contact.

The following files are provided as additional illustrations and test cases for the substructuring and matrix output options:

hertzcontact_2d_substr_xnode.inp	Substructure analysis with additional nodes retained by moving the **EQUATION* definition to the global level.
hertzcontact_2d_xnodes_gen1.inp	Substructure generation referenced by the analysis hertzcontact_2d_substr_xnode.inp.
hertzcontact_2d_substr_sload.inp	Substructure analysis with the displacement loading applied using the **SLOAD* option.
hertzcontact_2d_sload_gen1.inp	Substructure generation referenced by the analysis hertzcontact_2d_substr_sload.inp.
hertzcontact_3d_substr.inp	Three-dimensional analysis using substructuring.
hertzcontact_3d_gen1.inp	Substructure generation referenced by the analysis hertzcontact_3d_substr.inp.
hertzcontact_3d_sub_only.inp	Generates the substructure only; outputs the matrix computed during the substructure generation, the substructure matrix, to a results file.
hertzcontact_3d_sub_library.inp	Uses the substructure generated by the previous input file as a substructure database file; prints the substructure matrix to a results file after it has been read in as an element from the substructure file. The **ELEMENT MATRIX OUTPUT* option is used to output the matrix in this case.
hertzcontact_3d_res.inp	Restart job of problem hertzcontact_3d_sub_library.inp. It is necessary to provide both the restart file and the substructure database file for this job.
hertzcontact_3d_uel.inp	Uses the **USER ELEMENT* option to read in the substructure matrix output during its generation. This matrix is then used to complete the analysis.
hertzcontact_3d_uel2.inp	Again uses the **USER ELEMENT* option to read in the substructure matrix. The same analysis is completed again with the matrix output during its use rather than during its generation.
hertzcontact_2d_rsurf_unsym.inp	Two-dimensional model with rigid elements. This model uses the unsymmetric solver.
hertzcontact_2d_rsurf_unsym_gen1.inp	Substructure generation referenced by the analysis hertzcontact_2d_rsurf_unsym.inp.

hertzcontact_2d_symsub_unsym.inp	Uses a previously created symmetric substructure in a model that uses the unsymmetric solver.
hertzcontact_2d_unsorted.inp	A substructure model with unsorted node sets and unsorted retained degrees of freedom.
hertzcontact_2d_unsorted_gen1.inp	Substructure generation referenced by the analysis hertzcontact_2d_unsorted.inp.
hertzcontact_cpe6m.inp	Two-dimensional problem with rigid elements and CPE6M elements.
hertzcontact_cpe6m_auglagr.inp	Two-dimensional problem with rigid elements and CPE6M elements, augmented Lagrangian contact.
hertzcontact_cpe6m_substr.inp	Two-dimensional problem with CPE6M elements using substructuring.
hertzcontact_cpe6m_gen1.inp	Substructure generation referenced by the analysis hertzcontact_cpe6m_substr.inp.
hertzcontact_cpe6m_dyn.inp	Two-dimensional dynamic analysis using CPE6M elements.
hertzcontact_cpeg8.inp	Two-dimensional problem with rigid elements and CPEG8 elements.
hertzcontact_2d_substr_cpeg8.inp	Two-dimensional problem with CPEG8 elements using substructuring.
hertzcontact_2d_gen1_cpeg8.inp	Substructure generation referenced by the analysis hertzcontact_2d_substr_cpeg8.inp.
hertzcontact_cpeg8_dyn.inp	Two-dimensional dynamic analysis using CPEG8 elements.
hertzcontact_cpeg8_dyn_auglagr.inp	Two-dimensional dynamic analysis using CPEG8 elements and augmented Lagrangian contact.
hertzcontact_c3d10m.inp	Three-dimensional problem using C3D10M elements.
hertzcontact_c3d10m_auglagr.inp	Three-dimensional problem using C3D10M elements and augmented Lagrangian contact.
hertzcontact_c3d10m_auglagr_res.inp	A restart analysis from increment 2 of the analysis hertzcontact_c3d10m_auglagr.inp.
hertzcontact_c3d10m_substr.inp	Three-dimensional problem with C3D10M elements using substructuring.
hertzcontact_c3d10m_gen1.inp	Substructure generation referenced by the analysis hertzcontact_c3d10m_substr.inp.
hertzcontact_c3d10m_dyn.inp	Three-dimensional dynamic analysis using C3D10M elements.
hertzcontact_substr45.inp	A substructure model where the substructure has been rotated through an angle of 45°. The **EQUATION* option is used during the substructure definition, and the **TRANSFORM* option is used at the usage level.
hertzcontact_substr45_gen1.inp	Substructure generation referenced by the analysis hertzcontact_substr45.inp.
hertzcontact_2d_cload.inp	A two-dimensional model in which the two cylinders are initially apart, and the deformation is produced by a point load instead of a displacement boundary condition. The **CONTACT CONTROLS* option with the STABILIZE parameter is used to prevent rigid body motion until contact is established.
hertzcontact_2d_cload_auglagr.inp	An augmented Lagrangian contact model of the analysis hertzcontact_2d_cload.inp.
hertzcontact_2d_kincoup.inp	Two-dimensional problem with the displacement applied through a **KINEMATIC COUPLING* reference node.
hertzcontact_2d_substr_kincoup.inp	Two-dimensional problem using substructuring with the displacement applied to the top surface through the use of the **KINEMATIC COUPLING* option. The coupling reference node is one of the retained substructure nodes, providing a “handle” for displacing the model.

hertzcontact_2d_kincoup_gen1.inp	Substructure generation referenced by the analysis hertzcontact_2d_substr_kincoup.inp.
hertzcontact_2d_coupk.inp	Two-dimensional problem with the displacement applied to the top surface. The displacement of the top surface is controlled by a reference node through the use of the **COUPLING* and **KINEMATIC* options.
hertzcontact_2d_coupk_substr.inp	Two-dimensional problem using substructuring. The displacement is applied to the top surface through the use of the **COUPLING* and **KINEMATIC* options. The coupling reference node is one of the retained substructure nodes, providing a "handle" for displacing the model.
hertzcontact_2d_coupk_substrgen.inp	Substructure generation referenced by the analysis hertzcontact_2d_coupk_substr.inp.
hertzcontact_2d_coupd_substr.inp	Two-dimensional problem using substructuring. The displacement is applied to the top surface through the use of the **COUPLING* and **DISTRIBUTING* options. The coupling reference node is one of the retained substructure nodes, providing a "handle" for displacing the model. The distributing weight factors are calculated automatically through the tributary surface area.
hertzcontact_2d_coupd_substrgen.inp	Substructure generation referenced by the analysis hertzcontact_2d_coupd_substr.inp.

Note that in both *hertzcontact_3d_uel.inp* and *hertzcontact_3d_uel2.inp* the results file to be used is specified using the FILE parameter on the **USER ELEMENT* option.

Abaqus/Explicit input files

hertz2d.inp	Two-dimensional model.
hertz3d.inp	Three-dimensional model.
hertz2d_sc10.inp	Two-dimensional model with penalty stiffness equal to 10 times the default value.
hertz3d_sc10.inp	Three-dimensional model with penalty stiffness equal to 10 times the default value.
hertz3d_sc10_cms.inp	Three-dimensional model with penalty stiffness equal to 10 times the default value and contact mass scaling activated.
hertz_c3d10m.inp	Three-dimensional model using 10-node quadratic modified tetrahedral elements.
hertz_c3d10m_subcyc.inp	Three-dimensional model using 10-node quadratic modified tetrahedral elements for the sole purpose of testing the performance of the subcycling.
hertz_cpe6m.inp	Two-dimensional model using 6-node quadratic modified triangular elements.

References

- Irons, B., , and S. Ahmad, Techniques of Finite Elements, Ellis Horwood Ltd., Chichester England , 1980.
- Oden, J. T., , and N. Kikuchi, Fifth Invitational Symposium of the Unification of Finite Elements, Finite Differences, Calculus of Variations, H. Kardestuncer, Editor, University of Connecticut at Storrs , 1980.
- Timoshenko, S., , and J. N. Goodier, Theory of Elasticity, Second edition, McGraw-Hill, New York , 1951.

Figures

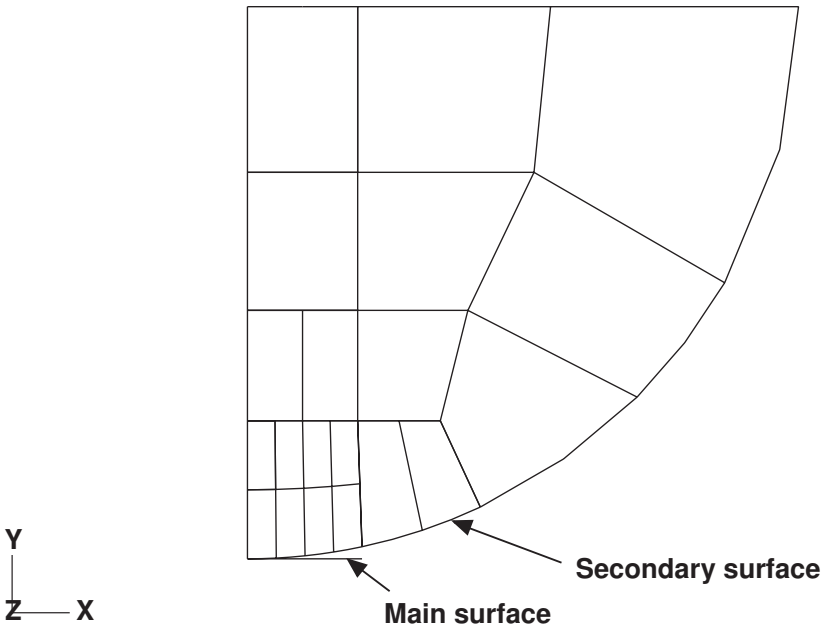


Figure 1: Mesh for the Hertz contact example, Abaqus/Standard.

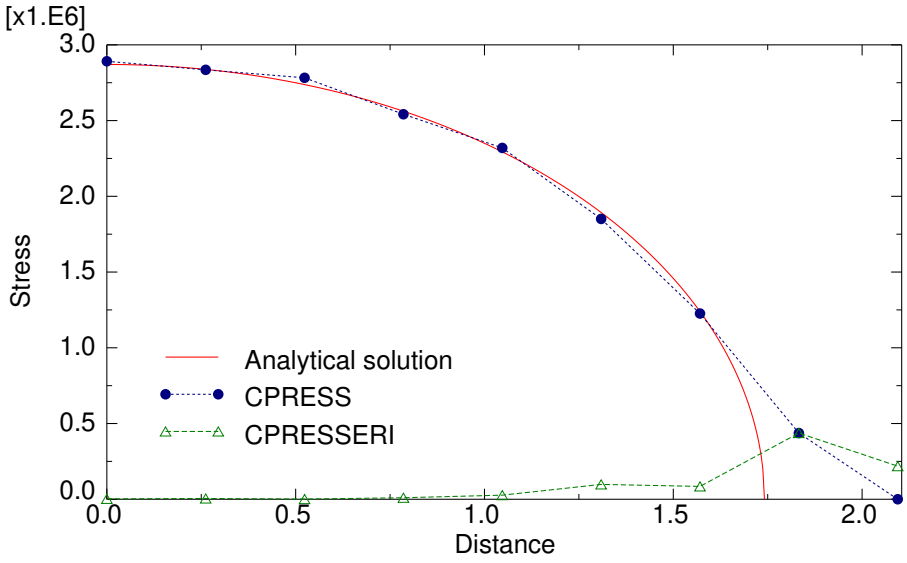


Figure 2: Contact pressure and contact pressure error indicator versus position for the Hertz contact (no friction) example, Abaqus/Standard.

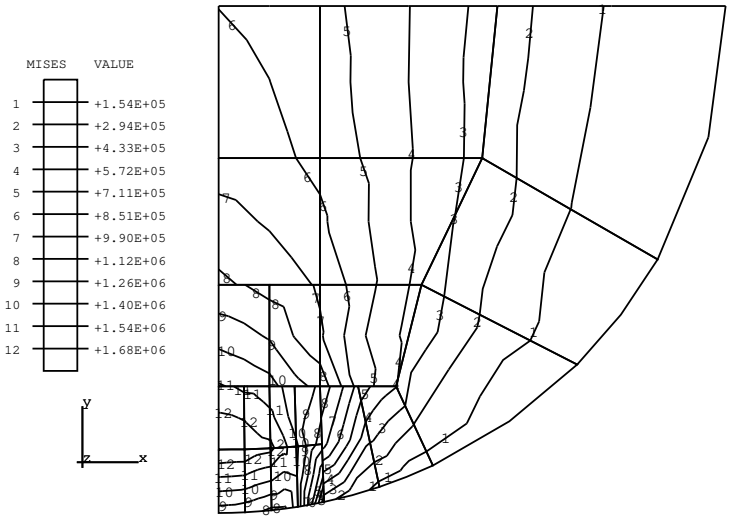


Figure 3: Mises stress distribution for the Hertz contact problem, Abaqus/Standard.

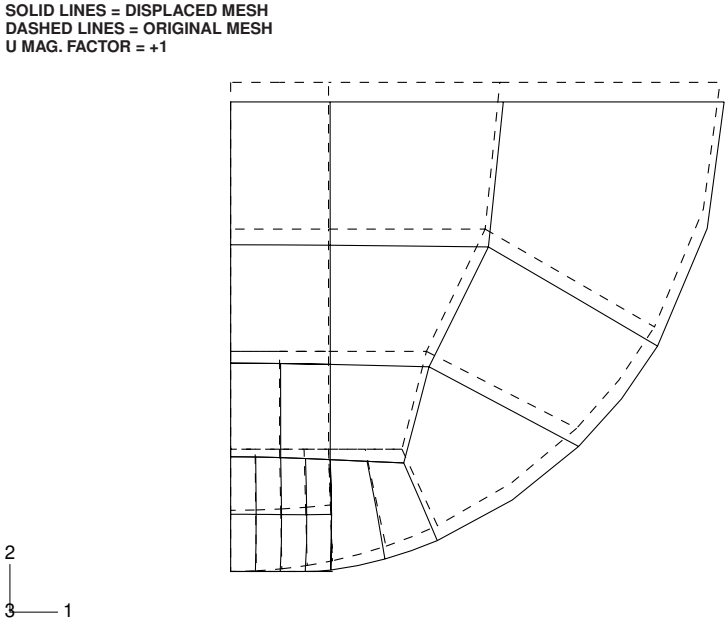


Figure 4: Displaced configuration for the Hertz contact problem, Abaqus/Standard.

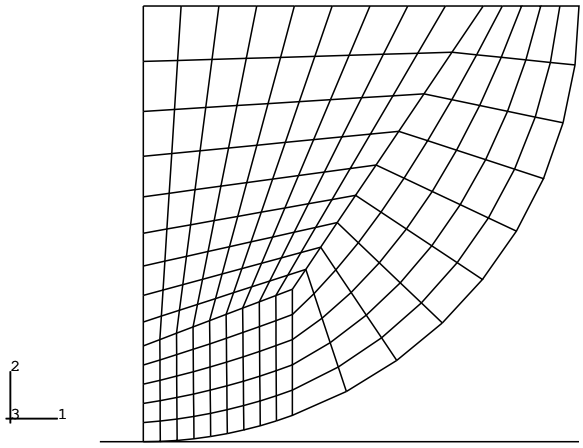


Figure 5: Mesh for the Hertz contact example using CPE4R elements, Abaqus/Explicit.

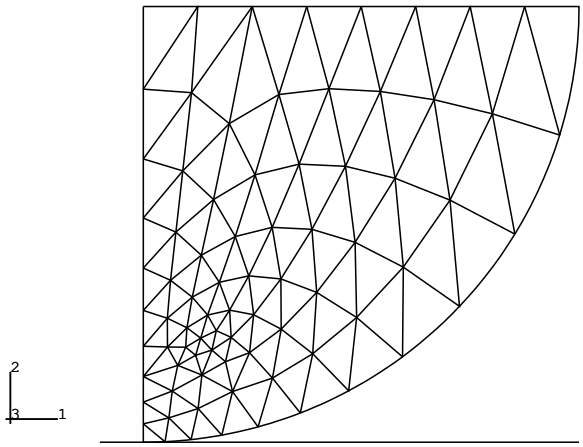


Figure 6: Mesh for the Hertz contact example using CPE6M elements, Abaqus/Explicit.

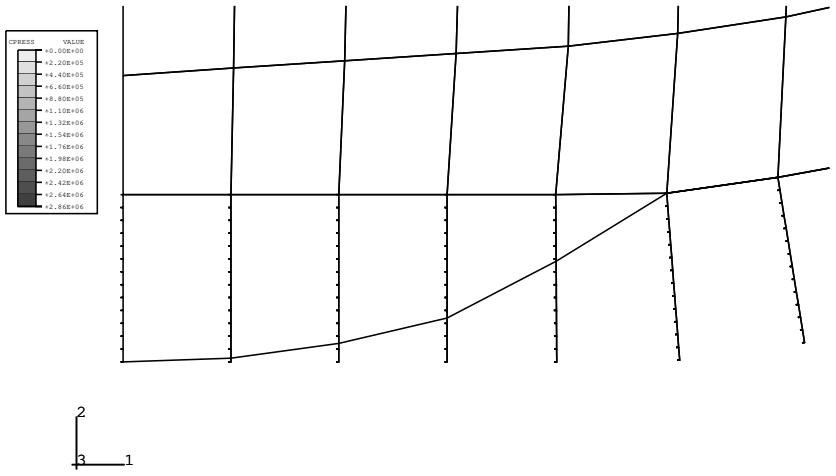


Figure 7: Contact pressure contour for the Hertz contact problem using CPE4R elements and kinematic contact, Abaqus/Explicit.

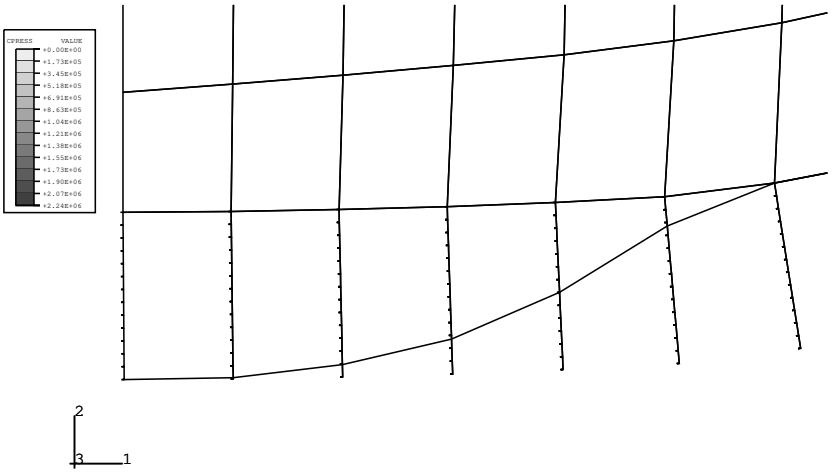


Figure 8: Contact pressure contour for the Hertz contact problem using CPE4R elements and penalty contact, Abaqus/Explicit.

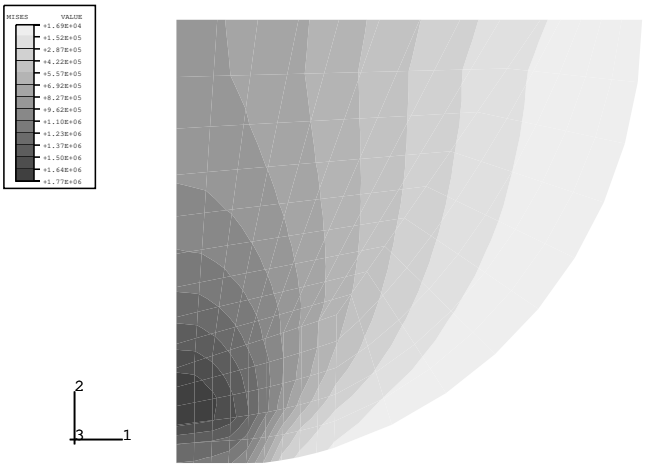


Figure 9: Mises stress distribution for the Hertz contact problem using CPE4R elements and kinematic contact, Abaqus/Explicit.

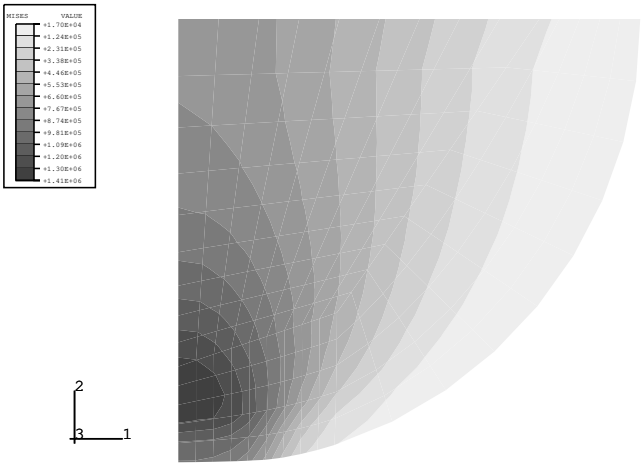


Figure 10: Mises stress distribution for the Hertz contact problem using CPE4R elements and penalty contact, Abaqus/Explicit.

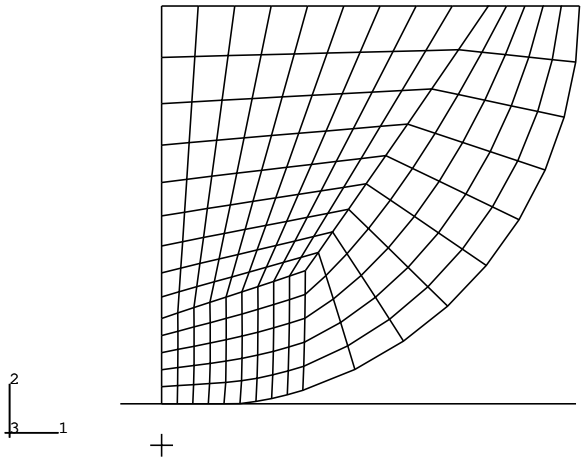


Figure 11: Displaced configuration for the Hertz contact problem using CPE4R elements and kinematic contact, Abaqus/Explicit.

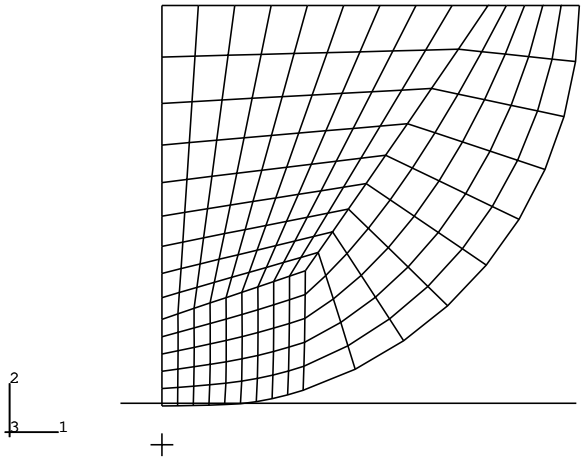


Figure 12: Displaced configuration for the Hertz contact problem using CPE4R elements and penalty contact, Abaqus/Explicit.

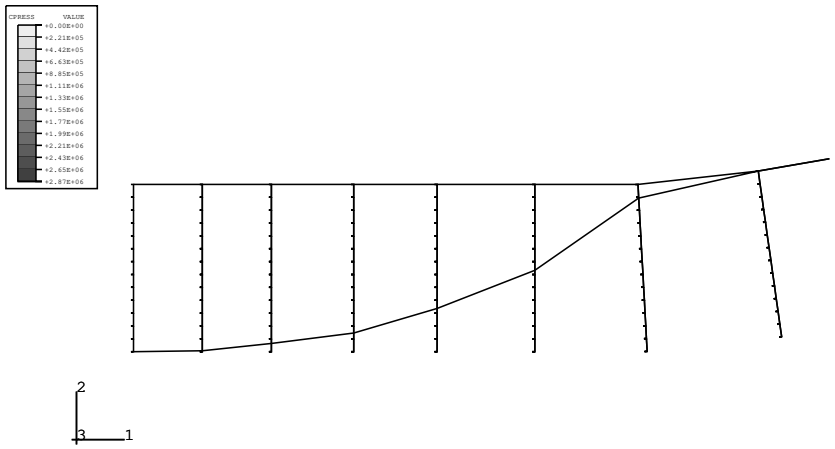


Figure 13: Contact pressure contour for the Hertz contact problem using CPE6M elements and kinematic contact, Abaqus/Explicit.

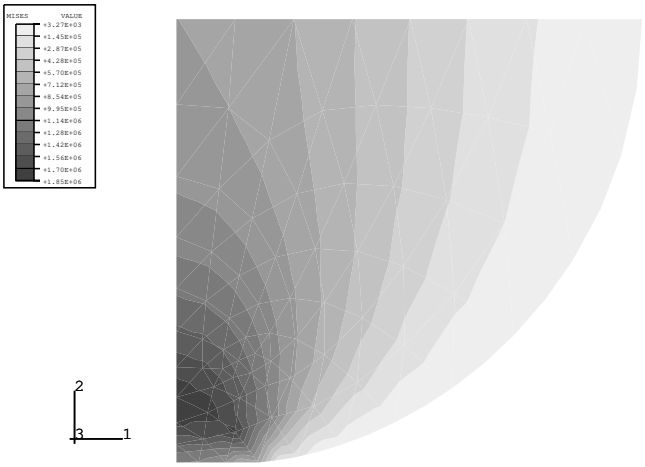


Figure 14: Mises stress distribution for the Hertz contact problem using CPE6M elements and kinematic contact, Abaqus/Explicit.

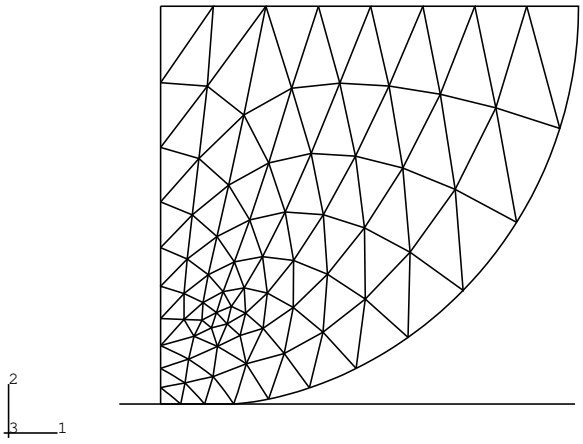


Figure 15: Displaced configuration for the Hertz contact problem using CPE6M elements and kinematic contact, Abaqus/Explicit.

Crushing of a pipe

Products: Abaqus/Standard

Overview

This example shows the crushing of a long, straight pipe between two flat, frictionless anvils.

Extreme accident analysis of piping systems sometimes requires knowledge of the behavior of a pipe section as it is crushed; this example discusses the simplest such investigation. The objectives are to establish the load-deflection response of the pipe and to describe the overall deformation of the section, since this may greatly affect fluid flow through the pipe. The example also provides a simple demonstration of the capabilities of Abaqus for modeling contact problems between deformable bodies and rigid, impenetrable surfaces.

Problem description

The dimensions of the pipe section segment and its material properties are shown in *Figure 1*. By symmetry only one quadrant of the pipe section needs to be modeled. A uniform mesh of fully integrated 8-node, plane strain, "hybrid" elements is used, with two elements through the thickness and eight around the pipe quadrant. No mesh convergence studies have been performed, but the reasonable agreement with experimental results suggests that the mesh is adequate to predict the overall response with usable accuracy.

The contact between the pipe and a flat, rigid anvil is modeled with a contact pair. The outside surface of the pipe is specified using a surface definition. The rigid anvil is modeled as an analytical rigid surface using a surface definition and a rigid body constraint. The mechanical interaction between the contact surfaces is assumed to be frictionless. The pipe is crushed by pushing down on the rigid anvil using a boundary condition to prescribe a downward vertical displacement.

In addition to the plane strain models, a continuum shell element model is provided for illustrative purposes. This model uses a uniform mesh of SC8R elements, with four elements stacked through the thickness and sixteen elements around the pipe quadrant. The anvil is modeled using continuum shell elements and then converted to a rigid body. No mesh convergence studies have been performed, since the results provide reasonable agreement with the experimental results. This model is more costly than the plane strain model since it uses more degrees of freedom.

Results and discussion

Figure 2 shows the load versus relative anvil displacement, compared to the experimental measurements of Peech et al. (1977). The "staircase" pattern of the predicted response is caused by the discrete contact that occurs in the model because contact is detected only at the nodes of the contact secondary surface. A finer mesh would provide a smoother response. The plane strain model predicts the experimental results with reasonably good accuracy up to a relative displacement of about 50.8 mm (2.0 in). Beyond this point the plane strain assumption no longer characterizes the overall physical behavior, and the model predicts a stiffer response than the "ring crush" experiments of Peech et al. (1977). Another analysis of the same problem (Taylor, 1981) shows the same discrepancy, which might, therefore, be attributable to incorrect assumptions about the material behavior.

Deformed configuration plots and contours of equivalent total plastic strain are shown in *Figure 3* and *Figure 4*. Large plastic strains develop near the symmetry planes where the tube is being crushed and extended severely. The "figure eight" shape is correctly predicted: the constriction of the pipe section associated with this geometry will certainly restrict flow.

The results for the continuum shell model are similar to the plane strain model and, hence, show reasonable agreement with the experimental results.

Input files

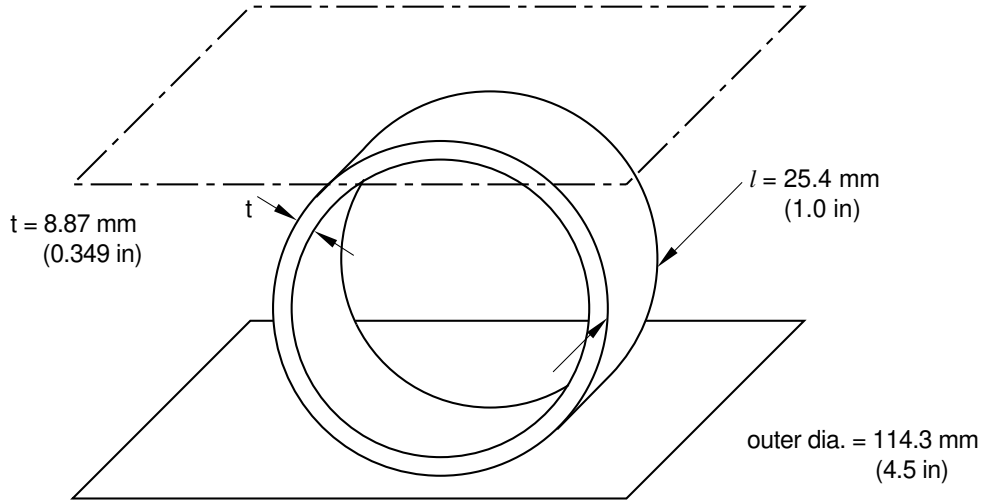
pipecrushing_cpe8h.inp	Plane strain modeling of the ring crush experiment using CPE8H elements.
pipecrushing_cpe6h.inp	CPE6H elements.
pipecrushing_cpe4i.inp	CPE4I elements.
pipecrushing_sc8r.inp	SC8R elements.

References

- Peech, J. M., R. E. Roener, S. D. Porofin, G. H. East, and N. A. Goldstein, “Local Crush Rigidity of Pipes and Elbows,” Proc. 4th SMIRT Conf. paper F-3/8, North Holland, 1977.
- Taylor, L. M., “A Finite Element Analysis for Large Deformation Metal Forming Problems Involving Contact and Friction,” Ph.D. Thesis, U. Texas at Austin, December 1981.

Figures

Pipe segment geometry:



Material behavior:

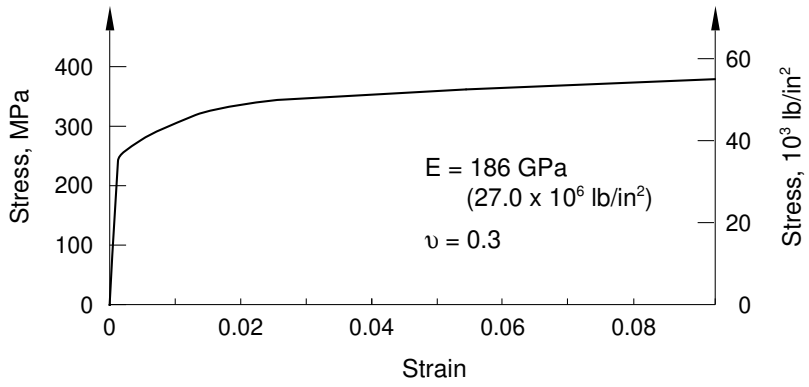


Figure 1: Pipe crush example.

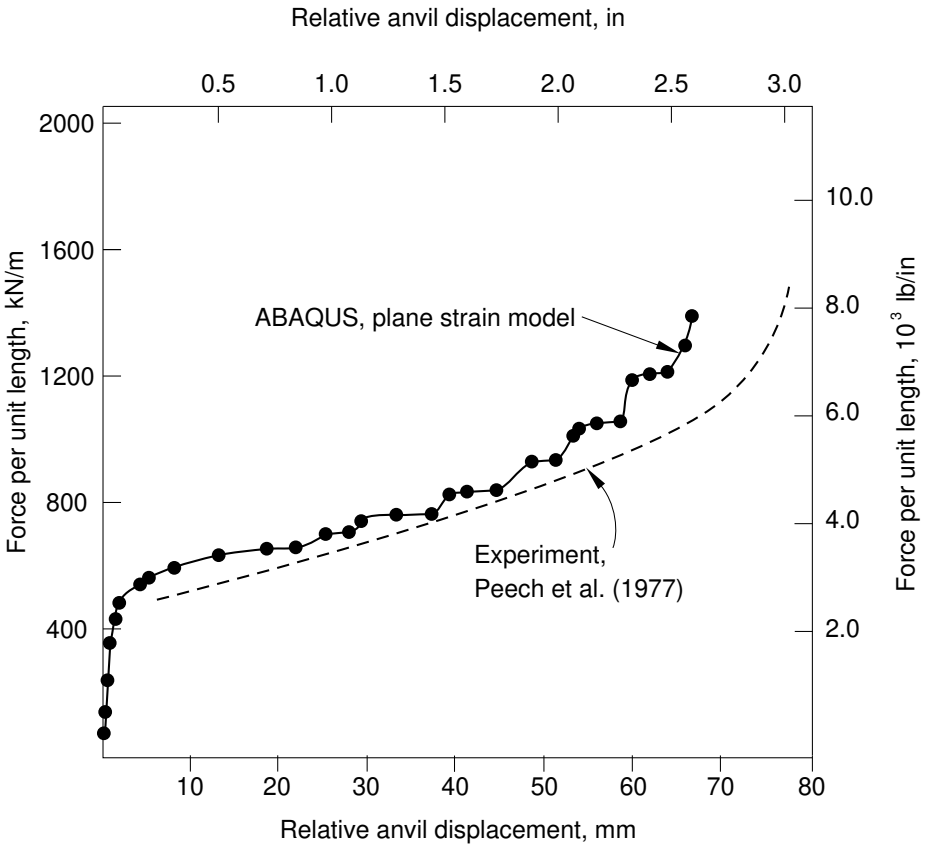


Figure 2: Force-displacement response for pipe crush case.

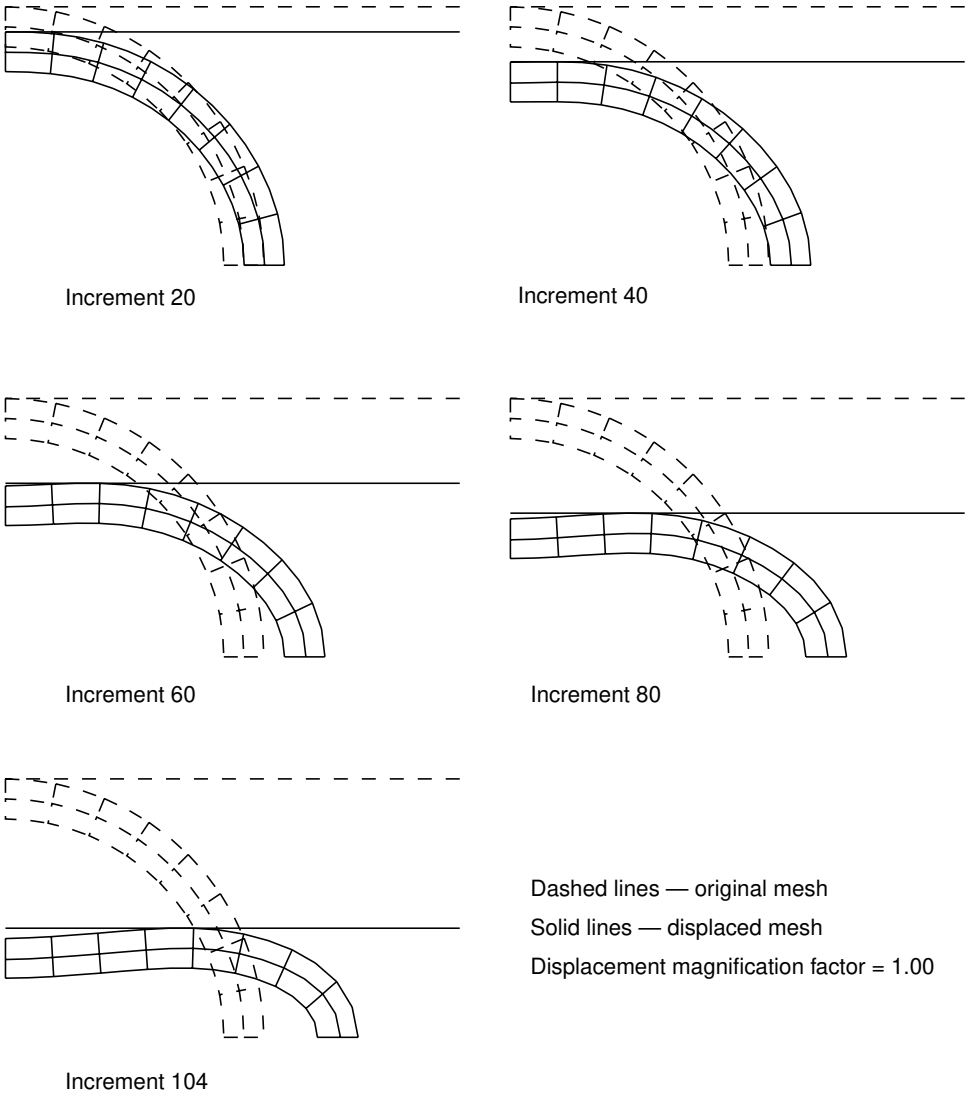


Figure 3: Progressive deformation of the pipe.

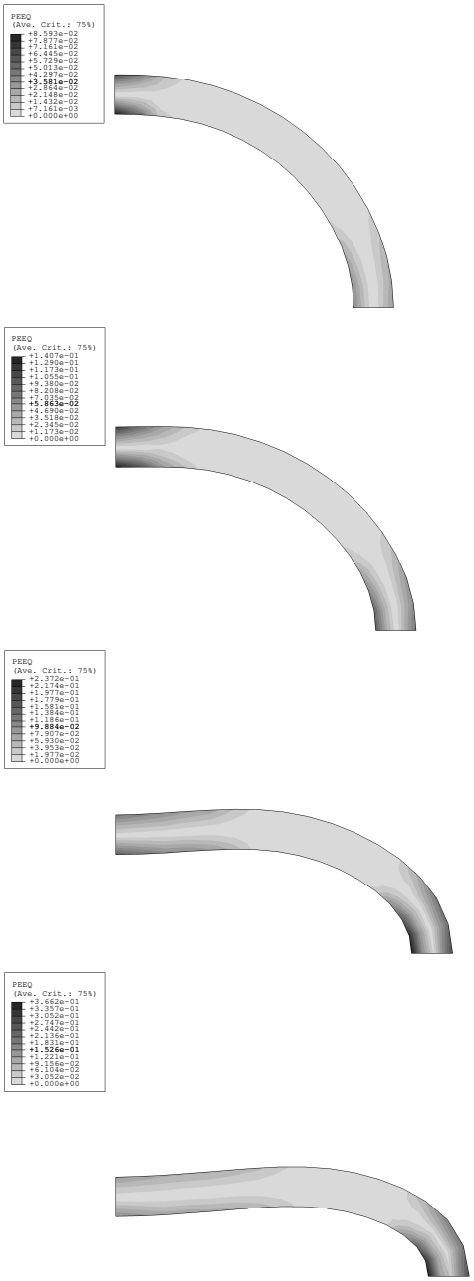


Figure 4: Equivalent plastic strain contours in the pipe.

Radial stretching of a cylinder

Products: Abaqus/Standard

Overview

This problem verifies and illustrates the use of axisymmetric and cylindrical elements under uniform radial displacement in Abaqus.
The analytical solution for stress components in cylindrical coordinates is used to verify the Abaqus quasi-static solution.

Problem description

The physical problem consists of a hollow cylinder with the inner edge constrained in the radial direction. The base of the cylinder is constrained in the axial direction. A uniform radial displacement is specified along the outer edge of the cylinder. *Figure 1* and *Figure 2* show the model geometry used in this analysis. In consistent units the inner and outer radii of the cylinder are 4.0 and 6.0, respectively, with a cylinder height of 2.0. A linear elastic, isotropic material with a Young's modulus of 2×10^{11}, a Poisson's ratio of 0.3, and a density of 1000 is specified. A 20×20 mesh is used to model the axisymmetric domain with CAX4R, CAX4, and CAX8 elements. The complete three-dimensional domain is modeled with CCL12 and CCL24 cylindrical elements. Mesh convergence studies have not been performed.

Results and discussion

The derived analytical solution for this problem is

$$\sigma_{rr} = (2\lambda + 2\mu) C_2 + \lambda C_1 - 2\mu C_3 / r^2,$$
$$\sigma_{\theta\theta} = (2\lambda + 2\mu) C_2 + \lambda C_1 + 2\mu C_3 / r^2,$$
$$\sigma_{zz} = 0,$$
$$\sigma_{rz} = 0,$$
$$C_1 = -\left[2\lambda / (\lambda + 2\mu)\right] \left[U_0 R_o / \left(R_o^2 - R_i^2\right)\right],$$
$$C_2 = U_0 R_o / \left(R_o^2 - R_i^2\right),$$
$$C_3 = -U_0 R_o R_i^2 / \left(R_o^2 - R_i^2\right),$$

where λ and μ are the Lamé parameters; R_o and R_i are the outer and inner radii, respectively; and U_0 is the applied uniform displacement of 0.2 units.

Figure 3 and *Figure 4* show the variation of the radial stress, σ_{rr}, and hoop stress, $\sigma_{\theta\theta}$, with respect to the radius of the hollow cylinder for the various elements. These stresses are compared to the analytical solution. The results for all elements agree well with the analytical solution.

Input files

radstr_holcyl_cax4r.inp	CAX4R axisymmetric model.
radstr_holcyl_cax4.inp	CAX4 axisymmetric model.
radstr_holcyl_cax8.inp	CAX8 axisymmetric model.
radstr_holcyl_ccl12.inp	CCL12 three-dimensional model.
radstr_holcyl_ccl24.inp	CCL24 three-dimensional model.

Figures

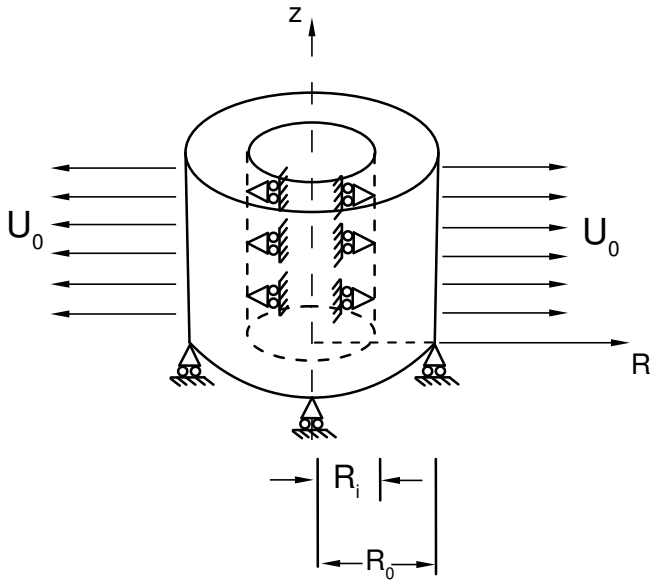


Figure 1: Three-dimensional representation of the problem.

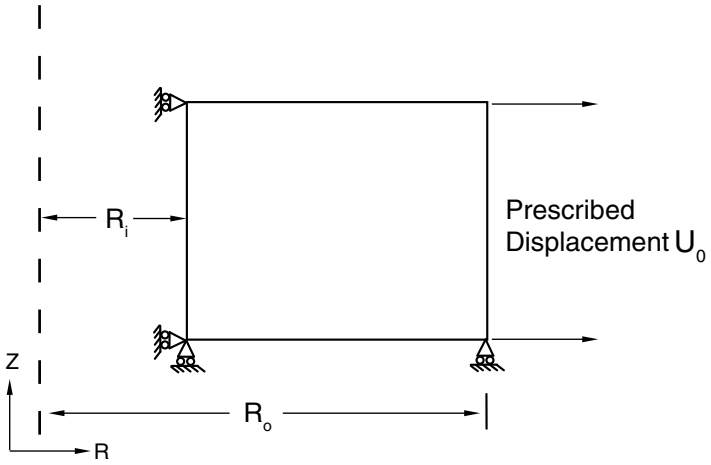


Figure 2: Equivalent axisymmetric model.

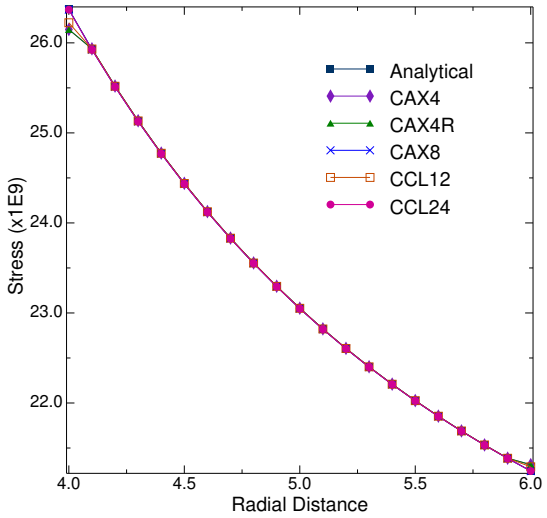


Figure 3: Variation of radial stress with radius.

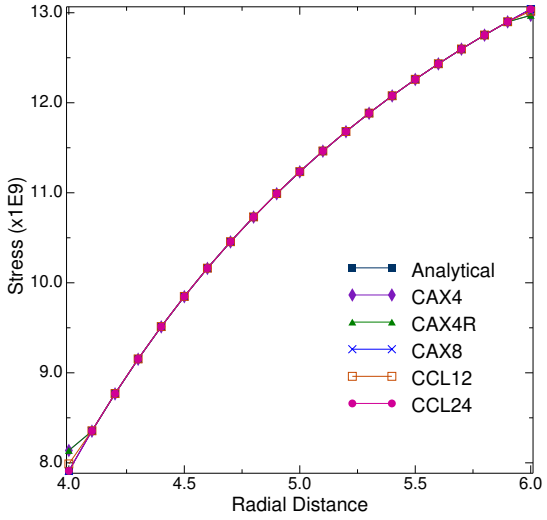


Figure 4: Variation of hoop stress with radius.

Buckling analysis

In this section:

- *Buckling analysis of beams*
- *Buckling of a ring in a plane under external pressure*
- *Buckling of a cylindrical shell under uniform axial pressure*
- *Buckling of a simply supported square plate*
- *Lateral buckling of an L-bracket*
- *Buckling of a column with general contact*

Buckling analysis of beams

Products: Abaqus/Standard

Overview

This example shows how Abaqus can be applied to the buckling analysis of beams.

Buckling studies such as this usually require two types of analyses.

Eigenvalue analysis is used to obtain estimates of the buckling loads and modes. The concept of eigenvalue buckling prediction is to investigate singularities in a linear perturbation of the structure's stiffness matrix. The resulting estimates will be of value in design if the linear perturbation is a realistic reflection of the structure's response before it buckles. For this to be the case, the structural response should be linear elastic. In other words, eigenvalue buckling is useful for "stiff" structures (structures that exhibit only small, elastic deformations prior to buckling). Such analysis is performed using the eigenvalue buckling procedure (*Eigenvalue Buckling Prediction*), with the "live" load applied within the step. The buckling analysis provides the factor by which the live load must be multiplied to reach the buckling load. Any preload must be added to the load from the eigenvalue buckling step to compute the total collapse load.

It is usually also necessary to consider whether the postbuckling response is stable or unstable and if the structure is imperfection sensitive. In many cases the postbuckled stiffness may not be positive. The collapse load will then depend strongly on imperfections in the original geometry ("imperfection sensitivity"). This is addressed by following the eigenvalue prediction with a load-displacement analysis of the structure. Typically this is done by assuming an imperfection in the original geometry, in the shape of the buckling mode, and studying the effect of the magnitude of that imperfection on the response. Material nonlinearity is often included in such collapse studies. This example illustrates these analyses for some simple, classical, beam problems.

Problem description

The objectives for this example include the study of buckling under the action of axial and transverse loads. Such studies are usually classified as follows:

1. Flexural buckling of axially compressed beams in flexural modes (Euler buckling).
2. Lateral buckling of beams that are loaded transversely in the plane of higher flexural rigidity. This is of importance in the design of beams without lateral supports in which the bending stiffness of the beam in the plane of loading is large in comparison with the lateral flexural rigidity. The plane configuration of the beam becomes unstable if the load is increased beyond the critical value.
3. Torsional buckling of beams subjected to uniform axial compression in torsional modes while their longitudinal axis remains straight. In general, torsional buckling is important for thin-walled columns having wide flanges and short lengths.

A column may buckle in any one of these modes. Only the lowest value is of practical interest in design calculations. In general cases, buckling failure may occur by a combination of torsion and bending, which is best addressed by a load-displacement study.

We consider slender, elastic straight beams, orientated along the x-axis, all with the I-section shown in *Figure 1*. The section dimensions are suitable for the study of flexural, lateral, and torsional instability problems. The beam is assumed to be made up of an isotropic material with Young's modulus 211 GPa and Poisson's ratio of 0.3125. The mesh consists of 20 B31OS or 10 B32OS beam elements spanning the 12 m length of the beam. This discretization should give good accuracy for the first several modes of buckling. Mesh convergence studies are not reported here.

A cantilever beam is considered for the Euler buckling problem. All degrees of freedom are restrained at the clamped end of the beam. The input data are shown in *beambuckle_b31os_isec_flex.inp*. An interesting extension of this buckling problem is to examine the response of the column far into the postbuckling range. This is the simplest of the classical "elastica" problems, an elastica being an elastic curve bent by some load (see Timoshenko and Gere, 1961). For this study an initial imperfection in the shape of the lowest buckling mode, with a peak magnitude of 10% of the beam thickness, is introduced. The Riks technique is used. An axial force, equal in magnitude to the critical load, is applied, and the analysis is stopped when the axial force becomes six times the applied load.

All components of displacement, and the rotation about the *x*-axis, are restrained at one of the support nodes for the lateral/torsional buckling problems. Displacements in the *y*- and *z*-directions, and rotation about the *x*-axis, are restrained at the other support node. Beam sections are tested using an I-section and an arbitrary section type. General beam sections are tested using an I-section, an arbitrary section, and a general section type. (A general beam section with linear response in combination with the open section beam elements requires that the warping constants be specified.)

beambuckle_b31os_isec_lat.inp shows the input data used for the eigenvalue buckling analysis. The distributed load is applied as load type PZ, with a magnitude of 1 N/m. A load-displacement analysis is then performed, with collapse being defined by large motion occurring under very small load increments. The model used must provide for switching to the buckling mode. A slight initial imperfection is used for this purpose. The first mode from the eigenvalue buckling analysis is scaled to have a maximum rotation equal to 1% of the flange width. The translational displacements are equally scaled and added to the nodal coordinates to define the perturbed or imperfect geometric data. The normal at each node is defined based upon the scaled rotations from the eigenvalue analysis. Since instabilities are expected, the Riks method is used. The analysis is terminated when the lateral displacement (u_z) of the middle node is greater than the flange width of the beam. The input for this load-displacement analysis is shown in *beambuckle_b31os_arbsec_lat.inp*.

The model used for the eigenvalue torsional buckling analysis is the same as that used for the lateral buckling analysis. Here, a concentrated axial load of 10 N is applied to one end of the beam. *beambuckle_b31os_tors_gsec.inp* shows the input used for this analysis.

Results and discussion

The critical flexural buckling load for mode *n*, as given by Timoshenko and Gere (1961), is

$$(P)_{\text{crf}} = \frac{\pi^2 EI}{4l^2}(2n-1)^2,$$

where *E* is Young's modulus, *I* is the moment of inertia, and *l* is the length of the beam.

The buckling load estimates provided by Abaqus are shown in *Table 1*. For practical purposes only the lowest mode is of significance, and a coarser mesh than used here would give that mode accurately.

For the elastica problem, the *x* and *y* positions of the tip of the column are shown as functions of the load in *Figure 2*. The deformed shape of the column is plotted in *Figure 3*.

The critical lateral buckling load is given by Timoshenko and Gere (1961) as

$$(q)_{\text{cr}} = \frac{\gamma_4\sqrt{EI_{22}GJ}}{l^3},$$

where *E* is Young's modulus, *G* is the shear modulus, *l* is the length of the beam, $I_{22} = \int_A x_1^2 dA$, and γ_4 is a dimensionless factor dependent upon the loading and on the ratio $\left(l^2GJ\right)/\left(E\Gamma_w\right)$, where Γ_w is the warping constant

$$\Gamma_w = \frac{t_f(h-t_f)^2b^3}{24},$$

and J is the torsion constant

$$J = \frac{2bt_f^3 + (h - t_f)t_w^3}{3}.$$

Here t_f is the thickness of the flange, t_w is the thickness of the web, h is the height of the cross-section, and b is the width of the flange. For our model, this gives a critical load of 62.5 N/mm. The eigenvalue buckling analysis with 20 linear open section beam elements predicts a critical load of 62.47 N/mm. The load-displacement analysis shows a severe loss of stiffness at a load very close to the expected critical value, as shown in *Figure 4*.

The critical torsional buckling load for mode n is given by Timoshenko and Gere (1961) as

$$(P)_{cr_T} = \frac{A}{I_0}\left(GJ + \frac{n^2\pi^2}{l^2}E\Gamma_w\right),$$

where A is the cross-sectional area and I_0 is the polar moment of inertia of the cross-section about the shear center. The torsional buckling load estimates provided by Abaqus are shown in *Table 2*.

Input files

beambuckle_b31os_isec_flex.inp	Element B31OS with **BEAM SECTION*, SECTION=I for the flexural eigenvalue buckling prediction.
beambuckle_b31os_isec_lat.inp	Element B31OS with **BEAM SECTION*, SECTION=I for the lateral eigenvalue buckling analysis.

Using the Lanczos solver

beambuckle_b31os_lanczos.inp	Same as beambuckle_b31os_isec_flex.inp, except that it uses **FREQUENCY*, EIGENSOLVER=LANCZOS for the eigenvalue buckling analysis in the given ranges.

Lateral buckling load-displacement analysis

beambuckle_b31os_load_isec.inp	Element B31OS with **BEAM SECTION*, SECTION=I.
beambuckle_b31os_dload_isec.inp	Element B31OS with **BEAM SECTION*, SECTION=I and pressure load.
beambuckle_b31os_arbsec_lat.inp	Element B31OS with **BEAM SECTION*,SECTION=ARBITRARY.
beambuckle_b31os_load_gseci.inp	Element B31OS with **BEAM GENERAL SECTION*, SECTION=I.
beambuckle_b31os_load_arbsec.inp	Element B31OS with **BEAM GENERAL SECTION*, SECTION=ARBITRARY.
beambuckle_b31os_load_gsecg.inp	Element B31OS with **BEAM GENERAL SECTION*, SECTION=GENERAL.
beambuckle_b32os_load_isec.inp	Element B32OS with **BEAM SECTION*, SECTION=I.
beambuckle_b32os_load_arbsec.inp	Element B32OS with **BEAM SECTION*,SECTION=ARBITRARY.
beambuckle_b32os_load_gseci.inp	Element B32OS with **BEAM GENERAL SECTION*, SECTION=I.
beambuckle_b32os_load_garbsec.inp	Element B32OS with **BEAM GENERAL SECTION*, SECTION=ARBITRARY.

beambuckle_b32os_load_gsecg.inp	Element B32OS with **BEAM GENERAL SECTION*, SECTION=GENERAL.

Torsional eigenvalue buckling analysis

beambuckle_b31os_tors_isec.inp	Element B31OS with **BEAM SECTION*, SECTION=I.
beambuckle_b31os_tors_gsec.inp	Element B31OS with **BEAM GENERAL SECTION*.
beambuckle_b31os_tors_gseci.inp	Element B31OS with **BEAM GENERAL SECTION*, SECTION=I.
beambuckle_b32os_tors_isec.inp	Element B32OS with **BEAM SECTION*, SECTION=I.

Elastica study

beambuckle_b21_elastica.inp	Element B21 with **BEAM GENERAL SECTION*, SECTION=GENERAL.
beambuckle_b21h_elastica.inp	Element B21H with **BEAM GENERAL SECTION*, SECTION=GENERAL.
beambuckle_b22_elastica.inp	Element B22 with **BEAM GENERAL SECTION*, SECTION=GENERAL.
beambuckle_b22h_elastica.inp	Element B22H with **BEAM GENERAL SECTION*, SECTION=GENERAL.
beambuckle_b23_elastica.inp	Element B23 with **BEAM GENERAL SECTION*, SECTION=GENERAL.
beambuckle_b23h_elastica.inp	Element B23H with **BEAM GENERAL SECTION*, SECTION=GENERAL.
beambuckle_b31_elastica.inp	Element B31 with **BEAM SECTION*, SECTION=I.
beambuckle_b31h_elastica.inp	Element B31H with **BEAM SECTION*, SECTION=I.
beambuckle_b31os_elastica.inp	Element B31OS with **BEAM SECTION*, SECTION=I.
beambuckle_b31osh_elastica.inp	Element B31OSH with **BEAM SECTION*,SECTION=I.
beambuckle_b32_elastica.inp	Element B32 with **BEAM SECTION*, SECTION=I.
beambuckle_b32h_elastica.inp	Element B32H with **BEAM SECTION*, SECTION=I.
beambuckle_b32os_elastica.inp	Element B32OS with **BEAM SECTION*, SECTION=I.
beambuckle_b32osh_elastica.inp	Element B32OSH with **BEAM SECTION*,SECTION=I.
beambuckle_b33_elastica.inp	Element B33 with **BEAM SECTION*, SECTION=I.
beambuckle_b33h_elastica.inp	Element B33H with **BEAM SECTION*, SECTION=I.
beambuckle_pipe21_elastica.inp	Element PIPE21 with **BEAM SECTION*,SECTION=PIPE.
beambuckle_pipe21h_elastica.inp	Element PIPE21H with **BEAM SECTION*,SECTION=PIPE.
beambuckle_pipe22_elastica.inp	Element PIPE22 with **BEAM SECTION*,SECTION=PIPE.
beambuckle_pipe22h_elastica.inp	Element PIPE22H with **BEAM SECTION*,SECTION=PIPE.
beambuckle_pipe31_elastica.inp	Element PIPE31 with **BEAM SECTION*,SECTION=PIPE.
beambuckle_pipe31h_elastica.inp	Element PIPE31H with **BEAM SECTION*,SECTION=PIPE.
beambuckle_pipe32_elastica.inp	Element PIPE32 with **BEAM SECTION*,SECTION=PIPE.
beambuckle_pipe32h_elastica.inp	Element PIPE32H with **BEAM SECTION*,SECTION=PIPE.

References

- Timoshenko, S. P., and J. M. Gere, Theory of Elastic Stability, 2nd Edition, McGraw-Hill, New York, 1961.

Tables

Table 1: Flexural buckling load estimates (values given in MN).

Eigenvector	Estimated	Theoretical	Direction $(n)^*$
	buckling load	buckling load	
1	0.4371	0.4398	*y* (1)
2	3.9267	3.9587	*y* (2)
3	7.4575	7.5182	*z* (1)
4	10.8670	10.9965	*y* (3)
5	21.1796	21.5530	*y* (4)
6	34.7394	35.6285	*y* (5)
7	51.3717	53.2228	*y* (6)
8	63.0448	67.6640	*z* (2)
9	70.8435	74.3360	*y* (7)
10	92.8553	98.9680	*y* (8)

$^*n =$ number of half sine waves

Table 2: Flexural and torsional buckling load estimates (values given in MN).

Eigenvector	Estimated	Theoretical	Mode (*n*)
	buckling load	buckling load	
1	1.7544	1.7704	Flexural - *y* (1)
2	6.4235	6.4134	Torsional (1)
3	7.0577	7.0814	Flexural - *y* (2)
4	13.1363	13.0300	Torsional (2)
5	16.0307	15.9330	Flexural - *y* (3)
6	24.5735	24.0590	Torsional (3)
7	28.8769	28.3260	Flexural - *y* (4)
8	29.7522	30.1110	Flexural - *z* (1)
9	41.1234	39.4980	Torsional (4)
10	45.8840	44.2590	Flexural - *y* (5)

Figures

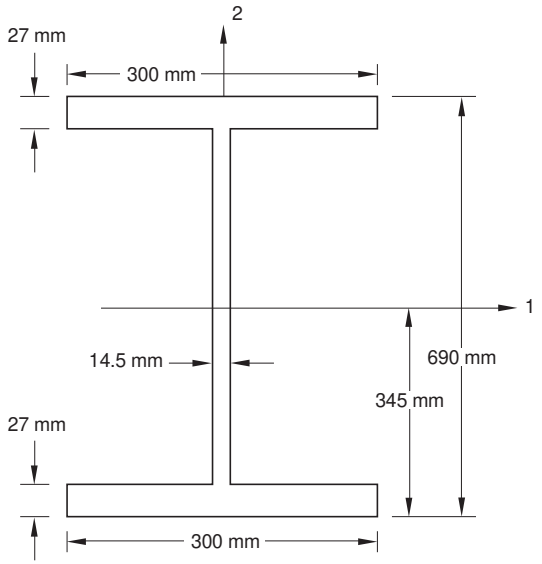


Figure 1: Beam cross-section details.

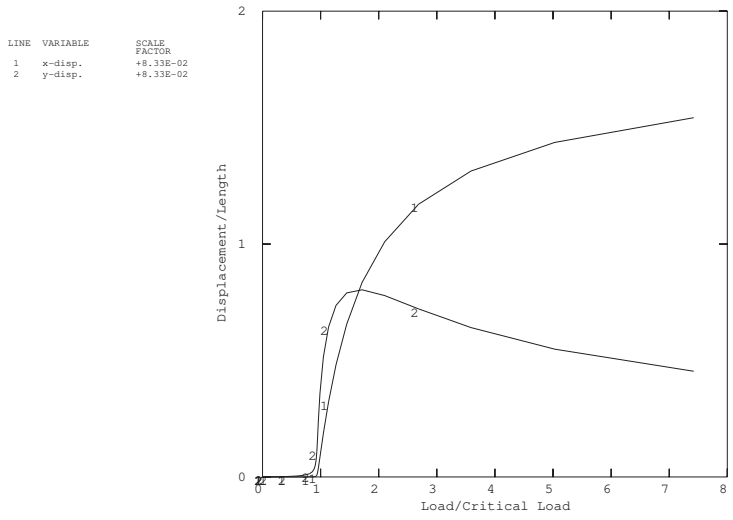


Figure 2: Elastica results.

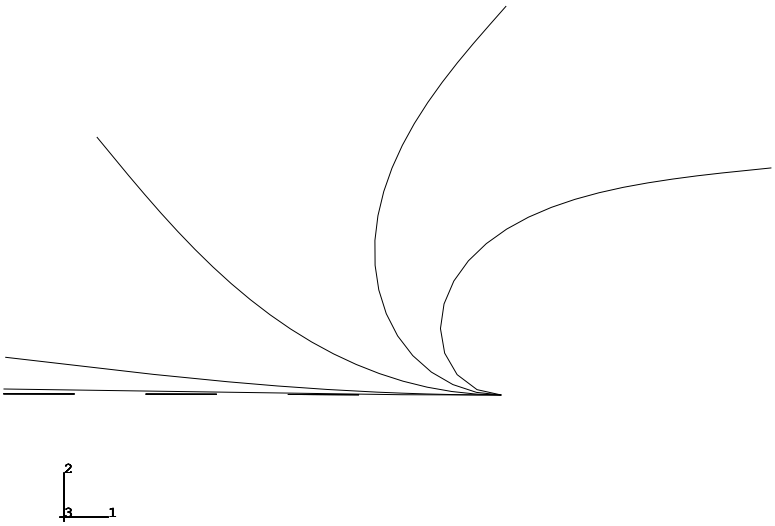


Figure 3: Progressive deformed configurations of elastica.

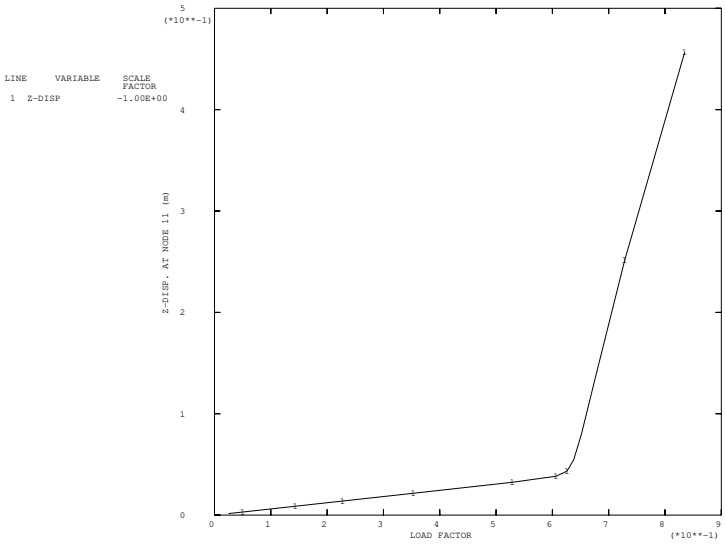


Figure 4: Load versus deflection curve for lateral buckling problem.

Buckling of a ring in a plane under external pressure

Products: Abaqus/Standard

Overview

This example shows the buckling of an elastic ring under axisymmetric pressure.

A particularly simple and interesting example of the asymmetric buckling of an axisymmetric structure under axisymmetric loading is the buckling of a thin, elastic ring under external pressure. The problem is interesting because the buckling load is strongly influenced by the follower force nature of the pressure: if this effect is neglected (the "radial loading" case), the prediction of the critical buckling load will be too high—Boresi (1955) shows that the error can be as much as 50% for very thin rings.

In problems of this geometric type the prebuckled deformation is axisymmetric (assuming no imperfections), while the buckling occurs as deformation in a periodic mode with respect to angular position:

$$w = A\cos k\theta,$$

where w is the radial displacement of a point at angular position θ, A is some arbitrary magnitude, and k is the mode number, $k = 2,3,4....$ Eigenvalue buckling load estimates are useful in design in such a case, because they are quite accurate if the structure is not very sensitive to imperfections. The buckling deformation can be arbitrarily chosen to be symmetric about $\theta = 0$ and will then be antisymmetric about $\theta = \pi/2k$.

For this case we know the lowest buckling mode ($k = 2$) has the smallest critical load, so a mesh of 45° extent should suffice for eigenvalue buckling estimation. This requires symmetric boundary conditions at $\theta =$ 45° during loading, but antisymmetry at $\theta =$ 45° during eigenvalue solution. This is easily accomplished with Abaqus, as shown below.

Following the eigenvalue buckling estimation, imperfection sensitivity is studied by introducing an imperfection into the radius in the form of the lowest buckling mode:

$$R = R_0 + A\cos 2\theta,$$

where R_0 is the radius of the perfect ring. The magnitude of the imperfection is usually chosen in the range of 1% to 10% of the thickness of the ring and the load-displacement response obtained. These results then show the sensitivity of the response to such an imperfection. For load-displacement analysis the antisymmetry condition no longer applies, since the response is no longer a pure bifurcation. As a result of this, a 90° model with symmetry conditions at both ends must be used.

Problem description

The problem is shown in *Figure 1*. The ring has a mean radius of 2.54 m (100 in), with a square cross-section of 25.4 × 25.4 mm (1 × 1 in). The material is assumed to be linear elastic, with Young's modulus of 206.8 GPa (30×10^6 lb/in^2) and Poisson's ratio 0.0. The ring is loaded by uniform external pressure.

Element choice

The obvious element choice for this case is a beam in a plane. Element types B21 and B22 are, therefore, used. For purposes of verification, the analyses are also done with shell elements S8R, S8R5, S9R5, STRI65 and STRI3. The axisymmetric elements with nonaxisymmetric deformation are ideally suited for this problem. Results are reported for shell elements SAXA1*n* and SAXA2*n* and continuum elements CAXA8*n* and CAXA8H*n* (*n* = 2, 3 or 4), where *n* is the number of Fourier modes used in the element. The lowest-order Fourier mode

possible for this problem is $n = 2$, since the buckling shape has a $\cos 2\theta$ circumferential displacement. Higher-order modes can be used, but they do not alter the solution.

Eigenvalue buckling load estimates

Several meshes are used for the eigenvalue buckling load estimates: three or five elements of type B21 in 45°; three B22 elements; one or two shell elements of type S8R, S8R5, S9R5; five or ten elements of type STRI3; three or six elements of STRI65; one element of type SAXA12 or SAXA22; and one element of type CAXA82 or CAXA8H2.

In all models symmetry boundary conditions are used at $\theta = 0$. Except for the SAXA and CAXA models, at $\theta = 45°$ a local coordinate system definition is used to obtain a local system with local x' radial to the ring and local y' tangential to the ring. In that local system the boundary conditions are symmetric ($u_{y'} = \theta_{x'} = \theta_{z'} = 0$) during load application and antisymmetric ($u_{x'} = u_{z'} = \theta_{y'} = 0$) during eigenvalue extraction.

In the SAXA and CAXA model the rigid body mode in the global x-direction is eliminated by forcing the radial displacements at a node in the $\theta = 0°$ plane and at the corresponding node in the $\theta = 180°$ plane to be identical with an equation constraint.

Eigenvalue buckling estimates are obtained by using the eigenvalue buckling procedure (*Eigenvalue Buckling Prediction*). This is a linear perturbation procedure in which the current stiffness is calculated using the rules for linear perturbation analysis. The stiffness matrix associated with the external pressure load is calculated. For a linear perturbation analysis, the magnitude of the pressure is immaterial, since the stiffness is proportional to the pressure. (A magnitude of 6895 Pa, 1 lb/in^2 is used.) Since deformation due to the pressure load is a uniform compression, except for the SAXA and CAXA models, symmetric boundary conditions are applied. For the eigenvalue buckling analysis we need to specify symmetric boundary conditions at $\theta = 0$ but antisymmetric at $\theta = 45°$. This is done by a complete specification of the buckling mode boundary conditions. If a second set of boundary conditions is specified this way, it is used during the buckling analysis. These boundary condition changes are not needed for the CAXA and SAXA elements. Only one eigenmode is requested, since the 45° sector has been chosen based on it being able to represent the lowest mode. Higher modes would require a different sector.

The reference solution to this problem, which assumes a very small ring thickness relative to the radius, is a critical pressure of $p_{cr} = \frac{E}{4(1-\nu)}\left(\frac{a}{R}\right)^3$, where E is Young's modulus, ν is Poisson's ratio, a is the ring thickness, and R is the mean radius, so that with the data chosen here the critical pressure is 0.05171 MPa (7.5 lb/in^2). The solutions obtained with the various Abaqus models are shown in *Table 1*. Except for the coarsest models, all of the models give the critical pressure quite accurately.

Results and discussion

The load-displacement response prediction requires 90° models, since the pure symmetry or antisymmetry condition at 45° is no longer valid. Meshes of five B21 beams and of two and three shell elements in a 90° arc are, therefore, used. A model of perfectly circular geometry is not useful, since it has no basis to switch into the postbifurcation mode. Various methods are commonly adopted to overcome this problem. Most typically some slight imperfection is introduced into the geometry. This imperfection may be random or may be chosen in the shape of the most critical buckling mode predicted by the eigenvalue analysis. The latter method is used here: presumably an imperfection in the shape of the lowest mode would be the most critical, so this seems to be a rational basis for investigating the sensitivity of the structure to imperfections. Thus, we generate the model with a radius

$$R = R_0 + A\cos 2\theta,$$

where R_0 is the nominal radius (2.54 m, 100 in) and A is the imperfection magnitude. This magnitude is chosen as 10%, 1%, and as 0.1% of the thickness. These values are all very small compared to the radius of the ring.

First, we compare the different models. *Figure 2* shows the response of the three different meshes for the 10% initial imperfection case. The two shell models are consistent, while the five-element beam model gives a stiffer response as it buckles. This is to be expected, since the beam element chosen uses linear interpolation. A finer mesh, or use of higher-order beam elements (B22 or B23), would probably improve the results.

The different imperfection magnitudes are compared in *Figure 3*, based on the two-shell-element model (since that model seems adequate from the above comparison). The figure shows the expected behavior: as the imperfection magnitude is reduced, the response becomes less smooth, with a sudden, sharp transition especially evident in the smallest imperfection modeled occurring at the load value predicted by the eigenvalue approach.

For the CAXA and SAXA elements an initial geometric imperfection is not possible, and results for these elements are not reported. Load-displacement results could be obtained, however, by introducing an initial imperfection in the loading.

Input files

ringbuckling_b21_buckle.inp	Eigenvalue buckling input data for B21.
ringbuckling_b21_static.inp	Static collapse input data for B21, where the imperfect mesh is generated by the ringbuckling_b21_meshgen.f Fortran program.
ringbuckling_b21_meshgen.f	Fortran program used to generate the mesh for ringbuckling_b21_static.inp.
ringbuckling_b21_5el.inp	Eigenvalue buckling input data for the five-element B21 model.
ringbuckling_b22_3el.inp	Eigenvalue buckling input data for the three-element B21 model.
ringbuckling_s8r_1el.inp	Eigenvalue buckling data for S8R, 1-element model.
ringbuckling_s8r_2el.inp	Eigenvalue buckling data for S8R, 2-element model.
ringbuckling_s8r_static.inp	Static collapse input data for S8R, where the imperfect mesh is generated by the ringbuckling_s8r_meshgen.f Fortran program.
ringbuckling_s8r_meshgen.f	Fortran program used to generate the mesh for ringbuckling_s8r_static.inp.
ringbuckling_s8r5_1el.inp	Eigenvalue buckling data for S8R5, 1-element model.
ringbuckling_s8r5_2el.inp	Eigenvalue buckling data for S8R5, 2-element model.
ringbuckling_s9r5_1el.inp	Eigenvalue buckling data for S9R5, 1-element model.
ringbuckling_s9r5_2el.inp	Eigenvalue buckling data for S9R5, 2-element model.
ringbuckling_stri3_5by2.inp	Eigenvalue buckling input data for STRI3, 5 × 2 mesh.
ringbuckling_stri3_10by2.inp	Eigenvalue buckling input data for STRI3, 10 × 2 mesh.
ringbuckling_stri65_2el.inp	Eigenvalue buckling input data for STRI65, 2-element model.
ringbuckling_stri65_6el.inp	Eigenvalue buckling input data for STRI65, 6-element model.
ringbuckling_caxa82.inp	Eigenvalue buckling input data for CAXA82.
ringbuckling_caxa8h2.inp	Eigenvalue buckling input data for CAXA8H2.
ringbuckling_saxa12.inp	Eigenvalue buckling input data for SAXA12.
ringbuckling_saxa22.inp	Eigenvalue buckling input data for SAXA22.

References

- Boresi, A. P., “A Refinement of the Theory of Buckling of Rings Under Uniform Pressure,” Journal of Applied Mechanics, vol. 77, pp. 99–102, 1955.

Tables

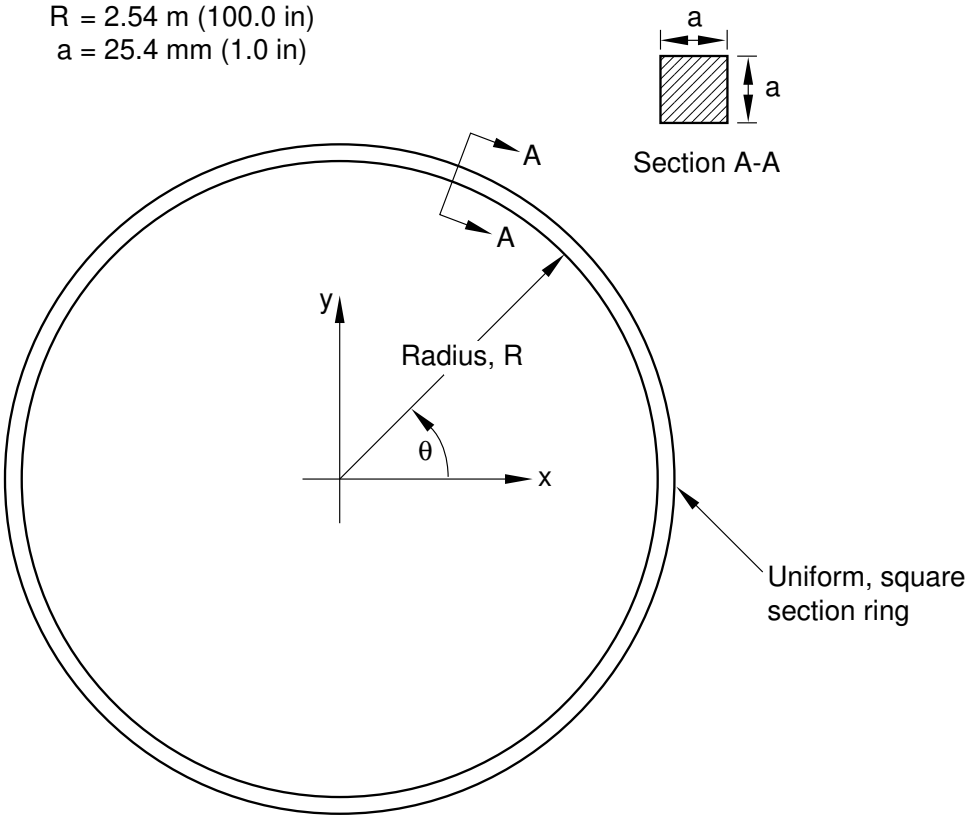
Table 1: Eigenvalue buckling estimates.

Element type	Number of elements in 45°	Critical pressure estimate (MPa)	Critical pressure estimate (lb/in^2)	Error
B21	3	0.0538	7.796	4.0%
	5	0.0524	7.605	1.4%
B22	3	0.0517	7.501	0.1%
S8R	1	0.0523	7.587	1.2%
	2	0.0517	7.505	0.1%
STRI3	5	0.0524	7.606	1.4%
	10	0.0519	7.526	0.3%
STRI65	3	0.0530	7.693	3.8%
	6	0.0517	7.505	0.1%
S8R5	1	0.0537	7.786	3.8%
	2	0.0519	7.523	0.3%
S9R5	1	0.0537	7.786	3.8%
	2	0.0519	7.523	0.3%
SAXA12	1 in 180°	0.0517	7.499	0.01%
SAXA22	1 in 180°	0.0517	7.499	0.01%
CAXA82	1 in 180°	0.0517	7.499	0.01%
CAXA8H2	1 in 180°	0.0517	7.499	0.01%

Figures

Geometry values used:

R = 2.54 m (100.0 in)
a = 25.4 mm (1.0 in)



Material: linear elastic

Young's modulus = 206.8 GPa
(30.0 x 10^6 lb/in^2)
Poisson's ratio = 0.0

Loading: uniform external pressure

Figure 1: Ring buckling problem.

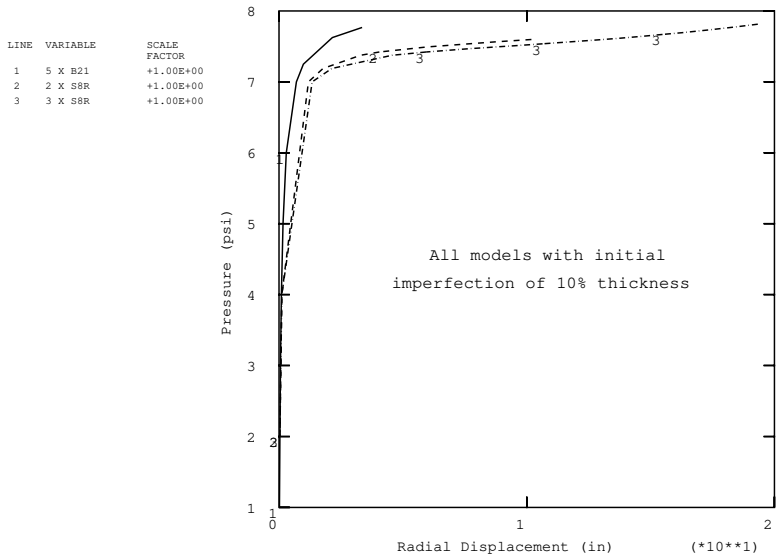


Figure 2: Pressure-displacement response for ring.

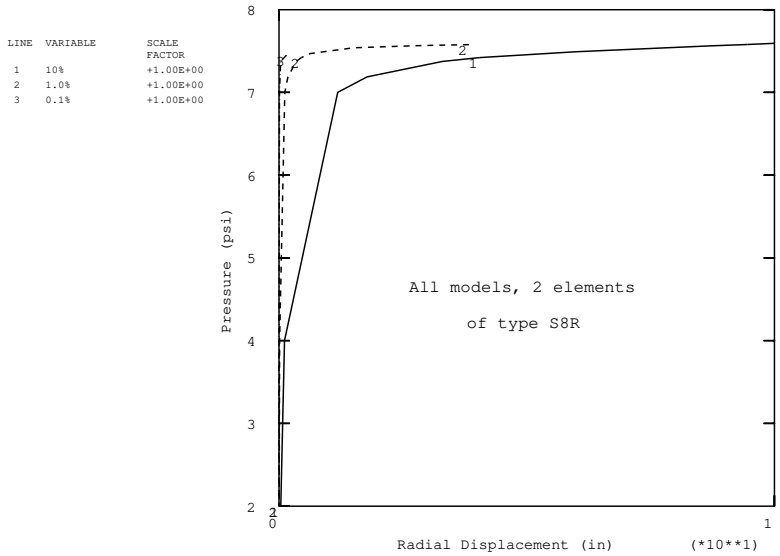


Figure 3: Pressure-displacement response with various initial imperfections.

Buckling of a cylindrical shell under uniform axial pressure

Products: Abaqus/Standard

Overview

This example illustrates the use of Abaqus to predict the elastic buckling instability of a “stiff” structure (a structure that exhibits only small, elastic deformations prior to buckling).
The example is a classic case of this type of problem; a detailed analytical discussion of the problem is available in Timoshenko and Gere (1961). This analytical solution allows the example to be used for verification of the numerical results.

The structural analyst often encounters problems involving stability assessment, especially in the design of efficient shell structures. Since the shell is usually designed to carry the loading primarily as a membrane, its initial response is stiff; that is, it undergoes very little deformation. If the membrane state created by the external loading is compressive, there is a possibility that the membrane equilibrium state will become unstable and the structure will buckle. Since the shell is thin, its bending response is much less stiff than its membrane response. Such buckling will result in very large deflections of the shell (even though the postbuckling response may be mathematically stable in the sense that the structure's stiffness remains positive). In many cases the postbuckled stiffness is not positive; in such cases the collapse load generally will depend strongly on imperfections in the original geometry; that is, the structure is “imperfection sensitive.” In some cases the buckling may be only a local effect in the overall response: the shell may subsequently become stiffer again and reach higher load levels usefully with respect to its design objective. Sometimes there are many collapse modes into which the shell may buckle. For all of these reasons shell collapse analysis is challenging. This example illustrates the standard numerical approach to such problems: eigenvalue estimation of bifurcation loads and modes, followed by load-deflection analysis of a model that includes imperfections.

Problem description

The problem consists of a long, thin, metal cylinder that is simply supported in its cross-section and loaded by a uniformly distributed compressive axial stress at its ends (*Figure 1*). The cylinder is sufficiently thin so that buckling occurs well below yield. (When buckling occurs in the plastic range, the problem can generally be studied numerically only by load-deflection analysis of models that include initial imperfections. The sudden change of deformation mode at collapse causes the elastic-plastic response to switch from elastic to yielding in some parts of the model and from yielding to elastic unloading at other points. Eigenvalue bifurcation predictions are then useful only as guidance for mesh design.)

In the particular case studied, the cylinder length is 20.32 m (800 in), the radius is 2.54 m (100 in), and the shell thickness is 6.35 mm (0.25 in). Thus, the radius to thickness ratio for the shell is 400:1.

The shell is made of an isotropic material with Young's modulus of 207 GPa (30×10^6 lb/in^2) and Poisson's ratio of 0.3.

Analysis procedure

In general, shell buckling stability studies require two types of analysis. First, eigenvalue analysis is used to obtain estimates of the buckling loads and modes. Such studies also provide guidance in mesh design because mesh convergence studies are required to ensure that the eigenvalue estimates of the buckling load have converged: this requires that the mesh be adequate to model the buckling modes, which are usually more complex than the prebuckling deformation mode. Using a mesh and imperfections suggested by the eigenvalue analysis, the second phase of the study is the performance of load-displacement analyses, usually using the Riks method to handle possible instabilities. These analyses would typically study imperfection sensitivity by perturbing the perfect

geometry with different magnitudes of imperfection in the most important buckling modes and investigating the effect on the response.

Eigenvalue buckling prediction

The key aspect of the eigenvalue analysis is the mesh design. For the particular problem under study we know that the critical buckling mode will be a displacement pattern with n circumferential waves (*Figure 2* shows a cross-section with $n = 2$ and $n = 3$) and m longitudinal half-waves, and we need to determine the values of n and m that represent the lowest critical stress. One approach would be to model the whole cylinder with a very fine mesh and to assume that we can then pick up the most critical mode. This approach would be computationally expensive and is not needed in this case because of the symmetry of the initial geometry. We need to model only one-quarter of a circumferential wave: the combination of symmetry boundary conditions at one longitudinal edge of this circumferential slice and antisymmetry boundary conditions at the other longitudinal edge during the eigenvalue extraction allows this quarter-wave model to represent the entire cylinder in the circumferential direction. A quarter-wave circumferentially subtends an angle of $\alpha = \pi/2n$. Likewise, we need only model one-half of the axial length, using either symmetry or antisymmetry at the midplane, depending on whether we are looking for even or odd modes m.

With this approach it is necessary to perform several analyses using different values of α and symmetry or antisymmetry at the midplane, instead of a single analysis with a very large model. Several small analyses are generally less expensive than one large analysis, since the computational costs rise rapidly with model size. In this particular example we consider the variation of α in the range of $\pi/6$ to $\pi/20$, corresponding to the range $n = 3$ to $n = 10$. We do not consider the cases of $n = 1$ and $n = 2$ because we know that these will not give lower critical loads.

The mesh chosen for the analysis of such a segment of the cylinder, using element type S4R5, is shown in *Figure 3*. Similar meshes with element types S4R, STRI3, STRI65, and S9R5 are also used. For the triangular elements each quadrilateral shown in *Figure 3* is divided into two triangles. The meshes using element types S9R5 and STRI65 have half the number of elements in the circumferential and axial directions as the meshes using the lower-order elements. No mesh convergence studies have been done, but all the meshes and elements used give reasonably accurate predictions of the critical load.

Eigenvalue buckling analysis is performed with Abaqus by first storing the stiffness matrix at the state corresponding to the "base state" loading on the structure, then applying a small perturbation of "live" load. The initial stress matrix resulting from the live load is calculated, and then an eigenvalue calculation is performed to determine a multiplier to the "live" load at which the structure reaches instability. In this example there is no load prior to the "live" load; therefore, the eigenvalue buckling (*Eigenvalue Buckling Prediction*) is the only step. During the buckling procedure one longitudinal edge has symmetry boundary conditions, and the other has antisymmetry boundary conditions, as shown in *Figure 3*. With these constraints a mesh subtending an angle of α can model modes with $n = \pi/2\alpha, \pi/\alpha, 2\pi/\alpha \ldots$ waves around the circumference of the cylinder. However, during the calculation of the initial stress matrix, both longitudinal edges must have symmetric boundary conditions (because the prebuckling response that creates this stress stiffness is symmetric). Thus, the boundary conditions associated with the "live" loading are specified under one boundary condition, and the boundary conditions associated with the buckling deformation are defined under a second boundary condition. If the second definition is not given, the boundary conditions are the same for the loading and for the buckling mode calculation. The loaded edge is simply supported. Since the number of longitudinal half-waves m can have odd or even values, the midlength edge is alternately modeled with symmetry and antisymmetry boundary conditions.

Load-displacement analysis on imperfect geometries

The example is continued by performing an incremental load-deflection analysis using the modified Riks method. For some problems linear eigenvalue analysis is sufficient for design evaluation, but if there is concern about material nonlinearity, geometric nonlinearity prior to buckling, or unstable postbuckling response (with associated

imperfection sensitivity), the analyst generally must perform a load-deflection analysis to investigate the problem further.

The mesh used for this phase of the analysis consists of eight rows of elements of type S4R5 in the circumferential direction between symmetry lines. (In the eigenvalue analysis antisymmetry boundary conditions are used, since the analysis is a linear perturbation method. But this load-deflection study allows fully nonlinear response, so the antisymmetry assumption is no longer correct.) Twenty elements are used along the length of the cylinder.

An imperfection in the form of the critical buckling mode (obtained in the previous analyses of the example) is assumed to be the most critical. The mesh is, therefore, perturbed in the radial direction by that eigenmode, scaled so that the largest perturbation is a fraction of the shell thickness. The studies reported here use perturbations of 1%, 10%, and 100% of the thickness. The following examples demonstrate two methods of introducing the imperfection.

The first method makes use of the model antisymmetry and defines the imperfection by means of a Fortran routine that is used to generate the perturbed mesh, using the data stored on the results file written during the eigenvalue buckling analysis. *bucklecylshell_stri3_n4.inp* shows the input data for the buckling prediction, *bucklecylshell_progpert.f* shows the Fortran routine used to generate the nodal coordinates of the perturbed mesh, and *bucklecylshell_postbucklpert.inp* shows the input data for the postbuckling analysis. The meshes for the buckling prediction analysis and the postbuckling analysis are different and are described in the "Input Files" section. The postbuckling analysis is performed using the static RIKS procedure (*Unstable Collapse and Postbuckling Analysis*).

The second method uses a geometric imperfection to define the imperfection, which requires that the model definitions for the buckling prediction analysis and the postbuckling analysis be identical. *bucklecylshell_s4r5_n1.inp* shows the input data for the buckling prediction, and *bucklecylshell_postbucklimperf.inp* shows the input data for the postbuckling analysis.

Results and discussion

The results for both analyses are discussed below.

Eigenvalue buckling prediction

The analytical solution given by Timoshenko and Gere assumes that the buckling eigenmode has n lobes or waves circumferentially and m half-waves longitudinally and provides a critical stress value for each combination of m and n. The mode that gives the minimum critical stress value will be the primary buckling mode of the shell: which mode is critical depends on the thickness, radius, and length of the cylinder. For the particular case studied here, the dependency of the critical stress values on m and n is illustrated in *Figure 4*: each node on the surface represents a possible buckling mode. *Table 1* shows the numerical values of these critical stresses for a number of mode shapes. For this geometry the minimum critical stress corresponds to a mode shape defined by $m = 1$ and $n = 4$; that is, one half-wave along the cylinder and four full waves around the circumference. *Figure 5* shows the (1, 4) buckling mode shape predicted with the mesh of S4R5 elements.

The $m = 1, n = 0$ mode corresponds to buckling of the cylindrical shell as an Euler column: for this mode the critical stress is more than 250 times the critical stress for $m = 1, n = 4$. For small numbers of axial half-waves (m) the critical stress changes rapidly with respect to the number of circumferential lobes (n). However, for higher values of m and n the critical stresses are not very much higher than the critical stress for $m = 1, n = 4$ and do not vary much from mode to mode, as can be observed in *Figure 4* and *Table 1*. This behavior exhibits itself in the finite element solutions, as shown—for example—in *Table 2*, where the results for element type S9R5 are given and compared to the analytical results of Timoshenko and Gere. The mode numbers (values of n and m) given in that table are estimated visually from inspection of deformed configuration plots of the eigenmodes. In several cases no identification is given (the mode number is listed as ``*''), because the mesh is too coarse to define any mode. As an example, consider the mesh for $\alpha = \pi/6$, which allows for an odd number of half-waves in the longitudinal direction. This mesh can yield eigenvectors that correspond to the (m, n) mode

shapes (3,1), (3,3), (3,5), . . . or (6,1), (6,3), (6,5), . . . However, as described earlier, the eigenvalues do not show an ascending pattern with the number of lobes either in the circumferential or longitudinal direction because of the geometry of this problem. Abaqus will estimate the eigenvalues in ascending order, from the closest eigenvalue to zero, unless a shift point is defined. For this case the analytical solution shows that the lower-order modes (among those that can be represented by the mesh) have very large eigenvalues: the eigenvalues reduce steadily as the number of longitudinal half-waves increases (see the analytical solution given in *Figure 4* and *Table 1*),

approaching a slightly higher value than the critical stress for $m = 1, n = 4$. Thus, for $\alpha = \pi/6$, the number of longitudinal half-waves in the eigenmodes corresponding to the lowest critical stress is very large; and, since the critical stresses for all of these high-order longitudinal eigenmodes are so similar, the eigenmode is rather indeterminate. The finite element mesh, however, has a fixed number of nodes longitudinally and cannot represent these very high numbers of half-waves with any amount of clarity. Thus, the eigenvector plots show many longitudinal modes—obviously too many for the mesh to represent accurately.

It should be emphasized that these remarks apply in the context of this case only. Nevertheless, the discussion offers some useful insight into more general problems of this class and illustrates some of the difficulties that can be encountered in buckling analysis.

The critical stress values in *Table 2* to *Table 4* for the various mode shapes correlate well with the analytical solution. *Figure 6* compares the eigenvalues obtained with different shell elements with the analytical solutions. Element type S9R5 provides the most accurate results among the shell elements studied. The accuracy of this element is particularly evident in the critical stresses corresponding to the higher-order modes. S4R5 and S4R elements predict somewhat higher critical loads than S9R5. STRI3 provides stiffer solutions compared to the quadrilateral elements due to the constant membrane strain representation.

The element STRI65 results correspond very closely with the analytical solutions. This element can represent linear stress variation (both in membrane and bending modes) and does not have any hourglass modes. Therefore, STRI65 is a robust and efficient element. In general, STRI65 is a good choice, particularly in problems that need very accurate modeling.

A close examination of the analytical solution reveals that there are several hundred modes for which the critical stress is within 15% of the ($m = 1, n = 4$) critical stress. Therefore, this example provides a severe test of the ability of the eigenvalue algorithm to predict nearly equal eigenvalues with distinctly different eigenvectors.

Load-displacement analysis on imperfect geometries

Figure 7 shows the applied load against the axial displacement of the node at a corner of the mesh plotted for the different initial imperfection values. The figure shows that the peak load is essentially the same as that predicted by eigenvalue analysis for the smaller initial imperfections (1% and 10% of the thickness). The larger imperfection (100% of thickness) reduces the peak load by about 12%. The analysis is completed with relative ease for an extensive portion of the postbuckling response.

Figure 8 shows the deformed shape of the cylinder well into the postbuckling response. The particular case shown has an initial imperfection of 1% of the thickness. The development of the postbuckling $n = 4, m = 1$ mode is very apparent. Higher axial modes are also evident: these may be mesh dependent but are not investigated further here.

Input files

bucklecylshell_stri3_n4_40.inp	Eigenvalue buckling prediction. The mesh uses STRI3 elements, with eight rows of elements in the circumferential direction describing an arc of $\pi/8$ radians and 40 elements along the cylinder length.
bucklecylshell_progpert.f	Fortran routine to access the results file and generate the nodal coordinates of a mesh, including a specified degree of geometric imperfection.
bucklecylshell_postbucklpert.inp	Postbuckling load-displacement analysis, with the nodal geometry defined by the Fortran routine of bucklecylshell_progpert.f.

bucklecylshell_s4r5_n1.inp	Eigenvalue buckling prediction. The mesh uses S4R5 elements, with eight rows of elements in the circumferential direction describing an arc of $\pi/4$ radians and 20 elements along the cylinder length.
bucklecylshell_ postbucklimperf.inp	Postbuckling analysis, with the imperfection defined by the **IMPERFECTION* option. The mesh is identical to the mesh described in bucklecylshell_s4r5_n1.inp.

S4 elements, symmetry boundary conditions:

bucklecylshell_s4_n3.inp	$n = 3$.
bucklecylshell_s4_n4.inp	$n = 4$.
bucklecylshell_s4_n5.inp	$n = 5$.
bucklecylshell_s4_n6.inp	$n = 6$.
bucklecylshell_s4_n7.inp	$n = 7$.
bucklecylshell_s4_n8.inp	$n = 8$.
bucklecylshell_s4_n9.inp	$n = 9$.
bucklecylshell_s4_n10.inp	$n = 10$.

S4 elements, antisymmetry boundary conditions:

bucklecylshell_s4_n3anti.inp	$n = 3$.
bucklecylshell_s4_n4anti.inp	$n = 4$.
bucklecylshell_s4_n5anti.inp	$n = 5$.
bucklecylshell_s4_n6anti.inp	$n = 6$.
bucklecylshell_s4_n7anti.inp	$n = 7$.
bucklecylshell_s4_n8anti.inp	$n = 8$.
bucklecylshell_s4_n9anti.inp	$n = 9$.
bucklecylshell_s4_n10anti.inp	$n = 10$.

S4R elements, symmetry boundary conditions:

bucklecylshell_s4r_n3.inp	$n = 3$.
bucklecylshell_s4r_n4.inp	$n = 4$.
bucklecylshell_s4r_n5.inp	$n = 5$.
bucklecylshell_s4r_n6.inp	$n = 6$.
bucklecylshell_s4r_n7.inp	$n = 7$.
bucklecylshell_s4r_n8.inp	$n = 8$.
bucklecylshell_s4r_n9.inp	$n = 9$.
bucklecylshell_s4r_n10.inp	$n = 10$.

S4R elements, antisymmetry boundary conditions:

bucklecylshell_s4r_n3anti.inp	$n = 3$.
bucklecylshell_s4r_n4anti.inp	$n = 4$.
bucklecylshell_s4r_n5anti.inp	$n = 5$.
bucklecylshell_s4r_n6anti.inp	$n = 6$.
bucklecylshell_s4r_n7anti.inp	$n = 7$.
bucklecylshell_s4r_n8anti.inp	$n = 8$.
bucklecylshell_s4r_n9anti.inp	$n = 9$.

bucklecylshell_s4r_n10anti.inp	$n = 10$.

S4R5 elements, symmetry boundary conditions:

bucklecylshell_s4r5_n3.inp	$n = 3$.
bucklecylshell_s4r5_n4.inp	$n = 4$.
bucklecylshell_s4r5_n5.inp	$n = 5$.
bucklecylshell_s4r5_n6.inp	$n = 6$.
bucklecylshell_s4r5_n7.inp	$n = 7$.
bucklecylshell_s4r5_n8.inp	$n = 8$.
bucklecylshell_s4r5_n9.inp	$n = 9$.
bucklecylshell_s4r5_n10.inp	$n = 10$.

S4R5 elements, antisymmetry boundary conditions:

bucklecylshell_s4r5_n3anti.inp	$n = 3$.
bucklecylshell_s4r5_n4anti.inp	$n = 4$.
bucklecylshell_s4r5_n5anti.inp	$n = 5$.
bucklecylshell_s4r5_n6anti.inp	$n = 6$.
bucklecylshell_s4r5_n7anti.inp	$n = 7$.
bucklecylshell_s4r5_n8anti.inp	$n = 8$.
bucklecylshell_s4r5_n9anti.inp	$n = 9$.
bucklecylshell_s4r5n10anti.inp	$n = 10$.

S9R5 elements, symmetry boundary conditions:

bucklecylshell_s9r5_n3.inp	$n = 3$.
bucklecylshell_s9r5_n4.inp	$n = 4$.
bucklecylshell_s9r5_n5.inp	$n = 5$.
bucklecylshell_s9r5_n6.inp	$n = 6$.
bucklecylshell_s9r5_n7.inp	$n = 7$.
bucklecylshell_s9r5_n8.inp	$n = 8$.
bucklecylshell_s9r5_n9.inp	$n = 9$.
bucklecylshell_s9r5_n10.inp	$n = 10$.

S9R5 elements, antisymmetry boundary conditions:

bucklecylshell_s9r5_n3anti.inp	$n = 3$.
bucklecylshell_s9r5_n4anti.inp	$n = 4$.
bucklecylshell_s9r5_n5anti.inp	$n = 5$.
bucklecylshell_s9r5_n6anti.inp	$n = 6$.
bucklecylshell_s9r5_n7anti.inp	$n = 7$.
bucklecylshell_s9r5_n8anti.inp	$n = 8$.
bucklecylshell_s9r5_n9anti.inp	$n = 9$.
bucklecylshell_s9r5_n10anti.inp	$n = 10$.

STRI3 elements, symmetry boundary conditions:

bucklecylshell_stri3_n3.inp	$n = 3$.
bucklecylshell_stri3_n4.inp	$n = 4$.
bucklecylshell_stri3_n5.inp	$n = 5$.

bucklecylshell_stri3_n6.inp	$n = 6$.
bucklecylshell_stri3_n7.inp	$n = 7$.
bucklecylshell_stri3_n8.inp	$n = 8$.
bucklecylshell_stri3_n9.inp	$n = 9$.
bucklecylshell_stri3_n10.inp	$n = 10$.

STRI3 elements, antisymmetry boundary conditions:

bucklecylshell_stri3_n3anti.inp	$n = 3$.
bucklecylshell_stri3_n4anti.inp	$n = 4$.
bucklecylshell_stri3_n5anti.inp	$n = 5$.
bucklecylshell_stri3_n6anti.inp	$n = 6$.
bucklecylshell_stri3_n7anti.inp	$n = 7$.
bucklecylshell_stri3_n8anti.inp	$n = 8$.
bucklecylshell_stri3_n9anti.inp	$n = 9$.
bucklecylshell_stri3_n10anti.inp	$n = 10$.

STRI65 elements, symmetry boundary conditions:

bucklecylshell_stri65_n3.inp	$n = 3$.
bucklecylshell_stri65_n4.inp	$n = 4$.
bucklecylshell_stri65_n5.inp	$n = 5$.
bucklecylshell_stri65_n6.inp	$n = 6$.
bucklecylshell_stri65_n7.inp	$n = 7$.
bucklecylshell_stri65_n8.inp	$n = 8$.
bucklecylshell_stri65_n9.inp	$n = 9$.
bucklecylshell_stri65_n10.inp	$n = 10$.

STRI65 elements, antisymmetry boundary conditions:

bucklecylshell_stri65_n3anti.inp	$n = 3$.
bucklecylshell_stri65_n4anti.inp	$n = 4$.
bucklecylshell_stri65_n5anti.inp	$n = 5$.
bucklecylshell_stri65_n6anti.inp	$n = 6$.
bucklecylshell_stri65_n7anti.inp	$n = 7$.
bucklecylshell_stri65_n8anti.inp	$n = 8$.
bucklecylshell_stri65_n9anti.inp	$n = 9$.
bucklecylshell_stri65_n10anti.inp	$n = 10$.

References

- Timoshenko, S. P., and J. M. Gere, Theory of Elastic Stability, 2nd Edition, McGraw-Hill, New York, 1961.

Tables

Table 1: Critical stresses versus mode shape, stresses given in GPa (from Timoshenko and Gere, 1961).

$n \backslash m$	1	2	3	4	5
0	75.08	64.29	51.86	40.81	32.04
1	116.7	24.45	27.84	26.25	23.05
2	1.478	4.741	7.832	9.769	10.53
3	0.388	1.251	2.389	3.478	4.331
4	0.281	0.478	0.913	1.417	1.908
5	0.479	0.298	0.449	0.681	0.942
6	94.65	0.329	0.309	0.401	0.533
7	1.757	0.495	0.314	0.308	0.360
8	3.022	0.794	0.414	0.316	0.305
9	4.875	1.251	0.510	0.394	0.322
10	7.473	1.898	0.878	0.537	0.395
$n \backslash m$	6	7	8	9	10
0	25.37	20.36	16.59	13.71	11.48
1	19.68	16.65	14.10	11.99	10.27
2	10.47	9.941	9.190	8.376	7.577
3	4.886	5.165	5.228	5.136	4.945
4	2.328	2.654	2.878	3.010	3.064
5	1.197	1.430	1.625	1.778	1.888
6	0.680	0.827	0.966	1.089	1.191
7	0.437	0.525	0.616	0.702	0.782
8	0.332	0.377	0.431	0.487	0.544
9	0.305	0.315	0.339	0.372	0.407
10	0.333	0.310	0.305	0.318	0.336

Table 2: Critical stresses – S9R5 element, stresses given in GPa.

α	Boundary condition at midlength of cylinder			
	SYMM		ASYMM	
	σ	(n, m)	σ	(n, m)
$\pi/6$	0.316	(*, *)	0.318	(*, *)
$\pi/8$	0.281	(4, 1)	0.317	(4, *)

α	Boundary condition at midlength of cylinder			
	SYMM		ASYMM	
	σ	(n, m)	σ	(n, m)
$\pi/10$	0.316	(*, *)	0.299	(*, 2)
$\pi/12$	0.310	(6, 3)	0.316	(6, *)
$\pi/14$	0.315	(7, 3)	0.309	(7, 4)
$\pi/16$	0.306	(8, 5)	0.316	(8, 4)
$\pi/18$	0.316	(9, 7)	0.306	(9, 6)
$\pi/20$	0.310	(10, 7)	0.309	(10, 8)

Table 3: Critical stresses – S4R5, S4R elements, stresses given in GPa.

α	Boundary condition at midlength of cylinder			
	SYMM		ASYMM	
	σ	(n, m)	σ	(n, m)
$\pi/6$	0.327	(*, *)	0.327	(*, *)
$\pi/8$	0.290	(4, 1)	0.326	(4, *)
$\pi/10$	0.326	(*, *)	0.308	(*, 2)
$\pi/12$	0.320	(6, 3)	0.326	(6, *)
$\pi/14$	0.327	(7, 3)	0.319	(7, 4)
$\pi/16$	0.317	(8, 5)	0.326	(8, 4)
$\pi/18$	0.326	(9, 7)	0.317	(9, 6)
$\pi/20$	0.322	(10, 7)	0.320	(10, 8)

Table 4: Critical stresses – STRI3 element, stresses given in GPa.

α	Boundary condition at midlength of cylinder			
	SYMM		ASYMM	
	σ	(n, m)	σ	(n, m)
$\pi/6$	0.359	(*, *)	0.355	(*, *)
$\pi/8$	0.285	(4, 1)	0.357	(4, *)
$\pi/10$	0.359	(*, *)	0.308	(*, 2)
$\pi/12$	0.321	(6, 3)	0.334	(6, 2)

α	Boundary condition at midlength of cylinder			
	SYMM		ASYMM	
	σ	(n, m)	σ	(n, m)
$\pi/14$	0.322	(7, 3)	0.321	(7, 4)
$\pi/16$	0.319	(8, 5)	0.325	(8, 4)
$\pi/18$	0.332	(9, 5)	0.319	(9, 6)
$\pi/20$	0.324	(10, 7)	0.326	(10, 8)

Table 5: Critical stresses – STRI65 element, stresses given in GPa.

α	Boundary condition at midlength of cylinder			
	SYMM		ASYMM	
	σ	(n, m)	σ	(n, m)
$\pi/6$	0.319	(*, *)	0.308	(*, *)
$\pi/8$	0.280	(4, 1)	0.315	(4, *)
$\pi/10$	0.326	(*, *)	0.298	(*, 2)
$\pi/12$	0.309	(6, 3)	0.328	(6, 2)
$\pi/14$	0.314	(7, 3)	0.308	(7, 4)
$\pi/16$	0.305	(8, 5)	0.315	(8, 4)
$\pi/18$	0.315	(9, 5)	0.305	(9, 6)
$\pi/20$	0.309	(10, 7)	0.308	(10, 8)

Figures

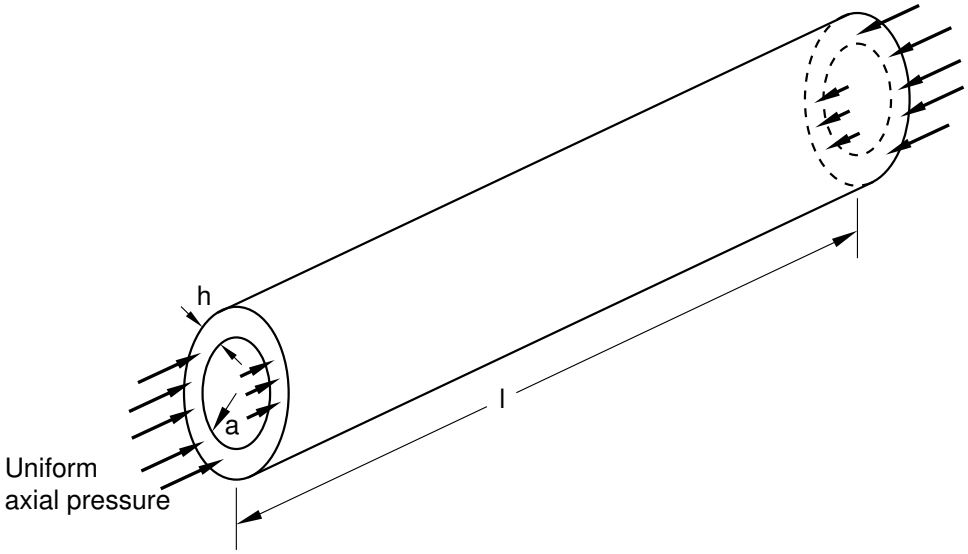


Figure 1: Cylindrical shell with uniform axial loading.

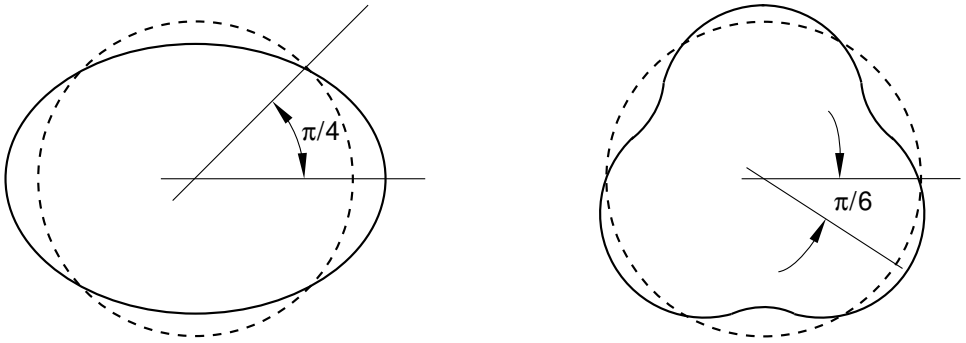


Figure 2: Cross-section deformation corresponding to n=2 and to n=3.

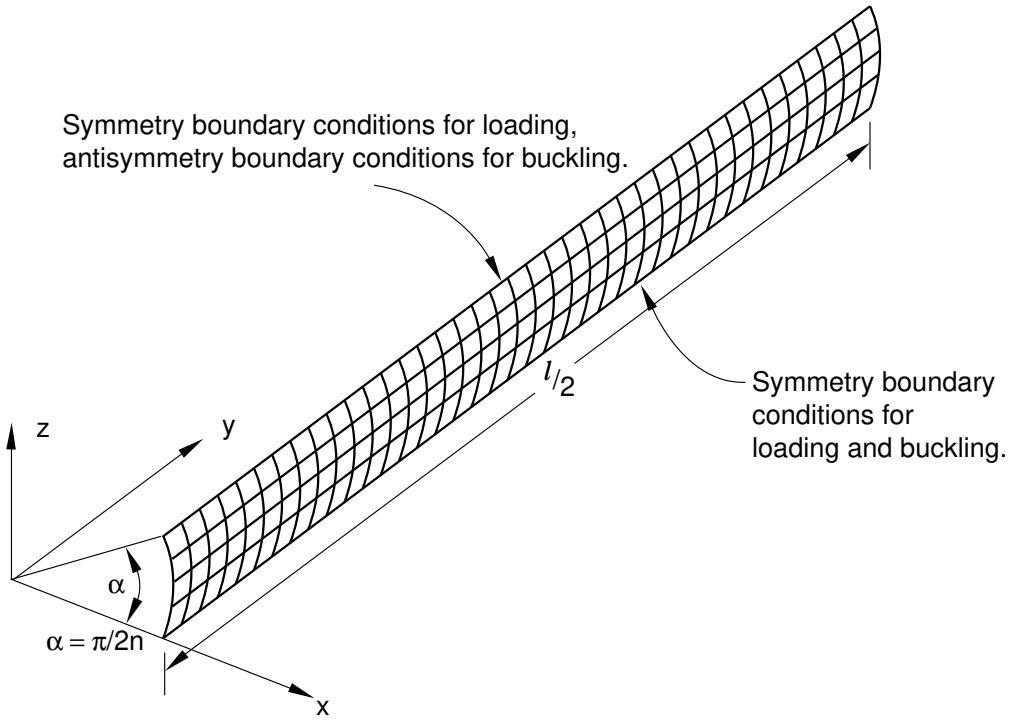


Figure 3: S4R5 mesh for eigenvalue buckling prediction.

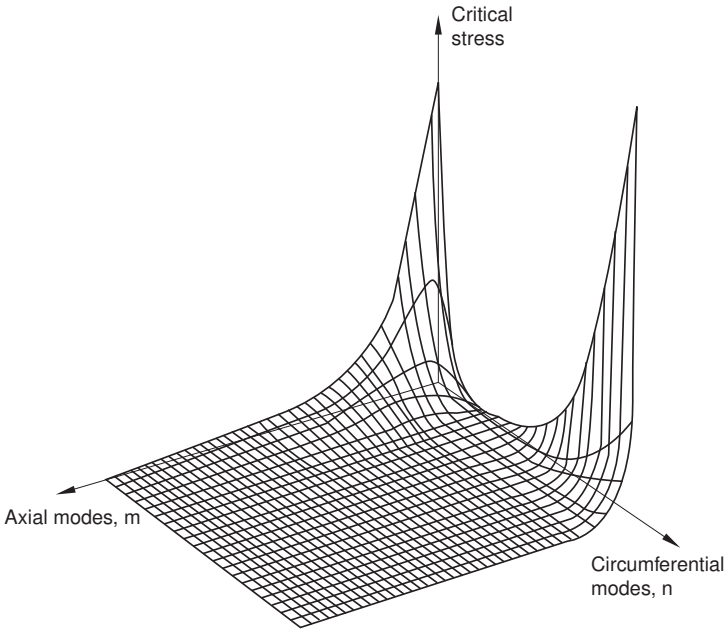


Figure 4: Critical stress for various buckling modes.

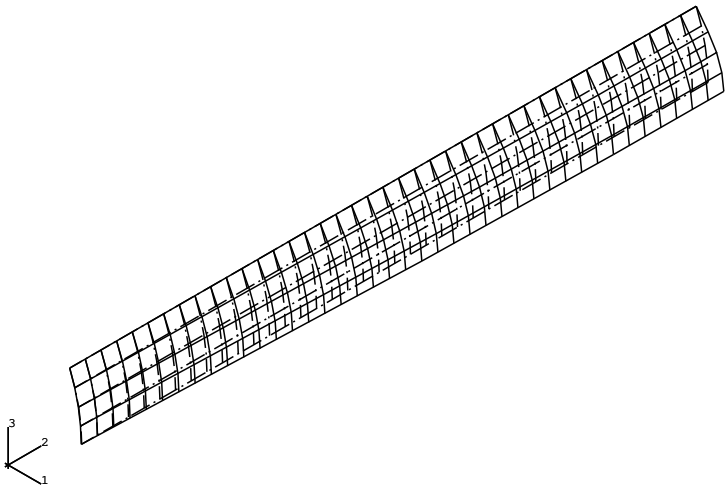


Figure 5: Buckling mode shape (m=1, n=4).

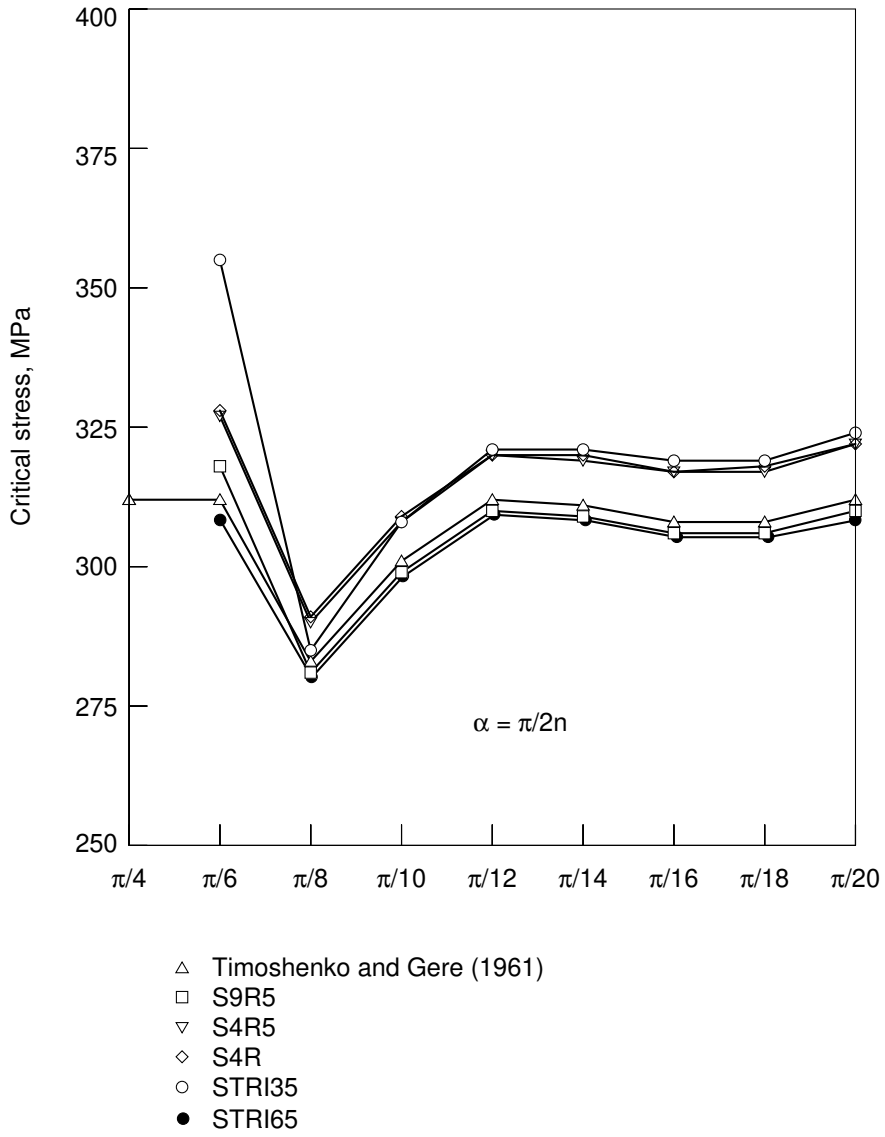


Figure 6: Critical stress versus subtended angle of quarter-wave model.

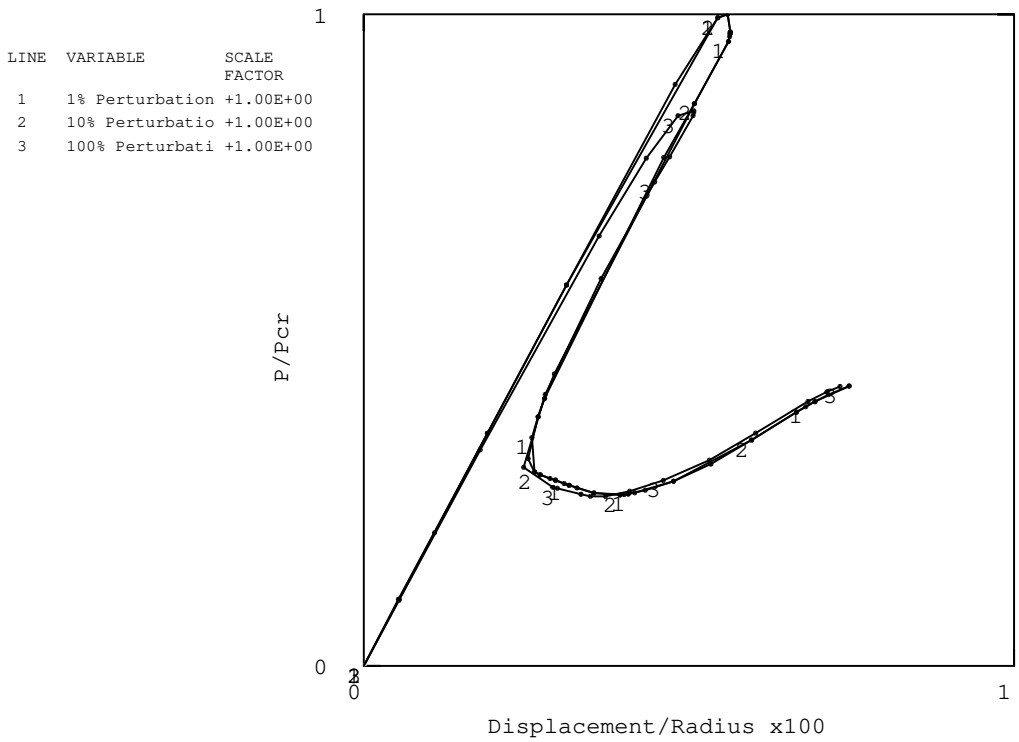


Figure 7: Applied load (normalized) versus axial displacement of an end node.

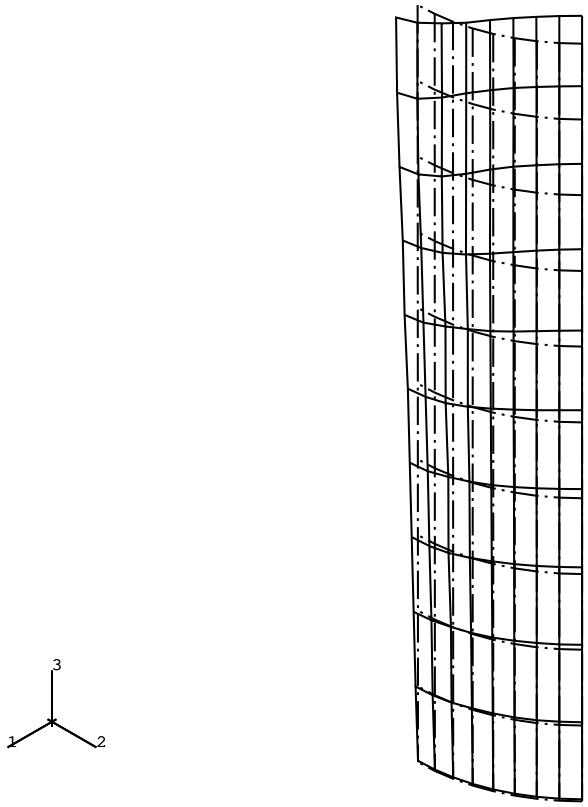


Figure 8: Postbuckled deformation (initial imperfection of 1% of thickness).

Buckling of a simply supported square plate

Products: Abaqus/Standard

Overview

This example shows the geometric collapse study of a stiff, shell-type structure using Abaqus.
The problem is that of a square, thin, elastic plate, simply supported on all four edges and compressed in one direction (see *Figure 1*).

The analytical solution for the buckling load for this case (see Timoshenko and Gere, 1961, Section 9.2) is

$$N^{\text{crit}} = \frac{4\pi^2 D}{b^2},$$

where N^{crit} is the critical value of the edge load per unit length of the edge, b is the length of each edge of the plate, and $D = Et^3/12\left(1-\nu^2\right)$ is the elastic bending stiffness of the plate, with Young's modulus E, Poisson's ratio ν, and plate thickness t.

The corresponding buckling mode is a transverse displacement of

$$w = A\cos\frac{\pi x}{b}\cos\frac{\pi y}{b}$$

in the coordinate system of *Figure 1*. Here A is an arbitrary magnitude.

Problem description

No particular units are used in this example; the values chosen are taken to be in a consistent set. The length of the edge of the square plate is 2 and the thickness is 0.01, so the plate is rather thin ($L/t = 200$). Since the solution is known to be symmetric, only one-quarter of the plate is modeled. Meshes of 2×2 or 4×4 elements are used. Since the form of the prebuckled and postbuckled solutions is rather smooth in this case, even these relatively coarse meshes should give reasonably accurate results for the buckling load.

The material is assumed to be isotropic elastic, with a Young's modulus of 10^8 and a Poisson's ratio of 0.3.

The boundary conditions on the model are

1. Symmetry about $x = 0$. This requires $u_x = \phi_y = \phi_z = 0$ on that edge of the mesh.
2. Symmetry about $y = 0$. This requires $u_y = \phi_x = \phi_z = 0$ on that edge of the mesh.
3. Simple support on the edge at $x = b/2$. This requires $u_z = \phi_x = 0$ on that edge of the mesh.
4. Simple support on the edge at $y = b/2$. This requires $u_z = \phi_y = 0$ on that edge of the mesh.

Loading

Two versions of the problem are used: one in which the plate is loaded in one direction by uniform edge loads, and one in which the plate is compressed by raising its temperature with the plate constrained in one direction against overall thermal expansion.

For the mechanically loaded case the edge loads are given as point loads on the edge nodes. Since the second-order elements (S8R5, S9R5, STRI65) use quadratic interpolation along their edges, consistent distribution of a uniform

load gives equivalent point loads in the ratio 1:4:1 at the corner, midside, and corner nodes, respectively (Simpson's integration rule). The first-order elements (S4R5, S4R, S3R, STRI3) are based on linear in-plane displacements so that the uniform edge loading gives equal point loads at the nodes on the edge.

Eigenvalue buckling prediction

Stiff shell collapse studies are typically begun with eigenvalue buckling estimates. Such estimates are usually accurate in cases of stiff shells—that is, when the prebuckle response is essentially linear; when the collapse is not catastrophic, so the structure is not excessively sensitive to imperfections; and when the response is elastic. As will be seen later, these conditions are fulfilled by this example.

Eigenvalue buckling estimates are obtained by using the eigenvalue buckling procedure (*Eigenvalue Buckling Prediction*). Since the eigenvalue buckling procedure is a linear perturbation procedure the size of the load is immaterial because the response is proportional to the magnitude of the load. Abaqus will predict the buckling modes and corresponding eigenvalues. In this case three modes are requested. The lowest buckling load estimates are shown in *Table 1*. All of the meshes except the 4×4 mesh of element type S3R give reasonable predictions. The S3R elements give a higher estimate of lowest buckling load because the constant bending strain approximation results in a stiffer response. The most accurate results are those provided by element types S8R5 and S9R5.

Load-displacement studies on imperfect geometries

The next phase of a typical collapse analysis is to perform a load-displacement analysis to ensure that the eigenvalue buckling prediction already obtained is accurate and, at the same time, to investigate the effect of initial geometric imperfection on the load-displacement response. In this way concerns about imperfection sensitivity (unstable postbuckling response) can be addressed. The eigenvalue analysis is useful in providing guidance about mesh design for these more expensive load-displacement studies: mesh convergence studies can be performed as part of the eigenvalue analysis, which is usually significantly less expensive than the load-deflection analysis.

For the load-displacement analysis the perfect geometry must be “seeded” with an imperfection to cause it to collapse. It is possible that a problem run with perfect geometry may never buckle numerically at reasonable load levels because the model has absolutely no prebuckled displacement in the postbuckled mode and, thus, no ability to switch to that mode. Presumably an imperfection in the form of the buckling mode would be the most critical. In this example, for simplicity, we use instead a bilinear imperfection:

$$z(x,y) = A^I \left(1 - \frac{2x}{b}\right) \left(1 - \frac{2y}{b}\right).$$

So long as the imperfection contains the mode into which the structure wishes to collapse, it is presumed that any imperfection will provide the necessary perturbation of the solution.

The imperfection magnitude, A^I, is taken as 0.1%, 1%, and 10% of the plate thickness. Since we expect a buckle at a load of about 90.4, the edge load is applied by requesting that the load be increased monotonically up to a value of 100, starting with an increment of 10. Normally the Riks method would be chosen if the postbuckling response is unstable. It is not necessary for this case.

In all cases where a sudden loss of stiffness is expected (as here, when the imperfection is very small) it is essential that equilibrium be satisfied closely; otherwise it is possible for the solution to fail to switch to the alternate branch of the solution. The default equilibrium tolerances used in Abaqus are rather tight by engineering standards, as experience shows that less demanding equilibrium control may fail to pick up the buckle in the case of almost perfect geometry.

Results and discussion

The numerical results for the mechanically loaded case are summarized in *Figure 2*, where the displacement of the center point of the plate is plotted as a function of compressive force. The case with the smallest imperfection (0.1% of the thickness) shows a very sharp loss of stiffness at an applied load of about 90. This is essentially the eigenvalue solution (90.4). As the initial imperfection magnitude is increased, the behavior becomes smoother, as would be expected. The plate shows positive stiffness up to the maximum loading applied, even when the imperfection is very small. Thus, in this case the buckling is not an unstable failure; the plate is, therefore, not very sensitive to imperfection. In cases of unstable postbuckling response it is usually easiest to approach the analysis by studying the larger imperfection magnitudes first, since then the response is smoothest.

The stress just at buckling with the smallest imperfection is about 9000. An interesting alternative case is where the edges parallel to the y-axis are restrained in the x-direction (that is, $u_x = \phi_z = 0$), and the temperature of the plate is raised. This should give the same prebuckled stress field in the plate; and, thus, critical temperature changes should be those that give the same critical stress. To investigate this case, we use a thermal expansion coefficient of 10^{-6} (strain per unit temperature rise) so that in the prebuckled state the critical stress should occur at a temperature of 90°. The results of such a thermally loaded case for the smallest imperfection studied are shown in *Figure 3*. The behaviors of the mechanically loaded case and the thermally loaded case are quite similar, with the thermally loaded case showing rather less displacement after buckling. This is to be expected, since thermal loading causes strain, whereas mechanical loading requires a specific stress state to retain equilibrium.

The same thermally loaded case is solved using the Riks approach to verify the Abaqus capability for using the Riks algorithm with thermal loading only. The temperature-displacement curves for the incremental static analysis versus the Riks analysis are very similar, with the smoother curve obtained by the Riks approach for strain levels between 0.5×10^{-3} and 2×10^{-3}. The Riks algorithm chooses smaller temperature increments, thus requiring more increments to apply the same total temperature rise.

Input files

S3R elements:

buckleplate_s3r_buckle.inp	Eigenvalue prediction of buckling under edge loading.
buckleplate_s3r_load.inp	Edge load-displacement response prediction.
buckleplate_s3r_thermbuckle.inp	Eigenvalue prediction of buckling under thermal loading.
buckleplate_s3r_loadthermal.inp	Thermal load-displacement response prediction.

S4 elements:

buckleplate_s4_buckle.inp	Eigenvalue prediction of buckling under edge loading.
buckleplate_s4_load.inp	Edge load-displacement response prediction.
buckleplate_s4_thermbuckle.inp	Eigenvalue prediction of buckling under thermal loading.
buckleplate_s4_loadthermal.inp	Thermal load-displacement response prediction.

S4R elements:

buckleplate_s4r_buckle.inp	Eigenvalue prediction of buckling under edge loading.
buckleplate_s4r_load.inp	Edge load-displacement response prediction.
buckleplate_s4r_thermbuckle.inp	Eigenvalue prediction of buckling under thermal loading.
buckleplate_s4r_loadthermal.inp	Thermal load-displacement response prediction.

S4R5 elements:

buckleplate_s4r5_buckle.inp	Eigenvalue prediction of buckling under edge loading.
buckleplate_s4r5_load.inp	Edge load-displacement response prediction.

buckleplate_s4r5_thermbuckle.inp	Eigenvalue prediction of buckling under thermal loading.
buckleplate_s4r5_loadthermal.inp	Thermal load-displacement response prediction.

S8R elements:

buckleplate_s8r_buckle.inp	Eigenvalue prediction of buckling under edge loading.
buckleplate_s8r_load.inp	Edge load-displacement response prediction.
buckleplate_s8r_thermbuckle.inp	Eigenvalue prediction of buckling under thermal loading.
buckleplate_s8r_loadthermal.inp	Thermal load-displacement response prediction.
buckleplate_postoutput.inp	**POST OUTPUT* analysis.

S8R5 elements:

buckleplate_s8r5_buckle.inp	Eigenvalue prediction of buckling under edge loading.
buckleplate_s8r5_load.inp	Edge load-displacement response prediction.
buckleplate_s8r5_thermbuckle.inp	Eigenvalue prediction of buckling under thermal loading.
buckleplate_s8r5_loadthermal.inp	Thermal load-displacement response prediction.
buckleplate_s8r5_riks.inp	Thermally loaded plate using the Riks algorithm.
buckleplate_s8r5_load_bigimp.inp	Edge load-displacement response prediction with an imperfection of 10%.
buckleplate_s8r5_load_smallimp.inp	Edge load-displacement response prediction with an imperfection of 0.1%.

S9R5 elements:

buckleplate_s9r5_buckle.inp	Eigenvalue prediction of buckling under edge loading.
buckleplate_s9r5_load.inp	Edge load-displacement response prediction.
buckleplate_s9r5_thermbuckle.inp	Eigenvalue prediction of buckling under thermal loading.
buckleplate_s9r5_loadthermal.inp	Thermal load-displacement response prediction.

STRI3 elements:

buckleplate_stri3_buckle.inp	Eigenvalue prediction of buckling under edge loading.
buckleplate_stri3_load.inp	Edge load-displacement response prediction.
buckleplate_stri3_thermbuckle.inp	Eigenvalue prediction of buckling under thermal loading.
buckleplate_stri3_loadthermal.inp	Thermal load-displacement response prediction.

STRI65 elements:

buckleplate_stri65_buckle.inp	Eigenvalue prediction of buckling under edge loading.
buckleplate_stri65_load.inp	Edge load-displacement response prediction.
buckleplate_stri65_thermbuckle.inp	Eigenvalue prediction of buckling under thermal loading.
buckleplate_stri65_loadthermal.inp	Thermal load-displacement response prediction.

References

- Timoshenko, S. P., and J. M. Gere, *Theory of Elastic Stability*, 2nd Edition, McGraw-Hill, New York, 1961.

Tables

Table 1: Eigenvalue buckling predictions. (Analytical solution: 90.38)

Mesh and element type	Edge load	Thermal load
2 × 2, S8R5	90.52	90.52
2 × 2, S8R	95.32	95.32
2 × 2, S9R5	90.52	90.52
2 × 2, STRI65	89.64	89.64
4 × 4, STRI3	90.47	90.47
4 × 4, S3R	115.92	115.92
4 × 4, S4R	92.80	92.80
4 × 4, S4R5	92.76	92.76
4 × 4, S4	92.35	92.35

Figures

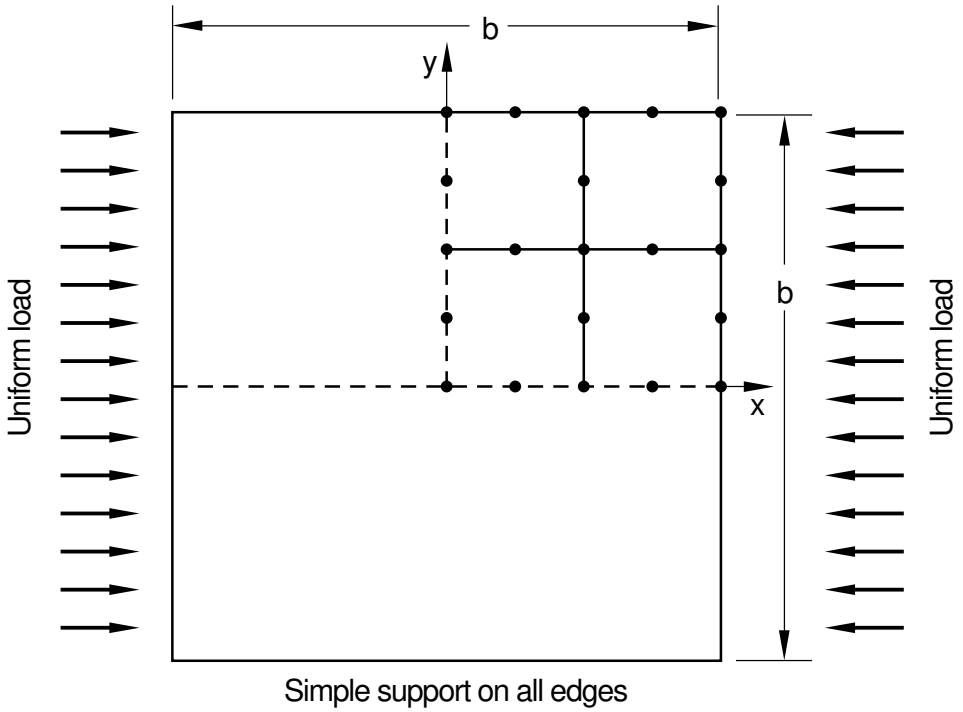


Figure 1: Square plate buckling study.

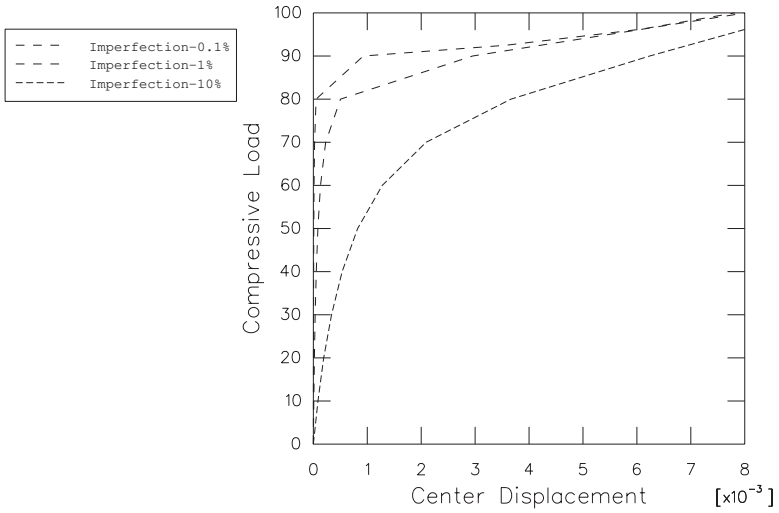


Figure 2: Square plate elastic buckling results.

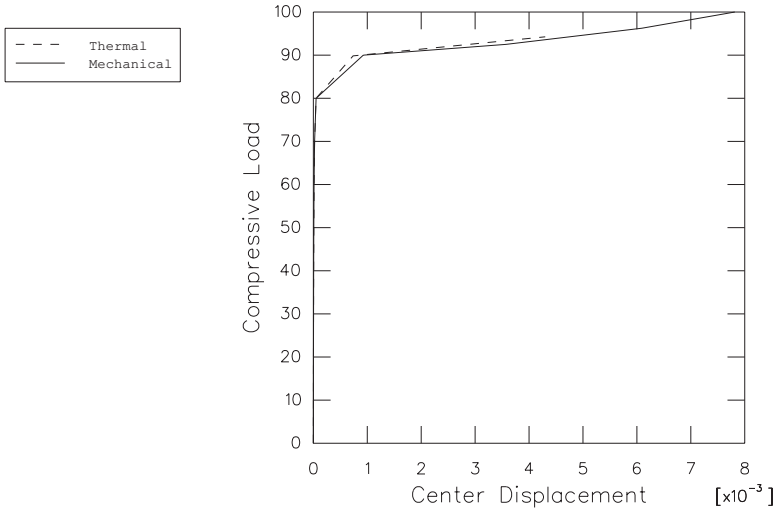


Figure 3: Comparison of mechanical and thermal loading results.

Lateral buckling of an L-bracket

Products: Abaqus/Standard

Overview

This problem considers the nonlinear postbuckling behavior of an aluminum “L” bracket plate that is clamped on one end and subjected to an in-plane load on the other.
This problem has been used to assess the behavior of various shell elements intended for use in geometrically nonlinear analyses (see Argyris et al., 1979; Simo et al., 1990). Here, the solution illustrates the postbuckling capabilities of the S4 element when subjected to in-plane bending.

Problem description

The bracket is shown in *Figure 1*. It is 240 mm long and 30 mm wide, with a thickness of 0.6 mm. The material is linear elastic with Young's modulus E=71240 MPa and Poisson's ratio ν=0.3. As shown, the bracket is loaded in tension.

The problem is modeled using fully integrated S4 shell elements with three different meshes: 17, 68, and 272 elements, as shown in *Figure 2*. For comparison, the finer meshes are also run with reduced-integration S4R shell elements. The reference solution is obtained with a refined mesh of second–order continuum elements. This continuum mesh uses 272 C3D20 elements in-plane and two through the thickness.

To trigger the lateral buckling mode of the bracket, a linear eigenvalue buckling analysis is performed for each model, with the resulting fundamental eigenmode added as an imperfection to the geometry for the nonlinear postbuckling analysis. For this geometry and loading the first eigenmode corresponds to out-of-plane buckling of the bracket when loaded in compression, opposite to the direction shown in *Figure 1*; the second buckling mode corresponds to tension, the relevant fundamental mode for this analysis. By default, Abaqus calculates both positive and negative eigenvalues, in ascending order of absolute value. To calculate only the positive eigenvalues, use the Lanczos eigensolver and restrict the range of eigenvalues of interest to positive values by setting the minimum eigenvalue of interest equal to zero. This method is particularly useful if the eigenmode is selected as an imperfection for a full geometrically nonlinear analysis; it ensures that the imperfection is appropriate for the direction of loading.

Results and discussion

The nonlinear buckling load predictions are compared with published results in *Table 1*. *Figure 3* and *Figure 4* show the postbuckling behavior for S4 and S4R elements for each of the meshes considered. These results compare well with the published results. Even the coarsest mesh (17 elements) produces reasonable results. However, a 17-element model with S4R elements (that is, with one element across the width of the bracket) cannot capture the buckling response due to its inability to represent in-plane bending accurately with a single element across the section. With 68 elements the S4 model has nearly converged on the reference solution obtained with a fine mesh of continuum elements, whereas S4R has not.

Input files

lbracket_buckle_17s4.inp	Eigenvalue extraction with the 17-element S4 mesh.
lbracket_postbuckle_17s4.inp	Postbuckling analysis with the 17-element S4 mesh.
lbracket_buckle_68s4.inp	Eigenvalue extraction with the 68-element S4 mesh.
lbracket_postbuckle_68s4.inp	Postbuckling analysis with the 68-element S4 mesh.

lbracket_buckle_272s4.inp	Eigenvalue extraction with the 272-element S4 mesh.
lbracket_postbuckle_272s4.inp	Postbuckling analysis with the 272-element S4 mesh.
lbracket_buckle_68s4r.inp	Eigenvalue extraction with the 68-element S4R mesh.
lbracket_postbuckle_68s4r.inp	Postbuckling analysis with the 68-element S4R mesh.
lbracket_buckle_272s4r.inp	Eigenvalue extraction with the 272-element S4R mesh.
lbracket_postbuckle_272s4r.inp	Postbuckling analysis with the 272-element S4R mesh.
lbracket_buckle_c3d20.inp	Eigenvalue extraction with the 544-element C3D20 mesh.
lbracket_postbuckle_c3d20.inp	Postbuckling analysis with the 544-element C3D20 mesh.

References

- Argyris, J. H., H. Balmer, J. St.
  Doltsinis, P. C. Dunne, M. Haase, M. Kleiber, G. A. Malejannakis, H. P. Mlejnek, M. Müller, and D. W. Scharpf, "Finite Element Method – The Natural Approach," *Computer Methods in Applied Mechanics and Engineering*, vol. 17/18, pp. 1–106, 1979.
- Simo, J. C., D. D. Fox, and M. S. Rifai, "On a Stress Resultant Geometrically Exact Shell Model. Part III: Computational Aspects of the Nonlinear Theory," *Computer Methods in Applied Mechanics and Engineering*, vol. 79, pp. 21–70, 1990.

Tables

Table 1: Comparison of bifurcation loads.

Mesh	S4R	S4	Simo et al.	Argyris et al.
17		1.22		
68	1.19	1.20	1.137	1.155
272	1.18	1.19		

Figures

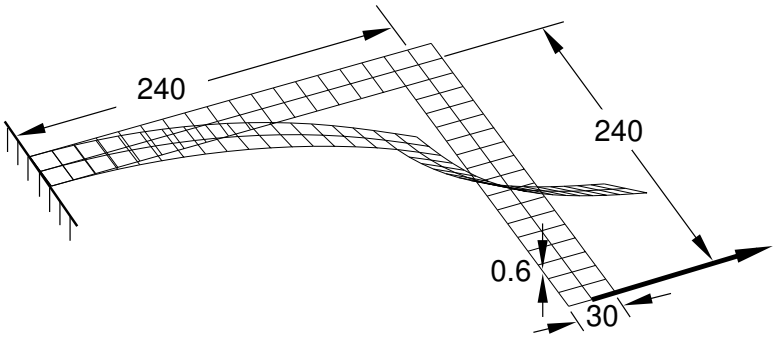


Figure 1: "L" bracket geometry.

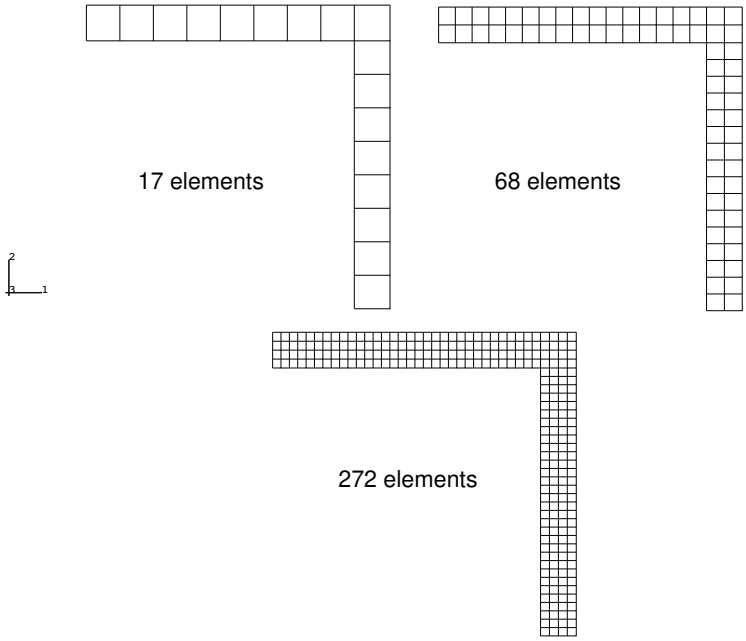


Figure 2: Meshes used.

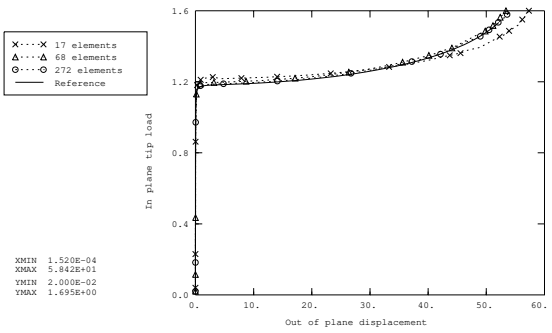


Figure 3: S4 postbuckling response.

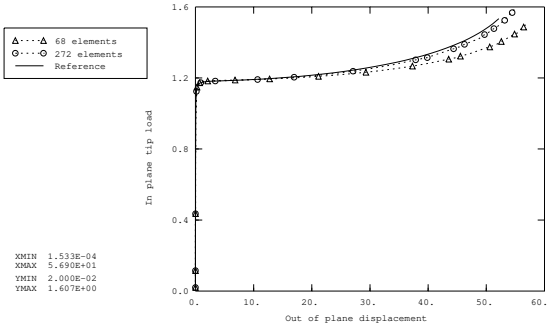


Figure 4: S4R postbuckling response.

Buckling of a column with general contact

Products: Abaqus/Explicit

Overview

This example illustrates the buckling of a column between two rigid platens.

Problem description

The column used in this example has an X-shaped section. The ends of the column are attached to two rigid platens. One of the platens is fixed in space, while the other is pushed and rotated 31° during 7 msec to buckle the column. The column has a height of 1.0 m, and each of the four flanges of the column is 0.2 m wide and 0.003 m thick.

Two node-based surfaces consisting of nodes at each end of the column are defined. Each node-based surface is attached to the appropriate rigid platen using a tie constraint.

In the primary analysis the general contact capability is used. 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 (see *About General Contact in Abaqus/Explicit*).

Additional models using penalty contact pairs and both penalty contact pairs (for all contact pairs involving rigid surfaces) and kinematic contact pairs (for all other contact pairs) are provided. The contact pair algorithm cannot use surfaces that have more than two facets sharing a common edge, so for these analyses self-contact of the column is modeled by defining double-sided contact surfaces on each of the four legs of the cross-section; each leg can contact itself and the adjacent legs. The contact definition is more straightforward with the general contact algorithm.

The column is made of steel, with a Young's modulus of 200 GPa and a Poisson's ratio of 0.3. The density is 7850 kg/m^3. A von Mises elastic, perfectly plastic material model is used with a yield stress of 250 MPa. Material failure is not considered in the primary analysis or the analyses that use contact pairs. An additional general contact analysis in which material failure is considered is provided to demonstrate the shell erosion capability in the general contact algorithm. The shear failure model is used in this test to specify that elements should be removed once their equivalent plastic strain reaches 40%.

The effects of initial geometric perturbations are also studied in this example. In a numerical buckling analysis of a configuration with a high degree of symmetry, buckling often does not initiate immediately when the bifurcation (branching) point in the equilibrium path is reached; small imperfections help to trigger buckling. When buckling does initiate in an explicit dynamics analysis with a high degree of symmetry (even under quasi-static conditions), the buckling mode often has a wavelength that spans only a few elements (a much shorter wavelength than would occur in reality). Geometric imperfections can be introduced into a model in Abaqus to achieve a more realistic solution. This seeding of imperfections is usually not necessary for cases without a high degree of symmetry. Designers may purposely introduce imperfection shapes that promote certain buckling modes to maximize energy absorption; for example, for car crash analysis.

In this example we first compute the buckling modes of the column by running a linear buckling analysis in Abaqus/Standard and store these modes in the results (`.fil`) file. We then use geometric imperfections in Abaqus/Explicit to read the buckling modes corresponding to the lowest eigenvalues, scale them, and use them to perturb the nodal coordinates of the column. The linear buckling analysis in Abaqus/Standard is performed in the presence of only an axial load to mimic the loading during the early part of the dynamic analysis when the buckling should initiate. When choosing the perturbation magnitudes, the goal is to seed the mesh with a deformation pattern that will allow the postbuckling deformation to proceed correctly. Under quasi-static conditions one would expect the postbuckling deformation to resemble the eigenmode corresponding to the

lowest eigenvalue, unless the lowest eigenvalues are closely spaced, in which case the postbuckling deformation is likely to be some combination of the lowest eigenmodes. Higher eigenmodes will tend to play an increased role in the postbuckling shape as the loading rate increases. An eigenmode number and a scaling factor to be applied to the corresponding eigenmode are given. Abaqus/Standard normalizes the eigenmodes such that the maximum deformation in the length units of the analysis (meters in this case) is 1.0. The first three eigenmodes are used in the seeding for this example, with the scaling factor monotonically decreasing as the mode number increases. Three separate input files, which are the same as the primary input file except for the use of geometric imperfections, are provided, with the lowest eigenmode scaled to 1%, 10%, and 100% of the shell thickness, respectively.

Results and discussion

All models that do not include material failure or initial imperfections give similar results, indicating that these results are not sensitive to the choice of contact algorithm. *Figure 1* shows the original configuration of the column. *Figure 2* shows the deformed shape of the column after 3.5 msec. *Figure 3* shows the deformed shape of the column after 7.0 msec. *Figure 4* shows the time history of the total kinetic energy, the total work done on the model, and the total internal energy. *Figure 5* shows the magnitude of the vertical (x-direction) reaction forces at the reference nodes of the top (`WALL1`) and bottom (`WALL2`) rigid platens. *Figure 6* shows the magnitude of the reaction moments about the y-axis at the reference nodes of the two rigid platens.

Figure 7 shows the final deformed shape of the column for the analysis with material failure and surface erosion (only the elements that have not failed are shown). In this analysis the bottom half of the column has less deformation in comparison to the analyses that do not consider material failure. Facets of failed elements do not participate in contact in this analysis; secondary nodes can be observed to pass through failed elements without generating contact forces.

Figure 8 and *Figure 9* show the deformed shapes of the column with a 10% seeded imperfection at 3.5 msec and 7.0 msec, respectively. Small initial imperfections significantly affect the results. The flange in the positive z-direction shows some buckling at 3.5 msec only when an initial imperfection is present. The postbuckling mode has a fairly short wavelength even with the seeded imperfections, due to dynamic effects. If the loading rate were decreased, the wavelength of the postbuckling mode would tend to increase. The incorporation of imperfections in the column also leads to a reduction in the work performed during its deformation. *Figure 10* shows plots of external work as a function of time for the column without any imperfection and for the column with imperfections of 1%, 10%, and 100%, respectively. The external effort needed to deform the column reduces as the amount of imperfection in the column increases, and even a small imperfection on the order of 1% introduced as a seed significantly reduces the energy spent in the buckling and crushing of the column.

The problems presented here test the features mentioned but do not provide independent verification of them.

Input files

sscxsec.inp	Primary analysis using the general contact capability.
sscxsec_cpair.inp	Model using a combination of penalty and kinematic contact pairs.
sscxsec_pnlty.inp	Model using penalty contact pairs.
sscxsec_erosion.inp	Model considering surface erosion due to material failure. This model uses the general contact capability.
sscxsec_bkl.inp	Eigenvalue buckling analysis.
sscxsec_imperf001.inp	Model using 1% imperfection.
sscxsec_imperf010.inp	Model using 10% imperfection.
sscxsec_imperf100.inp	Model using 100% imperfection.

Figures

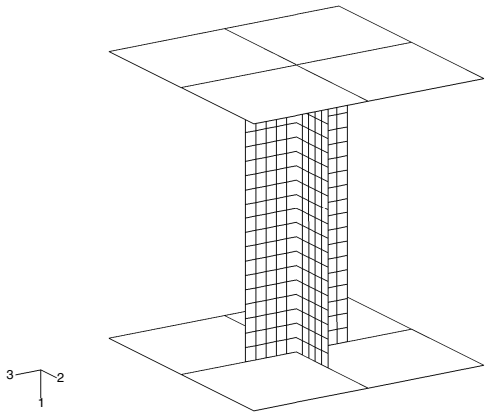


Figure 1: Initial configuration of the column.

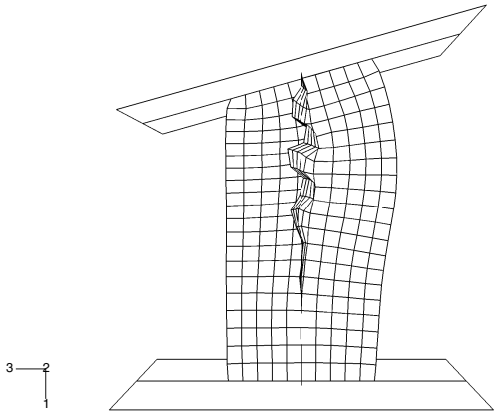


Figure 2: Deformed shape at 3.5 msec.

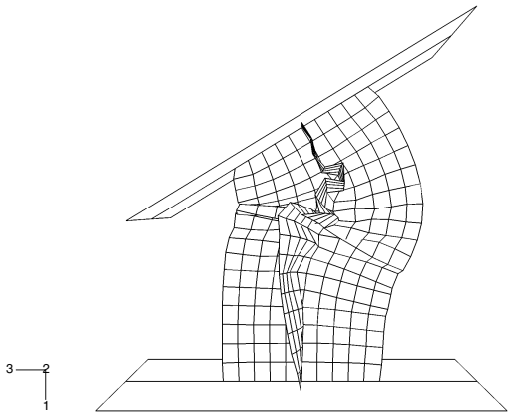


Figure 3: Deformed shape at 7.0 msec.

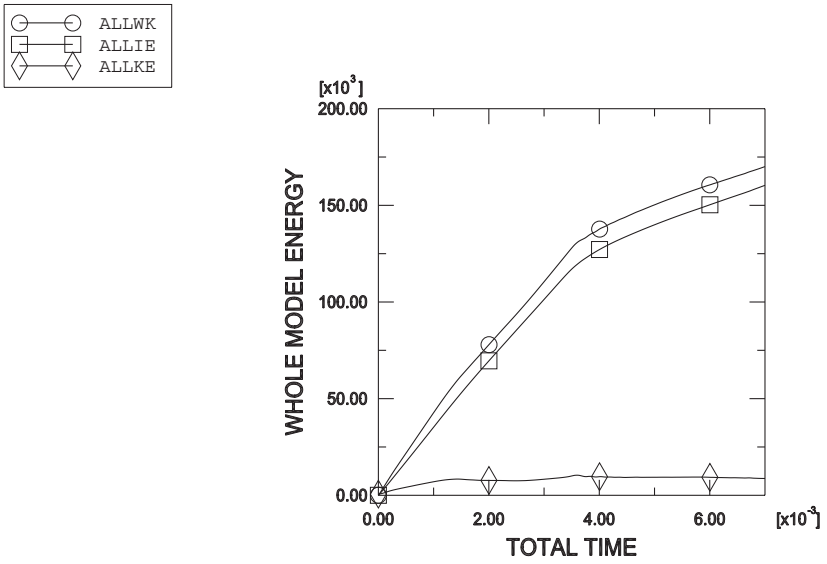


Figure 4: Time histories of the total kinetic energy, work done on the model, and internal energy.

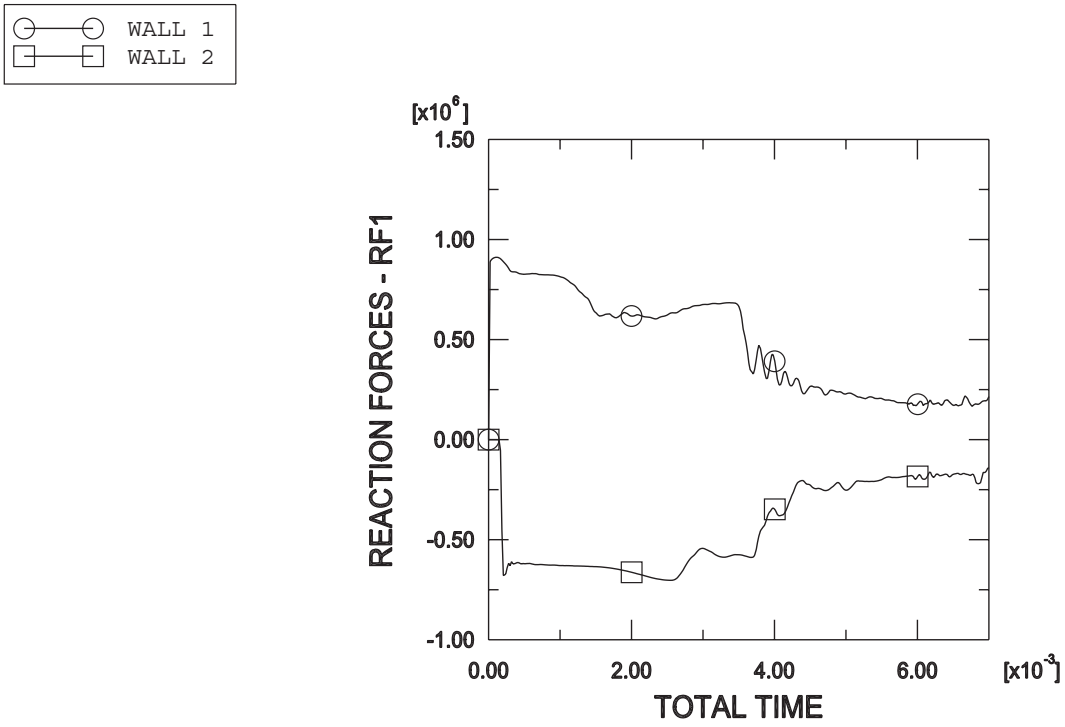


Figure 5: Magnitude of the vertical reaction forces on the rigid platens.

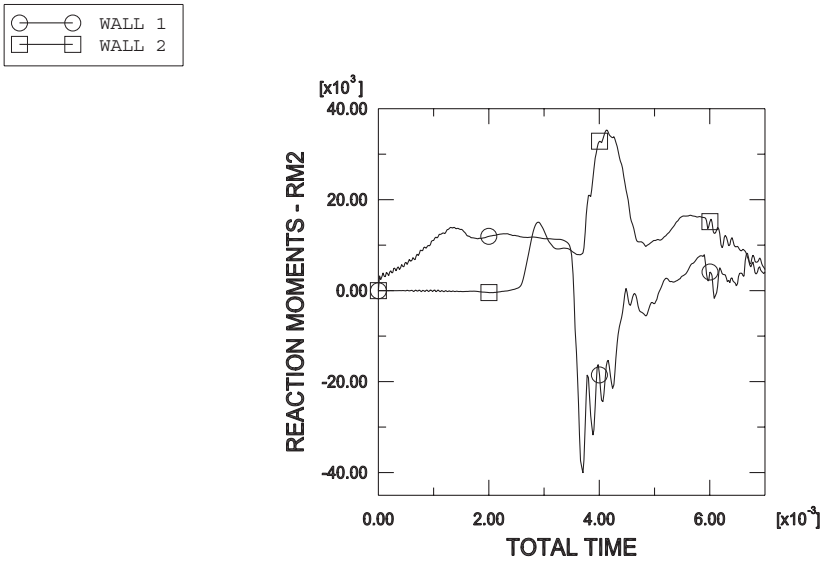


Figure 6: Magnitude of the reaction moments about the *y*-axis on the rigid platens.

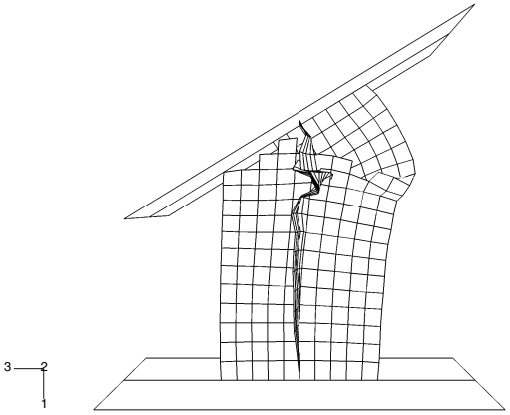


Figure 7: Deformed shape at 7.0 msec for the model with material failure and surface erosion.

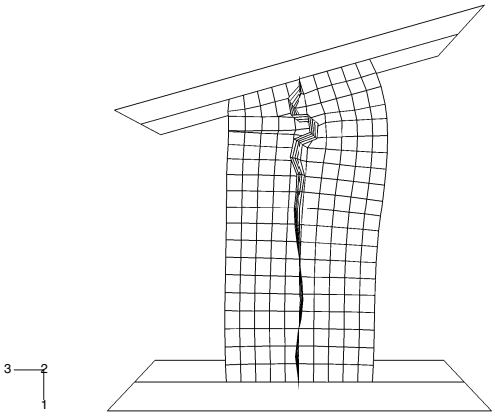


Figure 8: Deformed shape with 10% imperfection at 3.5 msec.

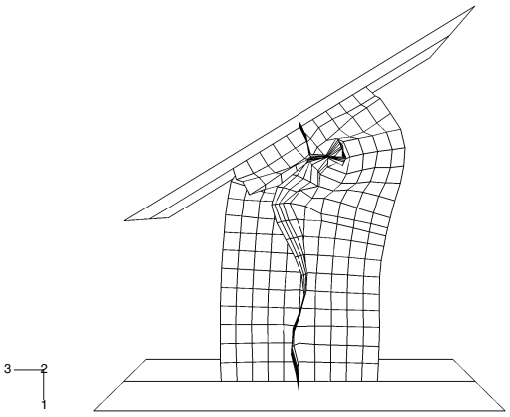


Figure 9: Deformed shape with 10% imperfection at 7.0 msec.

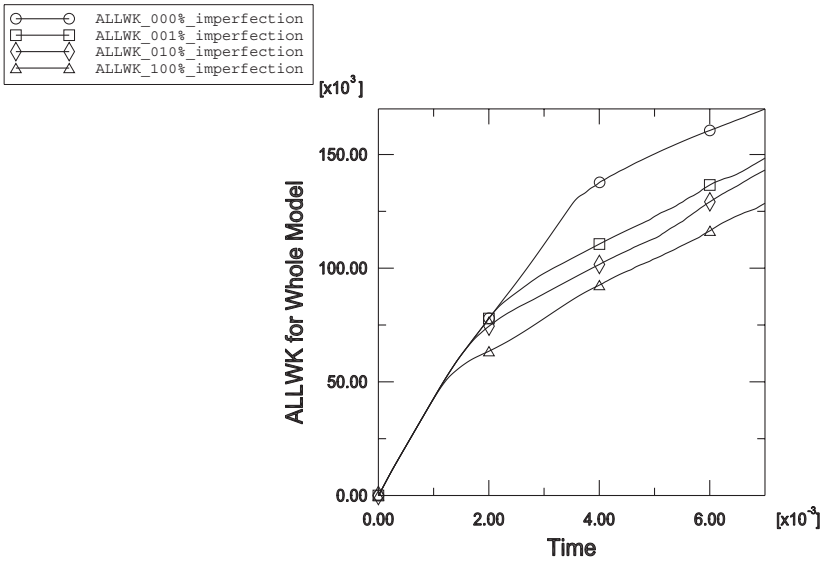


Figure 10: Time histories of external work for the column without any imperfection and for the column with imperfections of 1%, 10%, and 100% of shell thickness.

Dynamic stress/displacement analysis

In this section:

- *Subspace dynamic analysis of a cantilever beam*
- *Double cantilever elastic beam under point load*
- *Explosively loaded cylindrical panel*
- *Free ring under initial velocity: comparison of rate-independent and rate-dependent plasticity*
- *Large rotation of a one degree of freedom system*
- *Motion of a rigid body in Abaqus/Standard*
- *Rigid body dynamics with Abaqus/Explicit*
- *Revolute MPC verification: rotation of a crank*
- *Pipe whip simulation*
- *Impact of a copper rod*
- *Frictional braking of a rotating rigid body*
- *Compression of cylindrical shells with general contact*
- *Steady-state slip of a belt drive*
- *Crash simulation of a motor vehicle*
- *Truss impact on a rigid wall*
- *Plate penetration by a projectile*
- *Oblique shock reflections*

Subspace dynamic analysis of a cantilever beam

Products: Abaqus/Standard

Overview

This example shows the basic verification of the “subspace projection” procedure provided for solving mildly nonlinear dynamic problems.

The method in this example uses the eigenmodes of the system in its state at the start of the dynamic analysis as a set of global interpolation functions for the nonlinear problem. The discretized equations of motion are projected onto these eigenvectors and solved for the generalized modal accelerations, which are integrated by the central difference operator. The advantage of the subspace projection method in solving nonlinear dynamic problems is the relatively low cost of performing the analysis. However, the method is effective only if enough eigenmodes of the initial system can be extracted to provide a good basis for modeling the system's response throughout the dynamic event. This consideration usually limits the use of this method to mildly nonlinear cases, or to relatively small systems from which enough modes can be easily extracted to provide an accurate solution.

The example deals with the dynamic response of a cantilever beam subjected to a time varying base acceleration. The beam is rigidly supported at one end and has nonlinear elastic supports at the other end, as shown in *Figure 1*. The only nonlinearity in the problem is the contact between the beam and the elastic supports: geometric nonlinearity is neglected, and the response of the system is purely elastic. The problem has been analyzed by Shah et al. (1979) using a similar modal superposition method, with many modes, so that an accurate prediction of the response is available. The problem is also analyzed here using the standard direct, implicit integration method provided in Abaqus.

Problem description

The dimensions and material properties for the beam are given in *Figure 1*. The beam is modeled with 20 equal-sized linear beam elements (B21). One ITSUNI element models the nonlinear elastic supports at the end of the beam. The material definition of this element is given with the spring definition and describes an initial clearance on both sides of the beam in the vertical direction of 12.7×10^{-5} mm (0.5×10^{-5} in) and a spring rate of 35025 kN/m (2.0×10^{5} lb/in).

Since the relative motion of the beam with respect to the base is required, the base acceleration is introduced as a vertical distributed load applied to the entire length of the beam. The load changes with time, reaching its minimum of −7.005 N/m (−0.04 lb/in) at 0.011 sec and maximum of 7.005 N/m (0.04 lb/in) at 0.016 sec. This corresponds to an acceleration of −0.254 m/sec^2 (−10.0 in/sec ^{2}) at 0.011 sec and 0.254 m/sec^2 (10.0 in/sec^2) at 0.016 sec. The load is varied using the amplitude curve shown in *Figure 1*.

Analysis

The problem is analyzed using both the subspace projection method and the standard implicit integration method provided in Abaqus, using a fixed time increment. The subspace projection uses six eigenmodes. The choice of the number of eigenmodes used as the basis of the subspace solution determines the accuracy of the dynamic solution and is a matter of judgment on the part of the user, similar to choosing the number of finite elements in the mesh. If very few eigenmodes are specified, the solution will miss the high frequency response or will fail to represent nonlinearities accurately. The only reliable method of determining how many modes are needed is to repeat the analysis with more modes and observe the change in response. In this example only a small difference is noted between the solution obtained with two eigenmodes and that obtained using six eigenmodes.

The first step extracts the eigenmodes of the unloaded structure. The second step begins the dynamic analysis. The amplitude curve specifies that no load is applied until 0.001 sec. The analysis up to that time could be

performed in one increment since the structure is at rest over this time period. However, in this example two steps are used to reach 0.001 sec, the first of these two steps being over a very short time period, 10^{-6} sec. The purpose of this step is simply to obtain a solution point for plotting purposes at a time close to 0 sec. The second of these preliminary dynamic steps brings the analysis to 0.001 sec.

Results and discussion

Both analyses are run for 0.019 seconds of response with a time increment of 3.125×10^{-5} seconds. The calculated vertical displacements at node 10 (near the midspan of the beam) and at node 21 (at the supported end) are stored on the results file and are plotted as functions of time using the Abaqus postprocessing capability. This plot is shown in *Figure 2* and shows the results agreeing very closely with those obtained by Shah et al. (1979).

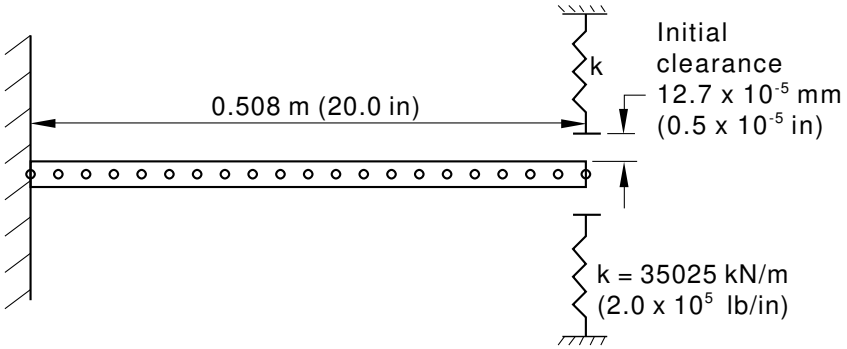
Input files

subdyncanti_itsuni.inp	Subspace procedure.
subdyncanti_itsuni_direct.inp	Direct, implicit procedure.
subdyncanti_itsuni_fvdepspring.inp	Identical to subdyncanti_itsuni.inp, except that field-variable-dependent nonlinear spring properties are used in the ITSUNI element.
subdyncanti_itscyl.inp	Identical to subdyncanti_itsuni.inp, except that the ITSCYL element is used.
subdyncanti_itscyl_direct.inp	Identical to subdyncanti_itsuni_direct.inp, except that the ITSCYL element is used.
subdyncanti_itscyl_fvdepspring.inp	Identical to subdyncanti_itsuni_fvdepspring.inp, except that the ITSCYL element is used.

References

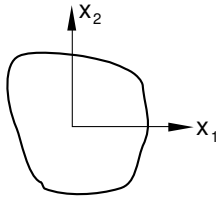
- Shah, V. N., G. J. Bohm, and A. N. Nahavandi, “Modal Superposition Method for Computationally Economical Nonlinear Structural Analysis,” *ASME Journal of Pressure Vessel Technology*, vol. 101, pp. 134–141, 1979.

Figures



Beam general section properties:

Young's modulus	206.84 GPa (30.0×10^6 lb/in^2)
Shear modulus	79.562 GPa (11.54×10^6 lb/in^2)
Density	4.275×10^4 kg/m^3 (0.004 lb-s^2/in^4)
Cross-sectional area	645 mm^2 (1.0 in^2)



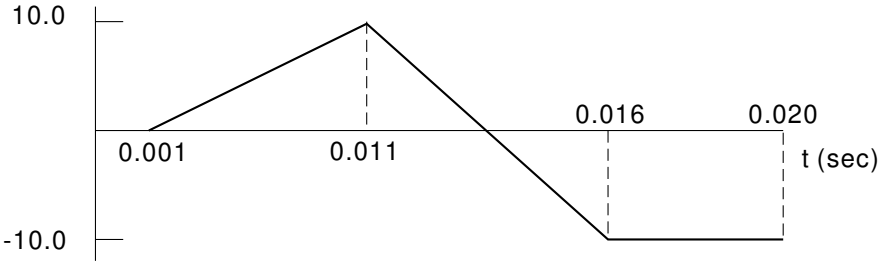


Figure 1: Beam geometry and amplitude curve of applied load.

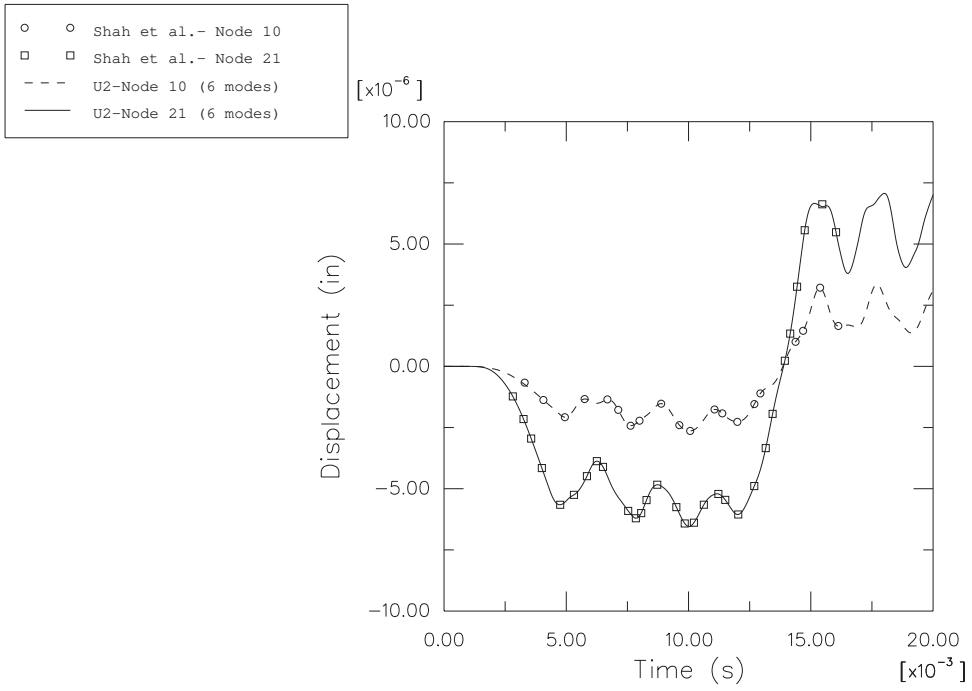


Figure 2: Cantilever displacement history.

Double cantilever elastic beam under point load

Products: Abaqus/Standard

Overview

This example illustrates the effect of time step choice on solution accuracy, compares direct and automatic time stepping, and verifies that the standard Newton and quasi-Newton solution techniques provide the same results in a relatively nonlinear case.

This example concerns the response of an elastic beam, built-in at both ends, subject to a suddenly applied load at its midspan (see *Figure 1*). The central part of the beam undergoes displacements several times its thickness, so the solution quickly becomes dominated by membrane effects that significantly stiffen its response.

A number of factors are involved in controlling solution accuracy in a nonlinear dynamic problem. First, the geometry must be modeled with finite elements, which involves a discretization error. In this example the beam is modeled with five elements of type B23 (cubic interpolation beam for planar motion). Since a 10 element model gives almost the same response, we assume that this model is reasonably accurate. Second, the time step must be chosen. This source of error is studied in this example by comparing results based on different time steps and different tolerances on the automatic time stepping scheme. Third, convergence of the nonlinear solution within each time step must be controlled. This aspect of solution control is common to all nonlinear problems.

The quasi-Newton solution technique can be less expensive in terms of computer time than the standard Newton technique because it avoids the complete recalculation of the Jacobian. Each newly computed Jacobian is based on the current Jacobian. This savings becomes significant in large models, in cases when the Jacobian is expected to vary smoothly over time. This example is too small for the quasi-Newton method to show significant savings in computer time, but it demonstrates that, with correctly chosen tolerances, the quasi-Newton method solves the nonlinear system with no loss in accuracy.

Problem description

The double cantilever beam has a span of 508 mm (20 in), with a rectangular cross-section 25.4 mm (1 in) wide by 3.175 mm (0.125 in) deep. The material is linear elastic, with a Young's modulus of 206.8 GPa (30×10^6 lb/in^2) and a density of 2710.42 kg/m^3 (2.5362×10^{-4} lb-s^2/in^4). Five elements of type B23 (cubic interpolation, beam in a plane) are used to model half the beam. The boundary conditions are that all displacements and rotations are fixed at the built-in end, with symmetry conditions ($u_x = \phi_z = 0$) at the midspan. A beam section is used with a 3-point Simpson rule for the cross-section integration. This integrates the section exactly since it is rectangular and remains linear elastic. Since the material response in this case is entirely linear, a general beam section would be preferred in a practical example, since it reduces the cost of the computation by avoiding numerical integration across the section.

Results and discussion

Nine different cases are run: fixed time steps of 25 μs, 50 μs, and 100 μs and automatic time stepping with half-increment residual tolerances of 44.48 N (10 lb), 222.4 N (50 lb), and 4448 N (1000 lb) using both the standard Newton and quasi-Newton solution techniques (the methods give almost identical results, as must occur since the default equilibrium tolerances are fairly stringent). Results for the displacement at the midspan are shown in *Figure 2* for the fixed time step cases and in *Figure 3* for the automatic time step cases. All of the results are based on the default integration operator in Abaqus: Hilber-Hughes, with $\alpha = -0.05$ (slight numerical damping). The loss of high frequency response with coarser time stepping and the generally high quality of the automatic time stepping solutions can be recognized, even for the case with the most coarse tolerance on the

half-increment residual (with a value of about three times the load). *Figure 4* shows peak entries in the half-increment residual vector at each time step for the fixed time step cases. This figure illustrates the value of the half-increment residual concept as an error indicator: the larger time increments increase the half-increment residual values dramatically. For the 50 μs time increment these residuals are initially large but decay with time because the slight numerical damping introduced in the integration operator removes the high frequency content in the solution with time.

In many nonlinear analyses it is informative to print the energy balance. In this case it allows us to assess how much energy has been lost through numerical damping. *Table 1* and *Table 2* show the energy values at the end of each of these runs and indicates that the most accurate solutions have energy errors of 0.7%, while the least accurate shows an energy balance error of 9.7%. The energy loss values for the automatic time increment runs suggest that these analyses are consistently more accurate than the analyses run with fixed time increments.

Input files

doublecant_haftol10_newton.inp	Automatic time stepping (HAFTOL=10) and the standard Newton solution technique.
doublecant_dt25_newton.inp	Fixed time stepping (DT=25 × 10^{-6}) and the standard Newton solution technique.
doublecant_haftol10_qnewton.inp	Automatic time stepping (HAFTOL=10) and the quasi-Newton solution technique.

Note that the restart option is invoked in the above input files. This is almost essential in any significant nonlinear problem. The output edit features are used extensively to control the printed output and to produce a results file. This allows the postprocessor to be used to generate time history plots, such as those shown in *Figure 2* and *Figure 3*.

doublecant_haftol50_newton.inp	Automatic time stepping (HAFTOL=50).
doublecant_haftol1000_newton.inp	Automatic time stepping (HAFTOL=1000).
doublecant_dt50_newton.inp	Fixed time stepping (DT=50 × 10^{-6}).
doublecant_dt100_newton.inp	Fixed time stepping (DT=100 × 10^{-6}).
doublecant_haftol50_qnewton.inp	Automatic time stepping (HAFTOL=50) with the quasi-Newton technique.
doublecant_haftol1000_qnewton.inp	Automatic time stepping (HAFTOL=1000) with the quasi-Newton technique.

Tables

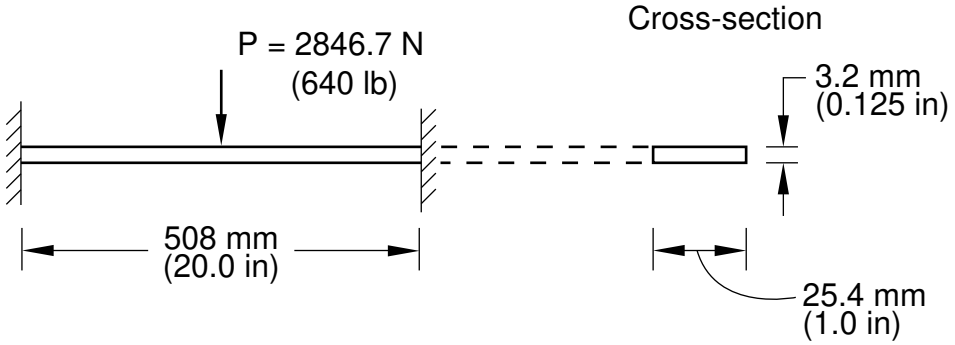
Table 1: Energy balance at end of run—analyses with fixed time increments.

Time increment	Kinetic energy		Strain energy		External work		Numerical energy loss
	N-m	lb-in	N-m	lb-in	N-m	lb-in	
25 μs	5.56	49.2	19.10	169	24.86	220	0.8%
50 μs	5.59	49.5	16.95	150	23.16	205	2.7%
100 μs	6.23	55.2	13.56	120	21.92	194	9.7%

Table 2: Energy balance at end of run—analyses with automatic time increments.

Half-increment tolerance	Kinetic energy		Strain energy		External work		Numerical energy loss
	N-m	lb-in	N-m	lb-in	N-m	lb-in	
44.5 N (10 lb)	4.80	42.5	20.23	179	25.20	223	0.7%
222 N (50 lb)	5.61	49.6	18.65	165	24.64	218	1.6%
4448 N (1000 lb)	4.77	42.2	15.49	137	21.93	194	7.6%

Figures



Material: elastic

Young's modulus = 206.8 GPa (30×10^6 lb/in^2)

density = 2714 kg/m^3 (2.54×10^{-4} lb s^2/in^4)

Figure 1: Double cantilever elastic beam.

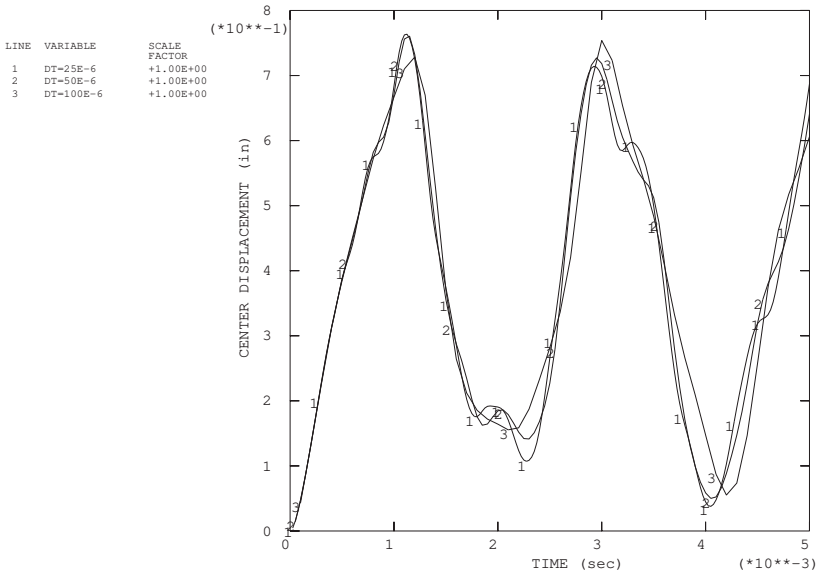


Figure 2: Fixed time step results for an elastic beam under point load.

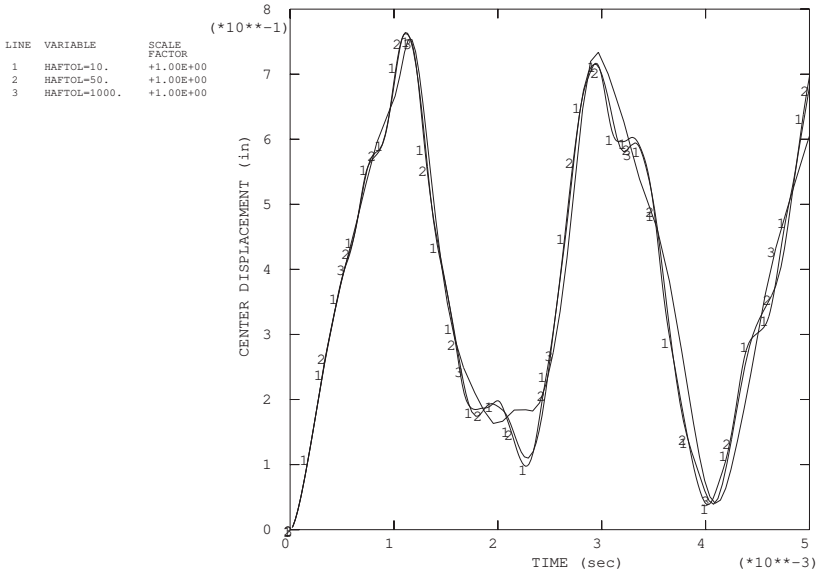


Figure 3: Automatic time step results for an elastic beam under point load.

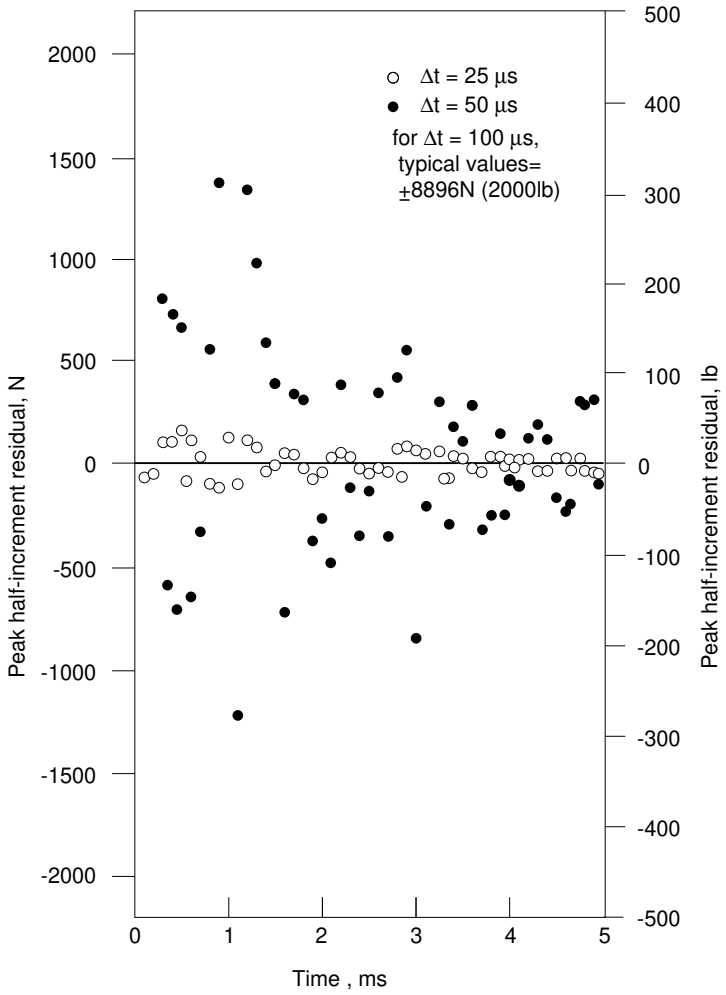


Figure 4: Peak half-increment residuals for the elastic beam under point load.

Explosively loaded cylindrical panel

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example shows the use of initial velocity conditions to model sudden, impulsive loadings arising from an explosion.

A cylindrical shell panel, firmly clamped on all four sides, is exposed to the detonation of a high explosive layer. In the course of the analysis a strong plastic hinge forms along the edge of the detonation area. Both experimental and numerical results for this problem have been reported by Leech (1966) and Morino et al. (1971).

Problem description

The panel is 319 mm (12.56 in) long and spans a 120° sector of a cylinder, with a midsurface radius of 74.6 mm (2.938 in) and a thickness of 3.18 mm (0.125 in). Only 60° of the panel is modeled because of the symmetry of the problem. Clamped boundary conditions are prescribed on three edges of the model, while the appropriate symmetry conditions are imposed along the remaining edge.

The shell is made from 6061-T6 aluminum alloy with a Young's modulus of 72.4 GPa (10.5×10^6 psi), a Poisson's ratio of 0.33, and a density of 2672 kg/m^3 (2.5×10^{-4} lb sec^2 in^{-4}). A von Mises elastic, perfectly plastic material model is used with a yield stress of 303 MPa (4.4×10^4 psi).

In the experiment the high explosive layer covers a 60° sector of the panel, extending 259 mm (10.21 in) from one end. Hence, there is no symmetry plane along the *y*-axis. All nodes in contact with the high explosive layer have been grouped in a node set named `BLAST`. The effect of the detonation is simulated by prescribing an initial inward radial velocity of 144 m/sec (5650 in/sec) to the nodes in this set.

All the relevant shell element types available in Abaqus/Standard are used in the simulation for comparative purposes and as a gauge of the relative merits of each element type for this class of problem. An 8×16 mesh is used for first-order elements, and a 4×8 mesh is used for second-order elements.

The Abaqus/Explicit analysis is performed using the finite-strain element, S4R, for three different mesh refinements (8×32, 16×32, and 32×64) and the small-strain elements, S4RS and S4RSW, for a 32×64 mesh. Geometrically equivalent analyses employing a shell offset with a value of 0.5 are performed using each of the quadrilateral shell elements in Abaqus/Explicit for a 32×64 mesh refinement. In addition, an analysis is performed with a 16×32 mesh of S4R elements using ENHANCED hourglass control.

Controls and tolerances

For the Abaqus/Standard analysis we choose to set the time integration accuracy control parameter (HAFTOL) to a very large (essentially infinite) value. This implies that we are choosing automatic control for the time stepping, but we are not controlling the accuracy of the time integration. The time increments will be limited only by the ability of the Newton scheme to solve the nonlinear equilibrium equations. This is a common technique for obtaining low-cost solutions for highly dissipative, strongly nonlinear cases. It is effective because the nonlinearities limit the time increments, and the high level of dissipation quickly removes the high frequency content from the solution. In practice it is desirable to verify the results with a second, more expensive, analysis in which a realistic value of HAFTOL is used. Default controls are used in Abaqus/Explicit.

Results and discussion

In both the experimental results and the Abaqus simulations, peak deflection occurs after about 400 μs. *Figure 1* shows deformed configuration plots for the S4R5 model and the S9R5 model after 400 μs of response time. *Figure 2* shows the deformed shapes at 400 μs for the three meshes used in the Abaqus/Explicit analysis.

The calculated values for the maximum deflection at a point midway along the centerline of the panel are reported for each of the analysis cases in *Table 1*. The experimental result for the maximum deflection reported by Morino et al. (1971) is also included for comparison. The mode of deformation in the problem is predominantly bending, and the second-order element models outperform the first-order element models for similar-cost analyses in Abaqus/Standard. These meshes are quite coarse, and improved performance is observed in Abaqus/Explicit upon mesh refinement. The results suggest that the 16 × 32 mesh of first-order elements provides a reasonably accurate solution for the maximum deflection. In addition, the results obtained using ENHANCED hourglass control closely match those obtained using the default hourglass control formulation.

Input files

Abaqus/Standard input files

exploadcylpanel_s3r.inp	S3R shell model.
exploadcylpanel_s4.inp	S4 shell model.
exploadcylpanel_s4r.inp	S4R shell model.
exploadcylpanel_s4r5.inp	S4R5 shell model.
exploadcylpanel_s8r.inp	S8R shell model.
exploadcylpanel_s8r5.inp	S8R5 shell model.
exploadcylpanel_s9r5.inp	S9R5 shell model.
exploadcylpanel_stri65.inp	STRI65 shell model.

Abaqus/Explicit input files

cylpa32x64.inp	S4R elements, fine mesh case.
cylpa8x32.inp	S4R elements, 8 × 32 mesh.
cylpa16x32.inp	S4R elements, 16 × 32 mesh.
cylpa16x32_enh.inp	S4R elements, 16 × 32 mesh, ENHANCED hourglass control.
cylpa32x64_s4rs.inp	S4RS elements.
cylpa32x64_s4rsw.inp	S4RSW elements.
cylpa32x64_offset.inp	S4R analysis, shell offset.
cylpa32x64_s4rs_offset.inp	S4RS analysis, shell offset.
cylpa32x64_s4rsw_offset.inp	S4RSW analysis, shell offset.
cylpa128x256.inp	Additional high mesh refinement case included for the sole purpose of testing the performance of the code.

References

- Leech, J. W., "Finite-Difference Calculation Method for Large Elastic-Plastic Dynamically-Induced Deformations of General Thin Shells," *Ph.D. Thesis, Dept. of Aeronautics and Astronautics, Massachussetts Institute of Technology, Cambridge, MA*, 1966.

- Morino, L., J. W. Leech, and E. A. Witmer, “An Improved Numerical Calculation Technique for Large Elastic-Plastic Transient Deformations of Thin Shells: Part 2—Evaluation and Applications,” *Journal of Applied Mechanics*, vol. 38, pp. 429–436, 1971.

Tables

Table 1: Maximum deflection along centerline of the panel at y=159.5 mm (6.28 in).

Code	Element Type	Mesh Size	POISSON	OFFSET	Maximum Deflection	
					mm	in
Abaqus/Standard	S4	8 × 16	0	0	28.7	1.13
Abaqus/Standard	S4	8 × 16	0.5	0	27.4	1.08
Abaqus/Standard	S4R	8 × 16	0	0	29.0	1.14
Abaqus/Standard	S4R	8 × 16	0.5	0	27.7	1.09
Abaqus/Standard	S4R5	8 × 16	0.5	0	30.5	1.20
Abaqus/Standard	S3R	8 × 16	0.5	0	31.2	1.23
Abaqus/Standard	S8R	4 × 8	0.5	0	31.2	1.23
Abaqus/Standard	S8R5	4 × 8	0.5	0	31.2	1.23
Abaqus/Standard	S9R5	4 × 8	0.5	0	31.5	1.24
Abaqus/Standard	STRI65	4 × 8	0.5	0	31.5	1.24
Abaqus/Explicit	S4R	8 × 32	—	0	26.2	1.03
Abaqus/Explicit	S4R	16 × 32	—	0	31.1	1.23
Abaqus/Explicit	S4R(enhanced hourglass)	16 × 32	—	0	30.7	1.21
Abaqus/Explicit	S4R	32 × 64	—	0	30.9	1.22
Abaqus/Explicit	S4R	32 × 64	—	0.5	29.8	1.18
Abaqus/Explicit	S4RS	32 × 64	—	0	31.1	1.23
Abaqus/Explicit	S4RS	32 × 64	—	0.5	29.2	1.15
Abaqus/Explicit	S4RSW	32 × 64	—	0	31.2	1.23
Abaqus/Explicit	S4RSW	32 × 64	—	0.5	29.3	1.15
Experimental	—	—	—	—	31.8	1.25

Fi

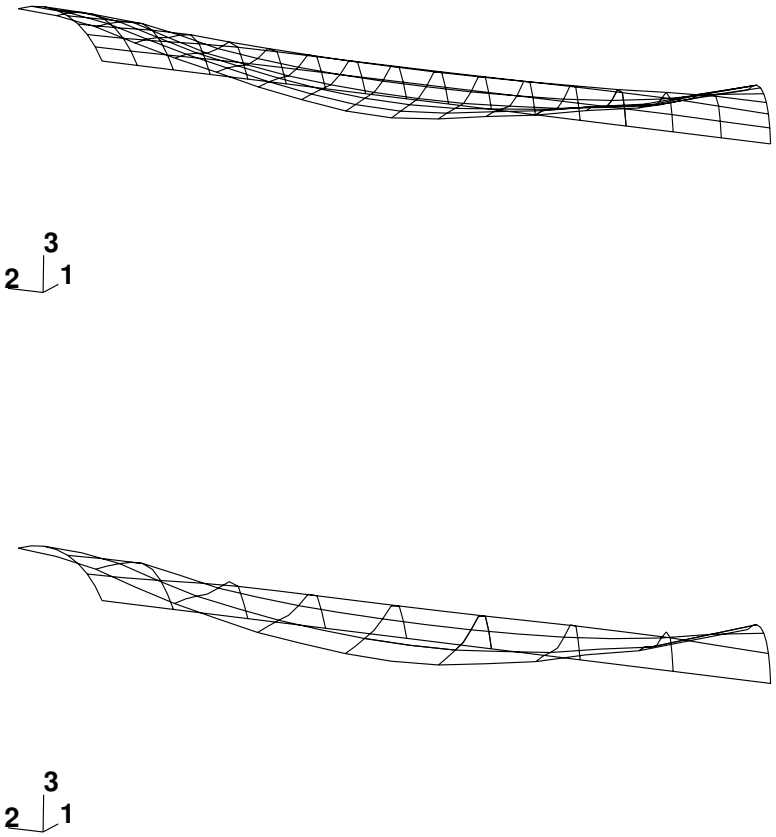


Figure 1: S4R5 and S9R5 deformed configurations at 400 μs (Abaqus/Standard).

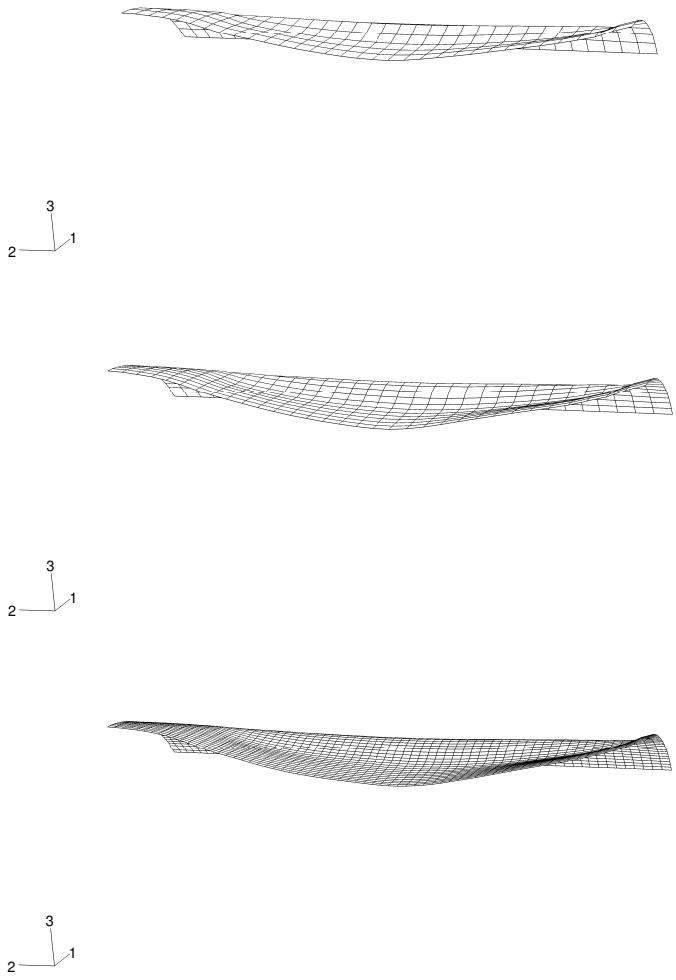


Figure 2: Deformed configurations for the 8 × 32, 16 × 32, and 32 × 64 meshes after 400 μs (Abaqus/Explicit).

Free ring under initial velocity: comparison of rate-independent and rate-dependent plasticity

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example shows the prediction of the transient response of a free circular ring subjected to a severe explosive loading over a 120° sector of its arc.

The problem in this example is interesting to study numerically because detailed, well-documented results of carefully performed experiments are available (Clark et al., 1962, and Witmer et al., 1963). Furthermore, the case is ideal experimentally because there are no boundary conditions: the ring is unconstrained. Thus, the only possible causes for discrepancy between analysis and experiment are the approximations in the geometric and time-stepping discretizations, the constitutive assumptions, and the initial velocity measurement.

In this case we find remarkably good agreement between the numerical results obtained with a strain-rate-dependent (viscoplastic) model and the experimental results. It is presumed that this level of agreement is somewhat fortuitous, since some of the parameters used in the constitutive model are chosen rather arbitrarily. Nevertheless, the trend of the response is so clearly followed by the numerical model that the analysis is certainly encouraging. The primary purpose of the analysis, aside from acting as a benchmark, is to illustrate the sensitivity of the results to different constitutive models, in this case by comparing rate-independent and rate-dependent plasticity models. To this end a reasonably fine geometric model and close tolerance on the automatic time stepping scheme are used to reduce the possibility of these discretizations giving rise to significant errors.

Problem description

The model is shown in *Figure 1*. The ring has an outer diameter of 152.4 mm (6 in) and thickness of 3.15 mm (0.124 in). The width of the ring is 30.36 mm (1.195 in). Half of the ring is modeled with 18 equal-sized elements, with symmetry boundary conditions at the ends of the model. B21 (linear interpolation beam for planar motion) elements are used in the Abaqus/Standard analysis; the Abaqus/Explicit analysis is first carried out with beam elements (B21) and then with shell elements (S4R). The cross-section integration (for material nonlinearity) is chosen as a seven-point Simpson rule: this should provide reasonable accuracy for a case like this where only a few cycles of reversal plasticity are expected.

The material is 6061–T6 aluminum alloy at room temperature. Its density is 2672 kg/m^3 (2.50×10^{-4} lb s^2/in^4). Young's modulus is assumed to be 72.4 GPa (10.5×10^6 lb/in^2), Poisson's ratio is 0.30, and the static yield stress is 295.1 MPa (42800 lb/in^2). Two plasticity models are used: one with no rate dependence, but isotropic strain hardening, with a constant tangent modulus of 542.6 MPa (78700 lb/in^2); and the standard elastic, viscoplastic model in Abaqus, with the static response assumed to be perfectly plastic and the yield stress given above. When the stress magnitude exceeds this static yield value, the plastic strain rate is given by

$$\dot{\bar{\varepsilon}}^{pl} = D\left(\frac{\bar{\sigma}}{\sigma^0} - 1\right)^p,$$

where $\bar{\sigma}$ is the magnitude of the stress, σ^0 is the static yield stress, $D =$ 6500 per second, and $p =$ 4.

Initial nodal velocities

The dynamic loading is prescribed by assigning initial velocities to the nodes in the 120° arc on which the explosive is detonated in the experiment. The values of these initial velocities are chosen as 174.1 m/s (6853 in/s) for all nodes except the node at the end of the arc (at the 60° point in the symmetric half-model), where a

value of 130.55 m/s (5139.7 in/s) is used. This is done because the velocity field contains a step discontinuity that cannot be reproduced exactly in the finite element model. We adjust the initial velocity at the node corresponding to the velocity discontinuity to match the total kinetic energy. This can be done analytically, since we know the element type (B21) chosen is based on linear interpolation, and so the velocity will vary linearly over each element. Alternatively we can match the energy by numerical trial and error (with some interpolation) by guessing values for this one nodal velocity and running one small dynamic increment, requesting the energy print. In this problem the value is chosen by trial and error, based on matching the initial kinetic energy in the discrete, finite element model to the actual initial kinetic energy in the experiment. The trials used are summarized in *Table 1*.

Solution controls in Abaqus/Standard

Automatic time stepping is used. An initial time step of 1μs is suggested, and the half-increment residual tolerance is set to 27600 N (6210 lb). This is based on a typical force value being the yield force in tension for the ring: about 27600 N (6210 lb). HAFTOL is set to this value to provide a dynamic solution of reasonable accuracy.

Results and discussion

The results for the two Abaqus/Standard analyses are shown in *Figure 2* and *Figure 3*. *Figure 2* shows the mean vertical diameter as a function of time, while *Figure 3* compares deformed shapes against the experimentally recorded shapes at 1.140 ms and at 2.580 ms. The results for the two-dimensional Abaqus/Explicit case using beam elements are shown in *Figure 4*. The original shape and the deformed shapes at 1.3 milliseconds and 2.6 milliseconds are shown. The results for the three-dimensional Abaqus/Explicit case using shell elements are shown in *Figure 5*. The original shape and the deformed shapes at 1.3 milliseconds and 2.6 milliseconds are shown. Results with pipe elements are consistent with those using beam elements.

These plots indicate that the analyses based on the rate-dependent yield model correlate quite well with the experiment: the configuration predictions in *Figure 3* are particularly strong evidence for this. However, as was pointed out above, whether 6061–T6 aluminum has much strain rate dependence is not well-established: the values used for D and p in the material model are rather arbitrary.

The sensitivity of structural problems of this type to rate dependence is apparent from the difference in the solutions shown here. This, combined with the difficulty of obtaining reliable measurements of the viscoplastic material behavior, points out a limitation on the reliability of such numerical solutions. It should be noted that the problem discussed here is an extreme case of high strain rates; larger, more massive structures (such as large pipes or automobile frames) should not see such high rates, except very locally.

The energy content at the end of the Abaqus/Standard runs is shown in *Table 2*. At this time (2.6 ms) in both cases about 74% of the total energy has been dissipated as plastic work. The total energy differs from the initial kinetic energy by only 0.02%, indicating that almost no numerical dissipation has occurred. This is because of the small values used for the half-increment residual tolerance and the consequent small time steps. The energy histories for the two-dimensional rate-independent Abaqus/Explicit case are shown in *Figure 6*. The energy histories for the three-dimensional rate-independent Abaqus/Explicit case are shown in *Figure 7*.

You can use a C++ program to reduce the amount of data in an output database by extracting results data from only specified frames and copying the data to a new output database that contains identical model data. An example of running this script for the output database generated by the three-dimensional rate-dependent case is given in *Decreasing the amount of data in an output database by retaining data at specific frames*.

Input files

Abaqus/Standard input files

freering_plastic.inp	Elastic-plastic model.

freering_viscoplastic.inp	Elastic, viscoplastic model.

Restart data are requested in both input files, as recommended in cases involving a fairly large number of time steps and nonlinearity to allow for recovery from unanticipated effects.

Abaqus/Explicit input files

ringb21.inp	Two-dimensional rate-independent case using beam elements.
ringb21_pipe_xpl.inp	Two-dimensional rate-independent case using pipe elements.
ringshell.inp	Three-dimensional rate-independent case using shell elements.
ringb21a.inp	Two-dimensional rate-dependent case using beam elements.
ringb21a_pipe_xpl.inp	Two-dimensional rate-dependent case using pipe elements.
ringshella.inp	Three-dimensional rate-dependent case using shell elements.
ringb31.inp	Three-dimensional rate-independent case using beam elements.
ringb31_pipe_xpl.inp	Three-dimensional rate-independent case using pipe elements.
ringb31a.inp	Three-dimensional rate-dependent case using beam elements.
ringb31a_pipe_xpl.inp	Three-dimensional rate-dependent case using pipe elements.

References

- Clark, E. N., R. H. Schmitt, and D. B. Ellington, “Explosive Impulse of Structures, Picatinny Arsenal,” *MIRP (33–616), G1–31*, no. 5 and 6, 1962.
- Witmer, E. A., H. A. Balmer, J. W. Leach, and T. H. H. Pian, “Large Dynamic Deformations of Beams, Rings, Plates and Shells,” *AIAA Journal*, vol. 1, no. 8, pp. 1848–1857, 1963.

Tables

Table 1: Initial velocity kinetic energy matching tests. Experimental kinetic energy value: 302.2 N-m (2675 lb-in).

Discrete model		60° node	
kinetic energy		radial velocity	
N-m	lb-in	m/s	in/s
287.7	2547.0	87.03	3426.5
303.3	2685.0	130.55	5139.7
306.2	2710.0	136.94	5391.5
311.6	2758.0	148.59	5850.0
The second row of the table is used in the analysis.			

Table 2: Energy totals at 2.6 ms.

Abaqus/Standard Model	Kinetic energy		Strain energy		Plastic work	
	N-m	lb-in	N-m	lb-in	N-m	lb-in
Viscoplastic	72.6	643	4.9	43.1	220.9	1955
Rate independent	74.4	659	2.2	19.1	221.9	1964

Figures

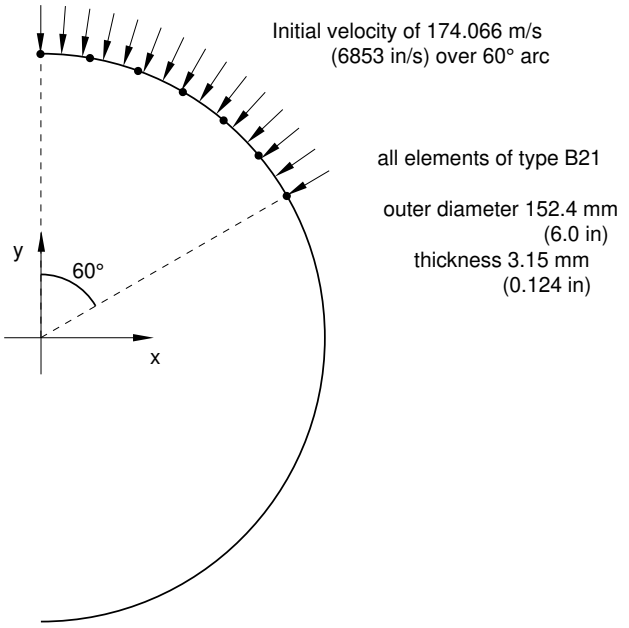


Figure 1: Mesh for Abaqus/Standard free ring problem.

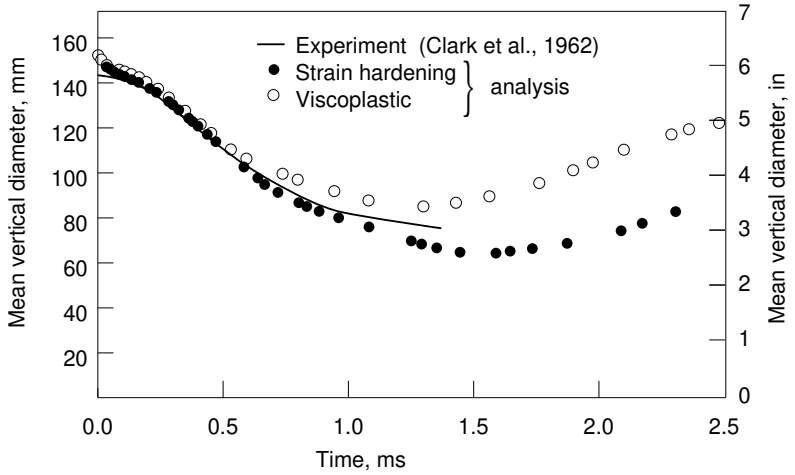


Figure 2: Mean diameter of the ring as a function of time, Abaqus/Standard.

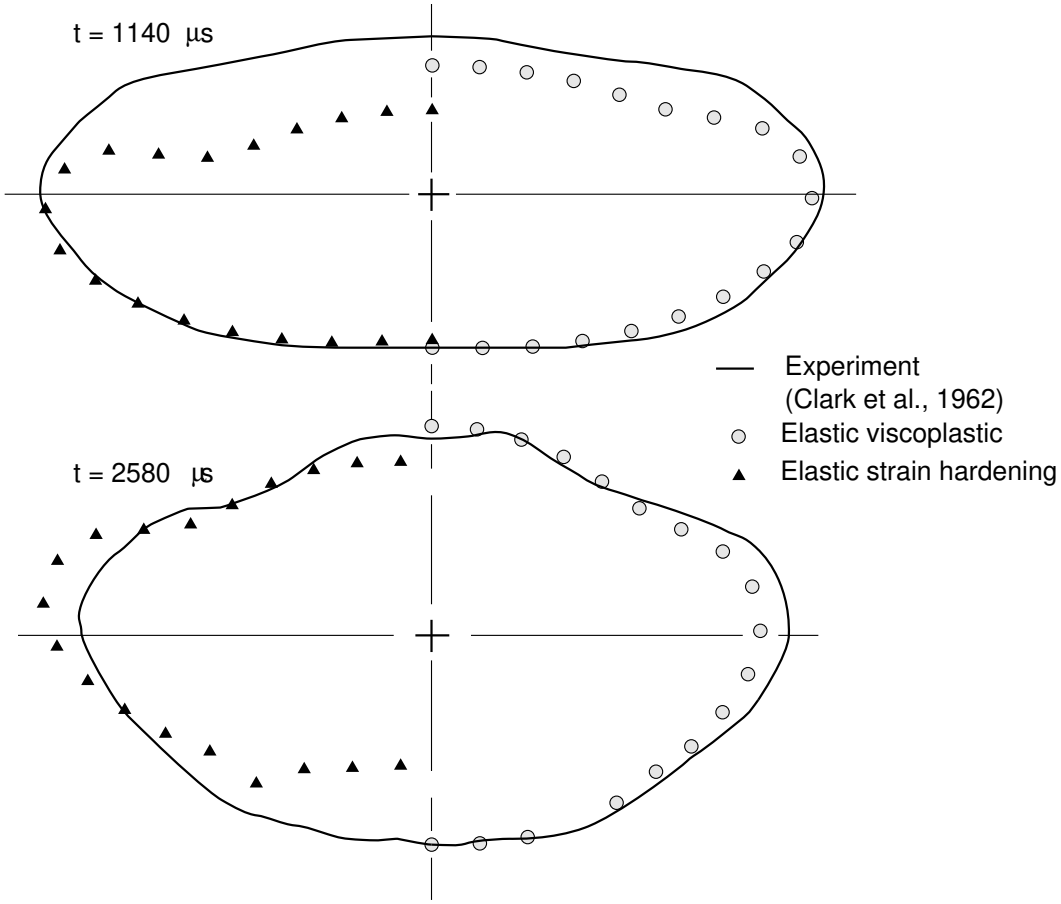


Figure 3: Comparison of predicted configurations for the ring, Abaqus/Standard.

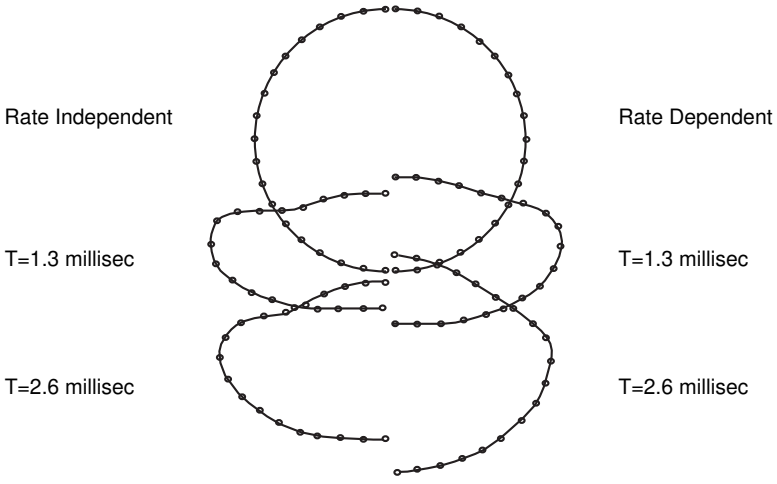


Figure 4: Original shape and deformed meshes for B21 elements, Abaqus/Explicit.

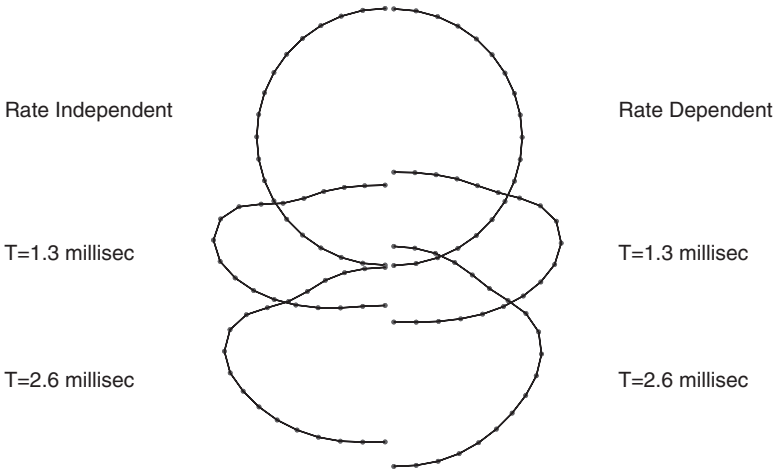


Figure 5: Original shape and deformed meshes for shell elements, Abaqus/Explicit.

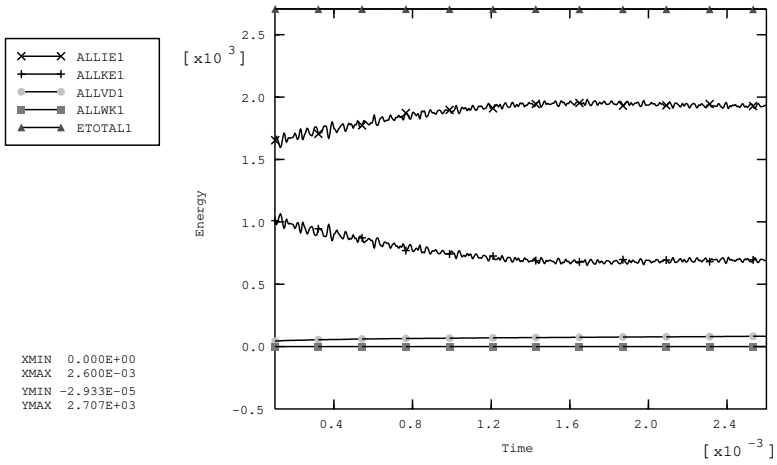


Figure 6: Energy histories for the beam model, Abaqus/Explicit.

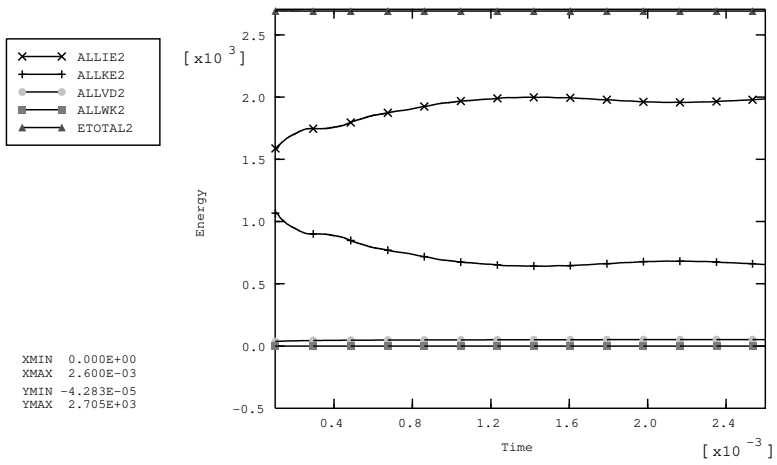


Figure 7: Energy histories for the shell model, Abaqus/Explicit.

Large rotation of a one degree of freedom system

Products: Abaqus/Standard

Overview

This problem shows an elementary example of a flexible-structure, large-rotation problem. Since it involves only one degree of freedom, it can be solved very simply in closed form. It, therefore, provides a convenient illustration of some aspects of geometrically nonlinear analysis.

Problem description

The problem is shown in *Figure 1*. A uniform rod, pinned at one end and free to slide in one direction at the other, is loaded so that it is initially compressed. We assume that the response of the rod is entirely linear elastic, so the only nonlinearity arises from rotation. We also assume that the initial height of the moving end of the rod above the horizontal position, h, is small compared to the horizontal distance between the supports, d, so that the strain in the rod is small so long as $\mathbf{0} \leq \boldsymbol{u} \leq \mathbf{2}\boldsymbol{h}$.

The solution clearly involves an instability, since a nonzero force is required to begin displacing the endpoint of the rod downward, but the force must drop back to zero as the rod becomes horizontal: this horizontal position is one of unstable equilibrium. Since the problem involves only one displacement variable, no bifurcation is possible, so the behavior is quite simple compared to what can happen in systems with many degrees of freedom whose response may involve instabilities. Moreover, the displacement variable is prescribed, so there are, in fact, no unknowns in this problem. To obtain a solution at regular displacement intervals, automatic time incrementation is switched off.

The structure exhibits nonlinear response throughout its deformation, unlike typical "stiff" shell-type structures that often behave in an almost linear fashion until they buckle. Therefore, this type of problem cannot be analyzed effectively with the eigenvalue buckling procedure. However, since an exact solution to the problem is readily developed (see below), the example is a useful illustration of a simple, geometrically nonlinear analysis.

Two simple models are possible with Abaqus—one using a single truss element of type T2D2, and one using a SPRING element. There are two differences between these two models. One is the way strain is measured. Because the truss element is usually used with the standard constitutive models in Abaqus, it uses logarithmic strain. With the spring, the "strain" is calculated from the change in distance between its ends. The second difference is that the force in the truss is calculated as the stress times the area, and the area is updated as the truss deforms, using the assumption that the truss is incompressible and so has constant volume. In the spring, the force is defined immediately by the spring rate that is given in the input data times the "strain." The exact solutions are, therefore, not the same for the two models, but they show only minor differences because the dimensions are chosen so that the strains are small throughout the deformation. The differences would be significant if large strains were involved.

Exact solution: truss model

The strain in the truss is assumed to be uniform, so the logarithmic strain definition gives

$$\varepsilon = \ln\left(\frac{l}{L}\right),$$

where l is the current length of the truss and L is its original length. From the geometry of *Figure 1* we have the results

$$l^2 = d^2 + (h-u)^2$$

and

$$L^2 = d^2 + h^2,$$

so the strain is given in terms of the displacement as

$$\varepsilon = \frac{1}{2}\ln\left(1 + \frac{u(u-2h)}{L^2}\right),$$

and its first variation is

$$\delta\varepsilon = \frac{1}{l^2}(u-h)\,\delta u.$$

We assume the material of the truss responds in a linear elastic manner, so the stress is

$$\sigma = E\varepsilon,$$

where E is Young's modulus. Assuming that the initial cross-sectional area is A and the material is incompressible, the virtual work statement is

$$\int_L \sigma\,\delta\varepsilon A\,dL = P\,\delta u.$$

Since the strain and stress are uniform, the integral over the volume of the truss is

$$\int_L \sigma\,\delta\varepsilon A\,dL = AL\sigma\,\delta\varepsilon.$$

Introducing the above results for ε, $\delta\varepsilon$, and σ, this equation gives

$$P = \frac{AEL}{l^2}(u-h)\ln\left(\frac{l}{L}\right).$$

This equation is the static equilibrium equation for the system and is shown in *Figure 2*.
largerotation1dof_truss.inp shows this problem, loaded by prescribing the displacement u throughout the step. This gives exactly the above solution.

Exact solution: spring model

The force in the spring is defined to be

$$F = K\Delta,$$

where K is the spring stiffness given in the input data and Δ is the change in length of the spring. From the discussion above, we have

$$\Delta = \left[d^2 + (h-u)^2\right]^{1/2} - L,$$

so the variation in the change in length is

$$\delta\Delta = -\frac{(h-u)}{l}\delta u.$$

The principle of virtual work gives

$$F\,\delta\Delta = P\,\delta u.$$

Using the force–relative displacement relation and the Δ–u relationship in this expression gives the equilibrium equation

$$P = -\frac{K}{l}(h-u)\left\{\left[d^2 + (h-u)^2\right]^{1/2} - L\right\}.$$

largerotation1dof_spring.inp shows this version of the problem, also loaded by prescribing the displacement. This gives exactly the above response.

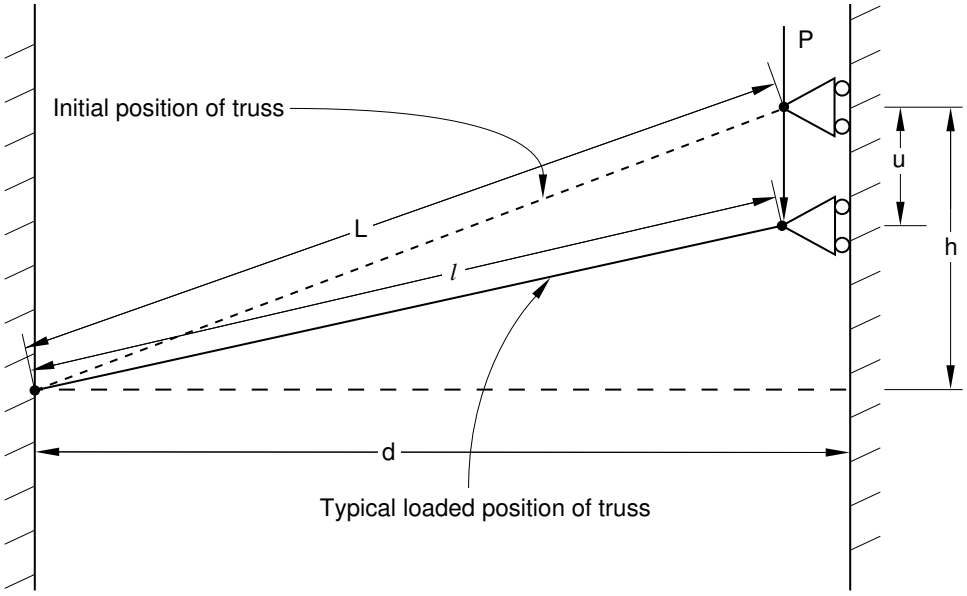
Results and discussion

The form of the equilibrium response is interesting because in some respects it typifies the response of some important practical cases. The initial response is stable and not very nonlinear. As the displacement increases, the system loses stiffness until a limit value of the load, P^c, is reached. The displacement at which this occurs is about 42% of h. Beyond that value the response is unstable (the system has negative stiffness) until, at a displacement of about 158% of h, it again becomes stable. (The critical displacement values and the corresponding load values can be estimated from the plot in *Figure 2* or can be computed exactly from the equilibrium equations given above.) For any load in the range $-P^c < P < P^c$ the system, thus, has three static equilibrium configurations, of which two are stable and one is unstable. Outside that range of loads the system has only one stable, static equilibrium configuration. We, thus, observe that, even in a simple elastic system with only one degree of freedom, uniqueness and stability of the solution are lost when geometric nonlinearity is introduced. In this simple case it is easy to obtain the equilibrium solution even in the unstable response phase by prescribing the only active degree of freedom of the system. In a more practical case the "Riks" algorithm must be used instead—such usage is illustrated in several other examples in this chapter.

Input files

largerotation1dof_truss.inp	Used to obtain the prescribed displacement results of *Figure 2* with the truss element.
largerotation1dof_spring.inp	Used to obtain the prescribed displacement results with the spring element.

Figures



Data used in the example:

Quantity	Value	Units
Young's modulus (E)	2000	force/length^2
Truss cross-sectional area (A)	1	length^2
Initial truss length (L)	10	length
Initial offset (h)	1	length

Figure 1: Elastic, large rotation truss example.

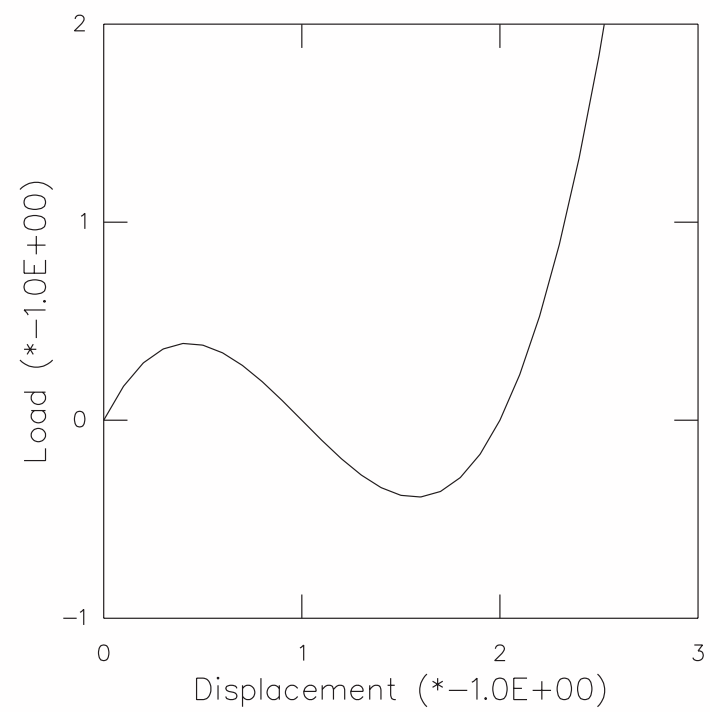


Figure 2: Load-displacement response for truss example.

Motion of a rigid body in Abaqus/Standard

Products: Abaqus/Standard

References:

- *Implicit dynamic analysis*
- *Rotary inertia element*

Overview

This problem illustrates the accuracy of the integration of rotations during implicit dynamic calculations on a rotating body whose rotary inertia is different in different directions.

We consider two cases of rigid body dynamics:

- force-free motion of a rigid body; and
- forced motion of a rigid body.

The Euler's equations for the motion of a rigid body in a rotating coordinate system attached to the body are

$$I_{11}\ddot{\phi}^1 + (I_{33} - I_{22})\dot{\phi}^2\dot{\phi}^3 = N_1,$$
$$I_{22}\ddot{\phi}^2 + (I_{11} - I_{33})\dot{\phi}^3\dot{\phi}^1 = N_2,$$
$$I_{33}\ddot{\phi}^3 + (I_{22} - I_{11})\dot{\phi}^1\dot{\phi}^2 = N_3.$$

In these relations $\dot{\boldsymbol{\phi}}$ is the body's angular velocity; $\ddot{\boldsymbol{\phi}}$ is its angular acceleration; I_{11}, I_{22}, I_{33} are the second moments of inertia along the principal axes of the body; and N_1, N_2, N_3 are the torque components acting on the rigid body.

Force-free motion of a rigid body

Problem description

The problem is shown in *Figure 1*. An arbitrary symmetric body whose rotary inertia about its axis of symmetry is different from its value along the two other principal axes spins around its axis of symmetry with an initial angular velocity $\boldsymbol{\omega}$ The body is modeled with a ROTARYI element whose second moments of inertia along its principal axes, $\mathbf{e}_\alpha$ ($\alpha = 1, 2, 3$), have the values $I_{11} = I_{22}$ and $I_{33} = 2I_{11}$. The axis of symmetry is $\mathbf{e}_3$. Dummy nodes are attached rigidly to the ROTARYI element along the principal axes by using a BEAMMPC so that their displacements can be tracked. Since ROTARYI elements have only rotational degrees of freedom, a MASS element is needed on top of the ROTARYI element to activate translational degrees of freedom at these dummy nodes. Initial conditions are taken from the analytical solution presented below.

For the force-free symmetric body $N_1 = N_2 = N_3 = 0$ and $I_{11} = I_{22}$; therefore, the Euler's equations reduce to

$$I_{11}\ddot{\phi}^1 + (I_{33} - I_{11})\dot{\phi}^2\dot{\phi}^3 = 0$$
$$I_{11}\ddot{\phi}^2 + (I_{11} - I_{33})\dot{\phi}^3\dot{\phi}^1 = 0$$
$$I_{33}\ddot{\phi}^3 = 0.$$

The last equation can be integrated to give $\dot{\phi}^3 = \omega$, where ω is a constant, defined as an initial condition of the problem.

To determine $\dot{\phi}^1$, we take the time derivative of the first equation:

$$I_{11}\frac{d\ddot{\phi}^1}{dt} + (I_{33} - I_{11})\ddot{\phi}^2\omega = 0.$$

Using the second equation to solve for $\ddot{\phi}^2$ gives

$$\frac{d\ddot{\phi}^1}{dt} + \left[\frac{(I_{33}-I_{11})\omega}{I_{11}}\right]^2\dot{\phi}^1 = 0.$$

Similarly,

$$\frac{d\ddot{\phi}^2}{dt} + \left[\frac{(I_{11}-I_{33})\omega}{I_{11}}\right]^2\dot{\phi}^2 = 0.$$

These equations describe simple harmonic motion with angular frequency

$$\Omega = \frac{(I_{33}-I_{11})}{I_{11}}\omega.$$

With appropriate initial conditions the solution is $\dot{\phi}^1 = -A\sin\Omega t$, where A is a constant. The corresponding solution for $\dot{\phi}^2$ can be found by substituting this solution for $\dot{\phi}^1$ into the first of the Euler equations, giving $\dot{\phi}^2 = A\cos\Omega t$. The corresponding initial conditions are $\dot{\phi}^1\big|_{t=0} = 0$, $\dot{\phi}^2\big|_{t=0} = A$, $\dot{\phi}^3\big|_{t=0} = \omega$. We also choose $\phi^1\big|_{t=0} = -A/\Omega$, $\phi^2\big|_{t=0} = 0$, and $\phi^3\big|_{t=0} = 0$. These initial rotation conditions give rise to the local orientation indicated in *Figure 1*. The directions of the principal axes of inertia of the body are defined. We choose $A = 0.25$, $I_{11} = I_{22} = 1$, $I_{33} = 2I_{11} = 2$, so that

$$\Omega = \frac{(I_{33}-I_{11})}{I_{11}}\omega = \omega = 2\pi.$$

Initial angular velocities, $\dot{\boldsymbol{\phi}}\big|_{t=0}$, must be applied to node 1, and translational velocities, $\mathbf{v}\big|_{t=0}$, must be applied to the dummy nodes lying along the legs of the axes of the body. The translational velocity components are obtained from

$$\mathbf{v} = \dot{\boldsymbol{\phi}} \times \mathbf{r},$$

where $\mathbf{r}$ is the vector connecting the center of the body (node 1) to one of the nodes along the principal axes (node 2, 3, or 4). This latter initial velocity calculation is performed internally by Abaqus for each dummy node as a result of applying the BEAMMPCs mentioned previously. The model is shown in *rigidbodymotion_free.inp*.

The dynamic response of the body subjected to the above initial conditions is tracked for two seconds. Large-rotation theory is used, so the principal axes of inertia rotate with the rotation of the ROTARYI element. Rigid body rotary inertia contributes nonsymmetric terms to the system matrix when the motion is in three dimensions. Therefore, we use the unsymmetric equation solver. Numerical damping is removed from the implicit dynamic operator.

Results and discussion

The harmonic response for the angular velocity relative to the global coordinate system is obtained in the Abaqus solution and is plotted in *Figure 2*, *Figure 3*, and *Figure 4*. These angular velocity values are obtained from node

1. Noting that $\dot{\phi}^z\big|_{t=0} = \dot{\phi}^3\big|_{t=0} \cos(A/\Omega) - \dot{\phi}^2\big|_{t=0} \sin(A/\Omega)$, $\dot{\phi}^z\big|_{t=0}$ can be calculated as 6.268. This is shown accurately in *Figure 4*.

The solutions for $\dot{\phi}^1$ and $\dot{\phi}^2$ obtained above indicate that the vector $\dot{\phi}^1 \mathbf{e}_1 + \dot{\phi}^2 \mathbf{e}_2$ is of constant magnitude and precesses about the body 3-axis with the angular frequency $\Omega = 2\pi$. The evolution of this vector with respect to the global coordinate system is plotted in *Figure 5* as an *X*–*Y* plot of the history of $\dot{\phi}^x$ versus the history of $\dot{\phi}^y$ for node 1. As expected, the result traces a circle of diameter *A*. *Figure 6* shows a similar plot of u^x versus u^y for node 4, viewed by looking down the global *z*-axis.

The precession described by Goldstein is relative to the body axes, which are themselves rotating in space at a frequency of 2π. In large-displacement analysis in Abaqus (geometric nonlinearities considered in the step) the principal axes of inertia rotate with the rotation of the node to which the ROTARYI element is attached. This explains why the period of the motion observed in the figures is 0.5 and not 1.0.

The analysis is completed in 200 increments, with each increment requiring only 1 iteration to satisfy the moment equilibrium criterion.

Input files

rigidbodymotion_free.inp	Implicit force-free motion analysis.

Figures

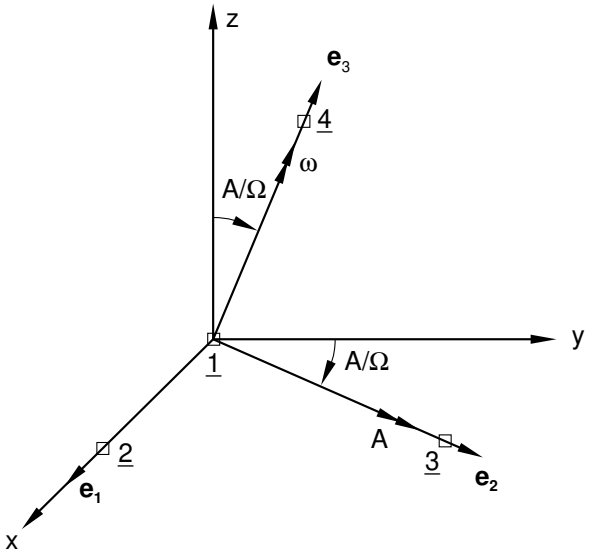


Figure 1: Rigid body rotation example.

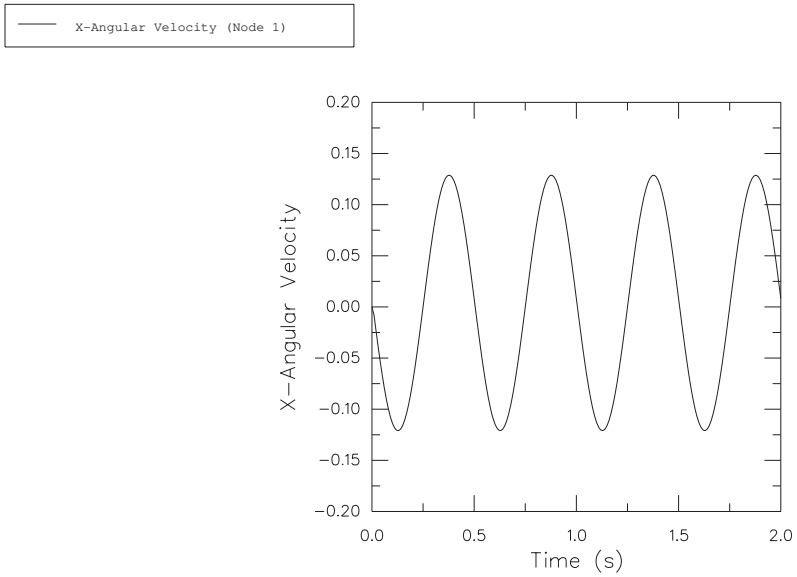


Figure 2: Angular velocity response (node 1).

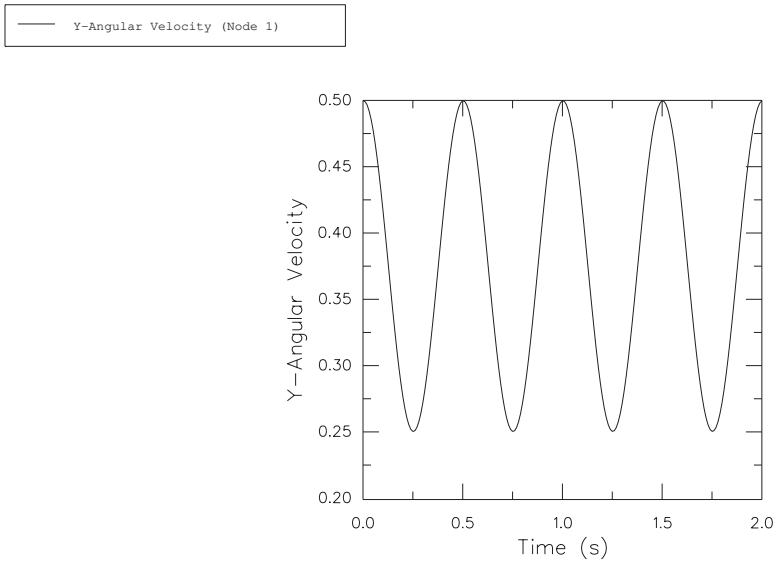


Figure 3: Angular velocity response (node 1).

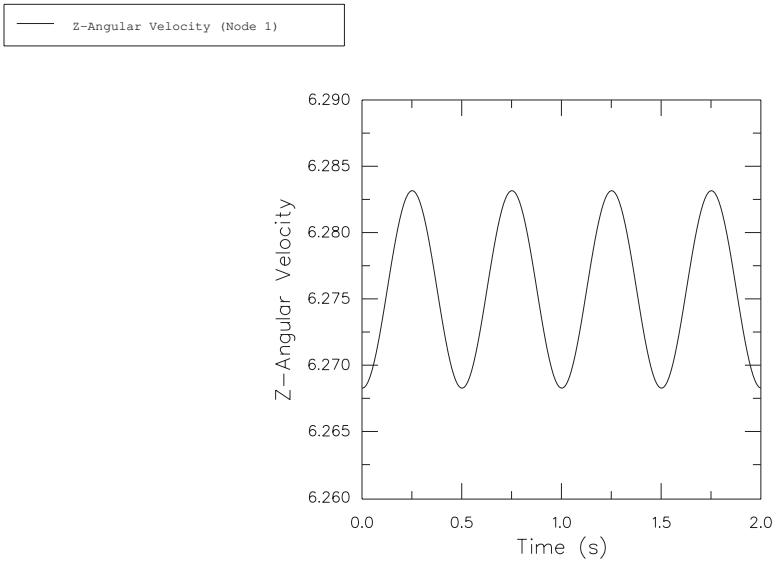


Figure 4: Angular velocity response (node 1).

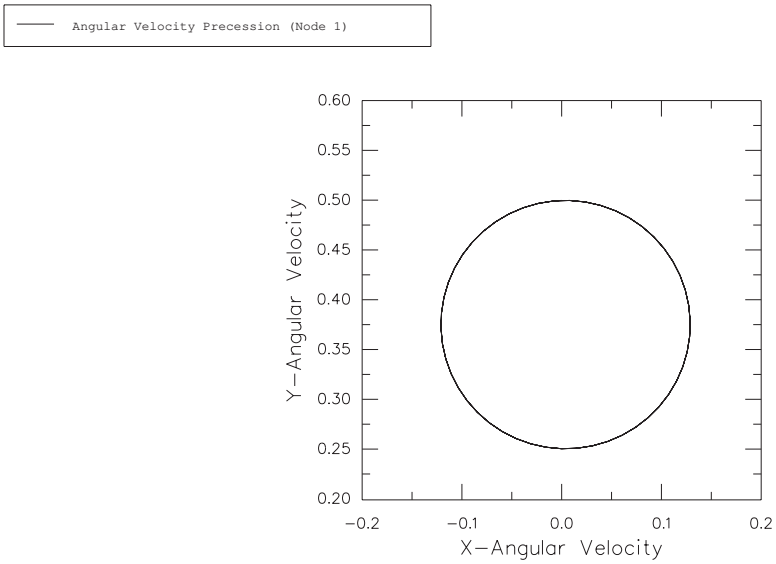


Figure 5: Precession of angular velocity (node 1).

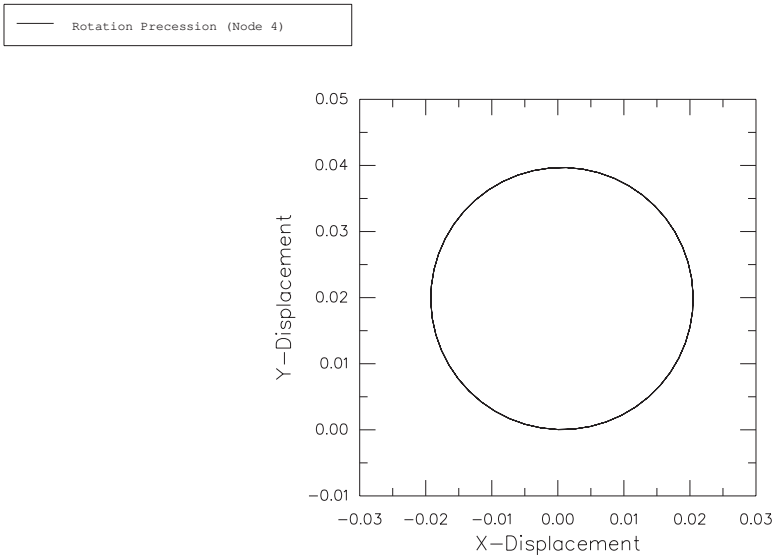


Figure 6: Precession of rotation (node 4).

Forced motion of a rigid body

Problem description

The top is modeled with a ROTARYI element, and a local coordinate system is used to prescribe the second moments of inertia along the principal axes $(\mathbf{e_1}, \mathbf{e_2}, \mathbf{e_3})$. A 2-node rigid beam element RB3D2 is used to connect the fixed point of the top, O, with its center of mass, C. The effect of the gravitational field is considered by applying a concentrated load of magnitude Mg in the z-direction at point C. The initial conditions for the angular velocity, $\dot{\phi}$, are prescribed in the global system of coordinates. For postprocessing and visualization purposes only, a second RB3D2 element is added at point C in a direction perpendicular to the axis of rotation. The Abaqus solution is compared to the analytical solution, which is outlined in the next section.

The problem is also solved using connector elements. A CONN3D2 element of type BEAM is used to model the top. A CONN3D2 element of type EULER is used to obtain the Euler angles.

Analytical solution

The solution for the motion of the symmetric top is described in Goldstein (1980), Whittaker (1988), and Macmillan (1936).

The analytical solution is described in terms of the Euler angles: θ, γ, ψ, where θ measures the inclination of the $\mathbf{e_3}$-axis from the vertical, γ measures the azimuth of the top about the vertical, and ψ is the rotation angle of the top around its own $\mathbf{e_3}$-axis. Since the system is conservative, the total energy is constant in time. By denoting $\omega_1 = \dot{\phi}^1, \omega_2 = \dot{\phi}^2$, and $\omega_3 = \dot{\phi}^3$, the energy conservation equation gives

$$E = KE + V = \frac{1}{2}\left({\omega_1}^2 + {\omega_2}^2\right) I_{11} + \frac{1}{2} I_{33} {\omega_3}^2 + Mgl\cos\theta = constant,$$

where $I_{11} = I_{22}$ since the body is symmetrical.

The energy equation can be arranged in the following form:

$$\dot{u}^2 = \left(1 - u^2\right)(\alpha - \beta u) - (b - au)^2,$$

where $u = \cos\theta$ and the constants a, b, α, β have the following form:

$$a = \frac{I_{33}}{I_{11}}\omega_3 \ ; \ b = \frac{K}{I_{11}} \ ; \ \alpha = \frac{2}{I_{11}}\left(E - \frac{I_{33}}{2}{\omega_3}^2\right) \ ; \ \beta = \frac{2Mgl}{I_{11}}.$$

In these relations K is the moment of momentum with respect to the z-axis. Its value is constant in time and is given by

$$K = I_{11}\omega_1\eta_1 + I_{22}\omega_2\eta_2 + I_{33}\omega_3\eta_3,$$

where in terms of Eulerian angles the direction cosines are

$$\eta_1 = \sin\theta\sin\psi \ ; \ \eta_2 = \sin\theta\cos\psi \ ; \ \eta_3 = \cos\theta.$$

The equation of motion for $u = \cos\theta$ is an elliptic function of time, and the integration is not straightforward since the function presents singularities.

We can arrange this equation in the following form $\dot{u}^2 = \beta f(u)$. The function $f(u)$ has two real roots $u_3 = \cos\theta_3$ and $u_2 = \cos\theta_2$ situated between $u = -1$ and $u = +1$. The third root $u_1 > u_2 > u_3$ is greater than +1. The top will move such that θ always remains between the roots θ_2 and θ_3, which are called turning angles.

The equation of motion for θ can be expressed in terms of these three roots as follows:

$$\dot{u}^2 = \beta(u - u_3)\ (u - u_2)\ (u - u_1).$$

By expressing the constants of integration in terms of the three roots, one can obtain the analytical solution of this equation by reducing the elliptic integral to a normal form. This solution is given in Macmillan (1936):

$$u = u_3 + 4h^{1/2}\sqrt{(u_1 - u_3)\ (u_2 - u_3)}\left[\frac{\sin\xi - h^2\sin 3\xi + h^6\sin 5\xi + \ldots}{1 - 2h\cos 2\xi + 2h^4\cos 4\xi - 2h^9\cos 6\xi + \ldots}\right]^2,$$

where

$$h = e^{\left(-\frac{\Omega_3}{\Omega_1}\pi\right)} \ ; \ \xi = \frac{\pi}{2\omega_1}\left(\frac{\sqrt{\beta}}{2}t + \Omega_3\right).$$

In the above equation Ω_1 and Ω_3 are elliptic integrals of the first kind and have the following expressions:

$$\Omega_1 = \frac{1}{\sqrt{(u_1 - u_3)}}\int_0^{\pi/2}\frac{d\phi}{\sqrt{1 - k^2\sin^2\phi}},$$

$$\Omega_3 = \frac{1}{\sqrt{(u_1 - u_3)}}\int_0^{\pi/2}\frac{d\phi}{\sqrt{1 - {k_1}^2\sin^2\phi}},$$

where

$$k^2 = \frac{u_2 - u_3}{u_1 - u_3} \ , \ {k_1}^2 = \frac{u_1 - u_2}{u_1 - u_3}.$$

The values of the elliptic integrals are usually tabulated in calculus books or in mathematical tables. As soon as the roots of the polynomial $f(u)$ are found, we know the solution for the equation of motion.

After determining $u = \cos\theta$ from the above equation, the remaining Euler's angles, γ and ψ, can be found from

$$\dot{\gamma} = \frac{b - a\cos\theta}{\sin^2\theta} \ ; \ \dot{\psi} = \frac{I_{11}a}{I_{33}} - \cos\theta\frac{b - a\cos\theta}{\sin^2\theta}.$$

The coordinates of the center of mass of the top in the x–y plane can be obtained if the first two Euler's angles θ and γ are known: $x = l\sin\theta\cos\gamma$ and $y = l\sin\theta\sin\gamma$.

Results and discussion

In this section we will present the comparative results between Abaqus and the analytical solution for two situations often discussed in the literature. Many different response characteristics are possible depending on the initial conditions and inertia properties.

Case 1

Let us consider first that the symmetric top is spinning about its own axis $\mathbf{e_3}$, which is fixed in some direction $\theta_0 = 20°$. At time $t = 0$ the symmetry or figure axis is released, and the top rotates around the $\mathbf{e_3}$-axis with angular velocity $\dot{\phi}^3 = 50$. In addition to the angular velocity around the symmetry axis, we prescribe an angular velocity $\dot{\phi}^1 = 0.5$ around the $\mathbf{e_1}$- or $\mathbf{e_x}$-axis. Usually the motion of the top is depicted by tracing the curve of the intersection of the $\mathbf{e_3}$-axis on a sphere of unit radius. This curve is called the "locus" of the figure axis. In our representation we will trace the projection of the locus in the *x*–*y* plane. According to the analytical solution, the ratio $u' = \frac{b}{a}$ lies between the roots u_2 and u_3, and the locus of the top axis exhibits loops (Goldstein, 1980).

We have chosen the length of the top axis $l = 1$ and $Mg = 20$. The initial velocities in Abaqus are prescribed in the global coordinate system; therefore, the two components of the angular velocities $\omega_y = -17.101$ and $\omega_z = 46.9846$ in the global system will create a resultant angular velocity $\dot{\phi}^3 = 50$ in the local system (*Figure 7*). The initial velocity in the global *x*-direction is the same as the initial velocity in the local $\mathbf{e_1}$-direction. The turning angles, obtained by solving the equation $f(u) = 0$, are $u_3 = 0.9517$ and $u_2 = 0.9112$ or $\theta_3 = 24.32°$ and $\theta_2 = 17.88°$, respectively. Based on the fact that the first Euler angle, θ, is equal to the spherical angle used in the polar representation, the variation of this angle in time is obtained in Abaqus from the displacements. The turning angles are reproduced accurately in Abaqus, and the analytical solution is in good agreement with the Abaqus solution. This comparison is shown in *Figure 8*.

The numerical damping coefficient ALPHA was taken equal to zero in the direct integration scheme used in Abaqus. It is worth mentioning that the analytical solution is an approximate solution since the accuracy of this solution will depend on the number of terms taken in the expansion series and on the accuracy with which the elliptic integrals are evaluated. The projection on the *x*–*y* plane of the top's locus is depicted in *Figure 9*, where the analytical solution and the Abaqus solution are shown. The locus exhibits loops along with precession in the counterclockwise direction. The Abaqus solution agrees with the analytical solution; however, the analytical solution is extremely sensitive to the values of the elliptic integrals taken from the tables.

The averaged precession frequency prediction can be found from the analytical solution for a fast top; that is, a top that has a large initial kinetic energy compared to the maximum change in the potential energy. The theoretical averaged precession frequency is

$$freq = \frac{Mgl}{I_{33}}\omega_3 = 0.4\ rad/s.$$

The total time for the complete precession in the *x*–*y* plane is 15 s, and the precession frequency given by Abaqus is, therefore, $freq = 2\pi/15 = 0.418\ rad/s$.

The change in the potential energy is reflected in the external work; due to the small applied force and small displacements, the exernal work has small values. Therefore, the total energy is approximately equal to the kinetic energy of the system. The total energy and the external work obtained in Abaqus are presented in *Figure 10* and *Figure 11*, respectively. For better visualization, the time variation of the external work is shown in *Figure 11* only for the first 3 s of the spinning process.

Case 2

A second case assumes that the top is spinning only about its own axis. For this case the ratio $u' = \frac{b}{a}$ coincides with one of the roots of the polynomial $f(u)$, and the locus of the top axis exhibits cusps touching circles (Goldstein, 1980). In this case we prescribe only the angular velocity around the $\mathbf{e_3}$-axis, $\dot{\phi}^1 = 50$. All of the other parameters are kept the same as before. The "turning" angles are obtained by solving again the equation $f(u) = 0$ with the new coefficients and are found to be $\theta_3 = 21.76°$ and $\theta_2 = 20°$, respectively. The variation in time of the first Euler angle, θ, is presented in *Figure 12* the first 3 s of the process. The projection of the top's locus on the *x*–*y* plane, obtained in Abaqus, is presented in the *Figure 13* where the analytical solution is also shown. The total energy and external work done for this case are presented in *Figure 14* and *Figure 15*.

Abaqus/Explicit is also used to study the forced motion of the rigid top presented in this section. Due to the explicit time integration, the running time is less in Abaqus/Explicit. The top is modeled using a rigid R3D4 element and a ROTARYI element. The rigid body reference node is identical to the node of the ROTARYI element.

The problem is also solved in Abaqus/Explicit and Abaqus/Standard using connector elements. The Euler angles are obtained directly (in radians) as output variable CPR. The solution obtained using connector elements agrees well with the analytical solution.

Input files

rigidbodymotion_forced_std.inp	Implicit forced motion analysis.
rigidbodymotion_verify.f	Code used to generate the analytical solution.
rigidbodymotion_forced_xpl.inp	Forced motion analysis with Abaqus/Explicit.
rigidbodymotion_conn_f_std.inp	Forced motion analysis in Abaqus/Standard, using connector elements.
rigidbodymotion_conn_f_xpl.inp	Forced motion analysis in Abaqus/Explicit, using connector elements.

References

- Goldstein, H., Classical Mechanics, Second Edition, Addison-Wesley, 1980.
- Fowles, G. R., Analytical Mechanics, Third Edition, Holt, Rinehart and Winston, 1977.
- MacMillan, W. D., Dynamics of Rigid Bodies, First Edition, McGraw-Hill Book, 1936.

Figures

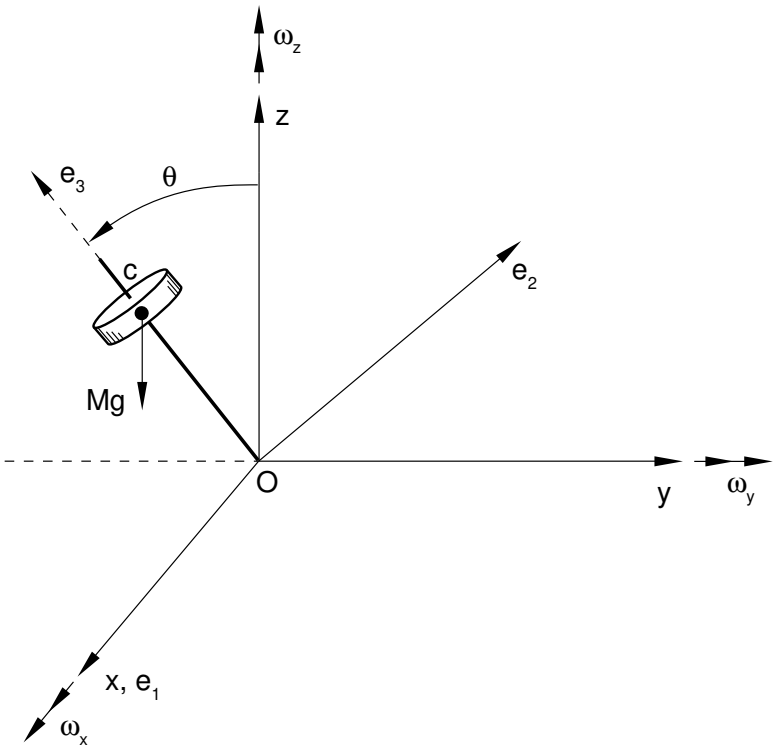


Figure 7: Symmetric top.

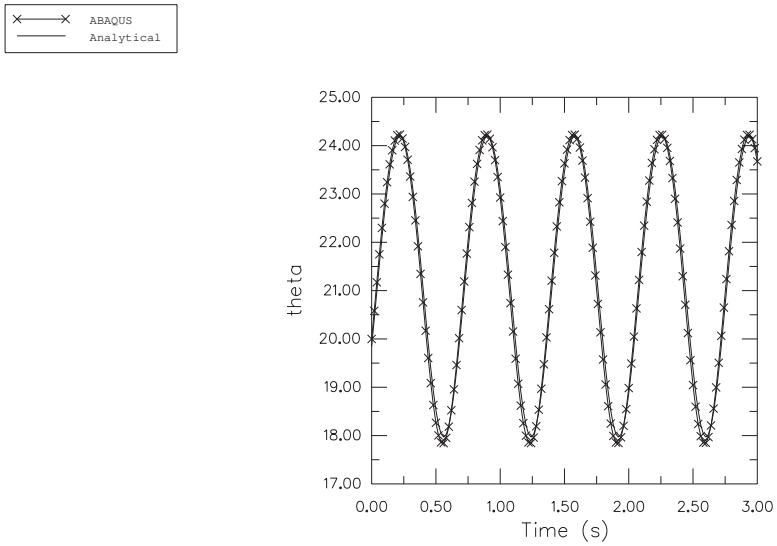


Figure 8: The variation of the first Euler angle, θ, for the first 3 s of the process—case 1.

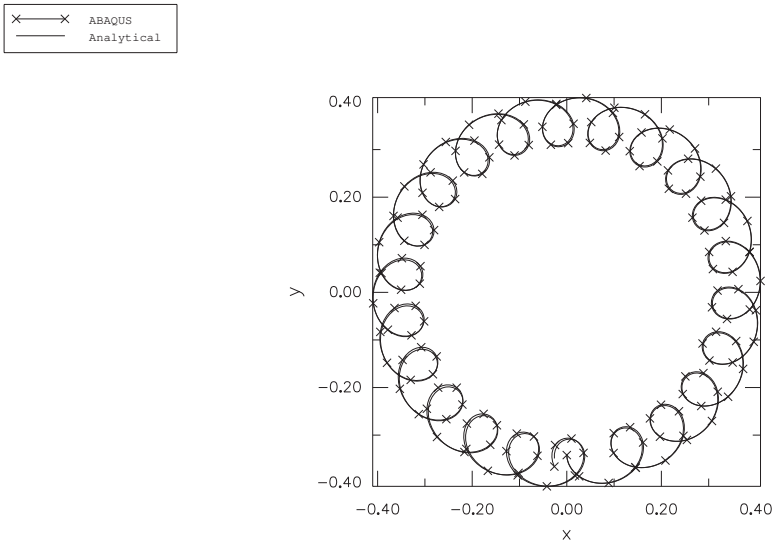


Figure 9: The locus of the top in the *x*–*y* plane—case 1.

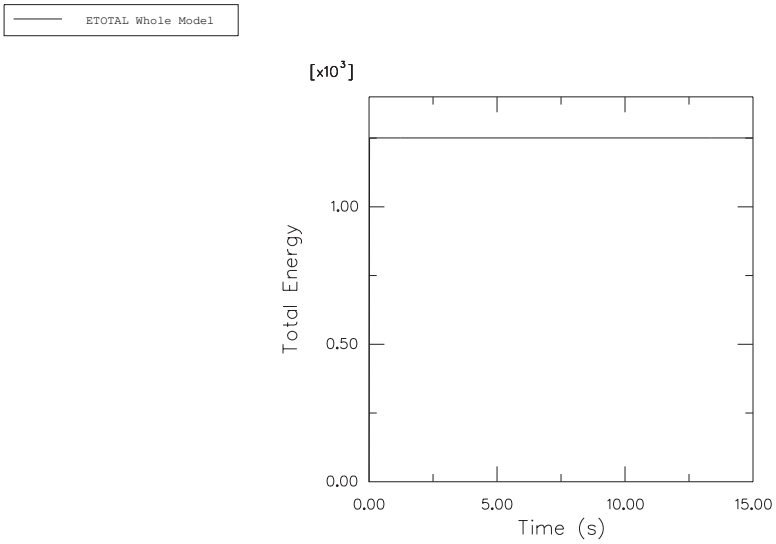


Figure 10: Total energy—case 1.

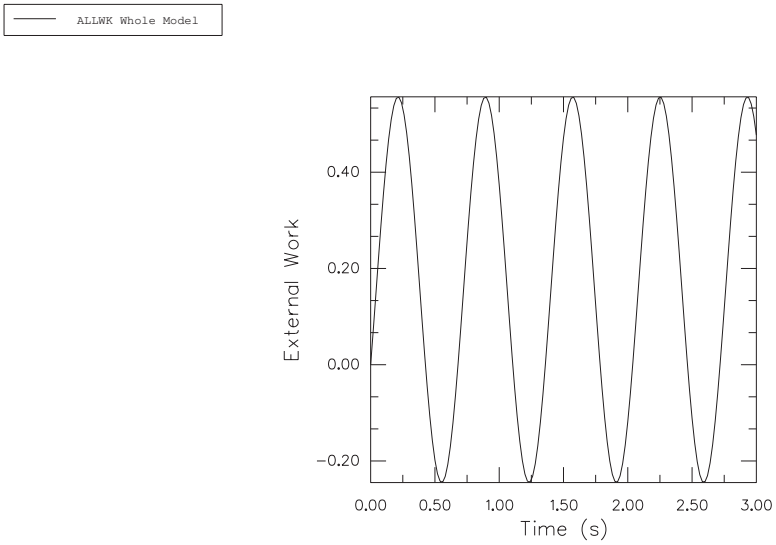


Figure 11: The external work done for the first 3 s of the process—case 1.

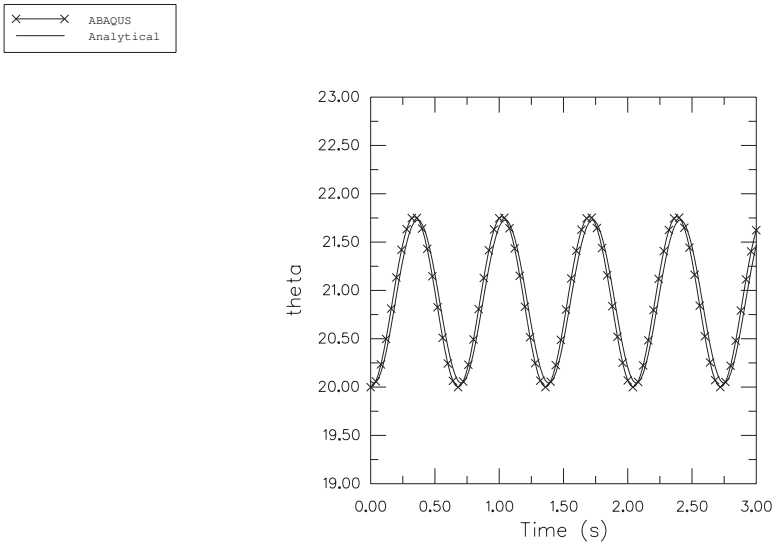


Figure 12: The variation of the first Euler angle, θ, for the first 3 s of the process—case 2.

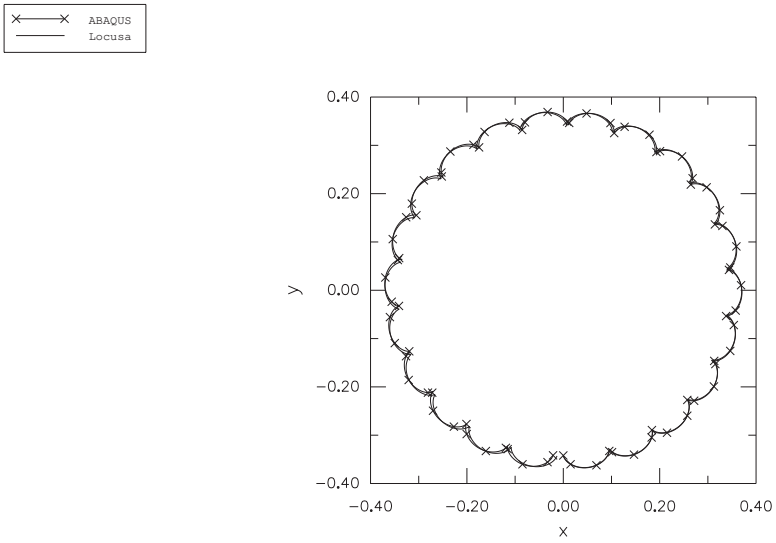


Figure 13: The locus of the top in the *x*–*y* plane—case 2.

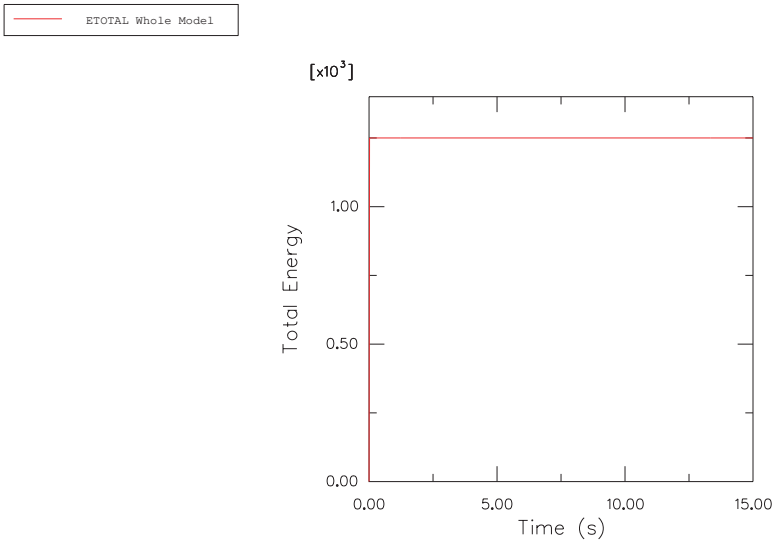


Figure 14: Total energy—case 2.

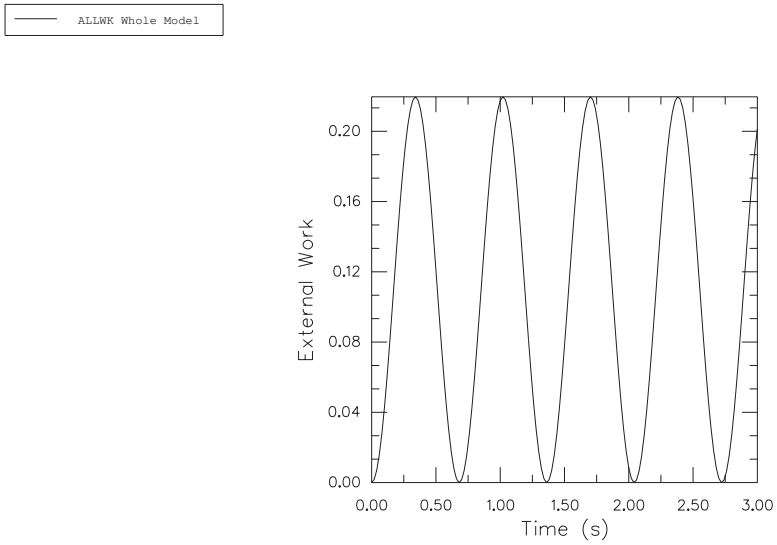


Figure 15: The external work done for the first 3 s of the process—case 2.

Rigid body dynamics with Abaqus/Explicit

Products: Abaqus/Explicit

Overview

This example verifies the rigid body dynamic behavior predicted with Abaqus/Explicit by comparison with analytical solutions.

Problem description

Figure 1 shows the geometry of the system considered. A single rigid body is under the action of two springs, with one attached to the rigid body and the other in contact with the rigid body. A point load is also applied to the rigid body. The rigid body is constrained at the reference node to undergo planar motion. Several two-dimensional and three-dimensional analyses based on this geometry are performed. For all cases a dummy continuum element is used to control the time incrementation.

In the first problem only rotation about the out-of-plane axis is allowed at the reference node and all the translational degrees of freedom are constrained. The inertial properties of the rigid body are represented with mass $M = 20$ and inertia about the axis normal to the plane of motion $I = 65$ at the reference node. The two springs each have a stiffness equal to 1.0×10^6. The mass, m, where the spring node comes in contact with the rigid body, is 5. The force applied, F, is 1.0×10^5. The initial angular velocity of the rigid body, w_0, is 10. The end of the spring that is in contact with the rigid body has an initial velocity such that contact is already established at time $t = 0$. The various quantities above are in a consistent set of units.

A variation of the first problem is considered (see *Figure 2*) in which the rigid body reference node location does not correspond to the center of mass of the rigid body. Point masses are specified on the rigid body surface nodes, $m_1 = 10$, $m_2 = 10$ (in three dimensions the surface node masses are each 5 since there are twice as many surface nodes); and the rotary inertia and the mass elements at the reference node are removed. The magnitude of the point masses is chosen such that the moment of inertia of the rigid body about the location of the pin constraint is the same as in the original problem; thus, the analytical solution for the rotational response is also the same.

Another variation of the original problem considered here, shown in *Figure 3*, is to allow translation parallel to the spring elements in addition to the rotation about the out-of-plane axis. The force applied is changed to $F = -1.0 \times 10^5$; the initial angular velocity, w_0, is 10; and the initial velocity, v_0, is 15. The initial velocity for the spring node in contact is chosen such that contact is already established at time $t = 0$.

A final variation of the problem is obtained by replacing the mass element and inertia element specified at the reference node with the surface masses forming the problem shown in *Figure 4*. The analytical solutions for the two active degrees of freedom are not the same for the last two problems since the reference node is allowed to translate.

Results and discussion

Figure 5 shows numerical solutions of the rotational response from the four analyses in which only rotation is allowed at the reference node and compares these solutions with a corresponding analytical solution based on the small-rotation assumption. For the problem shown in *Figure 3* the rotational and translational solutions are compared with the analytical solutions in *Figure 6* and *Figure 7*, respectively. Comparisons for the problem shown in *Figure 4* are presented in *Figure 8* and *Figure 9*. The results are in close agreement for all cases. The

deviations from the analytical solutions observed in *Figure 8* and *Figure 9* as the analysis progresses are the result of effects from the observed large rotations, which are not accounted for in the analytical solution.

Input files

rbd_2d_i_xybc.inp	Two-dimensional model with only a rotation active in the rigid body and a rotary inertia element at the reference node.
rbd_2d_i_xbc.inp	Two-dimensional model with one rotation and one translational degree of freedom active in the rigid body and a rotary inertia element at the reference node.
rbd_2d_sm_xybc.inp	Similar to rbd_2d_i_xybc.inp but with the rigid body modeled using point masses distributed on the surface nodes.
rbd_3d_i_xybc.inp	Three-dimensional analysis similar to rbd_2d_i_xybc.inp.
rbd_3d_sm_xybc.inp	Three-dimensional analysis similar to rbd_2d_i_xybc.inp but with the rigid body modeled using point masses distributed on the surface nodes.
rbd_2d_sm_xbc.inp	Similar to rbd_2d_i_xbc.inp but with the rigid body modeled using point masses distributed on the surface nodes.
rbd_3d_i_xbc.inp	Three-dimensional analysis similar to rbd_2d_i_xbc.inp.
rbd_3d_sm_xbc.inp	Three-dimensional analysis similar to rbd_2d_i_xbc.inp but with the rigid body modeled using point masses distributed on the surface nodes.

Figures

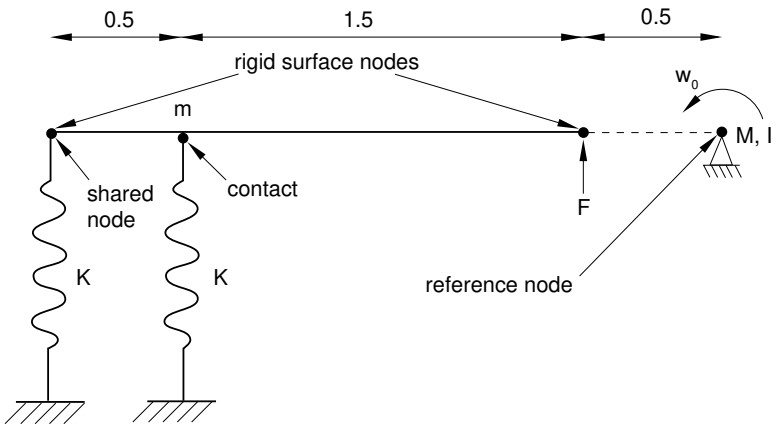


Figure 1: Rigid body with an inertia element and having only rotation about the out-of-plane axis active at the reference node.

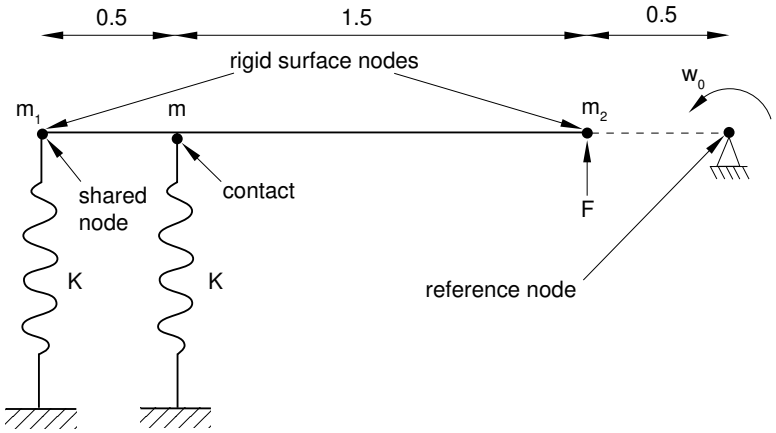


Figure 2: Rigid body with mass distributed at the surface nodes and having only rotation about the out-of-plane axis active at the reference node.

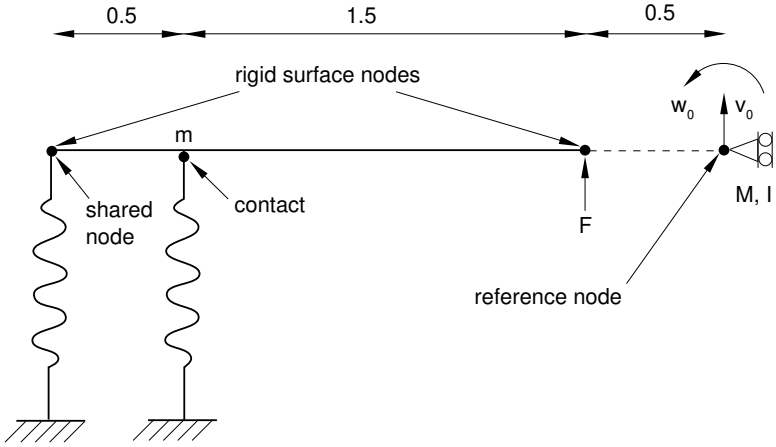


Figure 3: Rigid body with an inertia element and having one rotation and one translation active at the reference node.

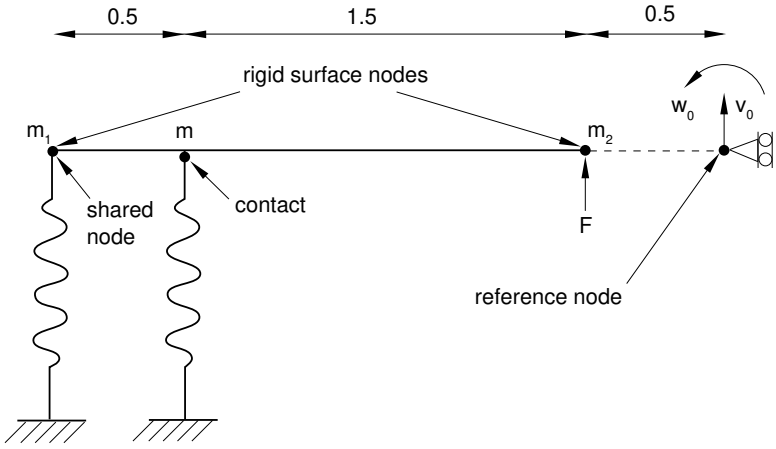


Figure 4: Rigid body with mass distributed at the surface nodes and having one rotation and one translation active at the reference node.

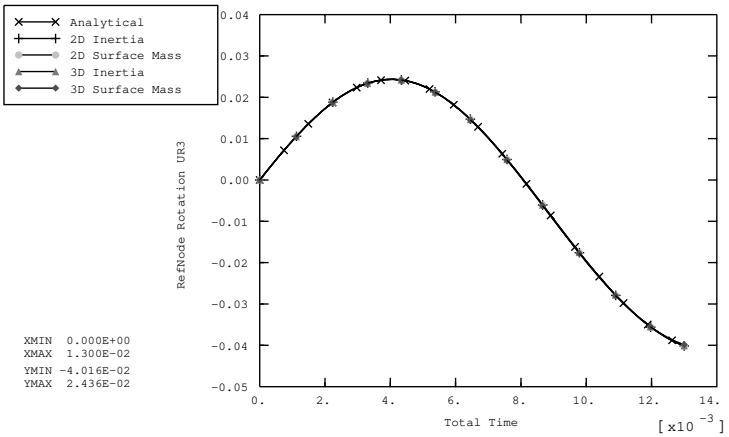


Figure 5: Predicted rigid body rotation compared with the analytical solution when only one rotational degree of freedom is active for the rigid body.

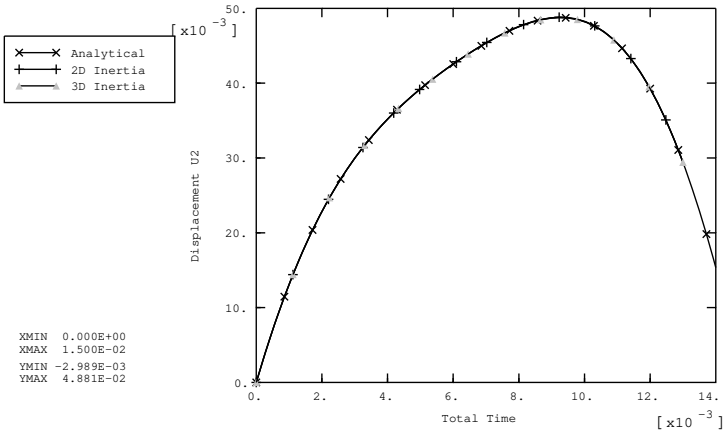


Figure 6: Predicted rigid body translation compared with the analytical solution when rotary inertia is specified and two degrees of freedom are active at the reference node.

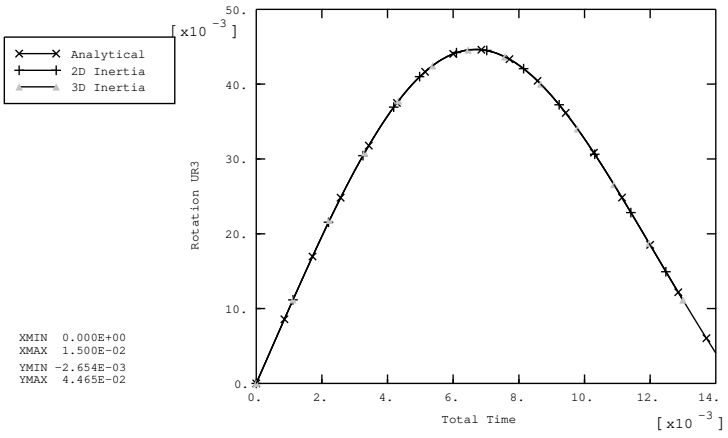


Figure 7: Predicted rigid body rotation compared with the analytical solution when rotary inertia is specified and two degrees of freedom are active at the reference node.

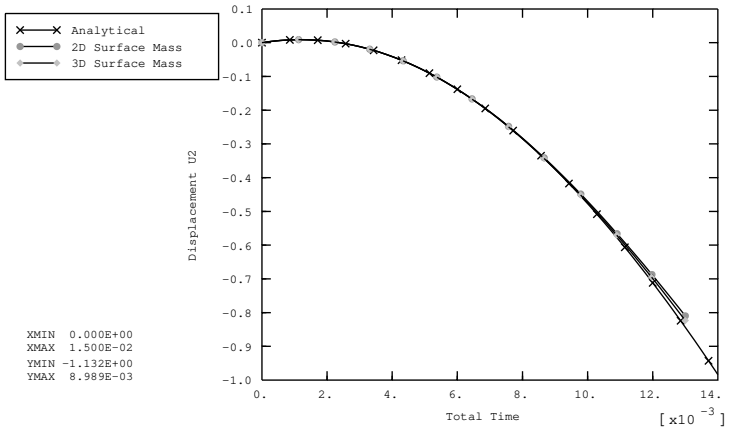


Figure 8: Predicted rigid body translation compared with the analytical solution when mass is distributed at the rigid body surface nodes and two degrees of freedom are active at the reference node.

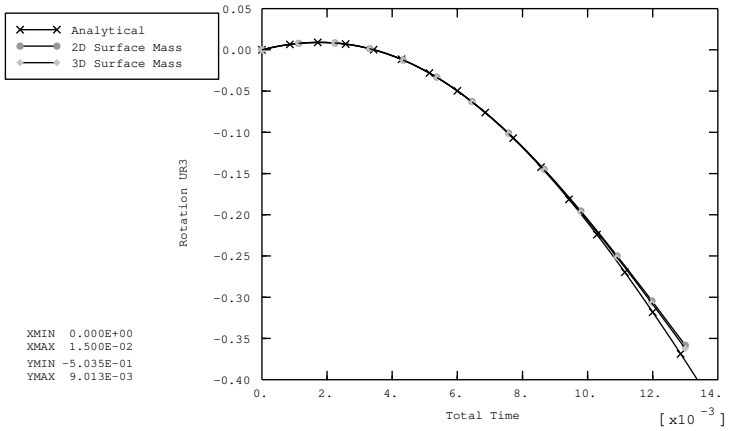


Figure 9: Predicted rigid body rotation compared with the analytical solution when mass is distributed at the rigid body surface nodes and two degrees of freedom are active at the reference node.

Revolute MPC verification: rotation of a crank

Products: Abaqus/Standard

Overview

This example illustrates and verifies the use of revolute joints (REVOLUTE MPC) in a simple elasto-kinematic system.

Problem description

Figure 1 shows the model after the revolutes have been rotated to create a crank. The crank is made of three segments, each 400 mm long. Initially they all lie along the global x-axis. The segments all have the same square cross-section, 20.3 mm × 20.3 mm, and are made of a material with a Young's modulus of 200 GPa and Poisson's ratio 0.0. The segments are connected by revolute joints whose axes are initially parallel to the global y-axis.

The segments of the crank are modeled with element type B31H. This “hybrid” beam element formulation is chosen because it provides rapid convergence of the nonlinear solution in cases of relatively stiff members undergoing large angular motions. The revolute joints are modeled with REVOLUTEMPCs (*General Multi-Point Constraints*). This requires separate nodes for the two sides of each joint and a third node for use in defining the revolute axis. Degree of freedom 6 at the third node represents the relative rotation in the joint. The line between the second and third nodes defines the initial direction of the axis of the joint.

Loading

The crank is built-in at one end. The revolutes are initially locked, and a load of −498.2 N is applied to the free end of the crank in the z-direction. The revolute joints are then subjected to opposing internal rotations of magnitude $\pi/2$, thus creating the crank-like geometry shown in *Figure 1*. Finally, the entire crank is rotated through an angle of $\pi/2$ about the global z-axis.

Results and discussion

The analysis includes an initial linear perturbation step in which the straight crank is loaded without considering geometric nonlinearity. This step is introduced mainly for verification of the model. Simple beam theory shows that the tip deflection at the end of the first step should be 107.95 mm. The analysis gives a value of 106.8 mm. The same loading with geometric nonlinearity included gives a tip displacement of 105.9 mm—slightly less because the bending of the beam in this case stiffens its response.

In the next step relative rotations are applied in the joints to make the bar into a crank, while the load remains on the tip. The 90° rotation is applied in four increments by prescribing the value of degree of freedom 6 at the relative rotation nodes (nodes 12 and 14). For comparison, we analyze a crank made with B31H elements with rigid joints. The tip displacement of 40.64 mm obtained in this analysis agrees exactly with that provided by the analysis with the revolutes, when the 400 mm displacement caused by the revolute rotation is taken into consideration.

The last step rotates the entire crank by 90° about the global z-axis. This is accomplished in four increments. The final configuration is shown in *Figure 2*.

Input files

revolutempc_joints.inp	Crank with revolute joints.
revolutempc.inp	Crank without revolute joints.

Figures

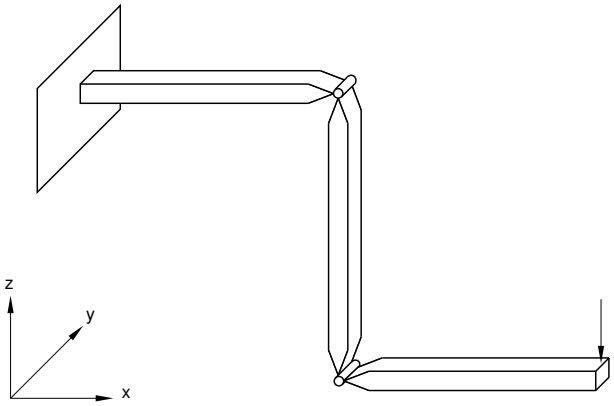


Figure 1: Crank: initial configuration.

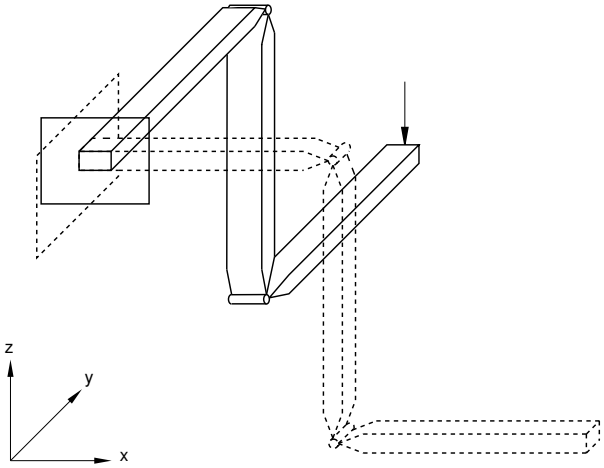


Figure 2: Crank: final (loaded) configuration.

Pipe whip simulation

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example simulates a pipe-on-pipe impact resulting from the rupture of a high-pressure line in a power plant. It is assumed that a sudden release of fluid could cause one segment of the pipe to rotate about its support and strike a neighboring pipe.

Problem description

The pipes have an outer diameter of 168.275 mm (6.625 in), with a 10.97 mm (0.432 in) wall thickness and a span of 1270 mm (50 in) between supports. The impacted pipe is assumed to be fully restrained at both ends, while the impacting pipe is allowed to rotate about a fixed pivot with an initial angular velocity of 75 radian/sec. We make use of symmetry boundary conditions to reduce the problem size by discretizing only the geometry to one side of the central symmetry plane.

Both pipes are made of steel with a Young's modulus of 207 GPa (30×10^6 psi), a Poisson's ratio of 0.3, and a density of 7827 kg/m^3 (7.324×10^{-4} lb sec^{2}in^{-4}). A von Mises elastic, perfectly plastic material model is used, with a yield stress of 310 MPa (45×10^3 psi).

S4R shell elements are used to discretize the pipes. A higher level of mesh refinement is used near the middle of the pipes, where the impact will take place. The mesh is shown in *Figure 1*. The contact surfaces are defined over the entire length of each pipe and then grouped into a single contact pair. Kinematic contact enforcement is used for the primary input file, although models that use penalty contact pairs and general contact are also provided. An additional analysis with enhanced hourglass control is performed.

Results and discussion

The deformed shapes at different stages of the analysis, shown in *Figure 2* through *Figure 4*, are in good agreement with the results reported by Ferencz (1989). The results of the analysis with enhanced hourglass control closely match the ones obtained with the default hourglass control.

A time history of the total kinetic energy, internal energy, and plastic dissipation over the duration of the analysis is shown in *Figure 5*. Near the end of the simulation the impacting pipe is beginning to rebound, having dissipated the majority of its kinetic energy by inelastic deformation in the crushed zone.

The results provided by the analysis based on penalty contact are approximately the same. The analysis costs using the alternative contact methods are increased by 2.5% as a result of a slightly smaller time increment with the penalty method.

Input files

Abaqus/Standard input file

pipewhip_std.inp	Contact pair analysis using S4R elements.

Abaqus/Explicit input files

pipewhip.inp	Contact pair analysis using S4R elements.
pipewhip_gcont.inp	General contact analysis using S4R elements.

pipewhip_enh.inp	Contact pair analysis using S4R elements with enhanced hourglass control.
pipewhip_enh_gcont.inp	General contact analysis using S4R elements with enhanced hourglass control.
pipewhip_s4rs.inp	Contact pair analysis using small-strain shell elements S4RS.
pipewhip_s4rs_gcont.inp	General contact analysis using small-strain shell elements S4RS.
pipewhip_s4rs_gcont_subcyc.inp	General contact analysis using small-strain shell elements S4RS with subcycling.
pipewhip_s4rsw.inp	Contact pair analysis using small-strain shell elements S4RSW.
pipewhip_s4rsw_gcont.inp	General contact analysis using small-strain shell elements S4RSW.
pipewhip_pnlty.inp	Contact pair analysis using penalty contact.

Four additional models are included with the Abaqus release for the sole purpose of testing the performance of the code (file names: *pipewhip_medium.inp*, *pipewhip_medium_gcont.inp*, *pipewhip_fine.inp*, and *pipewhip_fine_gcont.inp*).

References

- Ferencz, R. M., “Element-by-Element Preconditioning Techniques for Large-Scale, Vectorized Finite Element *ford University, Stanford,*

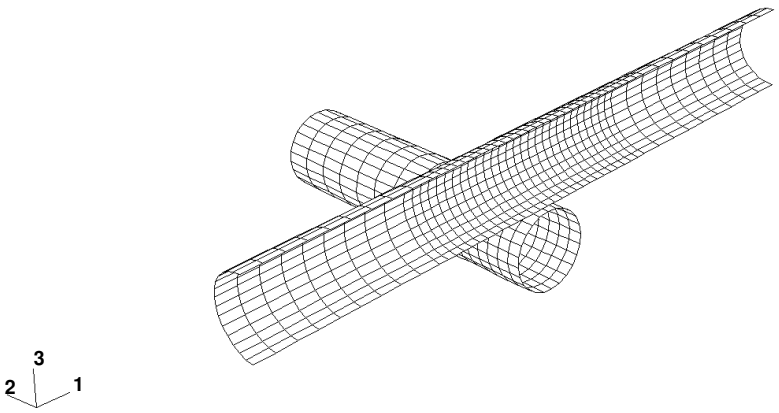


Figure 1: Undeformed mesh.

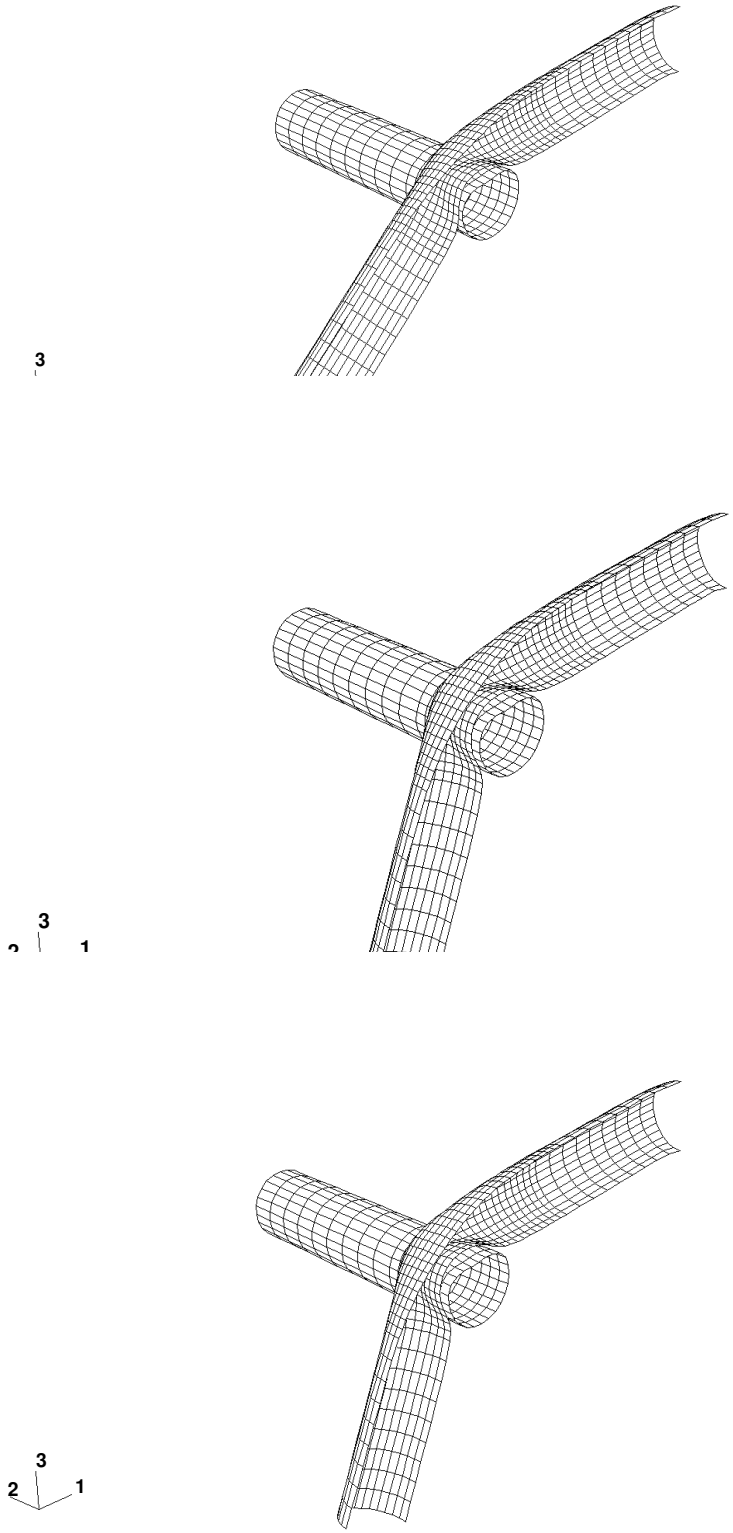


Figure 4: Deformed shape at 15 milliseconds.

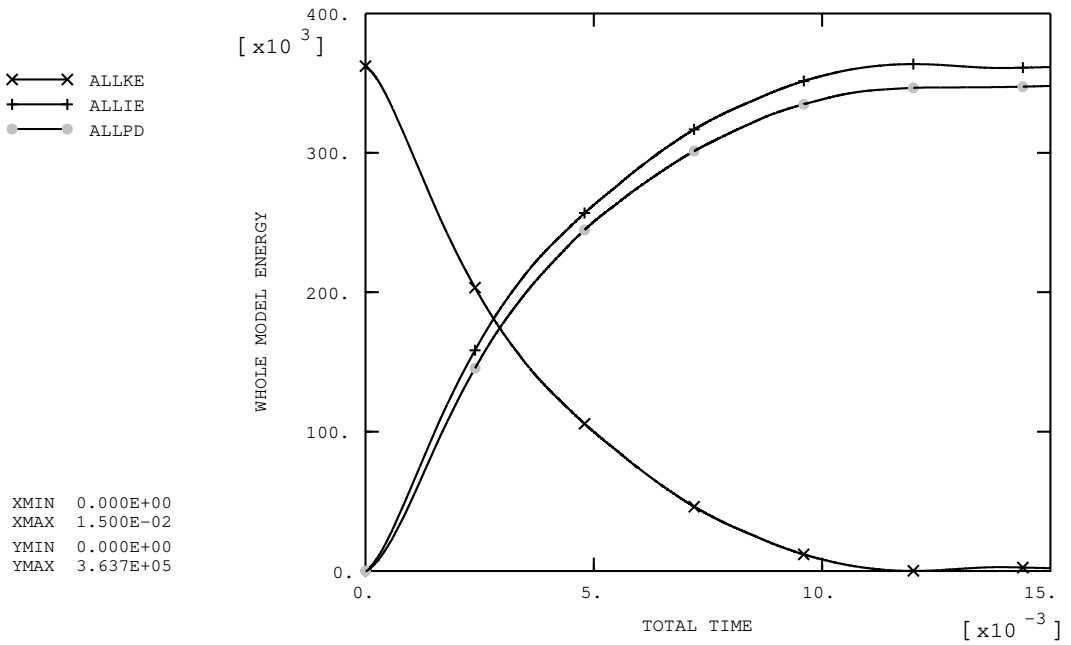


Figure 5: Time histories of the total kinetic energy, internal energy, and plastic dissipation.

Impact of a copper rod

Products: Abaqus/Explicit

Overview

This example simulates the high velocity impact of a copper rod onto a rigid wall.
Such tests are performed to determine the material constants for high-pressure equations of state. The test is sometimes described as the Taylor bar experiment. Extremely high plastic strains develop at the crushed end of the rod, resulting in severe local mesh distortion.

Problem description

The problem consists of a 32.4 mm long cylindrical rod with a radius of 3.2 mm, impacting a rigid wall with an initial velocity of 227 m/sec. The rod is made of copper, with Young's modulus of 110 GPa and Poisson's ratio of 0.3. The density is 8970 kg/m^3. A von Mises elastic, perfectly plastic material model is used with a yield stress of 314 MPa.

The rod is modeled first using a 10 × 36 mesh of axisymmetric quadrilateral elements (type CAX4R), as shown in *Figure 1*. Zero radial displacements are imposed along the symmetry axis. To simulate the impact of the rod on a (frictionless) rigid wall, zero axial displacements are prescribed at one end of the rod, while all other nodes are subjected to a 227 m/sec initial velocity. Although this technique is appropriate for modeling the crushing of the front end of the rod in the absence of friction or rebound, contact should be used if there are significant friction effects or if separation between the rod and the rigid wall is expected. Different hourglass control options are analyzed by modifying the section controls for the CAX4R element.

A three-dimensional analysis is also performed for the same problem. One quadrant of the rod is discretized, using 2700 solid elements of type C3D8R, with the appropriate boundary conditions prescribed on each of the two symmetry planes for the problem (see *Figure 4*). Again, zero longitudinal displacements are prescribed at one end of the rod, while all other nodes are subjected to a 227 m/sec initial velocity. Different hourglass control options and kinematic formulations are analyzed by modifying the section controls for the C3D8R element. Element section controls are used to modify the element formulation to reduce the analysis time. These options result in fewer element-level calculations and do not change the stable time increment size.

In addition, two- and three-dimensional analyses of the rod impact are performed using modified triangular (CAX6M) and tetrahedral (C3D10M) elements. The models for the modified element meshes are shown in *Figure 7* and *Figure 10*; these meshes incorporate the same number of nodes per side as the analogous quadrilateral and brick meshes.

The high velocity impact causes severe mesh distortion in elements near the front end of the rod, thereby dramatically reducing the stable time increment during the solution. Therefore, both the axisymmetric and the three-dimensional analyses are also performed with variable mass scaling to scale the masses of the elements that become very small. The scaling is defined such that the stable time increments do not fall below a prescribed minimum.

Eulerian elements have advantages over Lagrangian elements when handling severe element distortions. Therefore, a three-dimensional Eulerian analysis is also performed for the rod impact problem. The size of the Eulerian domain is 35.0 mm × 10.0 mm. The initial volume fraction of the copper material is specified such that the material occupies the same region as the rod in the three-dimensional Lagrangian analyses. The rod part of the Eulerian mesh is shown in *Figure 14*.

The area of interest of the rod impact problem is near the front end of the rod where large plastic deformation occurs. Less mesh resolution is needed in the rest of the Eulerian domain. For better computational efficiency, an Eulerian analysis is performed with adaptive mesh refinement. The analysis starts with a coarsely discretized

rod as shown in *Figure 13*. During the analysis, elements with equivalent plastic strain greater than 0.1 are refined and divided into subelements with the same size as those in *Figure 14*. For comparison, an Eulerian analysis with the coarse mesh shown in *Figure 13* is also performed.

Results and discussion

Table 1 shows the section control and mass scaling options used for the analysis.

For the axisymmetric model using CAX4R elements, the deformed shapes of the rod after 20 and 80 microseconds are shown in *Figure 2* and *Figure 3* for the combined hourglass control. The results for the three-dimensional model using C3D8R elements are shown in *Figure 5* and *Figure 6* for the orthogonal kinematic and combined hourglass section controls. The deformed shapes are also shown in *Figure 8* and *Figure 9* for the axisymmetric model using CAX6M elements and in *Figure 11* and *Figure 12* for the three-dimensional model using C3D10M elements. The results reproduce the behavior observed by Ferencz (1989).

From these figures, it is clear that extremely high plastic strains develop at the crushed end of the rod, close to the axis of symmetry, resulting in severe local mesh distortion. The shortening and widening of the bar are reported in *Table 2* for the different analysis cases. The values of the bar's spread are reported for the symmetric model, and the three-dimensional values are reported as the y-component displacement at node 91 for the model using C3D8R elements and at node 61 for the model using C3D10M elements.

The displacements and energies obtained from the analyses using different element types and section controls agree very well, except in the case of the model that uses C3D10M elements. These elements are slightly stiffer with the given mesh refinement, as demonstrated by the predicted shortening value of −12.71 in *Table 2*. This shortening value converges as the mesh is refined to the values obtained from the analyses that use other element types. Differences are less pronounced for the variations of the C3D8R element. Using the orthogonal kinematic and enhanced hourglass control formulations produces a solution similar to that for the analysis that uses the default section control parameters.

Without any mass scaling, the stable time increment for the problem is observed to reduce dramatically over the course of this analysis as a result of the large changes in element aspect ratio. Local mass scaling increases the stable time increment and, thus, reduces the total time of the simulation. A comparison of the stable time increment time histories for the unscaled and scaled cases is shown in *Figure 20*. The minimum allowable stable time increment chosen resulted in a 5.9% increase in the overall mass of the rod by the end of the simulation. Although this percentage is substantial, all the scaling is performed on the severely compressed elements near the rigid wall. Thus, the overall dynamics of the solution are unchanged, while the solution time is approximately one-third that of the unscaled case. The predicted maximum effective plastic strain for the scaled case is 5.876, which is 1.2% higher than the maximum obtained in the unscaled analysis (using the default section control options). Comparisons of kinetic energy and free end displacement time histories of the rod show excellent agreement and are presented in *Figure 19* and *Figure 20*, respectively.

The results for the Eulerian analyses with a coarse mesh and a fine mesh are shown in *Figure 15* and *Figure 16*. The results of the Eulerian analysis with adaptive mesh refinement are also shown in *Figure 17*. By comparing the final deformation of the rod, we find the results with adaptive mesh refinement are much more accurate than those without refinement. These results also agree very well with those obtained with a fully refined mesh. We can draw the same conclusion by comparing the energy results from these analyses, as shown in *Figure 21*.

Input files

rodimpac2d_cs.inp	Axisymmetric case using COMBINED hourglass control.
rodimpac2d_es.inp	Axisymmetric case using ENHANCED hourglass control.
rodimpac3d_ocs.inp	Three-dimensional case using the ORTHOGONAL kinematic and the COMBINED hourglass section control options.
rodimpac3d_oes.inp	Three-dimensional case using the ORTHOGONAL kinematic and the ENHANCED hourglass section control options.

rodimpac2d.inp	Axisymmetric case using the default section controls.
rodimpac3d.inp	Three-dimensional case using the default section controls.
rodimpac3d_aes.inp	Three-dimensional case using the default kinematic and the ENHANCED hourglass section control options.
rodimpac2dms.inp	Axisymmetric case using the default section controls with mass scaling.
rodimpac3dms.inp	Three-dimensional case using the default section controls with mass scaling.
rodimpac3d_cvs.inp	Analysis using the CENTROID kinematic and the VISCOUS hourglass section control options.
rodimpac3d_c3d4h.inp	Analysis using the elements C3D4H.
rodimpac2d_cax6m.inp	Analysis using the modified elements CAX6M.
rodimpac3d_c3d10m.inp	Analysis using the modified elements C3D10M.
rodimpac2d_j_c.inp	Test of the Johnson-Cook plasticity model for the axisymmetric case. The material properties used in this and the following three input files are taken from Johnson and Cook (1985).
rodimpac3d_j_c.inp	Test of the Johnson-Cook plasticity model for the three-dimensional case.
rodimpac2d_jcs.inp	Test of the Johnson-Cook shear failure model for the axisymmetric case.
rodimpac3d_jcs.inp	Test of the Johnson-Cook shear failure model for the three-dimensional case.
eulerian_rodimpact.inp	Three-dimensional case using a uniform Eulerian mesh.
eulerian_rodimpact_fine.inp	Three-dimensional case using a finer uniform Eulerian mesh.
eulerian_rodimpact_adapt.inp	Three-dimensional case using a uniform Eulerian mesh with adaptive mesh refinement.

Two additional models are included with the Abaqus release for the purpose of testing the performance of the code (file names: *rodimpac2d_fine.inp* and *rodimpac3d_fine.inp*).

References

- Ferencz, R. M., "Element-by-Element Preconditioning Techniques for Large-Scale, Vectorized Finite Element Analysis in Nonlinear Solid and Structural Mechanics," *Ph. D. Dissertation, Stanford University, Stanford, CA*, 1989.
- Johnson, G. R., and W. H. Cook, "Fracture Characteristics of Three Metals Subjected to Various Strains, Strain rates, Temperatures and Pressures," *Engineering Fracture Mechanics*, vol. 21, no. 1, pp. 31–48, 1985.

Tables

Table 1: Analysis options.

Analysis Case	Variable Mass Scaling	Section Controls	
		Kinematic	Hourglass
CAX4R	no	n/a	integral viscoelastic
CAX4R CS	no	n/a	combined
CAX4R ES	no	n/a	enhanced
CAX4R MS	yes	n/a	integral viscoelastic
C3D8R	no	average strain	integral viscoelastic

Analysis Case	Variable Mass Scaling	Section Controls	
		Kinematic	Hourglass
C3D8R MS	yes	average strain	integral viscoelastic
C3D8R OCS	no	orthogonal	combined
C3D8R OES	no	orthogonal	enhanced
C3D8R AES	no	average strain	enhanced
C3D8R CVS	no	centroid	viscous
CAX6M	no	n/a	n/a
C3D10M	no	n/a	n/a
C3D4H	no	n/a	n/a

Table 2: Shortening and spread of the rod.

Analysis Case	Shortening (mm)	Widening (mm)	Relative CPU Time	Relative Cost per Increment per Element
CAX4R	−13.10	6.098	1.0	1.0
CAX4R CS	−13.10	6.114	1.0	0.94
CAX4R ES	−13.16	5.556	0.75	1.01
CAX4R MS	−13.10	6.115	0.25	0.96
C3D8R	−13.10	5.737	3.0	0.47
C3D8R MS	−13.10	5.741	1.25	0.46
C3D8R OCS	−13.08	5.686	2.5	0.45
C3D8R CVS	−13.18	5.665	2.25	0.43
C3D8R OES	−13.19	5.683	3.0	0.49
C3D8R AES	−13.18	5.676	3.25	0.54
CAX6M	−13.19	5.173	0.5	2.14
C3D10M	−13.14	6.293	11.25	1.27
C3D4H	−13.07	5.951	7.0	0.43

Figures

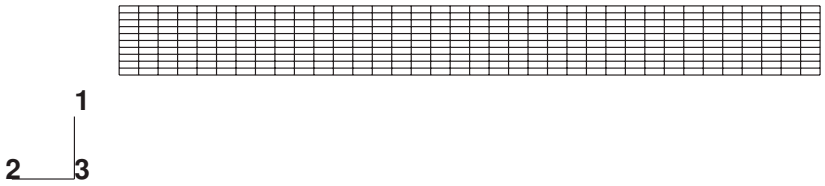


Figure 1: Original mesh (CAX4R model).

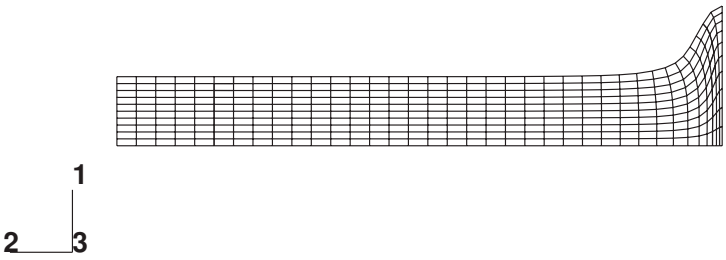


Figure 2: Deformed shape at 20 microseconds (CAX4R model using the combined hourglass control).

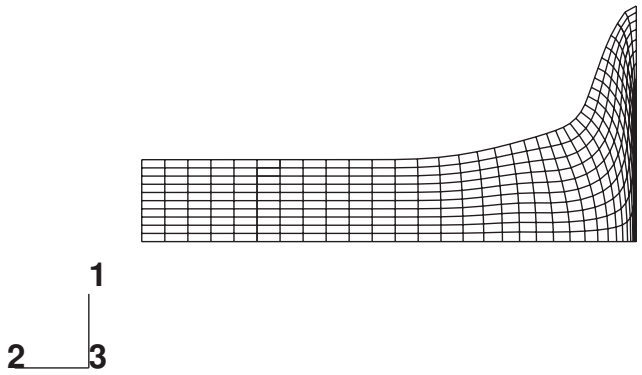


Figure 3: Deformed shape at 80 microseconds (CAX4R model using the combined hourglass control).

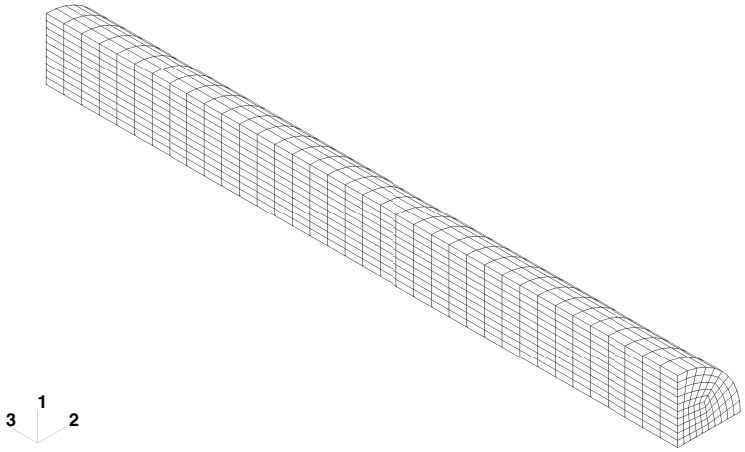


Figure 4: Original mesh (C3D8R model).

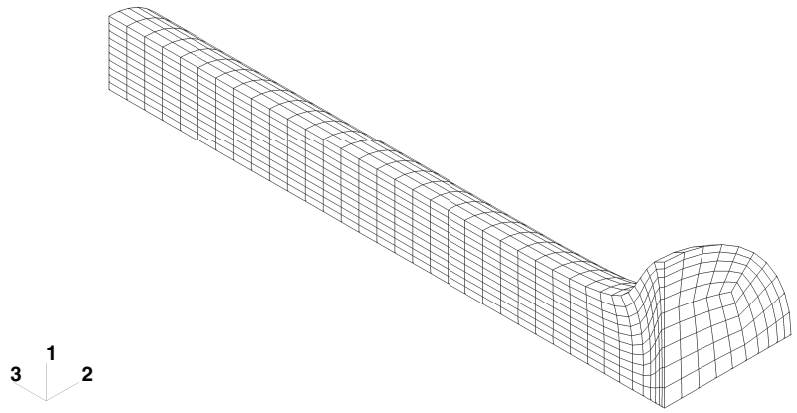


Figure 5: Deformed shape at 20 microseconds (C3D8R model using the orthogonal kinematic and combined hourglass section control options).

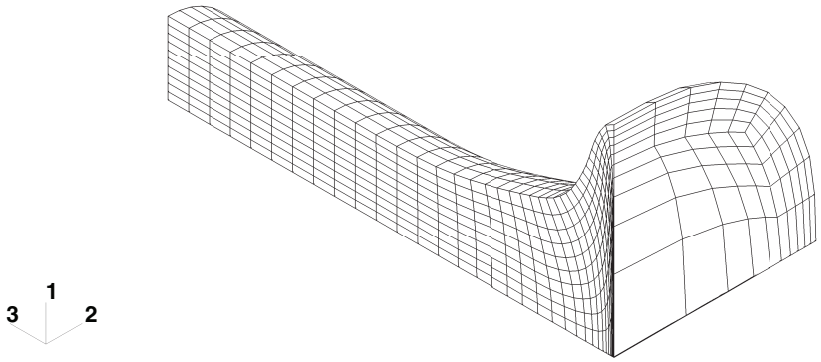


Figure 6: Deformed shape at 80 microseconds (C3D8R model using the orthogonal kinematic and combined hourglass section control options).

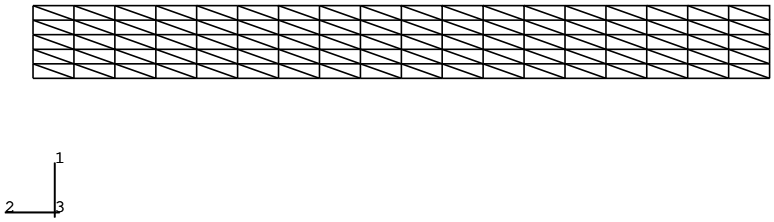


Figure 7: Original mesh (CAX6M model).

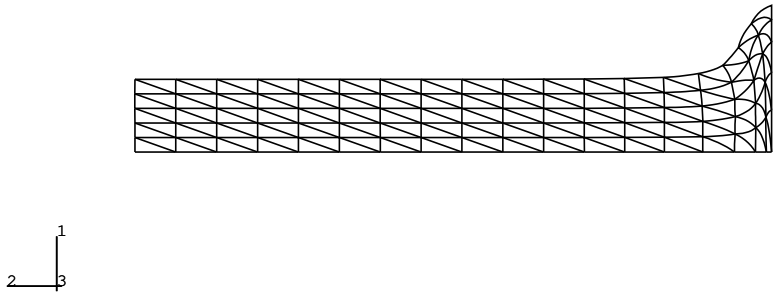


Figure 8: Deformed shape at 20 microseconds (CAX6M model).

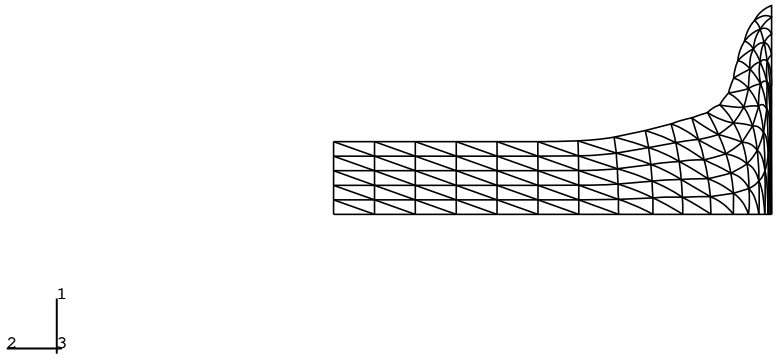


Figure 9: Deformed shape at 80 microseconds (CAX6M model).

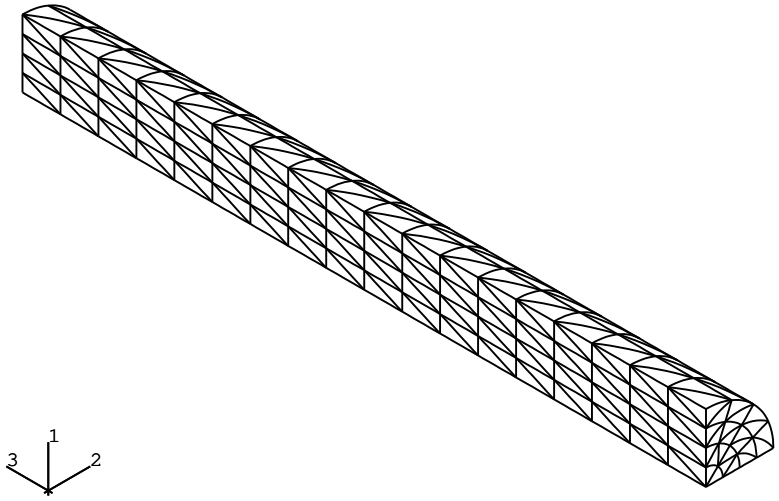


Figure 10: Original mesh (C3D10M model).

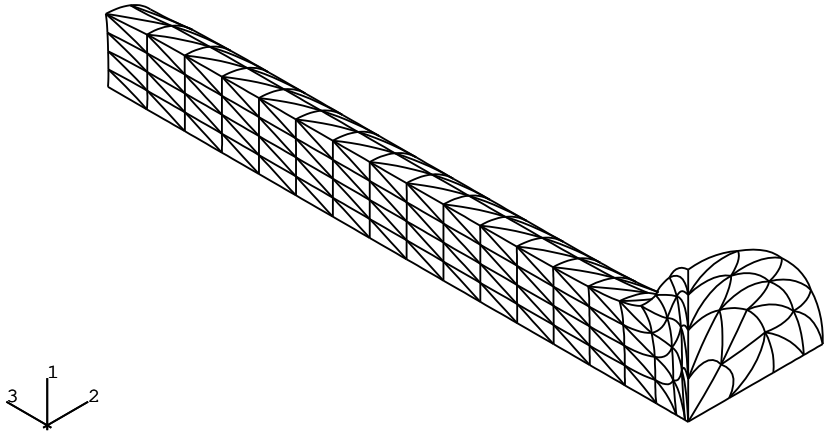


Figure 11: Deformed shape at 20 microseconds (C3D10M model).

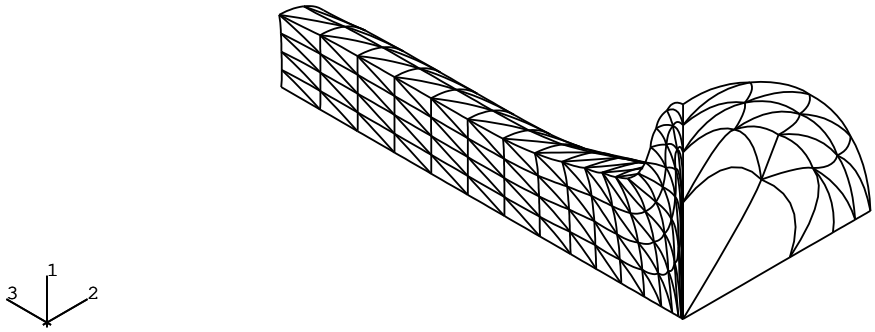


Figure 12: Deformed shape at 80 microseconds (C3D10M model).

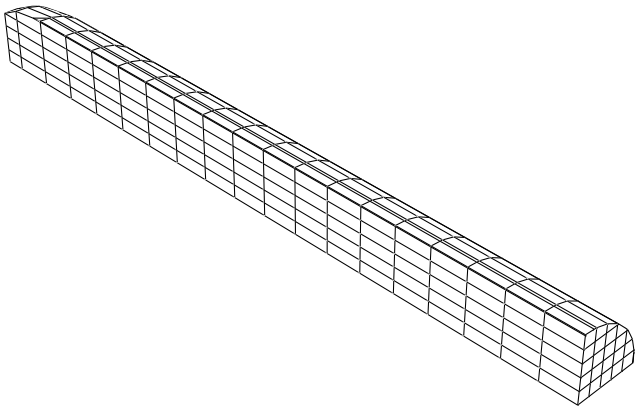


Figure 13: Coarse Eulerian mesh (only the rod part is shown).

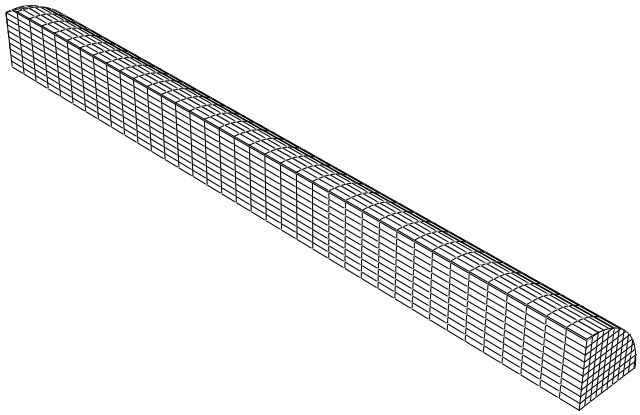


Figure 14: Fine Eulerian mesh (only the rod part is shown).

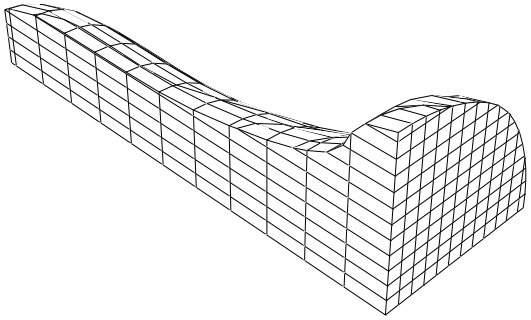


Figure 15: Deformation shape at 80 microseconds (coarse Eulerian mesh model).

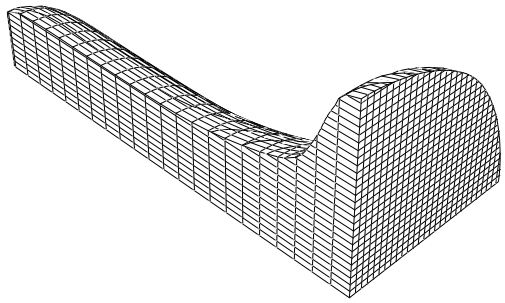


Figure 16: Deformation shape at 80 microseconds (fine Eulerian mesh model).

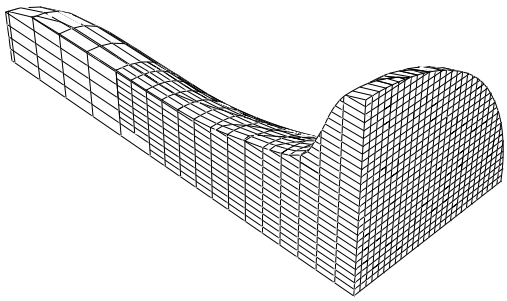


Figure 17: Deformation shape at 80 microseconds (adaptive Eulerian mesh model).

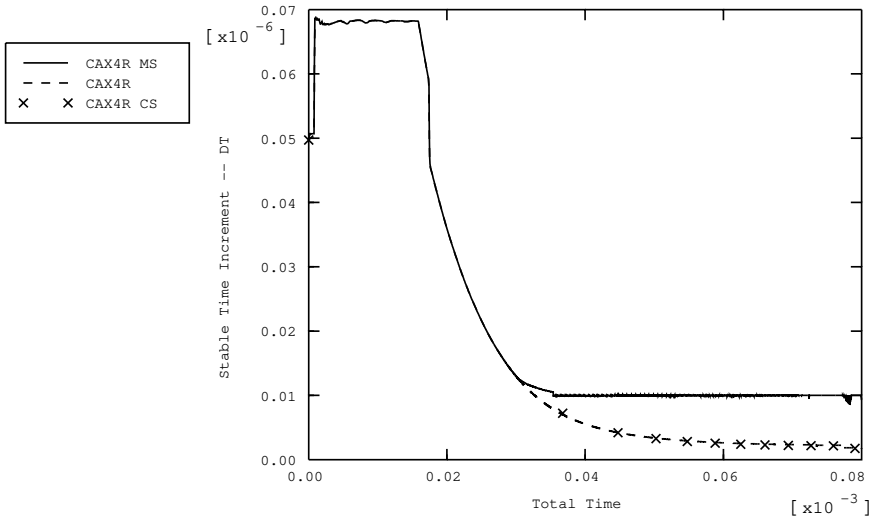


Figure 18: Time history of the stable time step size (see the table listing analysis options).

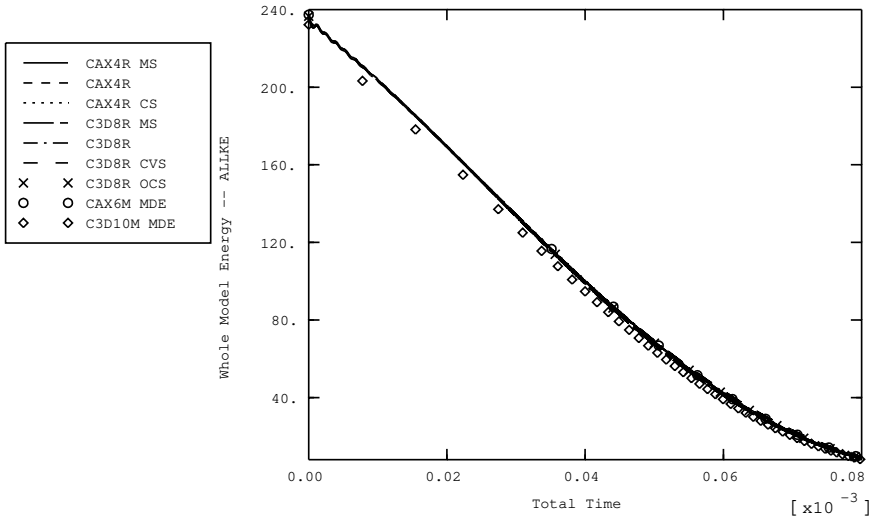


Figure 19: Time history of the total kinetic energy (see the table listing analysis options).

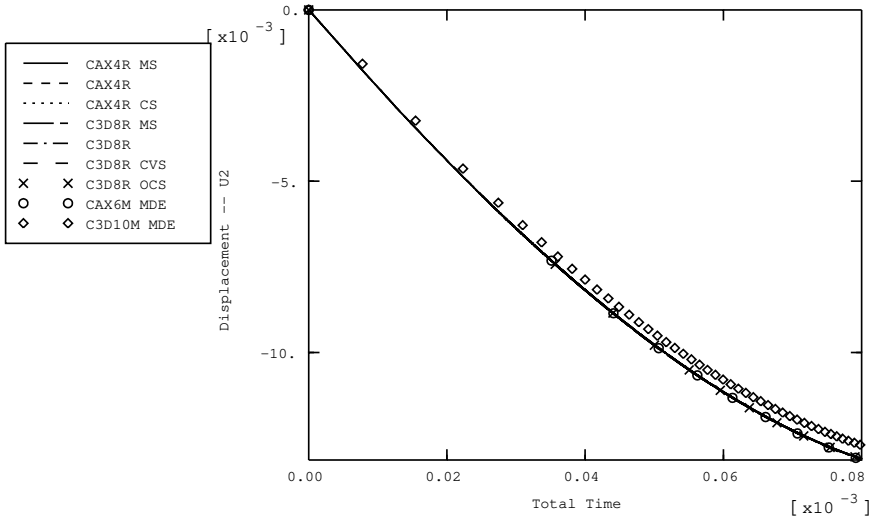


Figure 20: Time history of the free end displacement (see the table listing analysis options).

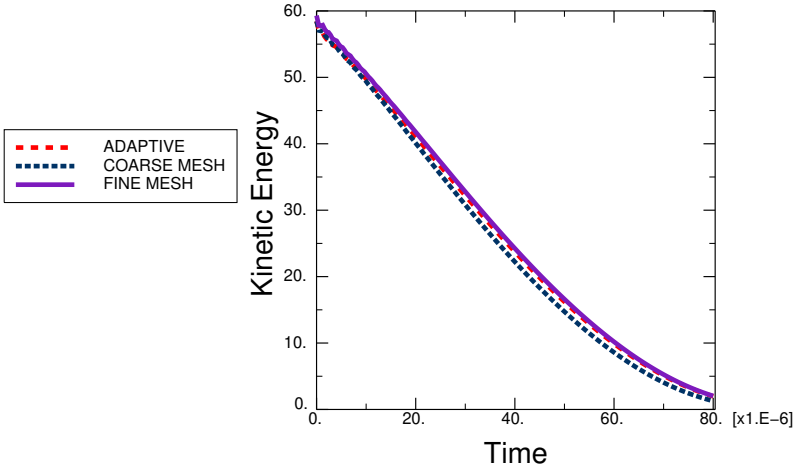


Figure 21: Time history of the total kinetic energy.

Frictional braking of a rotating rigid body

Products: Abaqus/Explicit

Overview

This example verifies the modeling of frictional braking of a rotating rigid drum.

Problem description

The problem consists of a rigid drum, initially rotating at $\omega_0 = 60$ rad/s about a fixed axis, that is brought to rest by frictional contact with a pad of hyperelastic material. The rigid drum has a radius R of 200 mm and width of 150 mm, total mass of 5 kg, and rotary inertia of 0.175 kg m^2 about its free axis of rotation. The deformable pad is a $100 \times 100 \times 50$ mm block of hyperelastic material, having a polynomial strain energy function of order $N = 1$ with constants $C_{10} = 0.552$ MPa, $C_{01} = 0.138$ MPa and $D_1 = 0.145 \times 10^{-6}$ MPa. A constant pressure $p = 0.350$ MPa is applied to the back of the pad to force it against the rigid drum. A Coulomb friction coefficient of 15% is assumed to exist between the pad and the drum. Both two-dimensional and three-dimensional idealizations of the problem are used for verification.

For two dimensions the rigid drum is modeled in two different ways:

1. The rigid drum is modeled as an analytical rigid surface using a planar analytical surface in conjunction with a rigid body constraint.
2. The rigid drum is discretized using 72 rigid elements of type R2D2.

The analytical rigid surface can yield a more accurate representation of two-dimensional curved punch geometries and result in computational savings. Contact pressure can always be viewed on the specimen surface, and the punch reaction force is available at the rigid body reference node. Results for the element facet representations are presented here.

For three dimensions the rigid drum is modeled in five different ways, as described below:

1. The rigid drum is modeled as an analytical rigid surface using a cylindrical or revolution analytical surface in conjunction with a rigid body constraint.
2. The rigid drum is discretized using 72 rigid elements of type R3D4.
3. Membrane elements of type M3D4R are used to model the drum, and they are included in the rigid body by referring to them in the rigid body constraint. A zero material density is specified for the membrane elements; and to make this model comparable to Case 1, a zero-thickness surface is used when defining the outer surface of the drum.
4. Shell elements of type S4R are used to model the drum, and they are included in the rigid body by referring to them in the rigid body constraint. A zero material density is specified for the shell elements; and to make this model comparable to Case 1, a zero-thickness surface is used when defining the outer surface of the drum.
5. Solid elements of type C3D4 are used to model the drum, and they are included in the rigid body by referring to them in the rigid body constraint. A zero material density is specified for the C3D4 elements.

The reference node of the rigid drum is located on the axis of rotation. Since we have chosen to place the reference node at the center of mass for the rigid body, a single MASS element and a ROTARYI element at the reference node are used to define the complete inertial properties for the rigid body.

The deformable pad is discretized into 10 equally spaced elements (CPE4R in two dimensions and C3D8R in three dimensions). A rigid plate has been added to the back face of the deformable pad, using R2D2 elements in two dimensions and R3D4 elements in three dimensions, to constrain these nodes to remain in a plane. This rigid plate is a second rigid body, firmly attached to the pad, and with the motion of its reference node constrained in all but the local *x*-direction. Hence, the pad is free to move toward the drum or away from it, but it can neither translate nor rotate in any other direction.

Results and discussion

The problem can be solved in closed form if we neglect the detailed behavior of the deformable pad. The normal contact force between the pad and the drum will be $f_n = pA =$ 3500 N, where $A =$ 0.01 m^2 is the area subjected to pressure loading, which leads to a tangential friction force of $f_t = 0.15 f_n =$ 525 N on the surface of the drum. The net torque about the axis of the drum is, therefore, $T = Rf_t =$ 105 Nm, leading to an angular deceleration of $\alpha = T/I =$ 600 rad/s. This should bring the drum to a complete stop over a time span of $\Delta t = \omega_0/\alpha =$ 0.10 seconds.

The following discussion of the results applies to the three-dimensional model of Case 1. An idealization of the problem is shown in *Figure 1*. A detail of the deformed shape of the brake pad at $t =$ 0.05 seconds is shown in *Figure 2*. A sequence of similar frames at different times in the analysis reveals intermittent stick and slip between the pad and the drum, leading to high frequency vibration of the pad. *Figure 3* is a time history plot of the total rotation of the drum, which is a very smooth curve. The time history plot for the angular velocity is shown in *Figure 4*, where we can clearly see the drum slowing down to an almost complete stop at $t =$ 0.10 seconds, followed by a steady rocking motion of the drum against the (still oscillating) pad. The slope of the left portion of the curve gives an average deceleration of 600 rad/s for the first 0.10 seconds, as expected. This is not so obvious from the time history plot of angular acceleration, shown in *Figure 5*, which is rather noisy. Similar noise levels are also observed in the time histories of the two components of the reaction force at the axis of the drum that are shown in *Figure 6*. Much of this noise is associated with intermittent stick and slip in friction and is not unusual for this type of problem. Despite the complex local behavior at this interface, the energy balance for the problem is maintained accurately, as shown in *Figure 7*.

The final results for all cases agree closely with the results from the three-dimensional Case 1.

Input files

braking2d_anl.inp	Two-dimensional Case 1 problem.
braking3d_rev_anl.inp	Three-dimensional Case 1 problem using an analytical rigid surface with TYPE=REVOLUTION.
braking3d_rev_anl_surffric.inp	Three-dimensional Case 1 problem using an analytical rigid surface with TYPE=REVOLUTION and surface property friction.
braking2d.inp	Two-dimensional Case 2 problem.
braking3d.inp	Three-dimensional Case 2 problem.
braking3d1.inp	Three-dimensional Case 3 problem.
braking3d2.inp	Three-dimensional Case 4 problem.
braking3d3.inp	Three-dimensional Case 5 problem.
braking3d_cyl_anl.inp	Three-dimensional Case 5 model using an analytical rigid surface with TYPE=CYLINDER.
braking3d_cyl_anl_surffric.inp	Three-dimensional Case 5 model using an analytical rigid surface with TYPE=CYLINDER and surface property friction.

Figures

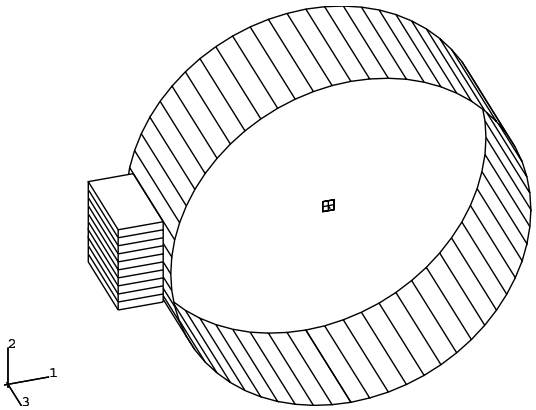


Figure 1: Three-dimensional idealization of the problem.

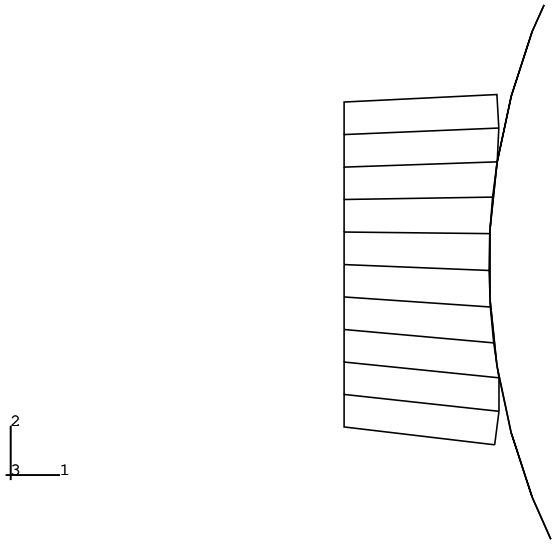


Figure 2: Deformed shape of the brake pad at $t = 0.05$ seconds.

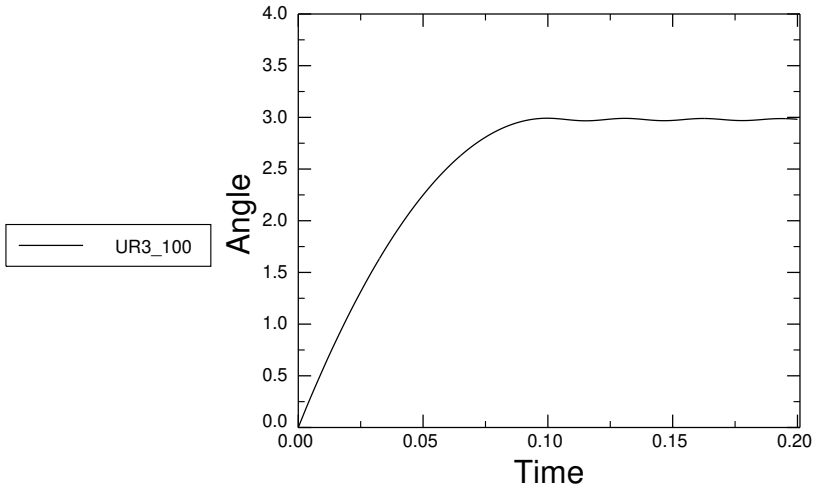


Figure 3: Time history plot of the total rotation of the drum.

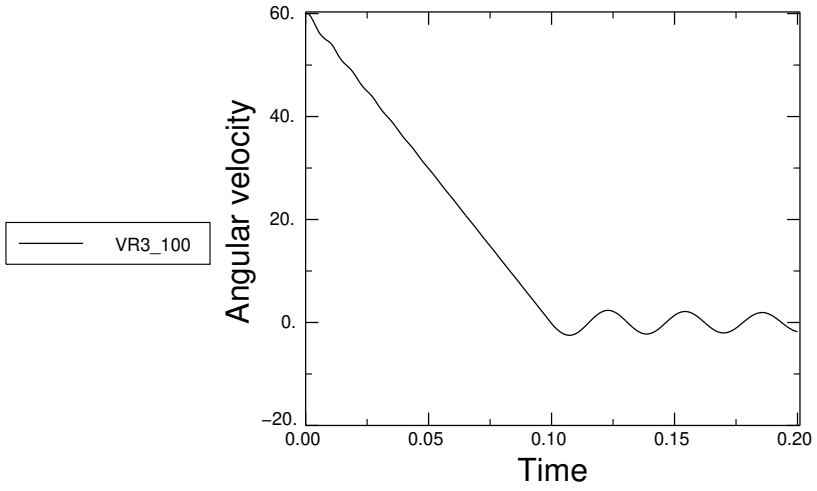


Figure 4: Time history plot of the angular velocity of the drum.

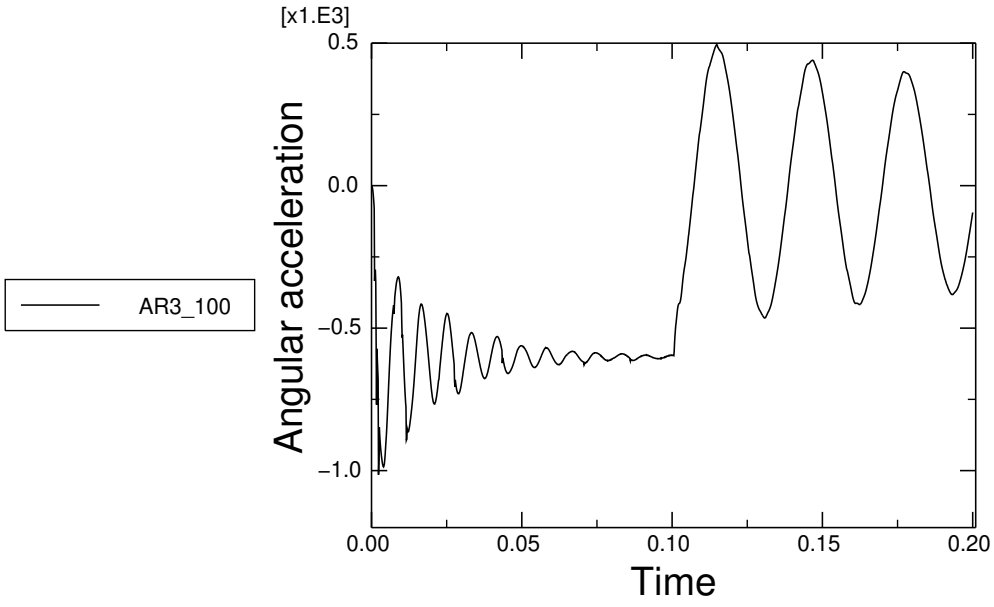


Figure 5: Time history plot of the angular acceleration of the drum.

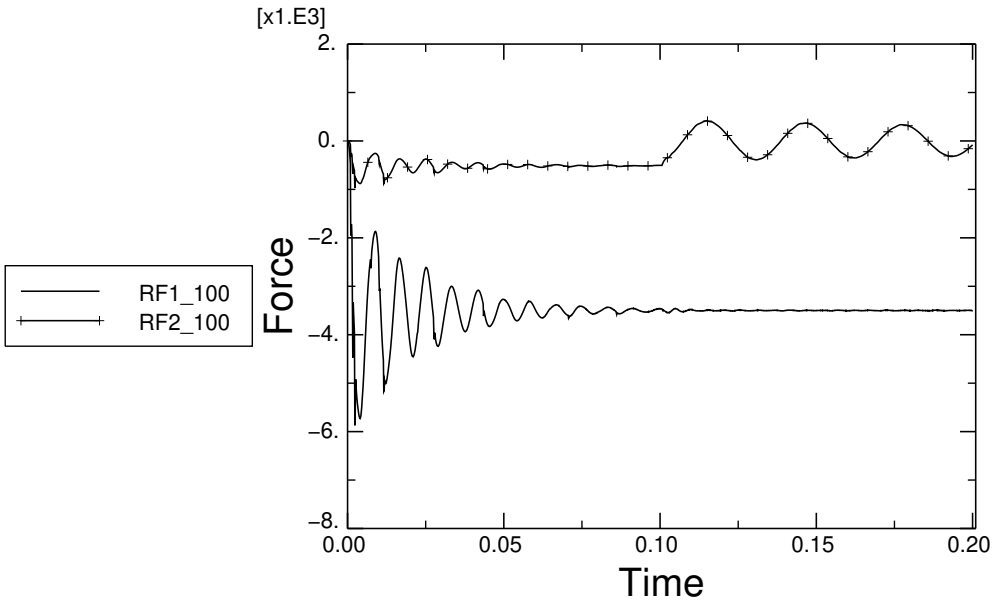


Figure 6: Time history plot of the reaction forces at the axis of the drum.

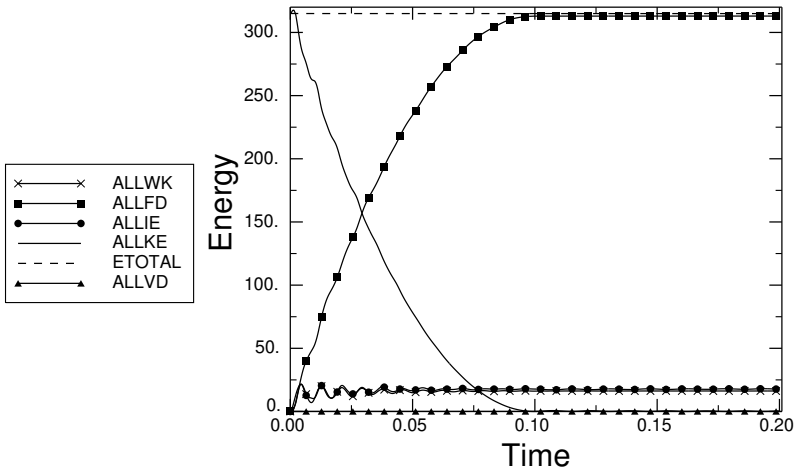


Figure 7: Time history plot of the energy balance.

Compression of cylindrical shells with general contact

Products: Abaqus/Explicit

Overview

This example models the compression of three interlocking cylindrical shells.

Problem description

In this example the shells are placed in a rigid box, and the top of the box is pushed downward at a constant velocity of 130 m/s for 10 ms. The shells are compacted into a volume approximately half the original volume of the box. *Figure 1* shows the original configuration of the cylinders in the box. The cylinders are shown from both the front and oblique views, with the front and right-side wall of the box removed.

This problem illustrates contact of double-sided shell surfaces. Models using each of the contact algorithms available in Abaqus/Explicit are provided. The primary model uses 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, models using penalty contact pairs and a combination of penalty and kinematic contact pairs are provided.

In the contact pair analyses self-contact interactions are not modeled since the three shells are not expected to undergo self-contact during the compression. Similar pair-wise definitions of contact are possible with the general contact algorithm and may result in minor improvements in computational efficiency.

“Bull-nose” extensions at the shell perimeters are present with the contact pair algorithm but not with the general contact algorithm; this difference between the two algorithms has some effect in this problem.

The element normals on several of the elements that make up the cylinders have been reversed to test the ability of Abaqus/Explicit to define the double-sided surface normals independently of the element normal.

The cylinders are made of steel, with a Young's modulus of 200 GPa, a Poisson's ratio of 0.3, and a density of 7850 kg/m^3. A von Mises elastic, linearly hardening plastic material model is used with a yield stress of 250 MPa.

Results and discussion

Figure 2 and *Figure 3* show the deformed shape of the cylinders after 5 and 10 msec, respectively. Results for the contact pair analyses are shown on the left of each figure; results for the general contact analysis are shown on the right. The effect of the “bull-nose” extensions at the shell perimeters is visible in the deformed shape plots for the contact pair analyses.

Figure 4 shows the time history of the total kinetic energy, the total work done on the model, the plastic dissipation, and the total energy balance for the model that uses the general contact algorithm. The other models give similar results.

This problem tests the features listed but does not provide independent verification of them.

Input files

shell_compact.inp	Primary analysis using the general contact capability.
shell_compact_cpair.inp	Model that uses contact pairs with kinematic contact.

shell_compact_cpair2.inp	Model that uses contact pairs with kinematic contact. The shell normals are reversed.
shell_compact_cpair3.inp	Model that uses contact pairs with kinematic contact. The NO THICK parameter is used when defining the surfaces for the lid and the center ring.
shell_compact_pnlty.inp	Analysis that uses contact pairs with penalty contact.
shell_compact_ef1.inp	External file referenced by these analyses.
shell_compact_ef2.inp	External file referenced by these analyses.
shell_compact_ef3.inp	External file referenced by these analyses.

Figures

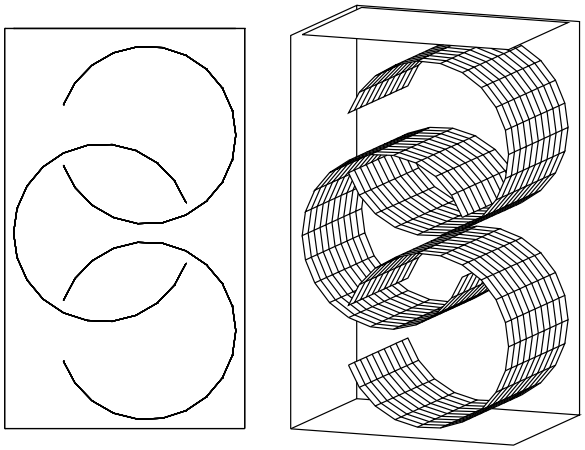


Figure 1: Initial configuration of the cylinders in the box from front and oblique views (front and right box walls removed).

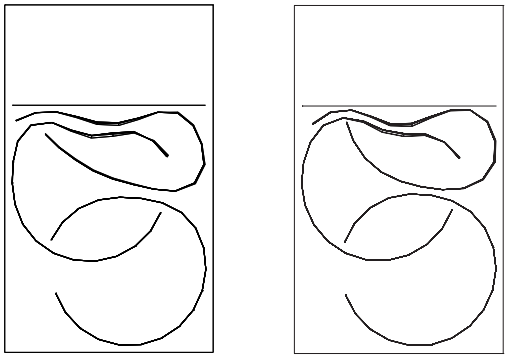


Figure 2: Deformed shape at 5.0 msec (contact pair analysis on the left, general contact analysis on the right).

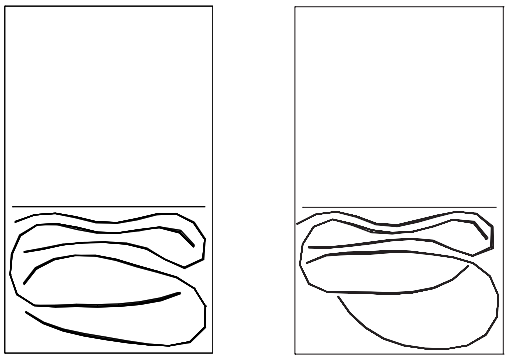


Figure 3: Deformed shape at 10.0 msec (contact pair analysis on the left, general contact analysis on the right).

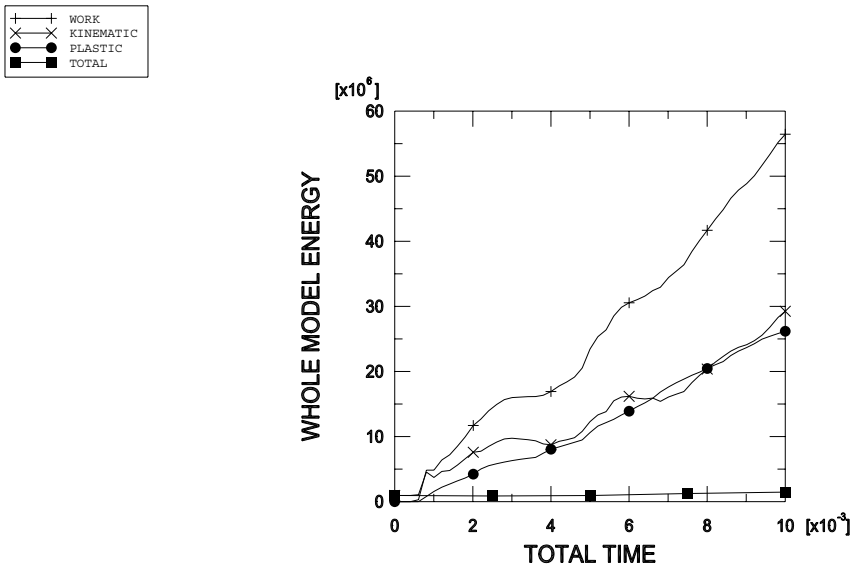


Figure 4: Time histories of the total kinetic energy, work, plastic dissipation, and internal energy.

Steady-state slip of a belt drive

Products: Abaqus/Explicit

Overview

This example predicts the steady-state slip of a belt drive.

Problem description

This problem consists of a pre-tensioned elastic belt wrapped 180° around a 1 m diameter rigid drum. The belt is fixed at one end and has a constant force of 50000 N applied at the other end. The interaction between the belt and the drum is governed by a Coulomb friction law with a coefficient of friction $\mu = 0.2$. The objective of the analysis is to predict the steady-state resisting moment as the drum is turned. This moment corresponds to the difference in forces at the two belt ends times the moment arm of 0.5 m. The difference in force is maximized at the steady-state slip condition, which can be simulated by prescribing a rotation of the drum.

The analysis is run in two steps: in the first step the belt is pre-tensioned while keeping the drum fixed, and in the second step the drum is accelerated to a prescribed angular velocity. The pre-tensioning force and the prescribed angular velocity are ramped up using a smooth-step amplitude curve. This amplitude definition provides a smooth loading rate, which is desirable in quasi-static or steady-state simulations. Mass proportional damping is used to further reduce oscillations in the response.

Results and discussion

The analytical solution for this problem can be found in many mechanical engineering handbooks. At the steady-state slip condition the ratio of the belt force at the tight end to the belt force at the loose end is given by

$$\frac{F_{tight}}{F_{loose}} = e^{\mu\beta},$$

where β is the wrap angle in radians. Since the drum is turned toward the end of the belt with the concentrated force, this end becomes the loose end. Thus, $F_{loose} = 50000$ N. Using the above relation, the force at the fixed end of the belt, $F_{tight} = 93723$ N. The steady-state resisting moment at the slip condition is then (93723 − 50000) × 0.5 = 21862 N-m.

The plot of reaction moment at the drum's reference node versus time in *Figure 2* has three distinct regions. The first region corresponds to the pre-tensioning step. The reaction moment gradually ramps to a negative value and remains constant at that value for the remainder of the first step. The second region corresponds to the portion of the second step in which the prescribed rotary acceleration of the drum is nonzero (the velocity is being ramped up). The reaction moment overshoots the analytical steady-state value of 21862 N because this reaction moment includes the rotary inertia of the drum as it is accelerated. The third region corresponds to a constant velocity of 20 rad/s of the drum. In this region rotary inertia no longer plays a role, and the predicted resisting moment solution oscillates slightly about the analytical value.

The analysis is performed in Abaqus/Explicit using contact pairs as well as general contact. The Abaqus/Explicit results show good agreement with the analytical solution.

Input files

pulley_rev_anl.inp	Three-dimensional model using **SURFACE*, TYPE=REVOLUTION and contact pairs.
pulley_rev_anl_gcont.inp	Three-dimensional model using **SURFACE*, TYPE=REVOLUTION and general contact.
pulley_seg_anl.inp	Two-dimensional model using **SURFACE*, TYPE=SEGMENTS.
pulley_cyl_anl.inp	Three-dimensional model using **SURFACE*, TYPE=CYLINDER and contact pairs.
pulley_cyl_anl_gcont.inp	Three-dimensional model using **SURFACE*, TYPE=CYLINDER and general contact.

Figures

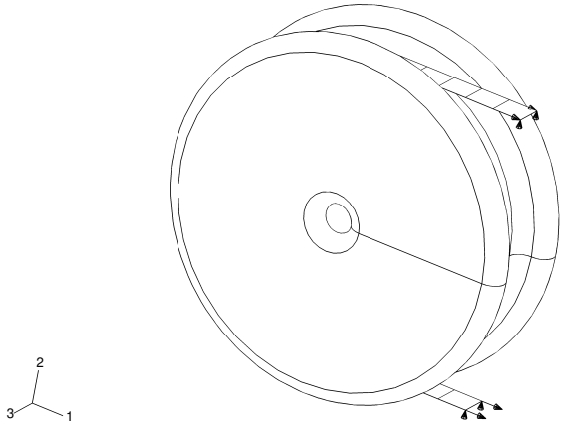


Figure 1: Three-dimensional belt on a rigid drum.

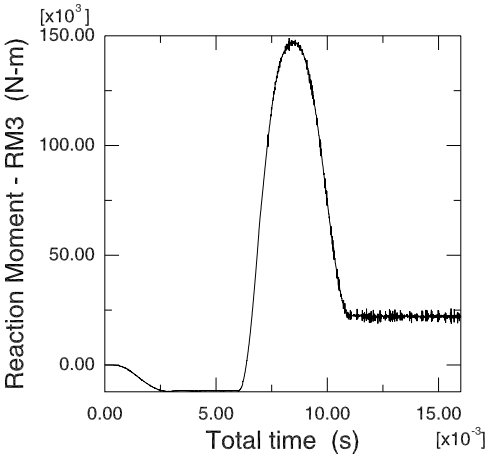


Figure 2: Reaction moment history at the drum's reference node.

Crash simulation of a motor vehicle

Products: Abaqus/Standard

Overview

This example is an elementary illustration of a motor vehicle crash simulation.
The case in this example is one for which experimental results are available (Mouldenhauer, 1980), thus providing verification of the numerical results.

Problem description

Figure 1 shows the structure, which is a scale model of a typical motor vehicle frame made of steel. The frame is moving forward at a speed of 13.89 m/s (50 km/h—about 31 miles/hour) when it collides against an oblique, rigid wall that is at 30° to its direction of motion. The objective of the analysis is to predict the history of deformation of the frame during the crash event.

The dimensions of the physical structure are shown in *Figure 1*. The finite element idealization is shown in *Figure 2*. First-order beam elements (element type B21) are used to model the frame.

The contact between the frame and a flat, rigid wall is modeled with a contact pair. The individual nodes of the frame that may be involved in contact with the wall are assigned to a node-based surface. Alternatively, the exterior surface of the frame could have been defined using an element-based surface. The rigid wall is modeled as an analytical rigid surface with a rigid body constraint in conjunction with a surface definition. The mechanical interaction between the node-based surface and the rigid surface is assumed to be frictionless.

No mesh convergence studies have been performed, but the reasonable comparison between the results of this analysis and the experimentally observed deformation suggests that the mesh is adequate, in the sense that major aspects of the behavior are predicted fairly well.

The frame is oriented along the x-axis, facing the rigid surface toward the left. The initial velocity of 13.89 m/s is prescribed for each node of the frame in the negative x-direction.

Controls and tolerances

This analysis clearly involves large deformations, so geometric nonlinearities must be included in the step definition.

The automatic time stepping algorithm for implicit dynamic integration requires that a half-increment residual tolerance be set. In an example like this we aim to obtain a solution of moderate accuracy and low computational cost. Also, this problem involves very large energy dissipation (caused by plastic deformation) and, consequently, the high frequency response will be damped rapidly. Thus, a half-increment residual tolerance that is an order of magnitude or two larger than actual typical forces should give acceptable results.

A typical force magnitude can be estimated by considering the force required to produce a fully plastic hinge in a member, based on a reasonable length of cantilever. The moment at a fully plastic hinge in a rectangular section is

$$M^0 = \tfrac{1}{4} h^2 w \sigma^0,$$

where σ^0 is the yield stress, h is the thickness of the section in the plane in which it bends, and w is the width of the section in the other direction. The force required to produce this moment in a cantilever of length L is

$$F^0 = M^0 / L.$$

Using the front segment of one of the side rails to compute F^0 for this problem gives a value of 135 N. Based on this calculation, we set the half-increment residual tolerance to 10000 N.

Material

The material has a Young's modulus of 213 GPa and a mass density of 7850 kg/m^3. It has an initial yield stress of 221.2 MPa, with isotropic hardening to a stress of 250 MPa at a plastic strain of 5.5×10^{-4} and perfect plasticity beyond that strain value.

The rigid surface is assumed to be frictionless.

Results and discussion

The implicit analysis requires about 420 increments to reach a stage in which the entire front of the frame is in contact with the rigid surface and the frame has essentially collapsed. At early stages of the analysis the time increments are very small because the initial impact initiates stress wave effects and these waves propagate throughout the model, carrying energy with them: small increments are required to model the dynamics accurately during this period. Later, the high frequency response is damped out by plastic yielding, and the time increment can be increased with no loss in accuracy.

Figure 3 shows the predictions of the deformed configuration at various times and provides an illustration of the history of the event. *Figure 4* compares the predicted configuration at 10 ms with the results of an experimental study. The correlation between the analysis and the experimental result is quite encouraging, especially considering the relatively coarse mesh. *Figure 5* shows the variation of the total kinetic energy, strain energy, and the plastic dissipation in the frame with respect to time. After 12.5 ms about one-fifth of the initial kinetic energy has been dissipated as plastic work.

Input files

autocrashsimulation.inp	Implicit analysis.

References

- Moldenhauer, H., “Oblique Impact of a Motor Vehicle (Crash simulation with Abaqus),” *Control Data Corporation, Frankfurt, W. Germany*, July 1980.

Figures

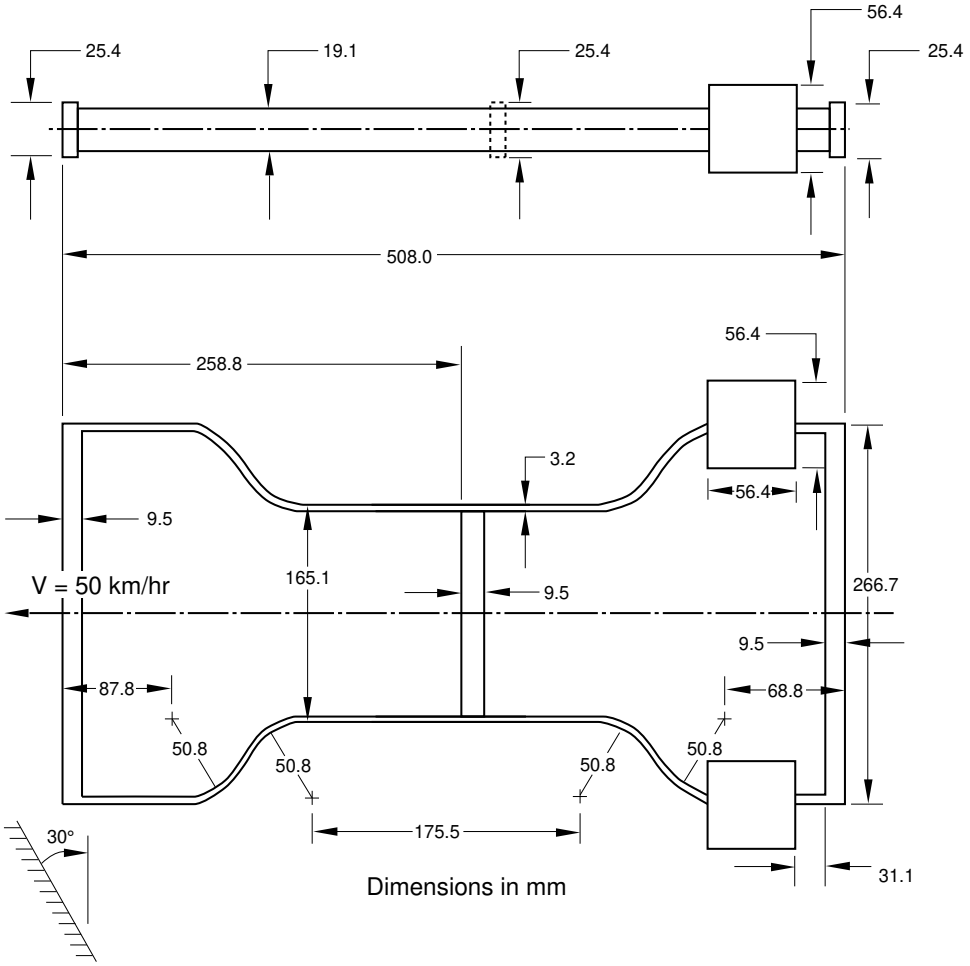
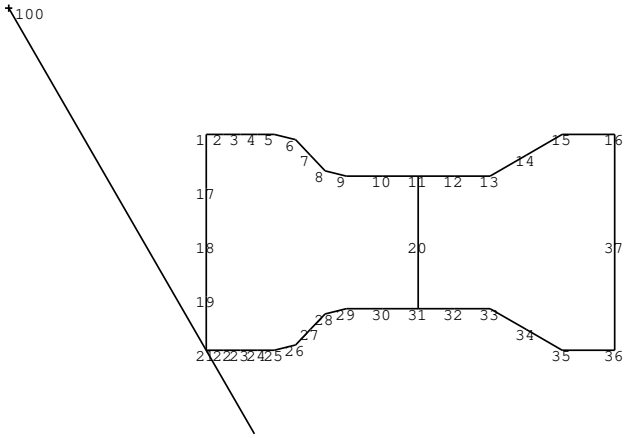
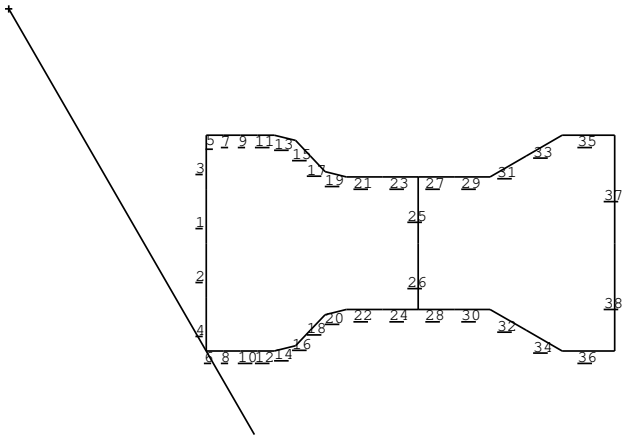


Figure 1: Motor vehicle frame crash study.



(a) Nodes



(b) Elements

Figure 2: Motor vehicle frame crash study: finite element model.

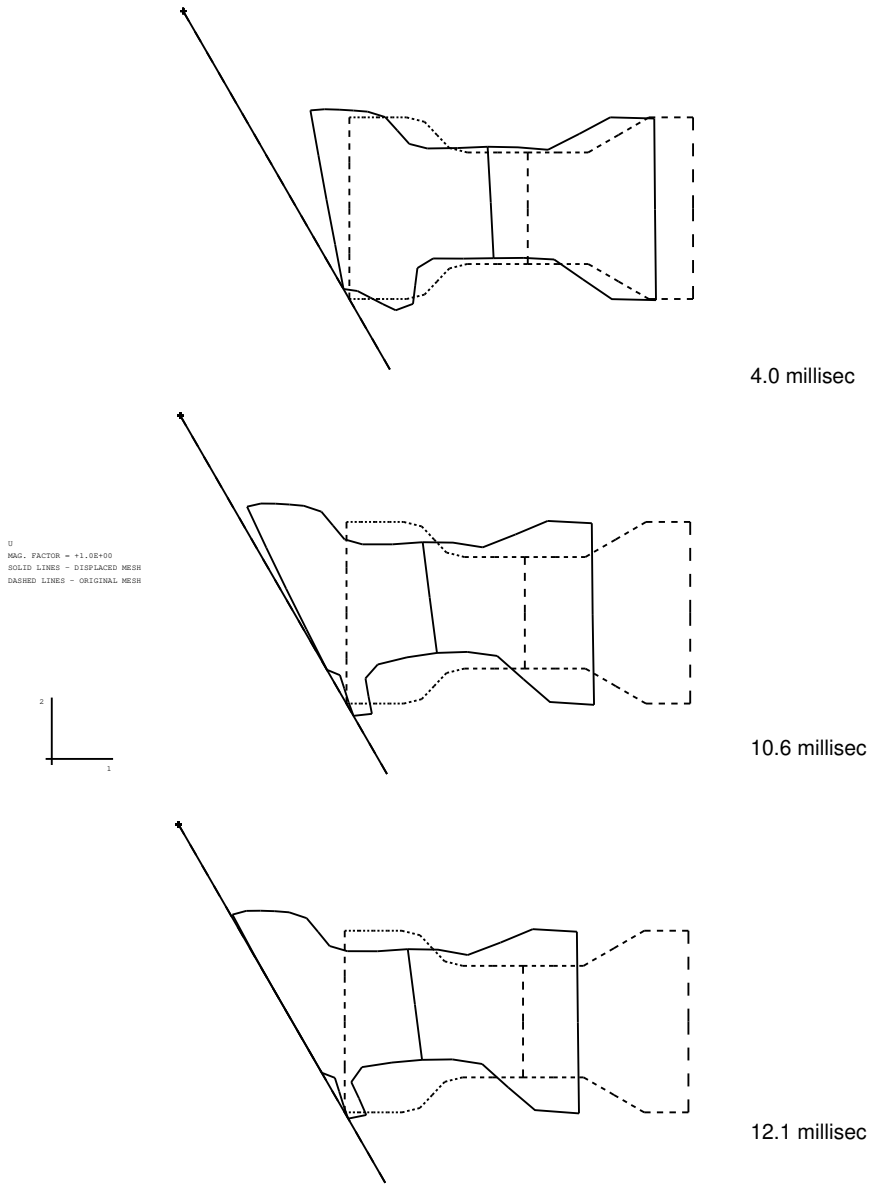


Figure 3: Deformation configurations.

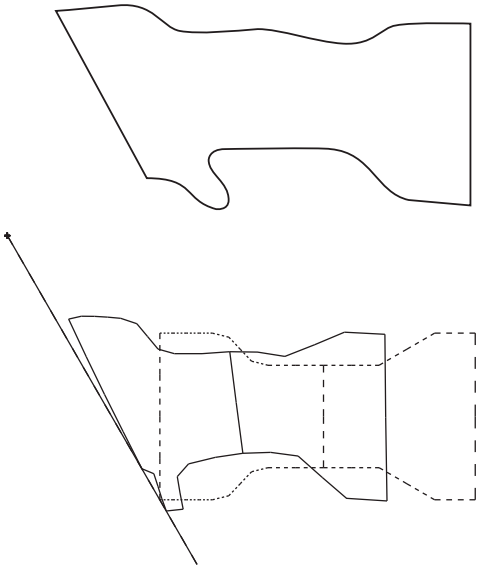


Figure 4: Comparison of measured and predicted configurations at 10 ms.

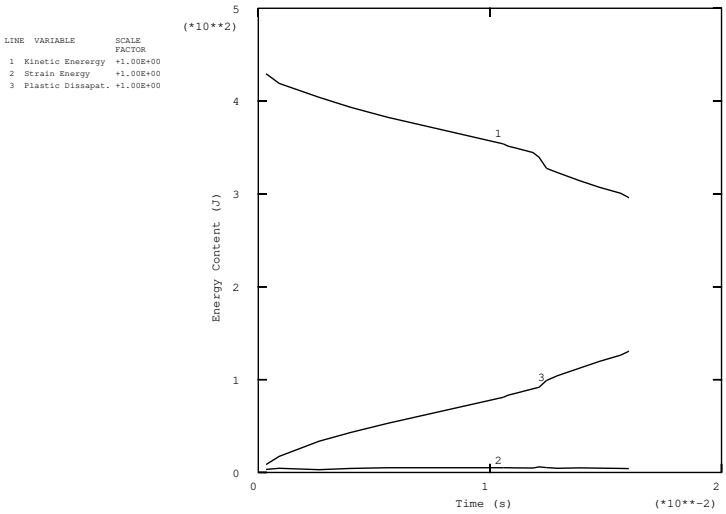


Figure 5: Total energy content throughout the solution.

Truss impact on a rigid wall

Products: Abaqus/Standard Abaqus/Explicit

Overview

This problem demonstrates characteristics of kinematic contact and penalty contact in Abaqus/Explicit and dynamic contact in Abaqus/Standard.

Problem description

The problem in this example investigates the dynamic response of a truss impacting a rigid wall. The analysis is completed with a coarse and a refined mesh as shown in *Figure 1* and *Figure 2*, respectively.

The truss has a length L=2 m and a cross-sectional area A=0.2 m^2. Boundary conditions act on the truss nodes to allow horizontal motion only, reducing the problem to one dimension. For the coarse mesh the truss is discretized using five T2D2 elements; 10 elements are used for the refined mesh analysis. The truss is made of steel, with Young's modulus of E=200 GPa, Poisson's ratio of ν=0.3, and density of ρ=7800 kg/m^3. The material remains linearly elastic. The initial velocity of the truss is v_0=1.5 m/s toward the rigid wall. The rigid wall is modeled using one R2D2 element. The wall is held in a fixed position. An initial clearance of 0.001 m between the truss and the wall is considered (see *Figure 3*); impact should occur at 6.67×10^{-4}s.

The analytical solution predicts that the kinetic energy of the truss will be converted entirely to strain energy as the truss is compressed during impact; this strain energy will then be converted entirely back to kinetic energy as the truss rebounds, so the truss will leave the wall with a uniform velocity of 1.5 m/s. After the initial contact is established, a stress wave will travel along the truss at a rate of $c = (E/\rho)^{1/2}$=5064 m/s. The analytical solution for the duration of the impact is $T = 2L/(c + v_0)$=7.9×10^{-4} s, during which time the contact force remains at a constant value of F = $\rho v_0 (c + v_0) A$=11.8×10^6 N. The momentum change of the truss corresponds to the contact force multiplied by the impact duration: $FT = 2\rho LAv_0$=9.36 kg m/s.

Two approaches are used to model the contact between the leading truss node and the rigid wall. In the first approach contact is defined using the default kinematic contact formulation in Abaqus/Explicit. The second approach uses the penalty contact formulation. The default penalty stiffness is used. The differences in these two contact formulations are discussed in greater detail in *Contact Constraint Enforcement Methods in Abaqus/Explicit* and in the results section that follows.

Results and discussion

Verification for this problem is provided by comparing the values of significant problem variables with the analytical solution. The numerical solutions are based on the default time incrementation except where noted.

Plots of kinetic energy are shown in *Figure 4*. Four stages of the solution (pre-impact, truss compression, truss re-expansion, and postimpact) are apparent in this plot. When penalty contact is used, the latter stages are delayed and changes in the slope at the transitions between these stages are smoothed. The onset of truss compression is advanced in time by one increment with kinematic contact. In each of the numerical solutions the kinetic energy is not entirely recovered upon rebound because of the numerical dissipation of energy and finite discretization. For the penalty contact solutions the dissipation of energy is primarily caused by small amounts of bulk viscosity (included by default in the Abaqus/Explicit element formulations) and viscous contact damping (included by default for penalty contact). For the kinematic contact solutions both the bulk viscosity and the contact algorithm itself contribute significantly to the loss of energy. The kinematic contact algorithm dissipates

the kinetic energy of the contact node upon impact, whereas the penalty contact algorithm converts the kinetic energy of the contact node into energy stored in the stretched penalty spring. These energy transfer considerations will be discussed further in the following paragraphs.

Velocity histories of the leading truss node (contact node) are plotted in *Figure 5*. The kinematic contact solutions for velocity closely match the analytical solution during pre-impact and during impact. The impact stage is less distinct in the velocity plots for penalty contact because some penetration occurs. All numerical solutions for the postimpact velocity show some oscillations that are not part of the analytical solution. These oscillations are associated with the energy dissipation and finite discretization. In the kinematic contact solutions a stress wave continues to pass through the truss during the postimpact phase, which periodically reduces the magnitude of the nodal velocity. This wave becomes narrower as the mesh is refined. With penalty contact a postimpact stress wave persists, which causes the postimpact nodal velocity to oscillate about approximately −1.5 m/s, where the negative velocity indicates movement in the negative x-direction. In all numerical solutions these velocity oscillations become more diffuse over time as a result of the bulk viscosity damping.

Contact force history solutions are plotted in *Figure 6*. For the kinematic contact tests Abaqus/Explicit gives very good estimates of the peak contact force and captures the steps in the contact force history quite well. However, it will be shown later that the contact force history with kinematic contact depends on the size of the time increment used in the analysis. The penalty contact force solutions produce reasonable estimates of the peak contact force, but because of the inherent numerical softening of the penalty method, extreme mesh refinement is needed to observe sudden jumps in contact force.

Figure 7 contains plots of external work. The external work remains zero in the analytical solution. Some external work associated with contact forces, which are treated as external forces in Abaqus/Explicit, can be observed in the numerical solutions. With penalty contact the external work accounts for the energy stored in the penalty springs during contact penetration and the energy dissipated by viscous contact damping. After the rebound the external work returns to a constant negative value as the penalty spring energy is recovered; the negative value corresponds to the amount of dissipation due to viscous contact damping. With kinematic contact a contact force first occurs in the increment just prior to the actual impact when a gap is still present; thus, penetration does not occur in the next increment. Therefore, the kinematic contact force does some work when contact is first established. This work corresponds to the kinetic energy of the contact node, and this energy is dissipated by the contact algorithm and is not recovered upon the rebound.

The energy dissipation caused by the bulk viscosity is plotted in *Figure 8*. This dissipation is greater with kinematic contact than with penalty contact because impacts in the kinematic contact formulation are not softened. Greater shock to the elements and increased element damping occur. Energy continues to dissipate after the rebound as a result of damping of stress waves that persist in the truss after the rebound.

Plots of strain energy are shown in *Figure 9*. The energy stored in penalty springs is not included in the strain energy reported by Abaqus/Explicit, because the contact forces are treated as external forces. Instead, the energy stored in penalty springs appears as negative external work, as mentioned previously. Some strain energy remains after the rebound in the numerical solutions, which is related to stress waves that remain in the truss.

An undesirable characteristic of the kinematic contact algorithm is that the initial impact force predicted for a given mesh over the contact region depends on the size of the time increment. The contact force results shown in *Figure 10* are based on analyses in which the time increment was scaled by 0.25. This scaling simulates the presence of a small element in the model that would control the time increment size. The kinematic contact algorithm will overestimate impact forces if the time increment is significantly lower than the stable time increments of elements near the contact region. Reducing the time increment causes the contact force to increase, because the approach speed of the leading node must be resolved over a shorter time interval to avoid penetration upon impact. *Figure 10* also shows that the time increment size has negligible influence on the contact force solution if the penalty contact formulation is used. Other solution variables discussed in this example have minimal dependence on the size of the time increment for both types of contact constraint methods.

To better understand these results, consider a single secondary node impacting a fixed rigid wall. *Figure 11* and *Figure 12* show such a contact secondary node as a circle in increment $\boldsymbol{\nu}$. Friction will not be considered.

In the kinematic contact formulation Abaqus/Explicit calculates a predicted penetration d_{penet}^{pred} (see *Figure 11*). This predicted penetration is equal to the movement of the node if no contact condition is enforced. Abaqus/Explicit then calculates the contact force, f_ν, in the normal direction according to

$$f_\nu = \frac{m d_{penet}^{pred}\big|_\nu}{\Delta t_{\nu+1}\frac{\Delta t_\nu + \Delta t_{\nu+1}}{2}}$$

and applies this force in the current increment. The contact force is applied before the contact is actually established. In the next increment, ν+1, the node contacts the surface of the opposing body without penetration (see *Figure 11*) and the loss of kinetic energy occurs. Although not shown in *Figure 11*, a contact force will also occur in increment ν+1 in the case of kinematic contact to eliminate the remainder of the velocity component normal to the surface.

Figure 12 shows the schematic for the penalty contact formulation. The contact force is first applied in increment ν+1, and some penetration of the node into the opposing surface occurs. The contact force $f_{\nu+1}$ is calculated according to

$$f_{\nu+1} = k d_{penet}^{cur}\big|_{\nu+1} + c v_{penet}^{cur}\big|_{\nu+1},$$

where k is the penalty stiffness calculated by Abaqus/Explicit, c is the viscous damping coefficient calculated from the default contact damping setting, and v_{penet}^{cur} is the penetration velocity. The penalty stiffness term can be envisioned physically as a spring attached between the penetrating node and the surface being penetrated. The energy is stored in this spring and is released as the node penetration reverses and decreases to zero (see *Figure 7*). The small amount of kinetic energy lost (see *Figure 4*) is the result of viscous effects of the elements, viscous contact damping, and strain energy remaining in the truss after separation (see *Figure 9*). As the mesh is refined, both formulations tend toward the analytical solution.

Input files

Abaqus/Standard input file

imp_ref_std.inp	Analysis of the refined model.

Abaqus/Explicit input files

imp_pnl_ref.inp	Analysis of the refined model using the penalty contact formulation.
imp_kin_ref.inp	Analysis of the refined model using the kinematic contact formulation.
impact_kin.inp	Analysis of the coarse model using the kinematic contact formulation.
impact_pnl.inp	Analysis of the coarse model using the penalty contact formulation.
imp_pnl_ref_sc.inp	Analysis of the refined model using the penalty contact formulation and a scaled time increment.
imp_kin_ref_sc.inp	Analysis of the refined model using the kinematic contact formulation and a scaled time increment.
impact_kin_sc.inp	Analysis of the coarse model using the kinematic contact formulation and a scaled time increment.
impact_pnl_sc.inp	Analysis of the coarse model using the penalty contact formulation and a scaled time increment.

Figures

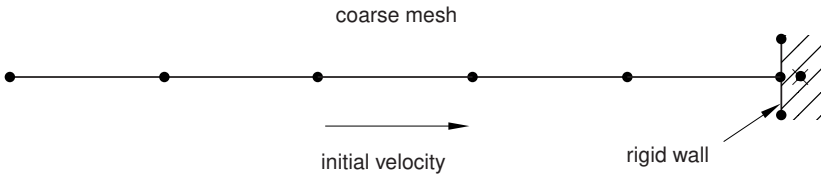


Figure 1: Coarse mesh model.

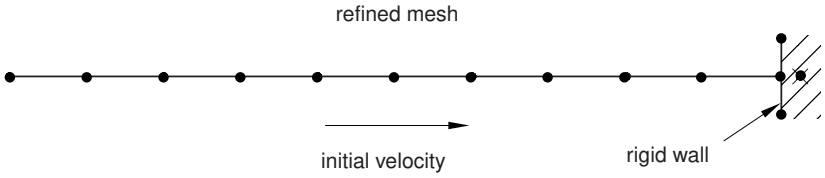


Figure 2: Fine mesh model.

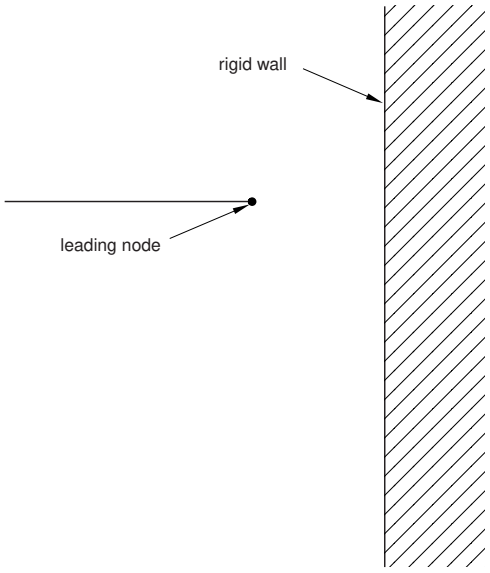


Figure 3: Initial gap.

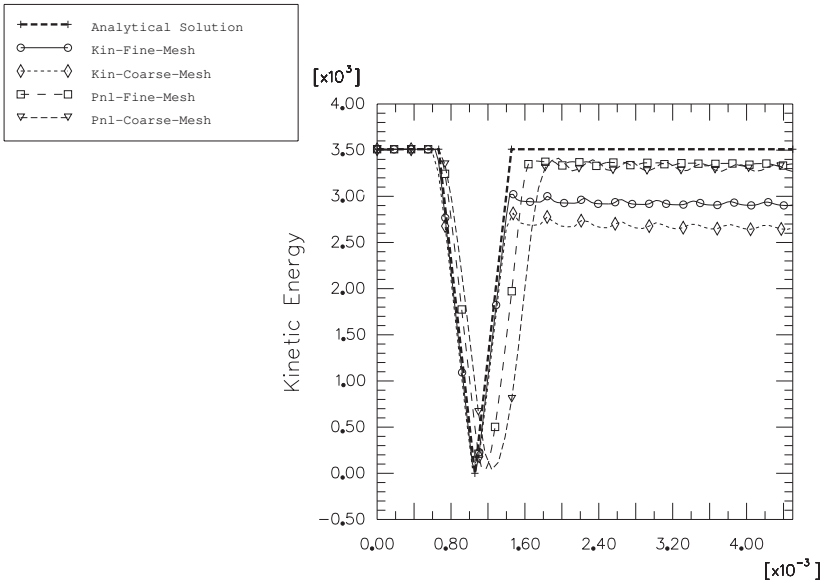


Figure 4: Kinetic energy.

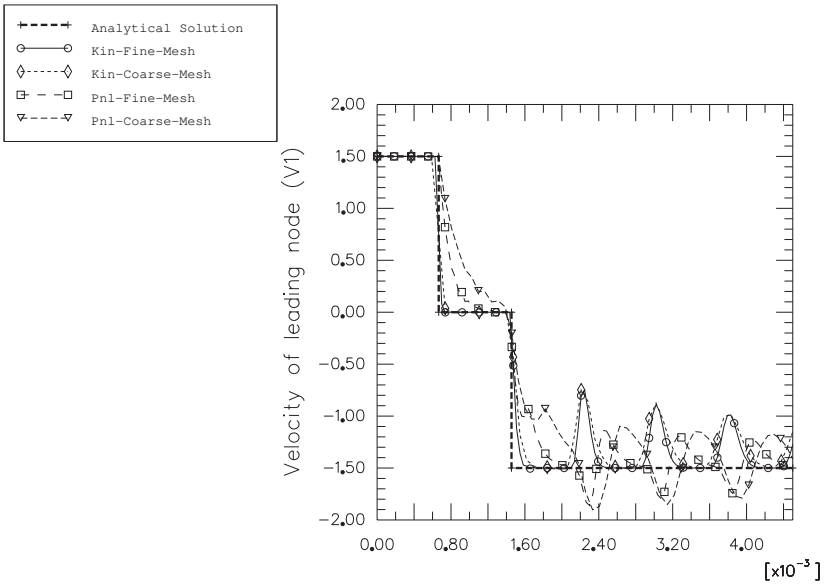


Figure 5: Velocity of leading node.

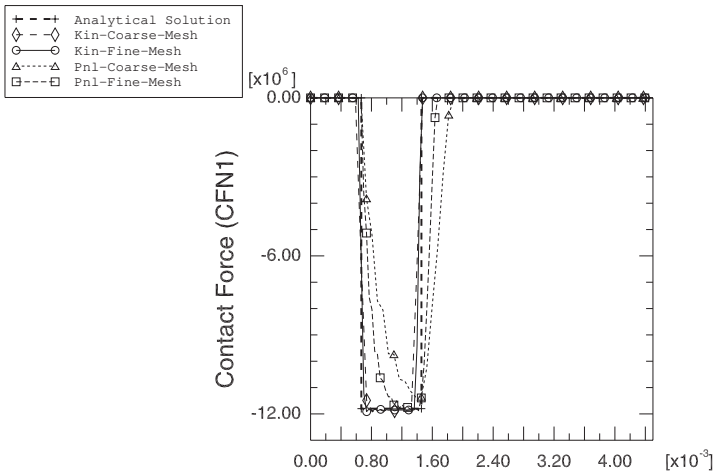


Figure 6: Contact force.

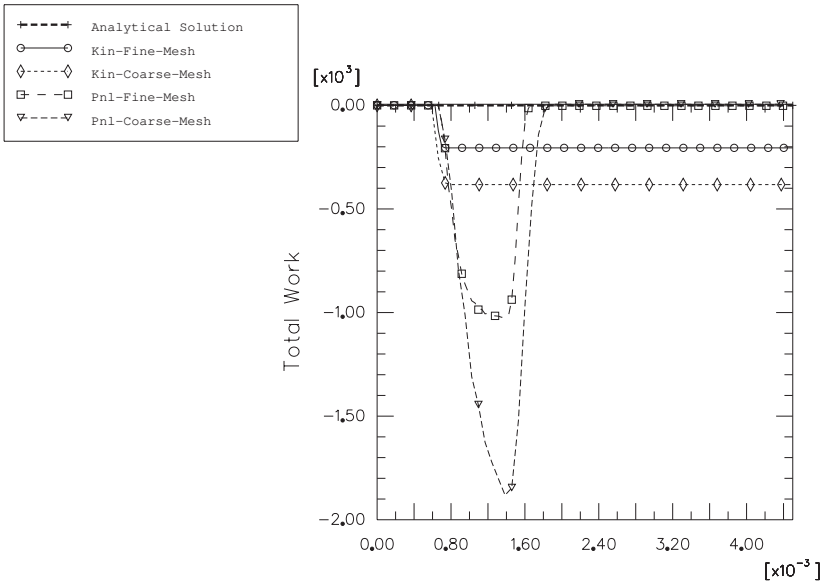


Figure 7: External work.

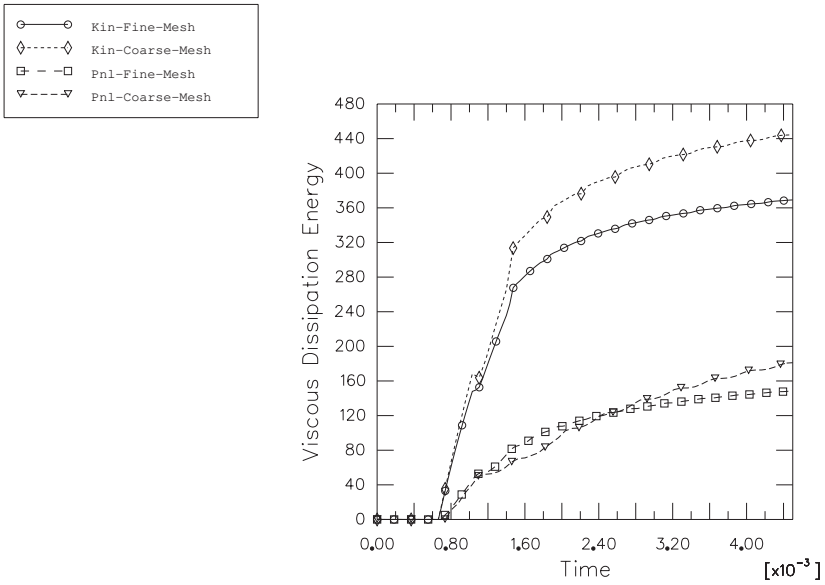


Figure 8: Viscous damping energy.

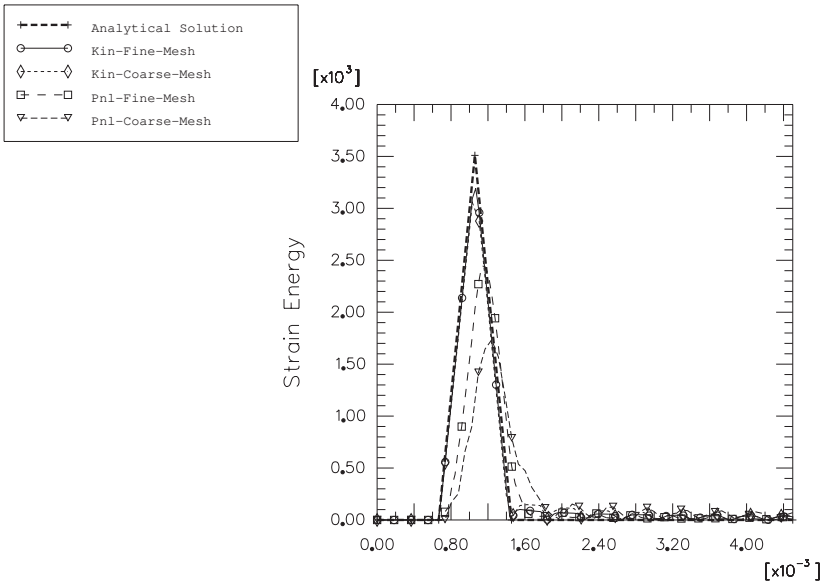


Figure 9: Strain energy.

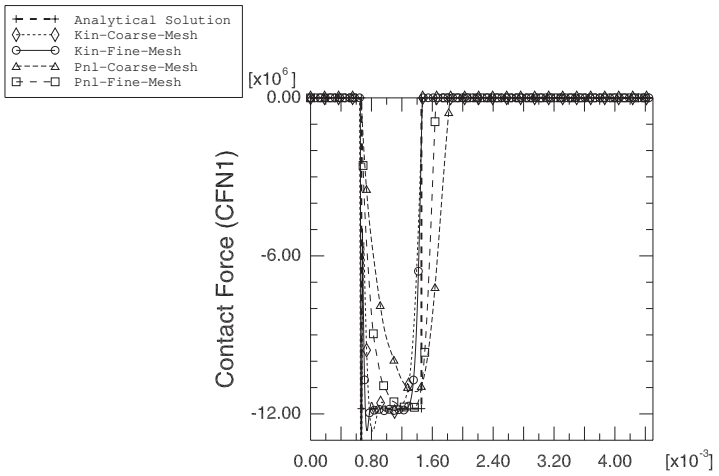


Figure 10: Contact force with scaled time increment.

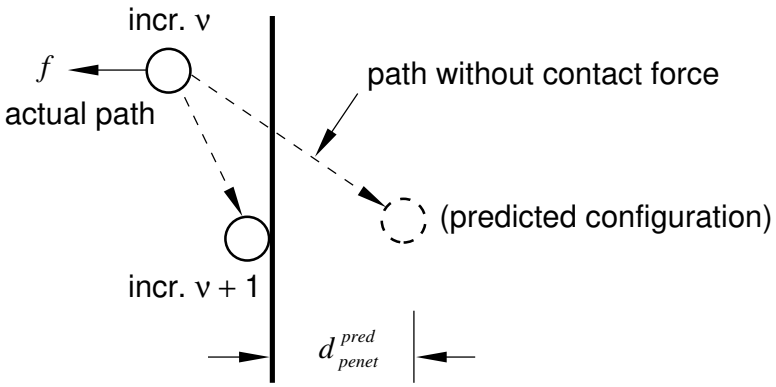


Figure 11: Schematic of kinematic contact formulation.

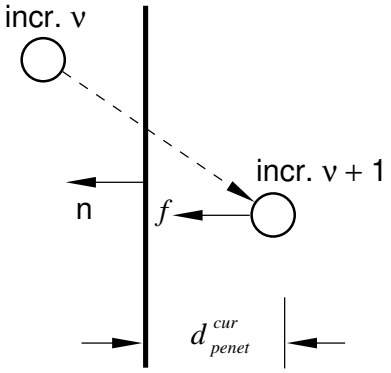


Figure 12: Schematic of penalty contact formulation.

Plate penetration by a projectile

Products: Abaqus/Explicit

Overview

This example compares three modeling approaches for the penetration of a plate by a high-speed projectile.

Problem description

This example consists of a two-dimensional axisymmetric plate penetrated by a high-speed projectile. The plate, which is made of Aluminum 2024–T4, has a thickness of 1.3 mm and is 50 mm in diameter. All degrees of freedom on the circumference of the plate are constrained. The outline of the model is shown in *Figure 1*. The projectile is modeled as an analytical rigid surface with a body diameter of 20 mm, an enclosed tip angle of 40°, and a mass of 0.11 kg attached to the reference node.

Figure 2 shows one of the finite element meshes used to model the plate, with 5 axisymmetric elements in the through-thickness direction and 50 axisymmetric elements in the radial direction. Nodes along the *Z*-axis in an axisymmetric model have no implicit constraints to remain at $r = 0$. In most axisymmetric problems it is appropriate to specify radial constraints for these nodes. However, in this example radial constraints for the nodes initially at the center of the plate are inappropriate, since the projectile will form a hole in the center of the plate. The tip of the projectile is assigned a small negative radial position to avoid the possibility of missed contact at the edge of the analytical surface due to numerical round-off. The nodes of the plate on the *Z*-axis will expand radially upon impact of the projectile, allowing projectile penetration and the formation of a hole. Element removal as a result of material failure is also modeled and further will contribute to enlargement of the hole. A potentially significant “petaling” mechanism, in which cracks emanate radially in the plate as the projectile passes through, is not studied in this example because of the two-dimensional nature of the axisymmetric models used.

If element failure and removal are not included in a high-speed impact problem such as this, the analysis will likely terminate prematurely as a result of severe element distortion. A node-based surface comprised of all the nodes on the plate is used for contact modeling purposes because element-based surfaces should not be defined over elements that fail and because nodes internal to the plate may become exposed once surrounding elements start failing. When all the elements attached to a node have failed, the node acts as a point mass and is still active in contact interactions. This aspect can be significant with respect to accurate modeling of momentum transfer in highly dynamic problems.

The results of interest are the velocity of the projectile at the end of the analysis and the work performed by the projectile, which is equal to the projectile's loss of kinetic energy. Analyses are conducted at initial projectile speeds of 400 m/s, 600 m/s, 800 m/s, and 1000 m/s. The projectile speed decreases by a small fraction in each analysis. The time period of the analysis is set such that the projectile penetration, assuming no decrease in velocity, is 55 mm. The results are compared to experimental results, as well as results obtained from analytical expressions based on simplifying assumptions commonly used for this type of problem. To determine the best modeling approach, three preliminary studies are conducted: a mesh convergence study, a comparison of the contact algorithms (kinematic and penalty), and a material model study. All preliminary studies are completed with an initial projectile velocity of 600 m/s. The parametric study capability of Abaqus is used to facilitate these studies.

Mesh convergence is studied for 5, 7, and 9 elements through the thickness and 50, 70, and 90 elements in the radial direction. Each mesh is biased toward the center of the disc, as in the 5 × 50 element mesh shown in *Figure 2*. At least four linear reduced-integration elements through the thickness should be used when bending may be significant. For high-speed impact problems such as this, bending may not be highly significant, because the

material may fail in shear prior to the occurrence of significant bending; however, fairly refined meshes are considered to provide examples of meshes that could also be used for low-speed impact studies. The input file *pp_mesh_study.inp* is parameterized for the mesh convergence study and is driven by the parametric study script *pp_mesh_study.psf*. The material is modeled with Mises plasticity with isotropic hardening and a plastic shear failure strain of 50%. The default kinematic contact algorithm is used for the mesh convergence study.

The effect on the results of the choice of the contact algorithm (kinematic or penalty) is investigated next, using the same material model and a mesh found to be efficient and accurate. The script *pp_con_study.psf* drives the parameterized input file *pp_con_study.inp*. We anticipate that the results will not differ significantly for the two contact algorithms.

The third preliminary study is completed to determine the influence of the material model on the analysis results. Material models of two basic types are considered: Mises plasticity with isotropic hardening and Mises plasticity without hardening but a higher yield stress. The hardening data for the first case are calculated using the function $\sigma = 44.2 + 29.2 * \epsilon^{0.338}$, where the units for stress are ksi and the strain is plastic strain. This function is consistent with data available from the Aluminum Association and represents an average of many tests. These hardening data are used for demonstration purposes; they may not be applicable to all situations. The perfectly plastic model is a simplification. Both models have similar strain energy for strains of about 15%. Plots of stress versus total log strain for both material models are shown in *Figure 3*. Two values of the equivalent plastic strain at failure (17% and 50%) are considered for each type of plasticity data. The value of 17% corresponds to the percent of total elongation at failure of a 2-inch specimen in a standard tensile test, as published by the Aluminum Association. The value of 50% is commonly used in high-rate dynamic analyses. Element failure is controlled with the shear failure model. The input file *pp_mat_1_study.inp* models the material with isotropic hardening, and the input file *pp_mat_2_study.inp* uses a perfectly plastic material model. These files are parameterized and are driven by the parametric study scripts *pp_mat_1_study.psf* and *pp_mat_2_study.psf*, respectively.

Analytical expressions based on simplifying assumptions for this type of problem are derived in Backman and Goldsmith. Two approaches, referred to as the energy method and the momentum method, give slightly different estimates for the final velocity of the projectile. With the energy method the resulting expression for the final velocity is

$$v_f = \left[v_i^2 - \frac{2W_d}{m}\right]^{\frac{1}{2}},$$

where m is the mass. The work done (W_d) is given by

$$W_d = \pi R^2 t \left[\rho\left(\frac{v_i R}{L}\right)^2 + \frac{\sigma}{2}\right],$$

where R is the radius of the projectile, L is the length of the conical nose, σ is the approximated yield strength of the material when modeled without hardening, t is the plate thickness, and ρ is the material density.

The momentum method gives the final velocity as

$$v_f = \frac{\pi \rho t R^2 v_i \sin\alpha}{m},$$

where α is the half cone angle.

Results and discussion

The results for the mesh convergence study, shown in *Table 1*, indicate that this problem is not highly sensitive to the mesh refinement in the radial or through-thickness direction for the meshes considered. The calculated decrease in projectile speed differs by about 2% between the mesh with the least (250) elements and the mesh with the most (810) elements. Analysis times for these cases, as reported in the status file, differ by a factor of approximately 8. The final configuration for the 250-element analysis is shown in *Figure 4*. Deformed meshes, with intact elements of the plate only, for the analyses with 250 elements and 810 elements are shown in *Figure*

5 and *Figure 6*, respectively. The predicted deformation of the plate is nearly identical for both meshes. The elements which have failed (not shown) correspond to roughly the inner 15% of the radius of the plate. Bending is not significant to the energy absorption of the plate under these high-speed impact conditions; thus, an even coarser mesh would tend to give a similar estimate of the projectile speed decrease but would give a less accurate prediction of the deformed shape of the intact elements. The 250-element mesh is used for the remainder of the studies.

Table 2 shows that the projectile speed decrease differs by only about 2% for the analyses with the penalty and kinematic contact formulations, respectively. This is not surprising, as the choice of the contact algorithm is not usually significant (exceptions are discussed in *Truss impact on a rigid wall* and *The Hertz contact problem*). The kinematic contact algorithm is used for the remaining studies.

The results from the material model study are shown in *Table 3*. The results obtained with the perfectly plastic material are quite close to the results obtained with the isotropic hardening material model for the same value of the failure strain; however, the failure strain does have a significant influence on the results. These results can be explained by consideration of the area under the stress-strain curve. The area under the stress-strain curve represents the ductility or energy absorbing potential of the material, and it is similar for both types of plasticity data, as can be seen in *Figure 3*. However, the choice of the failure strain can affect the energy absorbing capacity of the material significantly. The material model with hardening and a failure strain of 50% is used in the final study. In general, careful consideration should be given to the material model.

In the final study the initial velocity of the projectile is varied. *Figure 7* and *Figure 8* show deformed mesh plots of the plate (intact elements only) after projectile penetration with an initial velocity of 400 m/s and 1000 m/s, respectively. With a higher impact velocity there is less bending deformation of the surviving elements. This behavior is caused by the increased significance of inertial effects for higher impact speed. The decrease in projectile speed and the kinetic energy loss are shown in *Table 4* and *Table 5*, respectively. These tables compare Abaqus/Explicit results with experimental results and analytical expressions derived in Backman and Goldsmith based on simplifying assumptions. The experimental results are based on data presented by Backman and Goldsmith. The number of samples used for the experimental results is unknown. The numerical results for the decrease in projectile speed are within 8% of the energy method estimates, 40% of the momentum method estimates, and 30% of the experimental results.

Input files

pp_mesh_study.inp	Parameterized input file for the mesh study.
pp_mesh_study.psf	Python script to drive the mesh study.
pp_con_study.inp	Parameterized input file for the contact study.
pp_con_study.psf	Python script to drive the contact study.
pp_mat_1_study.inp	Parameterized input file to study effects of failure type and strain with an isotropic hardening material.
pp_mat_1_study.psf	Python script to drive the isotropic material study.
pp_mat_2_study.inp	Parameterized input file to study effects of failure type and strain with a Mises material.
pp_mat_2_study.psf	Python script to drive the Mises material study.
pp_velo_study.inp	Parameterized input file to analyze the penetration problem with projectile velocities of 400 m/s, 600 m/s, 800 m/s, and 1000 m/s.
pp_velo_study.psf	Python script to drive the velocity study.
pp_disc_rigid.inp	Input file to analyze the penetration problem using a discretized rigid projectile with a velocity of 250 m/s. A low velocity is chosen to allow sufficient penetration of the projectile so the balanced main-secondary approach with the main surface at r=0 can be verified.

References

- Backman, M. E., and W. Goldsmith, "The Mechanics of Penetration of Projectiles into Targets," *International Journal of Engineering Science*, vol. 16, pp. 1–91, 1978.

Tables

Table 1: Mesh study results.

Number of elements in radial direction	Number of elements through the thickness	Final velocity of missile (m/s)	Velocity drop (m/s)
50	5	597.79	2.21
70	5	597.76	2.24
90	5	597.74	2.26
50	7	597.76	2.24
70	7	597.77	2.23
90	7	597.78	2.22
50	9	597.76	2.24
70	9	597.79	2.21
90	9	597.74	2.26

Table 2: Contact algorithm study results.

Contact algorithm	Final velocity of missile (m/s)	Velocity drop (m/s)
Penalty	597.71	2.29
Kinematic	597.76	2.24

Table 3: Material model study results.

Material Model	Final velocity of missile (m/s)	Velocity drop (m/s)
Hardening with failure strain of 17% and allowing element deletion when the failure criterion is met	598.73	1.27
Hardening with failure strain of 50% and allowing element deletion when the failure criterion is met	597.79	2.21
Perfectly plastic with failure strain of 17% and allowing element deletion when the failure criterion is met	598.78	1.22
Perfectly plastic with failure strain of 50% and allowing element deletion when the failure criterion is met	597.93	2.07

Table 4: Velocity drop versus initial projectile speed.

Initial projectile velocity	Velocity drop (m/s)			
	Abaqus result	Energy method	Momentum method	Experiment
400	2.53	2.43	1.41	—
600	2.21	2.07	2.12	1.88

Initial projectile velocity	Velocity drop (m/s)			
	Abaqus result	Energy method	Momentum method	Experiment
800	2.12	2.04	2.83	2.4
1000	2.21	2.12	3.53	2.97

Table 5: Kinetic energy loss vs. initial projectile energy.

Initial projectile velocity	Kinetic energy loss (Nm)			
	Abaqus result	Energy method	Momentum method	Experiment
400	111.12	107.21	61.76	—
600	145.72	137.17	138.95	123.56
800	186.58	179.03	247.02	211.06
1000	242.99	233.06	385.97	321.72

Figures

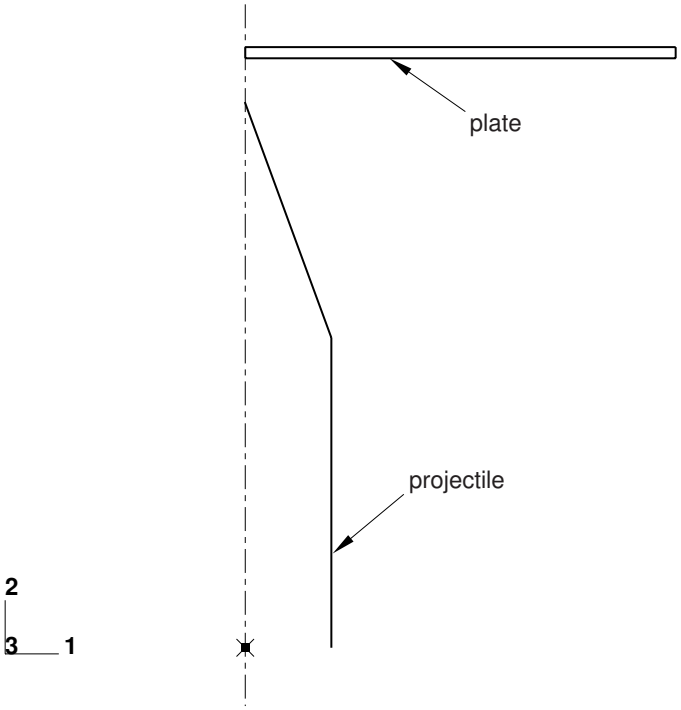


Figure 1: Plate penetration model outline.
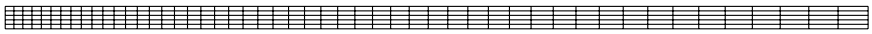


Figure 2: 5 × 50 element mesh for plate.

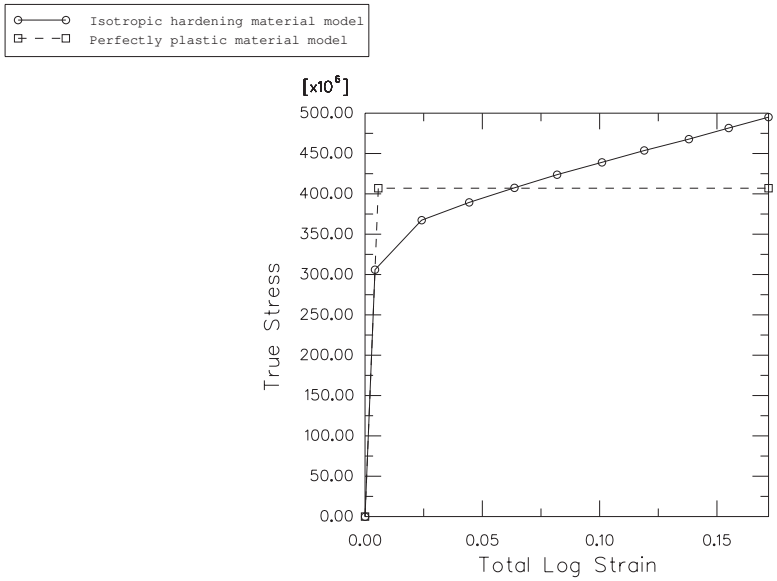


Figure 3: True stress vs. total log strain curve of material models.

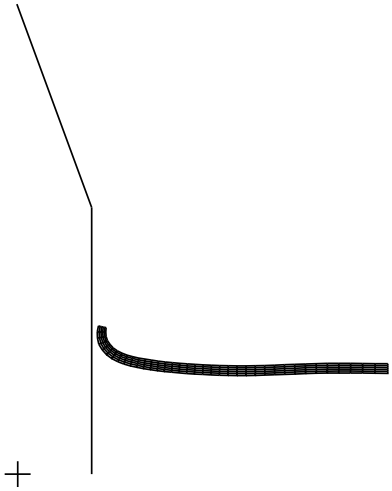


Figure 4: Final configuration (without failed elements) for analysis with 5 × 50 mesh and initial velocity of 600 m/s.

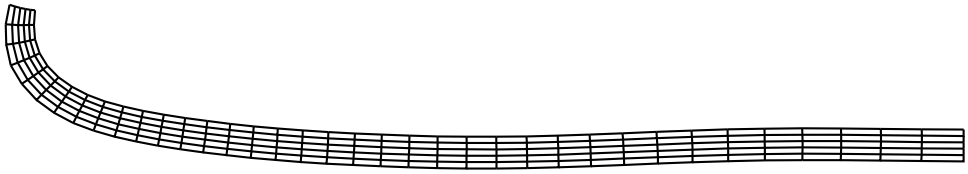


Figure 5: Deformed plot of intact elements for analysis with 5 × 50 element mesh and initial velocity of 600 m/s.

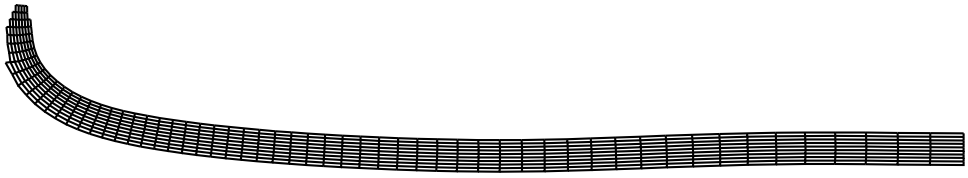


Figure 6: Deformed plot of intact elements for analysis with 9 × 90 element mesh and initial velocity of 600 m/s.

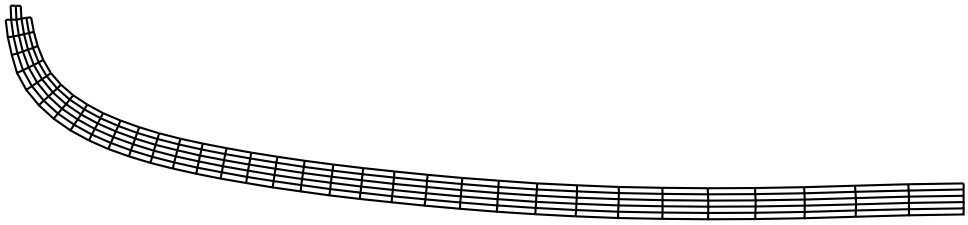


Figure 7: Deformed plot of intact elements for analysis with 5 × 50 element mesh and initial velocity of 400 m/s.



Figure 8: Deformed plot of intact elements for analysis with 5 × 50 element mesh and initial velocity of 1000 m/s.

Oblique shock reflections

Products: Abaqus/Explicit

Overview

This example illustrates the use of the ideal gas equation of state model and adaptive meshing in modeling shock wave interaction problems that involve both regular and Mach reflection processes.

Problem description

A plane shock wave in a gas with negligible viscosity and heat conductivity travels with constant velocity through a two-dimensional channel and encounters a wedge-shaped obstruction on the left wall (Amsden and Ruppel, 1981). A sequence of reflections occurs, depicted qualitatively in *Figure 1*(a) through *Figure 1*(e). The event is governed by the theory of regular shock reflection (Harlow and Amdsen, 1971). *Figure 1*(a) shows the incident shock wave (IS) moving through the channel toward the wedge. A shock wave (WS) is reflected from the wedge as shown in *Figure 1*(b). The flow Mach number and wedge angle are such that the shock remains attached at the wedge vertex. The wave configuration grows until the reflected shock strikes the right wall of the channel and is reflected back into the channel, as shown in *Figure 1*(c) (RS). Since the strength and angle of the incident shock wave (IS) are in the Mach reflection regime, a third shock called the Mach stem is formed (MS in *Figure 1*(d)). The intersection of the three shocks is called the triple point (T). This configuration cannot remain steady, and the Mach stem moves upstream against the incoming flow (*Figure 1*(e)) and eventually engulfs it, as shown in *Figure 1*(f).

The wedge half-angle is taken to be 15.13° in this example. A schematic of the model is shown in *Figure 2*; the model consists of two compartments separated by a diaphragm. Both compartments are filled with the same gas, at different initial states and velocities. The compartments are meshed with CPE4R elements. The left wall of the channel is modeled by a fixed analytical rigid surface, while the right wall is simulated by prescribing a symmetry boundary condition. The Abaqus/Explicit ideal gas equation of state model is used with a gas constant of 0.2 and a constant specific heat at constant volume of 0.5. These constants are not intended to represent any real gases. The gas in compartment A is initially at a unit density, a very small pressure, and zero velocity. Behind the incident shock in compartment B, a high energy gas with an initial density of 6 and an initial pressure stress of 1.2 flows toward compartment A at an initial velocity of 1.0. The diaphragm separating the compartments is removed instantaneously, causing a shock wave to propagate into compartment B.

Adaptive meshing

An elongated Eulerian adaptive mesh domain is used. The wedge-shaped obstruction is located in the middle portion of the domain where the shock refections take place. The Eulerian inflow and outflow boundaries are located far enough upstream and downstream from the obstruction to prevent undesired reflections. The mesh for the middle portion of the domain is held in place for the purpose of showing results by applying adaptive mesh constraints at the entry and exit planes of this subdomain. These constraints are in addition to spatial adaptive mesh constraints used at the Eulerian boundaries. Because the gas flow is substantial, the intensity of adaptive meshing must be increased to provide an accurate solution. The value of the MESH SWEEPS parameter is increased from the default of 1 to 5.

Results and discussion

The analysis is carried out over a time of 150. The vector plots of the velocity resultant in the middle portion of the domain are shown in *Figure 3*. From left to right the plots are at times t=0, 12, 17, 30, 90, and 120. The corresponding contour plots of the pressure stress are given in *Figure 4*. The maximum value of pressure stress

increased from the initial value of 1.2 to approximately 7.0 at the end of the analysis. The theory of regular reflection predicts that the half-angle of the wedge shock MS should be 48.5°. A measurement of pressure contour lines at intermediate times in *Figure 4* is in good agreement with this value.

Input files

ale_wedge_shock.inp	Analysis with adaptive meshing.

References

- Amsden, A. A., and H. M. Ruppel, "SALE-3D: A Simplified ALE Computer Program for Calculating Three-Dimensional Fluid Flow," *Los Alamos Scientific Laboratory*, 1981.
- Harlow, F. H., and A. A. Amsden, "Fluid Dynamics – A LASL Monograph," *Los Alamos Scientific Laboratory report LA-4700*, 1971.

Figures

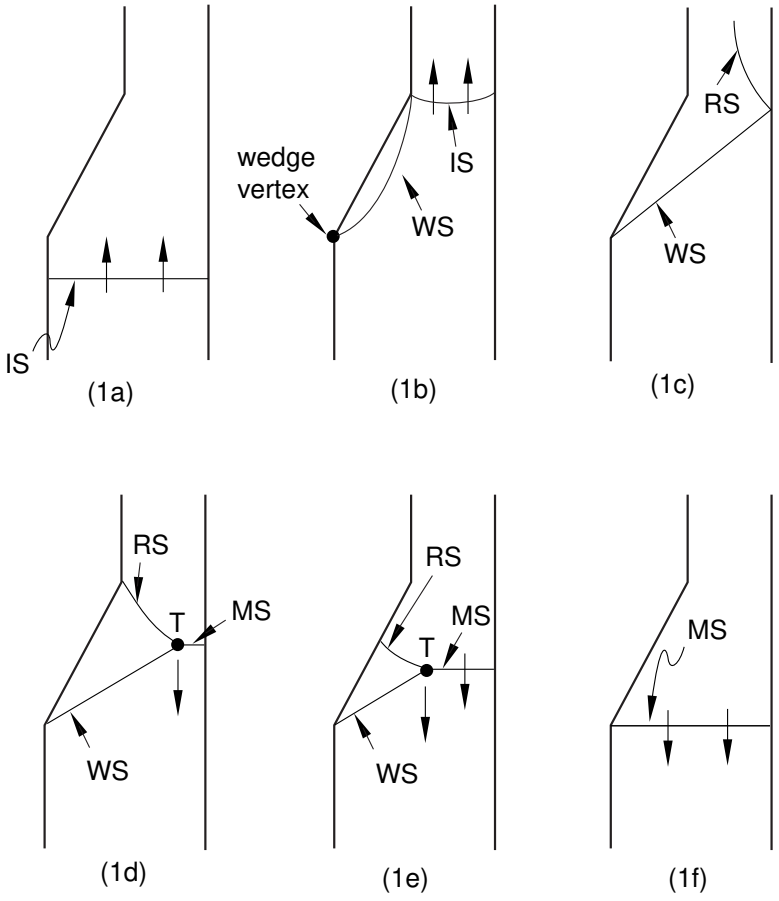


Figure 1: The sequence of shock reflections occurring when a plane shock wave encounters a wedge-shaped obstruction in a two-dimensional channel.

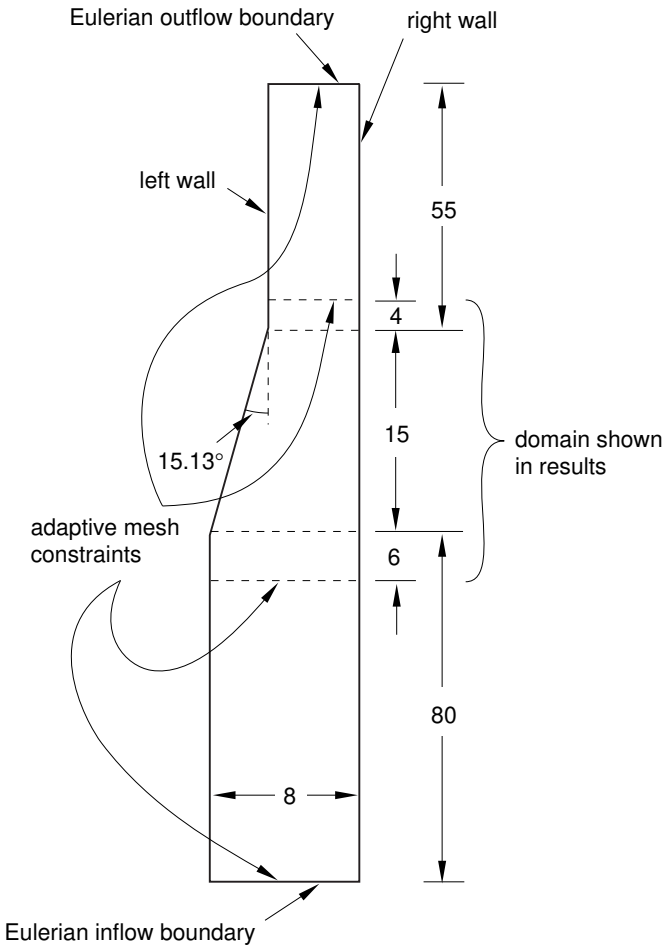


Figure 2: Schematic drawing of the model (CPE4R elements).

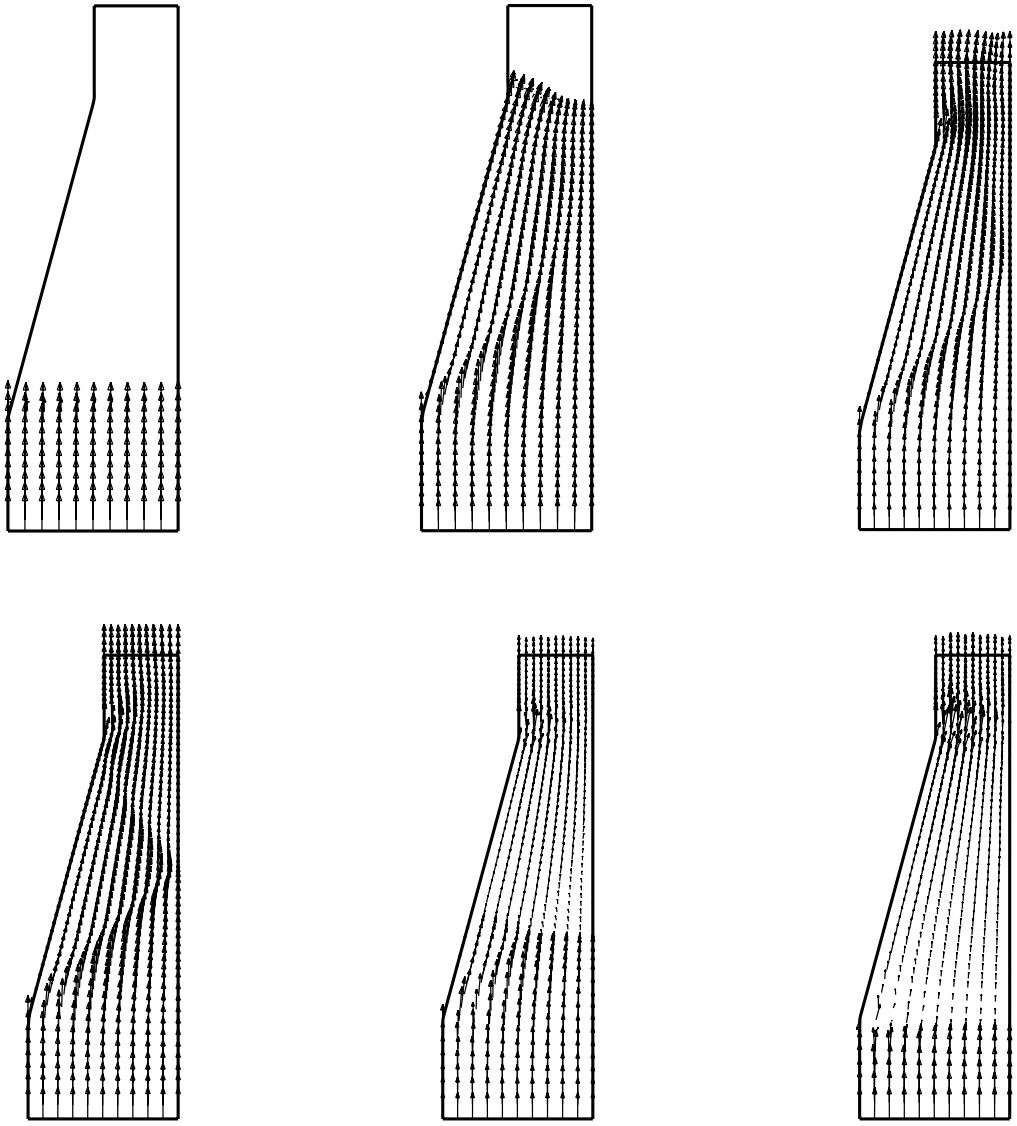


Figure 3: Vector plots of the velocity resultant in the middle portion of the domain for different intermediate times.

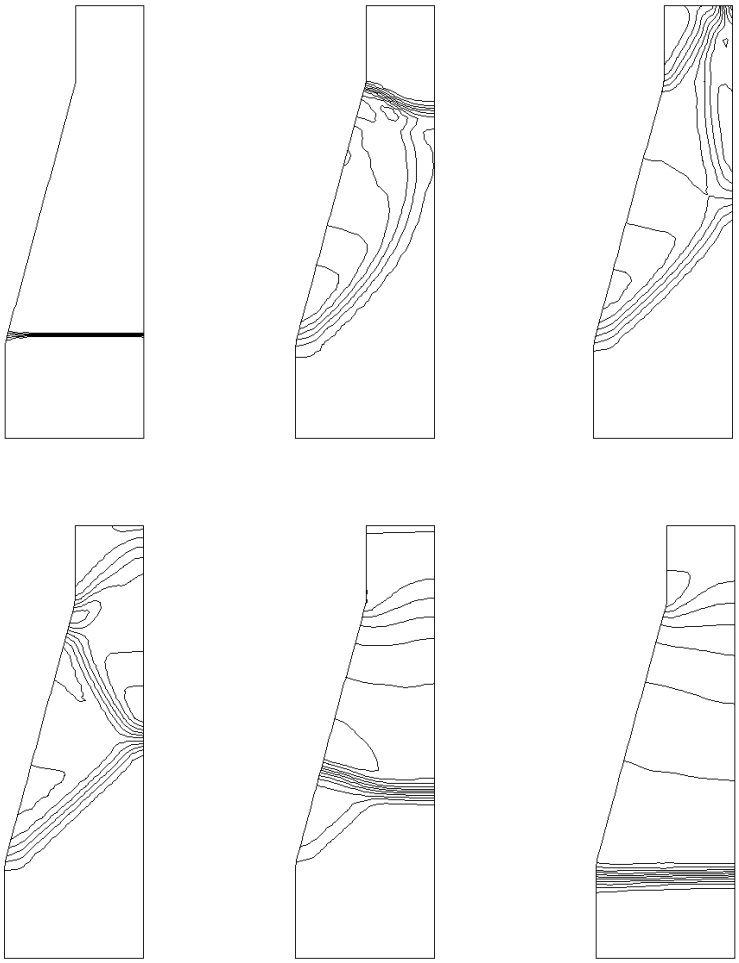


Figure 4: Pressure contours corresponding to the velocity resultant shown in the figure above.

Mode-based dynamic analysis

In this section:

- *Free vibrations of a spherical shell*
- *Eigenvalue analysis of a beam under various end constraints and loadings*
- *Vibration of a cable under tension*
- *Free and forced vibrations with damping*
- *Verification of Rayleigh damping options with direct integration and modal superposition*
- *Eigenvalue analysis of a cantilever plate*
- *Vibration of a rotating cantilever plate*
- *Response spectrum analysis of a simply supported beam*
- *Linear analysis of a rod under dynamic loading*
- *Random response to jet noise excitation*
- *Random response of a cantilever subjected to base motion*
- *Double cantilever subjected to multiple base motions*
- *Analysis of a cantilever subject to earthquake motion*
- *Residual modes for modal response analysis*

Free vibrations of a spherical shell

Products: Abaqus/Standard

Overview

This example uses modeling of the vibrations of a thin, elastic, spherical shell to verify the shell elements in Abaqus.

The first papers on the vibration of thin, elastic, spherical shells precede the general formulation of the classical bending theory of shells. The problem of free vibration of a "complete" spherical shell was first examined by Lamb (1882). More detailed treatments were given by Baker (1961) and Silbiger (1962).

Problem description

Thickness to radius ratios (h/R) of 1/100 and 1/20 are considered. Although the shell is "thin" in either case, the thicker shell illustrates the significance of bending effects.

All applicable shell elements in Abaqus are used. For the axisymmetric case using SAX1 or SAX2 elements and the asymmetric-axisymmetric case using SAXA11, SAXA12, SAXA13, SAXA14, SAXA21, SAXA22, SAXA23, or SAXA24 elements, a well-refined mesh is used, with 80 nodes located at equal intervals along the circumference.

The meshes for the complete spherical shell using general shell elements use an identical number of elements for both the first-order and second-order formulations. Mesh convergence has not been studied. For the triangular shell elements each quadrilateral has been split into two triangles, without any consideration of preserving mesh symmetry. The mesh used with the second-order elements is shown in *Figure 1*.

Analytical solution

Based on the membrane theory of shells, it is known that the natural frequency spectrum of a hollow, thin, elastic sphere consists of two infinite sets of modes and that one set of an infinite number of modes is spaced within a finite frequency interval. The mode shapes of the shell are expressed in terms of Legendre polynomials of degree n. For each value of n there are two distinct frequencies. The smaller of the two frequencies forms the "lower branch." The second or "upper branch" modes are primarily extensional. The first 10 frequencies are given in *Table 1*.

The $n = 0$ mode consists of purely radial vibration. Its frequency lies well above all of the frequencies associated with modes in the lower branch. It can be seen in the table that the frequencies of the upper branch increase without limit as n increases but that those of the lower branch approach the limit:

$$\lim_{n\to\infty} f = \frac{1}{2\pi}\sqrt{\frac{E}{\rho R^2}},$$

where f is the frequency of vibration, E is the modulus of elasticity, ρ is the mass density, and R is the radius of the sphere. Such a limiting situation is a result of the membrane theory employed (Kalnins, 1964). Membrane theory is accurate only for very thin shells and for low mode numbers. The Abaqus shell elements account for membrane and bending effects, so we should expect good agreement only in membrane-type modes.

If only axisymmetric modes are considered, there is a distinct mode shape for each value of frequency. However, a model based on general shell elements allows for nonaxisymmetric modes. Interestingly, for the spherical shell the frequencies corresponding to nonaxisymmetric modes are identical to the frequencies of the axisymmetric modes. This is a consequence of the spherical symmetry of the shell. Corresponding to each value of n there are

$2n$+1 linearly independent modes. To verify this, we have chosen to model the entire sphere, although the problem can be analyzed more economically by modeling a partial sphere using symmetry and antisymmetry boundary conditions. In addition, because of the multiple modes of identical frequency, this problem serves as a good test for the eigenvalue-eigenvector algorithms.

Results and discussion

Table 2 summarizes the results obtained using the axisymmetric shell elements SAX1 and SAX2 for the first 10 modes. For the lower-order modes and the thinner shell case, the results agree well with membrane theory.

The natural frequency of the ninth mode for $h/R =$ 0.05 is significantly different from that predicted by membrane theory and is in agreement with Kalnins (1964). Membrane theory is clearly accurate for small values of h/R and for the lower-order modes. The mode $n = 1$ corresponds to rigid body translation and is not shown in the table. In the axisymmetric case each frequency has a distinct mode shape and the eigenvalue iterations converge rapidly.

Table 3 and *Table 4* summarize the results obtained using the asymmetric-axisymmetric shell elements SAXA1*N* and SAXA2*N* (*N*=1, 2, 3 or 4). In this case for each value of *n* there are *n*+1 modes instead of $2n$+1, as predicted analytically. This is because, in the asymmetric-axisymmetric element formulation, symmetry with respect to the *r*–*z* plane at $\theta = 0$ is assumed. However, for each *n* the number of modes computed is limited by *N*+1, where *N* is the number of Fourier interpolation terms used.

Recall that, in the full models using general shell elements, there are $2n$+1 modes for each value of *n*. To improve convergence in the eigenvalue iteration, we have, therefore, specified a higher number of trial vectors to be used. We calculate 18 eigenvalues to get the modes up to $n = 3$. For higher-order modes such as $n = 9$, at least 100 or more eigenvalues need to be calculated. To keep this qualification test within a reasonable computational time, we have restricted the number of eigenvalues to 20. It implies that the bending effects will not be visible to the same extent as in the axisymmetric case. For this reason results from the general shell models are reported here only for the thin shell case with $h/R =$ 0.01.

Table 5 provides the results for second-order shell elements; *Table 6* provides the results for first-order shell elements. In these tables we list the first 20 eigenvalues, except the first six rigid body modes.

When second-order shell elements are used, the first five values (7 through 11) are almost identical to the membrane solution for the $n = 2$ case. The first-order mesh uses the same number of elements as the second-order mesh. Nevertheless, except for S3R elements, the results are quite accurate: the error is less than 2% for the first five eigenvalues. For S3R elements the maximum error is around 5% because these elements use a constant bending strain approximation. The accuracy can be increased by further refining the mesh. Eigenvalues 12 through 18 correspond to the mode $n = 3$. It is observed that $2n$+1 modes are recovered, as predicted by the analytical solutions.

We also notice that the first-order triangular elements show more variance in eigenvalues corresponding to a given value of *n* than the quadrilaterals. This is a consequence of orientation effects of the triangular element. The accuracy could be improved by designing the mesh to be spherically symmetric.

Figure 2 illustrates the modes $n = 2$ and $n = 3$ obtained with any of the shell models used.

Input files

freevibsphere_s3r.inp	S3R element model.
freevibsphere_s4.inp	S4 element model.
freevibsphere_s4_thick.inp	S4 element model ($h/R =$ 0.05).
freevibsphere_s4r.inp	S4R element model.

freevibsphere_s4r_thick.inp	S4R element model ($h/R = 0.05$).
freevibsphere_s4r5.inp	S4R5 element model.
freevibsphere_s8r.inp	S8R element model.
freevibsphere_s8r_thick.inp	S8R element model ($h/R = 0.05$).
freevibsphere_s8r5.inp	S8R5 element model.
freevibsphere_s9r5.inp	S9R5 element model.
freevibsphere_stri3.inp	STRI3 element model.
freevibsphere_stri65.inp	STRI65 element model.
freevibsphere_sax1.inp	SAX1 element model.
freevibsphere_sax1_thick.inp	SAX1 element model ($h/R = 0.05$).
freevibsphere_sax2.inp	SAX2 element model.
freevibsphere_sax2_thick.inp	SAX2 element model ($h/R = 0.05$).
freevibsphere_saxa11_thin.inp	SAXA11 element model ($h/R = 0.01$).
freevibsphere_saxa12_thin.inp	SAXA12 element model ($h/R = 0.01$).
freevibsphere_saxa13_thin.inp	SAXA13 element model ($h/R = 0.01$).
freevibsphere_sax14_thin.inp	SAXA14 element model ($h/R = 0.01$).
freevibsphere_saxa21_thin.inp	SAXA21 element model ($h/R = 0.01$).
freevibsphere_saxa22_thin.inp	SAXA22 element model ($h/R = 0.01$).
freevibsphere_saxa23_thin.inp	SAXA23 element model ($h/R = 0.01$).
freevibsphere_saxa24_thin.inp	SAXA24 element model ($h/R = 0.01$).

References

- Baker, W. E., "Axisymmetric Modes of Vibration of Thin Spherical Shells," *Journal of Acoustic Society of America*, vol. 33, pp. 1749–1758, 1961.
- Kalnins, A., "Effect of Bending on Vibration of Spherical Shells," *Journal of Acoustic Society of America*, vol. 36, pp. 74–81, 1964.
- Lamb, H., "On the Vibrations of a Spherical Shell," *Procedures of the London Mathematical Society*, vol. 14, pp. 50–56, 1882.
- Silbiger, A., "Nonaxisymmetric Modes of Vibration of Thin Spherical Shells," *Journal of Acoustic Society of America*, vol. 34 862, 1962.

Tables

Table 1: Natural frequencies in cycles/sec based on membrane theory. ($E = 180.0 \times 10^9$, $\nu = 1/3$, $\rho = 7670.0$.)

Mode	Lower spectrum	Higher spectrum
0	–	445.0
1	0.0	545.18

Mode	Lower spectrum	Higher spectrum
2	187.34	748.02
3	222.57	995.37
4	236.56	1256.58
5	239.56	1522.62
6	247.37	1791.24
7	249.80	2060.92
8	251.41	2331.42
9	252.54	2602.36
10	253.35	2873.62

Table 2: Natural frequencies with axisymmetric shell elements.

Mode(n)	Membrane theory	h/R=0.01		h/R=0.05	
		SAX1	SAX2	SAX1	SAX2
2	187.34	187.26	187.36	187.72	187.82
3	222.57	222.30	222.69	225.19	225.57
4	236.56	236.15	236.95	245.35	246.09
5	239.56	243.12	244.41	264.61	265.76
6	247.37	247.43	249.30	289.13	290.66
7	249.80	250.76	253.29	321.84	323.68
8	251.41	253.99	257.25	364.00	366.02
9	252.54	257.66	261.69	415.81	417.88
10	253.35	262.18	267.00	445.14	445.14

Table 3: Natural frequencies with first-order asymmetric-axisymmetric shell elements.

Eigenvalue number	SAXA11	SAXA12	SAXA13	SAXA14
4	187.26	187.26	187.26	187.26
5	187.35	187.35	187.35	187.35
6	222.30	187.41	187.41	187.41
7	222.53	222.30	222.30	222.30
8	236.15	222.53	222.53	222.53
9	236.51	222.73	222.73	222.73
10	243.12	236.15	222.76	222.76
11	243.59	236.51	236.15	236.15
12	247.43	236.84	236.51	236.51
13	248.01	243.12	236.83	236.83
14	250.76	243.59	237.03	237.03
15	251.45	244.03	243.12	237.04

Table 4: Natural frequencies with second-order asymmetric-axisymmetric shell elements.

Eigenvalue number	SAXA21	SAXA22	SAXA23	SAXA24
4	187.36	187.36	187.36	187.36
5	187.36	187.36	187.36	187.36
6	222.69	187.36	187.36	187.36
7	222.69	222.69	222.69	222.69
8	236.94	222.69	222.69	222.69
9	236.95	222.69	222.69	222.69
10	244.41	236.94	222.69	222.69
11	244.41	236.95	236.95	236.95
12	249.29	236.95	236.95	236.95
13	249.30	244.41	236.95	236.95
14	253.29	244.41	236.95	236.95
15	253.30	244.41	244.41	236.95

Table 5: Natural frequencies with second-order general shell elements S8R, S8R5, S9R5, and STRI65.

Eigenvalue number	S8R	S8R5	S9R5	STRI65
7	187.37	187.36	187.36	187.38
8	187.37	187.36	187.36	187.38
9	187.38	187.36	187.36	187.38
10	187.38	187.37	187.37	187.38
11	187.38	187.37	187.37	187.38
12	222.66	222.63	222.63	222.74
13	222.66	222.63	222.63	222.75
14	222.66	222.63	222.63	222.75
15	222.74	222.70	222.70	222.76
16	222.74	222.70	222.70	222.81
17	222.74	222.70	222.70	222.81
18	222.81	222.77	222.77	222.84
19	236.81	236.66	236.68	237.14
20	236.93	236.80	236.80	237.24

Table 6: Natural frequencies with first-order general shell elements S4R, S4R5, S4, STRI3, and S3R.

Eigenvalue number	S4R	S4R5	S4	STRI3	S3R
7	189.97	189.97	189.86	187.32	190.19
8	189.97	189.97	189.86	188.76	190.66
9	190.05	190.05	190.04	188.76	190.66
10	190.05	190.05	190.06	189.97	192.25

Eigenvalue number	S4R	S4R5	S4	STRI3	S3R
11	190.05	190.05	190.06	189.97	192.25
12	223.71	223.70	225.66	223.85	229.55
13	223.71	223.70	225.74	224.16	230.82
14	223.71	223.70	225.74	224.16	230.82
15	227.90	227.89	228.59	227.51	233.47
16	227.90	227.89	228.59	228.71	234.32
17	227.90	227.89	228.61	228.71	234.82
18	231.43	231.37	233.57	229.06	234.82
19	233.48	233.45	237.24	239.45	252.14
20	233.59	233.45	242.00	239.50	252.14

Figures

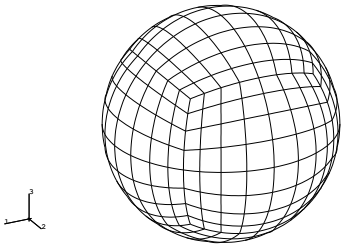
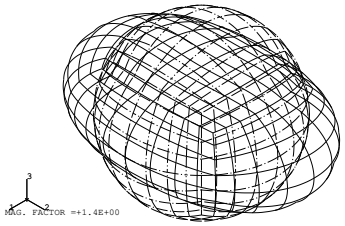


Figure 1: Spherical shell model, with second-order quadrilaterals.



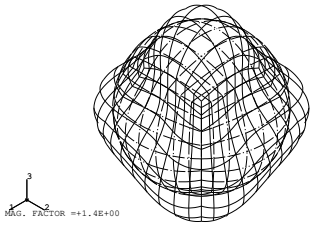


Figure 2: Modes $n =$ 2,3 of spherical shell.

Eigenvalue analysis of a beam under various end constraints and loadings

Products: Abaqus/Standard

References:

- *Static Stress Analysis*
- *Linear Constraint Equations*
- *Vibration of a cable under tension*

Overview

This example shows how to use the eigenvalue capability in Abaqus with a variety of options.

This example uses two simple beam structures: a cantilever with various supports at the tip, and a beam with both ends simply supported. In some cases the beam is preloaded in an initial static step, and the eigenvalues of the preloaded structure are then obtained. The preloaded structure analysis requires that geometric nonlinearities be included in the step so that Abaqus will form the initial stress matrix.

For the cantilever a variety of end conditions are used: a free end, a simple support, and a stiff, vertical spring support. In addition, cases are run with open and closed gap conditions at the end. In one case the beam is made up of separate segments, connected with an equation constraint.

Problem description

The beam has a length of 127 mm (5 in), and a solid circular cross-section with a radius of 2.54 mm (0.1 in). Young's modulus is 187 GPa (27×10^6 lb/in^2), and the density is 8015.19 kg/m^3 (7.5×10^4 lb-s^2/in^4). The finite element model consists of 10 equal-sized cubic interpolation beam elements of type B23.

Boundary conditions and loadings

The following cases are analyzed:

1. Beam with both ends simply supported (see *Figure 1*):
   a. Unstressed structure.
   b. Structure prestressed by an axial force. The pretension force is 4448 N (1000 lb).
2. Cantilever beam (see *Figure 2* to *Figure 5*):
   a. Simple cantilever.
   b. Pretensioned cantilever. The pretension force is 44482 N (10000 lb).
   c. Gap condition at the free end—gap open. This case is the same as B1 above.
   d. Gap condition at the free end—gap closed. This case is the same as B5 below.
   e. Cantilever with a simple support at the end.
   f. Cantilever with a spring support at the end. A stiff spring (stiffness 1.75127×10^3 MN/mm (10^7 lb/in)) is used, so that this case also corresponds to B5 above.
   g. Cantilever beam with a simple support at the end. This case is the same as B5, but now the beam is defined geometrically as several separate segments, joined together kinematically by an equation constraint.

Results and discussion

The results are given in *Table 1* for the three lowest modes for all cases. In most cases they are compared to exact solutions, taken from Timoshenko (1937). As would be expected, with a 10 element mesh with cubic interpolation, the lowest three modes agree closely with the exact solutions. The pretensioned cases show the expected increase in frequencies over the same cases without pretensioning.

Input files

eigenbeam_simple.inp	Basic simply supported case.
eigenbeam_pretension_simple.inp	Pretensioned, simply supported case.
eigenbeam_cant.inp	Basic cantilever case.
eigenbeam_pretension_cant.inp	Pretensioned cantilever case.
eigenbeam_closedgap.inp	Cantilever with a closed gap.
eigenbeam_cant_opengap.inp	Cantilever with an open gap at the end.
eigenbeam_roller.inp	Cantilever with a roller support.
eigenbeam_cant_springsup.inp	Cantilever with a spring support at the free end.
eigenbeam_cant_equation.inp	Cantilever made up of two segments joined with the **EQUATION* option.

References

- Timoshenko, S., Vibration Problems in Engineering, D. Van Nostrand Company, Inc., New York, 2nd, 1937.

Tables

Table 1: Three lowest vibration frequencies of a beam.

Case		Frequencies (Hz)		
		Mode 1	Mode 2	Mode 3
A1.	Abaqus	596.1	2384.6	5367.6
	Timoshenko	596.1	2384.3	5364.7
A2.	Abaqus	882.7	2716.9	5711.9
	Timoshenko	883.0	2717.1	5709.6
B1.	Abaqus	212.4	1330.8	3727.2
	Timoshenko	212.3	1330.7	3726.4
B2.	Abaqus	1137.9	3624.4	6694.1
B3.	Abaqus	212.4	1330.8	3727.2
	(same as case B1)			
B4.	Abaqus	931.2	3018.2	6300.7
	(same as case B5)			
B5.	Abaqus	931.2	3018.2	6300.7
	Timoshenko	931.4	3018.0	6295.8

Case		Frequencies (Hz)		
		Mode 1	Mode 2	Mode 3
B6.	Abaqus	931.2	3017.9	6299.6
	(same as case B5)			
B7.	Abaqus	931.2	3018.2	6300.7
	(same as case B5)			

Figures

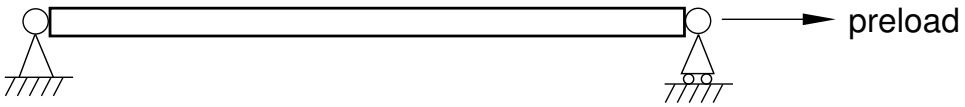


Figure 1: Beam with simply supported ends. For Case A1 the preload is zero; for Case A2 the preload is 4448 N.

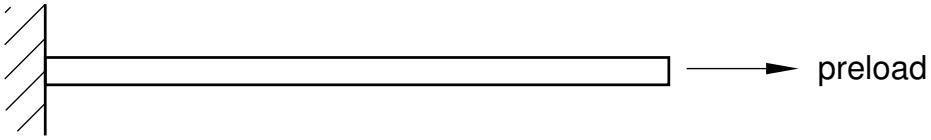


Figure 2: Cantilever beam. For Case B1 the preload is zero; for Case B2 the preload is 44482 N.

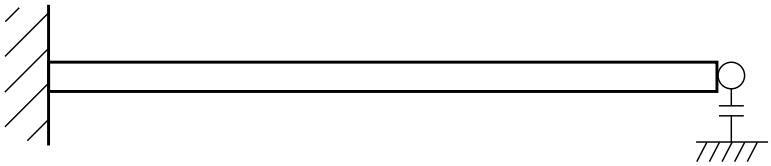


Figure 3: Cantilever beam with gap condition. For Case B3 the gap is open; for Case B4 the gap is closed.

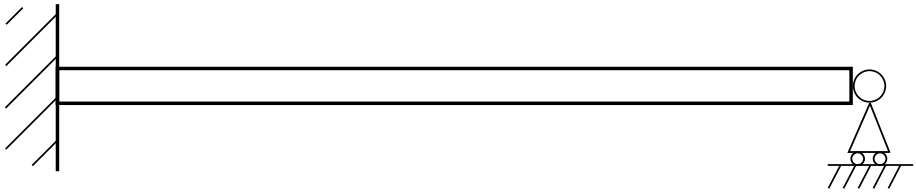


Figure 4: Cantilever beam with simply supported end. For Case B5 the beam is a single set of elements; for Case B7 the beam is defined as several separate segments joined with an equation constraint.

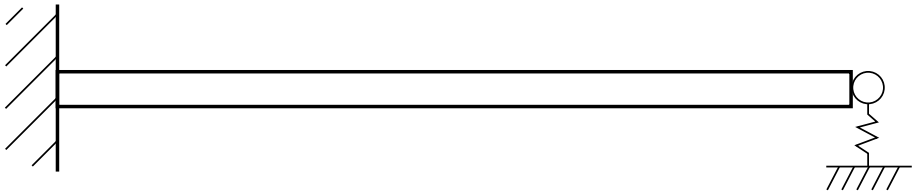


Figure 5: Cantilever beam with stiff spring support, Case B6.

Vibration of a cable under tension

Products: Abaqus/Standard

Overview

This example verifies that the frequencies of small vibrations about a prestressed, predeformed configuration of a cable under tension compare well with the analytical solution.

This “violin string” problem is a simple case in which a structure's frequencies depend on the state of prestress existing when the vibrations occur. In such cases the analysis is done by preloading the structure in one (or several) static steps and then requesting eigenvalue extraction. In some cases the static preload may involve considerable nonlinearity, although this is not the case in this simple problem.

Problem description

The truss model is shown in *Figure 1*. The cable is modeled using 13 truss elements of type T2D2 (two-dimensional, 2-node, linear interpolation). A tensile force of 2224 N (500 lb) is first applied to the cable in a static step. In the first increment of this step the model has one singular degree of freedom at each node, because the unstressed cable has no stiffness associated with transverse displacement. As soon as the cable has some tension, it offers stiffness to transverse motion through the initial stress terms. Thus, the user must take care to constrain these singular degrees of freedom initially, and remove the constraints once the tensile stress has been created. Alternatively, very weak springs could be used. In fact, this example is set up so that these temporary constraints are not needed because the cable is aligned exactly parallel to one of the global axis directions. Thus, the stiffness will be initially identically zero in the other global axis direction. Abaqus will recognize this and automatically eliminate those degrees of freedom in the initial increment. In both steps of this analysis, the preload and the eigenvalue extraction, the initial stress effect is obtained by considering geometric nonlinearities.

The first four eigenvalues are requested. The data also specify that only frequencies up to 1000 Hz should be extracted. The eigenvalue extraction will, therefore, terminate when four frequencies have been calculated or when convergence has been achieved for one mode whose frequency is above 1000 Hz, whichever condition occurs first.

vibrationcable_b21.inp is a model using 13 B21 elements. The loading and the boundary conditions for this problem are the same as the truss model. Four eigenvalues are requested.

Results and discussion

Four distinct frequencies are obtained. The frequencies are given in *Table 1*, where they are compared to the exact solution, taken from Thomson (1965). As might be expected, the lowest frequency is predicted very accurately, with the error growing for the higher modes. A finer mesh would provide more accuracy in the higher modes. The beam model results are very close to the truss model results.

Input files

vibrationcable_t2d2.inp	T2D2 elements.
vibrationcable_b21.inp	B21 elements.
vibrationcable_elmatrix.inp	Element matrices output in the beam example.

References

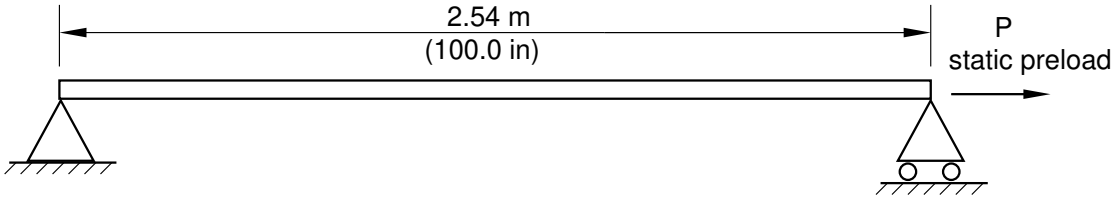
- Thomson, W. T., Vibration Theory and Applications, Prentice-Hall, New Jersey, 1965.

Tables

Table 1: Natural frequencies for preloaded cable, Hz.

Mode	Exact	Abaqus	Error	Abaqus	Error
	(Thomson, 1965)	T2D2	T2D2	B21	B21
1	74.7	74.3	–0.5%	74.3	–0.5%
2	149.	148.	–1.2%	148.	–1.2%
3	224.	219.	–2.4%	219.	–2.4%
4	299.	287.	–4.1%	287.	–4.1%

Figures



Cross-section area: 1.979 mm^2 (3.0677 x 10^{-3} in^2)
Young's modulus: 206.84 GPa (30.0 x 10^6 lb/in^2)
Density: 7801.0 kg/m^3 (7.3 x 10^{-4} $lb\text{-}s^2/in^4$)
Static preload: 2224.0 N (500.0 lb)

Figure 1: Preloaded cable vibration example.

Free and forced vibrations with damping

Products: Abaqus/Standard

Overview

This example verifies the frequency-dependent spring and dashpot elements available in Abaqus.

There are several different mechanisms that can cause damping in a system. In linear viscous damping the damping force is directly proportional to the velocity. In many cases such simple expressions for the damping forces are not available directly. However, it is possible to obtain an equivalent viscous damping coefficient by equating the loss of kinetic and strain energy to the energy dissipation. Hysteretic and viscoelastic damping are two important damping mechanisms that are more complex than linear viscous damping. In the frequency domain these mechanisms can be simulated by using dashpots with viscous damping coefficients that depend on the forcing frequency. Frequency-dependent springs will also be needed for modeling viscoelastic damping.

To illustrate how to model viscous, hysteretic, and viscoelastic damping mechanisms, springs and dashpots with constant and frequency-dependent properties will be used in frequency domain dynamic analyses of one- and two-degree-of-freedom discrete mass-spring-dashpot systems. In addition, viscous damping is modeled in the time domain by using a constant dashpot coefficient.

Abaqus also allows for spring and dashpot properties that depend on temperature and user-defined field variables. This dependence provides an easy means to vary material properties of springs and dashpots during time-domain analysis. In doing perturbation analysis (such as frequency-domain steady-state dynamic analysis) with Abaqus, temperature and field variable variations are not permitted within an analysis step. However, since the base state temperature and field variable values for each perturbation analysis step can be changed, it is possible to perform a multiple-step perturbation analysis that uses different temperature- and field-variable-dependent material properties that correspond to the base state temperature and field variable values. This dependence feature will be illustrated in analyses 2 and 3 described below. These two analyses employ both the direct-solution and the subspace-based steady-state dynamic procedure in Abaqus.

The one- and two-degree-of-freedom mass-spring-dashpot systems are shown in *Figure 1*. The following dynamic analyses are performed: (1) free vibration of the one-degree-of-freedom system after it is given an initial displacement and then released; (2) steady-state response to applied harmonic loading of the one-degree-of-freedom model with viscous damping; (3) steady-state response to applied harmonic loading of the one-degree-of-freedom model with hysteretic damping; and (4) steady-state response to applied harmonic loading of the two-degree-of-freedom model with viscoelastic damping. In all cases the forcing function is applied to the point mass closest to the anchor point, and numerical results are compared to the exact solutions for the system.

Problem description

The basic constant parameters of the analysis models are as follows:

Spring constant, k	5253.8 N/m (30 lb/in)
Damping coefficient, c	21.02 N/m-s (0.12 lb/in-s)
Mass, m	4.536 kg (0.02588 lb-s^2/in)

SPRING1 and DASHPOT1 elements are used in analyses 1–3. SPRING2 and DASHPOT2 elements are used in analysis 4.

In analysis 1 the model is the one-degree-of-freedom system shown in *Figure 1*. The initial displacement is 25.4 mm (1 in), so the force in the spring is initially 133.4 N (30 lb). The problem is run in two steps: a static step, wherein the initial displacement is imposed, and a dynamic step, during which the structure is allowed to oscillate.

The dynamic step is run with automatic time stepping, using two different values for the half-increment tolerance: 44.48 N (10 lb) and 4.448 N (1 lb). The higher value of the tolerance should give moderately accurate results, while the lower value should result in a more accurate solution. *Implicit Dynamic Analysis Using Direct Integration* gives guidelines for choosing a value for the half-increment tolerance for realistic, multiple-degree-of-freedom systems.

In analysis 2 a harmonic loading of the form $P = \overline{P}\cos(\Omega t)$ is applied to the single-degree-of-freedom system, where Ω is the circular frequency. The equation of motion for this system is

$$m\ddot{u} + c\dot{u} + ku = P.$$

The direct-solution and the subspace-based steady-state dynamic procedures are used to calculate the steady-state vibrations in this system with low and high viscous damping coefficients, $c =$ 0.12 and 0.24. The dashpot coefficient in this model is defined as a function of the first field variable, and the change of the field variable value is carried out in a dummy general static step placed between two direct-solution steady-state dynamic steps.

Analysis 3 is identical to analysis 2 in all aspects except that hysteretic damping is modeled instead of linear viscous damping. Hysteretic damping, also known as structural or solid damping, is observed in the vibration of many solid materials and can be attributed to internal friction. This form of damping produces a hysteresis loop in the force-displacement plot for each loading cycle that is proportional to the amplitude and tends to stay constant with rising forcing frequency. The energy loss is proportional to the displacement amplitude squared for both viscous damping and for hysteretic damping. This fact suggests that structurally damped systems subjected to harmonic excitation can be modeled as viscously damped systems with an equivalent coefficient of viscous damping that is inversely proportional to the frequency: see Denhartog (1985). The equation of motion for this one-degree-of-freedom system is, thus, written readily as

$$m\ddot{u} + \frac{\gamma k}{\Omega}\dot{u} + ku = P,$$

where γ is a damping coefficient and Ω is the forcing frequency. The equivalent viscous damping coefficient is $c_{eq} = \frac{\gamma k}{\Omega}$. For harmonic motion we have the relationship $\dot{u} = i\Omega u$; therefore, $\frac{\gamma k}{\Omega}\dot{u} = i\gamma k u$, where $i = \sqrt{-1}$ is the imaginary number. Hence, the equation of motion can also be written as

$$m\ddot{u} + k(1 + i\gamma)u = P.$$

Abaqus also allows direct specification of structural damping; however, this direct specification can be used only in modal-based analysis and is accurate only for small damping values. See *Material Damping* for further discussion. In this analysis the effects of low damping (γ=0.125) and high damping (γ=0.25) are compared, following the same procedure as used in analysis 2. The data set containing the frequency-dependent dashpot coefficients at intervals of 0.05 Hz over the frequency range of 0 to 10 Hz is included (file *vibration_dampdata1.inp*).

Analysis 4 involves a two-degree-of-freedom system with viscoelastic damping. Viscoelastic materials are often used in a structure to improve the damping characteristics of the structure or its components. In a one-dimensional test specimen made of linear viscoelastic material, an applied cyclic stress $\sigma^* = \sigma_0 \exp(i\Omega t)$ will result in a steady-state cyclic strain response, $\epsilon^* = \epsilon_0 \exp(i(\Omega t - \delta))$ with the same frequency but out of phase by the phase angle δ. The phase angle is also known as the loss angle and is a function of frequency. The damping ability of the material is dependent on it and not on the stress and strain amplitude. The ratio of the stress and strain defines the complex modulus, $E^*(\Omega) = \sigma^*/\epsilon^*$, where the real part is termed the storage modulus and the imaginary part the loss modulus. The equation of motion for the steady-state forced vibration of a single-degree-of-freedom viscoelastic system of mass *m* is simply

$$m\ddot{u} + K^*(\Omega)u = P,$$

where $K^*(\Omega)$ is the complex stiffness proportional to the complex modulus $E^*(\Omega)$. Making use of the substitution $\dot{u} = i\Omega u$ for harmonic motion, we can rewrite the equation of motion as

$$m\ddot{u} + c(\Omega)\dot{u} + k(\Omega)u = P,$$

where $k(\Omega) = \Re(K^*(\Omega))$ and $c(\Omega) = \frac{1}{\Omega}\Im(K^*(\Omega))$. Referring to *Frequency Domain Viscoelasticity*, we can identify that the equivalent viscous damping coefficient is $c(\Omega) = \Re(\kappa^*)k_\infty$ and the spring stiffness is $k(\Omega) = (1 - \Omega\Im(\kappa^*))k_\infty$, where $\kappa^* = \kappa^*(\Omega)$ is the Fourier transform of the nondimensional relaxation function $\kappa(t)$ and k_∞ is the long-term spring stiffness. The equation of motion for viscoelastic damping resembles the one for hysteretic damping to the extent that viscoelastic damping can also be simulated in discrete mass-spring-dashpot systems using frequency-dependent springs and dashpots. This form of damping is simulated in the two-degree-of-freedom discrete mass-spring-dashpot system shown in *Figure 1* with the following parameters: $m_1 = m_2 = m$ and $K_1{}^* = K_2^*$, such that the real and imaginary moduli are $k_1(\Omega) = k_2(\Omega) = k(\Omega) = (1 - \Omega\Im(\kappa^*))k_\infty$, and $c_1(\Omega) = c_2(\Omega) = c(\Omega) = \Re(\kappa^*)k_\infty$. The frequency dependence of $\kappa^*(\Omega)$ assumes the power law formula $\kappa^*(\Omega) = \kappa_1^* f^{-b}$, where b is a real constant, κ_1^* is a complex constant, and $f = \Omega/2\pi$ is the frequency in cycles/time. The equation of motion for the two-degree-of-freedom system is now readily developed and can be written as

$$\begin{bmatrix} m & 0 \\ 0 & m \end{bmatrix} \begin{Bmatrix} \ddot{u}_1 \\ \ddot{u}_2 \end{Bmatrix} + \begin{bmatrix} K_1^* + K_2^* & -K_2^* \\ -K_2^* & K_2^* \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \end{Bmatrix} = \begin{Bmatrix} P_1 \\ P_2 \end{Bmatrix}.$$

Since harmonic loading of the form $P_j = \overline{P}_j \exp(i\Omega t)$ produces the harmonic oscillation $u_j = \overline{u_j^*} \exp(i\Omega t)$ with the complex oscillation amplitude $\overline{u_j^*} = \overline{u}_{jR} + i\overline{u}_{jI}$, the equation of motion for the loading parameters used in this analysis, $P_1 = \overline{P}\cos(i\Omega t)$ and $P_2 = 0$, can be rewritten in terms of the real and imaginary parts of the oscillation amplitudes as follows:

$$\begin{bmatrix} 2k - m\Omega^2 & -k & 2\Omega c & -\Omega c \\ -k & k - m\Omega^2 & -\Omega c & \Omega c \\ 2\Omega c & -\Omega c & -2k + m\Omega^2 & k \\ -\Omega c & \Omega c & k & -k + m\Omega^2 \end{bmatrix} \begin{Bmatrix} \overline{u}_{1R} \\ \overline{u}_{2R} \\ -\overline{u}_{1I} \\ -\overline{u}_{2I} \end{Bmatrix} = \begin{Bmatrix} \overline{P} \\ 0 \\ 0 \\ 0 \end{Bmatrix}.$$

The frequency-dependent spring and dashpot properties are generated by a Fortran program using the basic model constants for the mass, m, and for the spring, k_∞. In addition, the parameters b=1.38366, $\Re(\kappa_1^*)$=2.3508 $\times 10^{-2}$, and $\Im(\kappa_1^*)$=6.5001 $\times 10^{-2}$ are used. This form of the power law dependence of frequency of $\kappa^*(\Omega)$ does not describe the viscoelastic properties for all frequencies accurately. In particular, this formula is incorrect for low frequencies since the stiffness becomes negative. Therefore, values computed using this formula for frequencies below 0.77 Hz are discarded in this analysis. The frequency-dependent data for the dashpot coefficients and for the spring stiffness are written at intervals of 0.035 Hz over the frequency range of 0.77 to 14 Hz.

Results and discussion

For analysis 1 the exact solution for damped free oscillation is

$$u = u^0 \exp(-\eta\omega t)\left[\cos\omega\sqrt{1-\eta^2}\,t + \frac{\eta}{\sqrt{1-\eta^2}}\sin\omega\sqrt{1-\eta^2}\,t\right],$$

where $\omega = \sqrt{k/m}$ is the natural frequency of the undamped system (34.05 rad/s in this analysis), $\eta = c/2\sqrt{mk}$ is the ratio of damping to critical damping (0.068 in this analysis), and u^0 is the initial displacement of 25.4 mm (1 in).

The exact solution and the Abaqus solutions obtained using the different values of the half-increment tolerance are plotted in *Figure 2*. The tighter tolerance provides the more accurate solution, showing a slight phase shift later in the response. The looser tolerance shows considerably more phase shift, as expected.

At any time during the analysis Abaqus can provide a summary of the energy present in the structure, as well as quantities such as viscous and plastic dissipation. Summation of the various energy quantities yields an energy balance. Comparison of this balance with the initial strain energy of the system yields the energy lost due to numerical damping in the time integration operator. *Table 1* is a summary of all the energy terms at the end of the problem ($t = 0.7$ seconds). Since the initial strain energy is 1.695 N-m (15 lb-in), the numerical damping loss is 1% for the smaller value of the tolerance and 9.1% for the larger value.

For analysis 2 the steady-state response of the viscously damped single-degree-of-freedom system subjected to a cosine forcing function is given by

$$u = \overline{u}\cos(\Omega t - \phi),$$

where the amplitude of oscillation is

$$\overline{u} = \frac{\overline{P}}{\sqrt{\left(k - m\Omega^2\right)^2 + (c\Omega)^2}} = \frac{\overline{P}}{k}\frac{1}{\sqrt{\left(1 - (\Omega/\omega)^2\right)^2 + (c\Omega/k)^2}},$$

with $\omega = \sqrt{k/m}$, and the phase angle of the response is

$$\phi = \arctan\frac{c\Omega/k}{1 - (\Omega/\omega)^2}.$$

The response calculated by the direct-solution and subspace-based steady-state dynamic procedures are in exact agreement with these solutions. The amplitude and the angle of phase lag for the frequency range of 0 to 10 cycles/time are shown in *Figure 3* and *Figure 4*, respectively.

For analysis 3 the steady-state response of the structurally damped single-degree-of-freedom system subjected to a cosine forcing function can be obtained from the solutions for the viscously damped case by replacing the constant dashpot coefficient with the equivalent frequency-dependent dashpot coefficient,

$$c = k\gamma/\Omega,$$

so the amplitude of oscillation is

$$\overline{u} = \frac{\overline{P}}{k}\frac{1}{\sqrt{\left(1 - (\Omega/\omega)^2\right)^2 + \gamma^2}}$$

and the phase angle of the response is

$$\phi = \arctan\frac{\gamma}{1 - (\Omega/\omega)^2}.$$

The Abaqus solutions obtained by the direct-solution and subspace-based steady-state dynamic procedures are again in exact agreement with these analytical results, as shown in *Figure 5* and *Figure 6*. Comparing these results with those in *Figure 3* and *Figure 4* for viscous damping, two differences are apparent. First, resonance (maximum amplitude) occurs at $\Omega = \omega$ and not at $\Omega < \omega$ as in analysis 2. Second, the phase angle for $\Omega = 0$ is $\phi = \arctan\gamma$ instead of zero as in analysis 2; therefore, motion with structural damping, where the energy dissipation is rate independent, will never be in phase with the forcing function.

For analysis 4 the steady-state response of the two-degree-of-freedom system with viscoelastic damping is obtained numerically by solving the system of four equations for the real and imaginary parts of the response at the two nodes, from which the response amplitudes

$$|\overline{u}_j| = \sqrt{u_{jR}^2(\Omega) + u_{jI}^2(\Omega)}$$

and the phase angles

$$\phi_j = \arctan \frac{-u_{jI}}{u_{jR}}$$

are obtained. The solutions obtained by the direct-solution steady-state dynamic procedure in Abaqus are also in exact agreement with the numerical solutions. The results are presented in *Figure 7* and *Figure 8*, which show the amplitudes and angles of phase lag of the response, respectively, at the free nodes for the frequency range of 0.77 to 14 cycles/time. It is important to realize that small intervals must be used in the frequency sweep to obtain results with high accuracy (in particular for the peak response). Furthermore, when the frequency dependence is nonlinear, such as exhibited in this system, the quality of the solution also depends upon the accuracy of the frequency-dependent spring and dashpot data used in the calculations. Abaqus assumes that the properties vary linearly over each frequency interval; consequently, a small interval size should be used in the discretization of the data to minimize interpolation errors.

Input files

vibration_1dof_dyn_haft1.inp	One-degree-of-freedom time-integration dynamic analysis with HAFTOL set to 4.448 N (1 lb).
vibration_1dof_ssdyn_viscous.inp	One-degree-of-freedom direct-solution and subspace-based steady-state dynamic analysis with viscous damping.
vibration_1dof_ssdyn_hyster.inp	One-degree-of-freedom direct-solution and subspace-based steady-state dynamic analysis with hysteretic damping.
vibration_2dof_ssdyn_visco.inp	Two-degree-of-freedom direct-solution steady-state dynamic analysis with viscoelastic damping.
vibration_1dof_dyn_haft2.inp	Problem with HAFTOL set to 44.48 N (10 lb).
vibration_dampdata1.inp	Frequency-dependent damping coefficients used in analysis 3.
vibration_dampdata2.inp	Frequency-dependent damping coefficients used in analysis 4.
vibration_springdata.inp	Frequency-dependent spring stiffness used in analysis 4.

References

- Denhartog, J. P., Mechanical Vibrations, Dover, 1985.

Tables

Table 1: Energy balance at 0.7 seconds.

		Solution with half-increment tolerance = 4.448 N (1 lb)	Solution with half-increment tolerance = 44.48 N (10 lb)
Kinetic energy	N-m	0.0472	0.0033
	lb-in	0.418	0.029
Strain energy	N-m	0.0490	0.1943
	lb-in	0.434	1.720

		Solution with half-increment tolerance = 4.448 N (1 lb)	Solution with half-increment tolerance = 44.48 N (10 lb)
Dissipated energy	N-m	1.5817	1.3445
	lb-in	14.000	11.900
Total energy	N-m	1.6780	1.5421
	lb-in	14.852	13.649
Energy loss through numerical damping	N-m	0.0167	0.1526
	lb-in	0.148	1.351

Figures

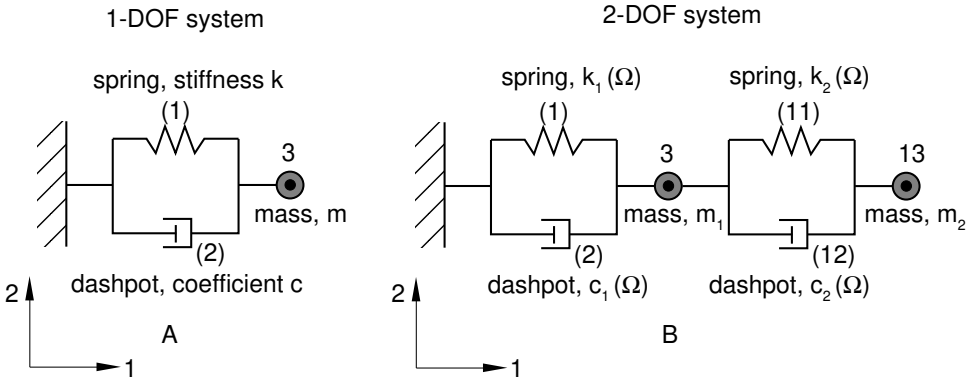


Figure 1: One- and two-DOF spring-mass-dashpot systems.

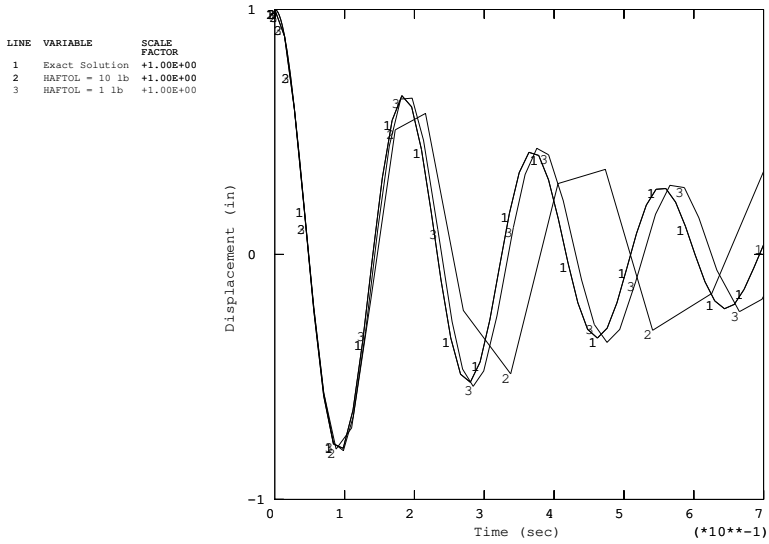


Figure 2: Displacement-time response for one-DOF spring-mass-dashpot example.

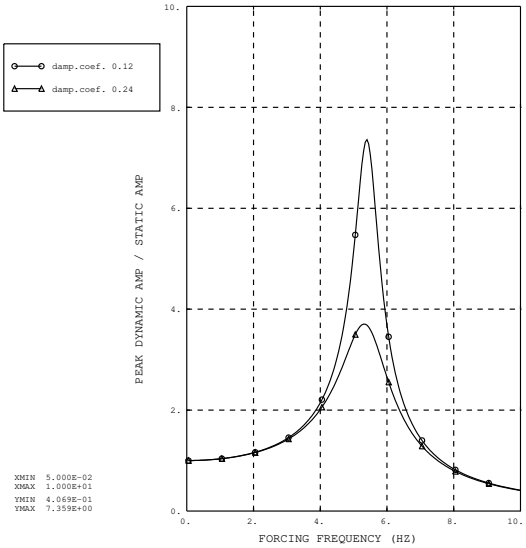


Figure 3: Peak amplitude response for viscous damping.

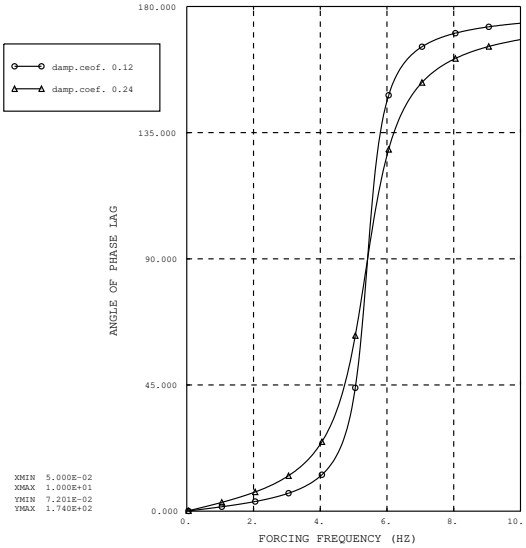


Figure 4: Phase angle response for viscous damping.

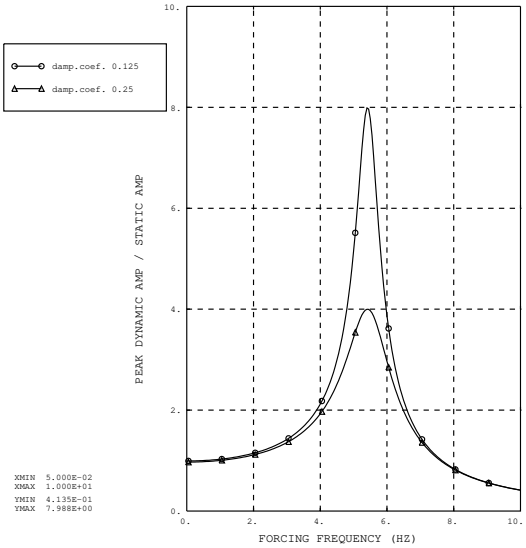


Figure 5: Peak amplitude response for hysteretic damping.

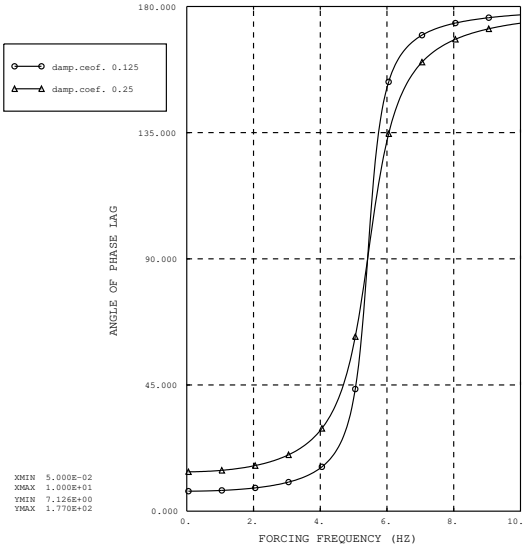


Figure 6: Phase angle response for hysteretic damping.

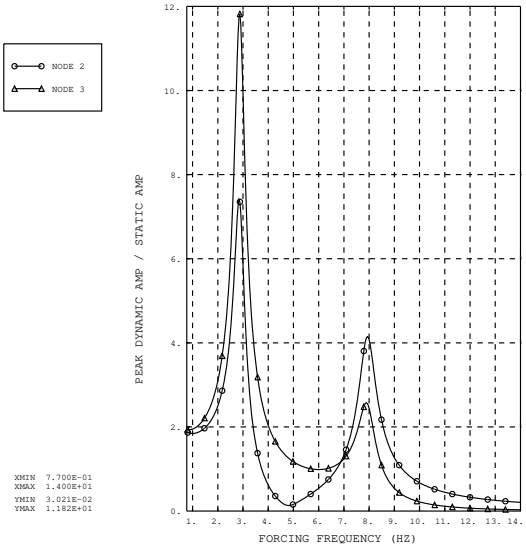


Figure 7: Peak amplitude response for viscoelastic damping.

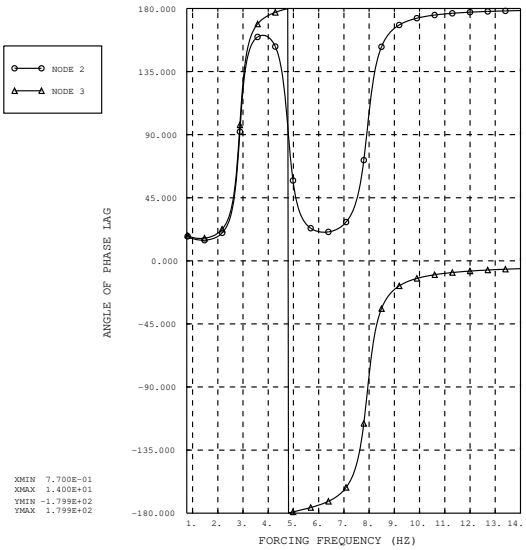


Figure 8: Phase angle response for viscoelastic damping.

Verification of Rayleigh damping options with direct integration and modal superposition

Products: Abaqus/Standard

Overview

This example verifies the Rayleigh damping options in Abaqus for direct integration and modal superposition procedures. The Abaqus results are compared with an exact solution for a simple problem.

For direct integration Rayleigh damping is defined with damping in the material definition for those elements in which mass and stiffness proportional damping is desired. In a SIM-based modal dynamic analysis Rayleigh damping can be introduced in the material definition (same as for direct integration) and as modal damping in the step definition. If the SIM architecture is not used, only modal damping can be used in a modal dynamic analysis. In this example only the latter form of damping is represented. For direct integration analysis Rayleigh damping can be introduced in any stress-based element, but it is not available for spring elements; dashpot elements should be used in parallel with spring elements for this purpose (see *Free and forced vibrations with damping*). Elements with nonhomogeneous material damping properties are dealt with by taking a volume average of the damping coefficients. Stiffness proportional damping in nonlinear analysis is discussed in *Material Damping*.

The example is the simplest dynamic system: a massless truss connecting a point mass to ground. The mass is obtained by giving the material in the truss a density so that the lumped mass of the truss gives the correct point mass at the free end of the truss. The truss is initially stretched and then let go so that it undergoes vibrations of small amplitude. This is a linear problem; consequently, the response can be predicted using either the direct integration or modal dynamic procedures. These solutions are compared with each other and to the exact solution of the equation of motion.

Problem description

Figure 1 shows the geometry. The model consists of a single truss element, type T3D2, constrained at one node and free to move only in the x-direction at its other node. The truss's mass matrix is lumped so that the system is equivalent to a spring and a lumped mass. The cross-sectional area of the truss is 645 mm^2 (1 in^2), and its length is 254 mm (10 in). It is made of linear elastic material, with Young's modulus 69 GPa (10^7 lb/in^2). The density of the truss provides a lumped mass at the unrestrained end of 2.777×10^5 kg (1585 lb-s^2/in).

In each case the mass is displaced by 25.4 mm (1 in) in an initial static step. It is then released in the dynamic (or modal dynamic) step, and the displacement response history is saved on a file for postprocessing. The time histories are plotted; and the logarithmic decrement, δ, of the peak response is calculated graphically and compared with the theoretical value.

Results and discussion

The equation of motion for the system is

$$m\ddot{u} + c\dot{u} + ku = 0,$$

where m is the mass, c the damping, k the stiffness, and u the displacement.

Rayleigh damping defines the damping as $c = \alpha m + \beta k$, where α is the mass damping factor and β is the stiffness damping factor.

Assuming a solution of the form $u(t) = u_0 \exp \lambda t$, we have

$$\lambda = \frac{-c}{2m} \pm \sqrt{\left(\frac{c}{2m}\right)^2 - \omega_u^2},$$

where $\omega_u = \sqrt{k/m}$ is the undamped frequency of vibration (25.118 rad/sec for the parameters of this example). Critical damping occurs when the value of c causes the discriminant of this equation to be zero, so

$$c_c = 2m\omega_u = 2\sqrt{km}.$$

We define the damping ratio, ξ, as the ratio of damping to critical damping:

$$\xi = \frac{c}{c_c} = \frac{\alpha}{2\omega_u} + \frac{\beta\omega_u}{2}.$$

The relationships in this equation are often used as a basis for choosing α and β.

The equation defining λ can be rewritten

$$\lambda = \omega_u \left(-\xi \pm \sqrt{\xi^2 - 1}\right).$$

We choose the damping in this case to be less than critical, so $\xi < 1$ and the system can vibrate. The initial conditions are $u(0) = 1$ and $\dot{u}(0) = 0$, so the dynamic part of the motion is

$$u(t) = \exp(-\xi\omega_u t)\left(\frac{\xi}{\sqrt{1-\xi^2}}\sin\omega_d t + \cos\omega_d t\right),$$

where $\omega_d = \omega_u\sqrt{1-\xi^2}$ is the damped frequency of the system.

The amplitudes of this oscillatory equation before and after one period of vibration, $T = 2\pi/\omega_d$, have the ratio

$$\frac{u(t)}{u(t+2\pi/\omega_d)} = \exp\left(2\pi\xi\frac{\omega_u}{\omega_d}\right),$$

so the logarithmic decrement over n cycles of response is

$$\delta = \frac{1}{n}\ln\left(\frac{u(t)}{u(t+nT)}\right) = \left(2\pi\xi\frac{\omega_u}{\omega_d}\right).$$

Table 1 shows the values of δ calculated from Abaqus for the various test cases examined, together with their corresponding exact solution. A sample time history from which the logarithmic decrements are calculated is shown in *Figure 2*. All the Abaqus runs use fixed time increments of .01 seconds. The integrator used in the modal method is exact, so the results of that analysis are exact. The integrator used in the direct integration method is not exact; however, since the period of the system is 0.25 seconds, the time increment chosen gives 25 increments per cycle, so those results are also quite accurate.

Input files

rayleighdamping_direct_alpha.inp	Direct integration analysis, $\alpha = 1.00472$, $\beta = 0.0$.
rayleighdamping_modal_alpha.inp	Modal superposition analysis, $\alpha = 1.00472$, $\beta = 0.0$.
rayleighdamping_direct_beta.inp	Direct integration analysis, $\alpha = 0.0$, $\beta = 1.59248 \times 10^{-3}$.
rayleighdamping_modal_beta.inp	Modal superposition analysis, $\alpha = 0.0$, $\beta = 1.59248 \times 10^{-3}$.
rayleighdamping_direct.inp	Direct integration analysis, $\alpha = 1.00472$, $\beta = 1.59248 \times 10^{-3}$.

rayleighdamping_modal.inp	Modal superposition analysis, $\alpha = 1.00472$, $\beta = 1.59248 \times 10^{-3}$.
rayleighdamping_beam_alpha.inp	Direct integration analysis using **BEAM GENERAL SECTION*, $\alpha = 1.00472$, $\beta = 0.0$.
rayleighdamping_beam_beta.inp	Direct integration analysis using **BEAM GENERAL SECTION*, $\alpha = 0.0$, $\beta = 1.59248 \times 10^{-3}$.
rayleighdamping_beam.inp	Direct integration analysis using **BEAM GENERAL SECTION*, $\alpha = 1.00472$, $\beta = 1.59248 \times 10^{-3}$.
rayleighdamping_shell_alpha.inp	Direct integration analysis using **SHELL GENERAL SECTION*, $\alpha = 1.00472$, $\beta = 0.0$.
rayleighdamping_shell_beta.inp	Direct integration analysis using **SHELL GENERAL SECTION*, $\alpha = 0.0$, $\beta = 1.59248 \times 10^{-3}$.
rayleighdamping_shell_.inp	Direct integration analysis using **SHELL GENERAL SECTION*, $\alpha = 1.00472$, $\beta = 1.59248 \times 10^{-3}$.
rayleighdamping_substr_alpha.inp	Direct integration analysis using substructures, $\alpha = 1.00472$, $\beta = 0.0$.
rayleighdamping_substr_alpha_gen1.inp	Substructure generation referenced in the analysesrayleighdamping_substr_alpha.inp and rayleighdamping_overide.inp.
rayleighdamping_substr_beta.inp	Direct integration analysis using substructures, $\alpha = 0.0$, $\beta = 1.59248 \times 10^{-3}$.
rayleighdamping_substr_beta_gen1.inp	Substructure generation referenced in the analysis rayleighdamping_substr_beta.inp.
rayleighdamping_substr.inp	Direct integration analysis using substructures, $\alpha = 1.00472$, $\beta = 1.59248 \times 10^{-3}$.
rayleighdamping_substr_gen1.inp	Substructure generation referenced in the analysis rayleighdamping_substr.inp.
rayleighdamping_override.inp	Tests override of damping properties on the **SUBSTRUCTURE PROPERTY* option.
rayleighdamping_usr_element.inp	Uses Rayleigh damping with user elements in direct integration dynamics (**DYNAMIC*).

Tables

Table 1: Exact versus graphical logarithmic decrements.

Damping parameters		Damping ratio, ξ	Logarithmic decrement		
Mass	Stiffness		Exact	Direct integration	Modal superposition
α	β				
1.00472	0.0	0.02	0.1257	0.1253	0.1257
0.0	1.59248×10^{-3}	0.02	0.1257	0.1253	0.1257
1.00472	1.59248×10^{-3}	0.04	0.2514	0.2499	0.2514

Figures

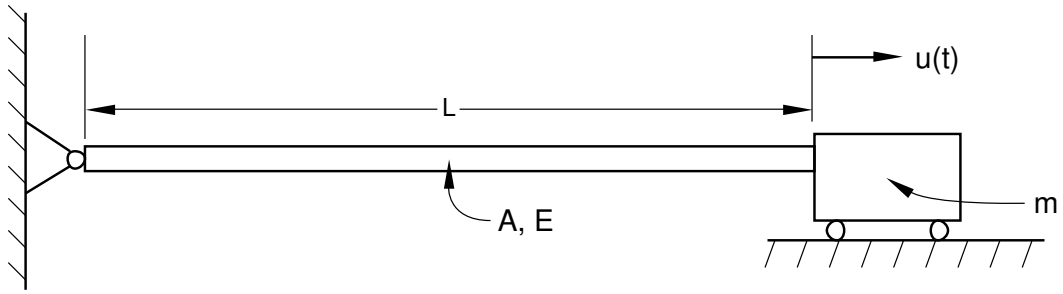


Figure 1: Truss-mass vibration system.

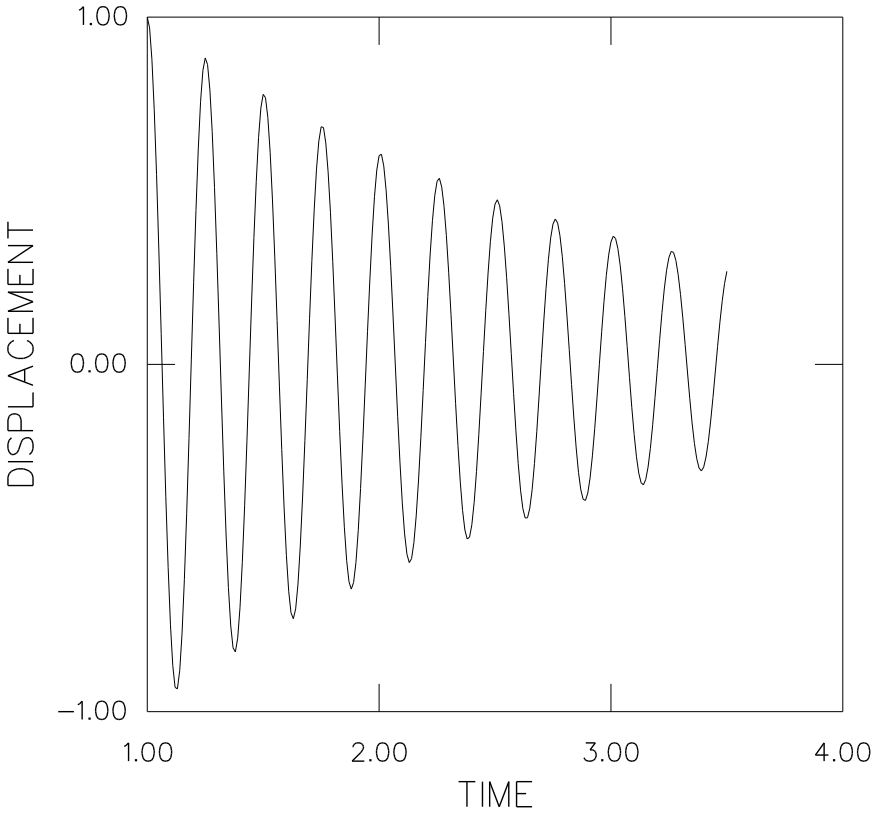


Figure 2: Sample time history.

Eigenvalue analysis of a cantilever plate

Products: Abaqus/Standard

Overview

This example, using a simple plate problem, verifies the linear vibration capability for shell elements.
The structure in this example is a cantilever plate, half as wide as it is long, with a width to thickness ratio of 100 to 1. The analysis is done with three different meshes; the finer meshes exercise the eigenvalue routines on relatively large models.

Problem description

The properties of the plate are shown in *Figure 1*. The analyses involve three different meshes: 2 × 4, 5 × 10, and 10 × 20, where the smaller number of elements is used across the width of the plate. The following shell elements are used with each mesh: S3R, S4R5, S8R5, S9R5, STRI65, STRI3, S4R, S4, and S8R. The meshes used with the triangular elements are based on dividing each rectangle into two triangles.

Results and discussion

The series solution developed by Barton (1951) is used by Zienkiewicz (1971) for a study similar to this example. Here a thinner plate is used than the one described by Zienkiewicz (1971), because the theoretical solution is a thin plate solution, and we wish to ensure that element types STRI65, S9R5, S8R5, S4R5, S8R, S4R, and S4 (which include transverse shear strain energy in penalty form) provide comparable results. If the thicker plate was used, the shear flexibility in these elements would cause their predictions to be different from the thin-plate solutions.

The second-order shell elements (S9R5, STRI65, S8R5, and S8R) all give essentially convergent values for the first four frequencies, even with the 2 × 4 mesh. (Here we mean convergence with respect to the number of elements used and base this conclusion on the observation that the frequency values are not changing significantly as the mesh is refined.) S8R shows some reduction in frequency in the fourth mode as the mesh is refined: presumably this is caused by transverse shear flexibility affecting the result. For the first-order elements (S4R5, S4R, S4, S3R, and STRI3) all the meshes give quite good values for the frequencies, except for S3R elements. Due to constant bending strain approximations, S3R elements require a finer mesh for good accuracy, which is evident from the results. For the same number of degrees of freedom the second-order elements give better results for the higher modes than the first-order elements. The mode shapes are shown in *Figure 2*.

Input files

eigenvalueplate_s3r_coarse.inp	Element type S3R, 2 × 4 mesh.
eigenvalueplate_s3r_fine.inp	Element type S3R, 5 × 10 mesh.
eigenvalueplate_s3r_finer.inp	Element type S3R, 10 × 20 mesh.
eigenvalueplate_s4_coarse.inp	Element type S4, 2 × 4 mesh.
eigenvalueplate_s4_fine.inp	Element type S4, 5 × 10 mesh.
eigenvalueplate_s4_finer.inp	Element type S4, 10 × 20 mesh.
eigenvalueplate_s4r_coarse.inp	Element type S4R, 2 × 4 mesh.
eigenvalueplate_s4r_fine.inp	Element type S4R, 5 × 10 mesh.
eigenvalueplate_s4r_finer.inp	Element type S4R, 10 × 20 mesh.

eigenvalueplate_s4r5_coarse.inp	Element type S4R5, 2 × 4 mesh.
eigenvalueplate_s4r5_fine.inp	Element type S4R5, 5 × 10 mesh.
eigenvalueplate_s4r5_finer.inp	Element type S4R5, 10 × 20 mesh.
eigenvalueplate_s8r_coarse.inp	Element type S8R, 2 × 4 mesh.
eigenvalueplate_s8r_fine.inp	Element type S8R, 5 × 10 mesh.
eigenvalueplate_s8r_finer.inp	Element type S8R, 10 × 20 mesh.
eigenvalueplate_s8r5_coarse.inp	Element type S8R5, 2 × 4 mesh.
eigenvalueplate_s8r5_fine.inp	Element type S8R5, 5 × 10 mesh.
eigenvalueplate_s8r5_finer.inp	Element type S8R5, 10 × 20 mesh.
eigenvalueplate_s9r5_coarse.inp	Element type S9R5, 2 × 4 mesh.
eigenvalueplate_s9r5_fine.inp	Element type S9R5, 5 × 10 mesh.
eigenvalueplate_s9r5_finer.inp	Element type S9R5, 10 × 20 mesh.
eigenvalueplate_stri3_coarse.inp	Element type STRI3, 2 × 4 mesh.
eigenvalueplate_stri3_fine.inp	Element type STRI3, 5 × 10 mesh.
eigenvalueplate_stri3_finer.inp	Element type STRI3, 10 × 20 mesh.
eigenvalueplate_stri65_coarse.inp	Element type STRI65, 2 × 4 mesh.
eigenvalueplate_stri65_fine.inp	Element type STRI65, 5 × 10 mesh.
eigenvalueplate_stri65_finer.inp	Element type STRI65, 10 × 20 mesh.

References

- Barton, M. V., “Vibrations of Rectangular and Shear Plates,” *Journal of Applied Mechanics*, vol. 18, pp. 129–134, 1951.
- Zienkiewicz, O. C., The Finite Element Method in Engineering Science, McGraw-Hill, London, 1971.

Tables

Table 1: Frequencies of the first four modes, in Hertz.

Mode	1	2	3	4
Series Solution	84.6	363.8	526.6	1187.0
S3R				
2 × 4 (90)	91.5	539.9	653.7	1811.8
5 × 10 (396)	86.8	401.1	549.8	1374.9
10 × 20 (1386)	85.1	367.8	532.1	1210.0
S4				
2 × 4 (90)	84.7	367.5	610.6	1324.1
5 × 10 (396)	84.0	361.9	535.7	1198.9
10 × 20 (1386)	83.9	360.8	525.6	1179.5
S4R				
2 × 4 (90)	84.2	357.2	609.5	1257.5
5 × 10 (396)	83.9	360.4	535.3	1189.7

Mode	1	2	3	4
10 × 20 (1386)	83.8	360.4	525.4	1177.2
S4R5				
2 × 4 (90)	84.2	356.3	609.3	1251.6
5 × 10 (396)	83.9	360.4	535.3	1189.6
10 × 20 (1386)	83.8	360.5	525.4	1177.5
S8R				
2 × 4 (222)	83.8	361.2	525.5	1183.8
5 × 10 (1086)	83.9	360.4	522.5	1172.9
10 × 20 (3966)	83.8	359.7	522.2	1170.9
S8R5				
2 × 4 (270)	83.8	360.6	523.8	1176.6
5 × 10 (1386)	83.8	360.6	522.4	1173.7
10 × 20 (5166)	83.8	360.5	522.2	1173.2
S9R5				
2 × 4 (270)	83.8	360.6	523.8	1176.6
5 × 10 (1386)	83.8	360.6	522.4	1173.7
10 × 20 (5166)	83.8	360.5	522.2	1173.2
STRI3				
2 × 4 (90)	81.6	298.9	473.7	928.2
5 × 10 (396)	83.5	348.2	514.1	1130.0
10 × 20 (1386)	83.7	357.4	520.3	1163.0
STRI65				
2 × 4 (270)	84.1	368.1	524.0	1229.1
5 × 10 (1386)	83.9	360.9	521.8	1175.4
10 × 20 (5166)	83.8	360.5	522.2	1172.9
The grid size specification is followed by the number of degrees of freedom in the model.				

Figures

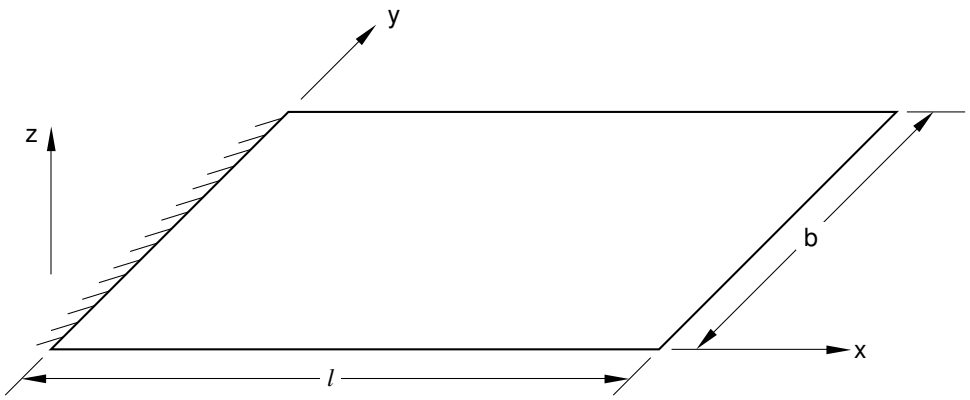
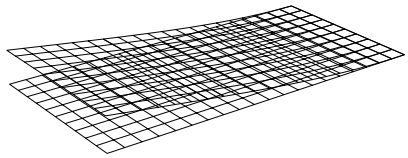


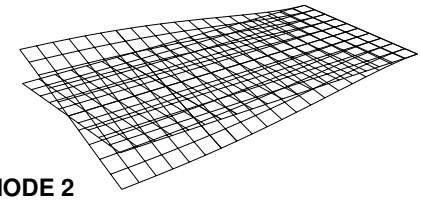
Plate properties:

Width, b	25.4 mm (1.0 in)
Length, l	50.8 mm (2.0 in)
Thickness	0.254 mm (0.01 in)
Young's modulus	206.8 GPa (30.0 x 10^6 lb/in^2)
Poisson's ratio	0.3
Density	7827.0 kg/m^3 (7.324 x 10^{-4} lb-s^2/in^4)

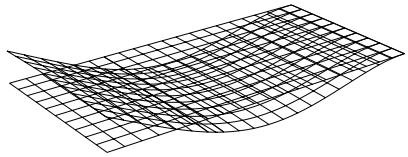
Figure 1: Cantilever plate.



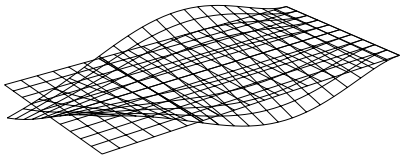
MODE 1



MODE 2



MODE 3



MODE 4

Figure 2: Mode shapes for vibrating cantilever plate.

Vibration of a rotating cantilever plate

Products: Abaqus/Standard

Overview

This example verifies the centrifugal load stiffness effect present in vibration problems when the structure is undergoing small vibrations in a rotating coordinate frame.

The most common example of vibration problems is the study of the vibrations of components of rotating machines, such as the blades on turbines and compressors. In such cases two effects that are not present in vibration problems in fixed coordinate systems become important: the initial stressing of the structure caused by the centrifugal loading and the "load stiffness" effect caused by the line of action of the centrifugal load changing if the vibration causes motion in the plane normal to the axis of rotation. In most conventional designs of rotating machines the initial stress effect is a stiffening effect, and the load stiffness effect is a softening effect. In the vibration of blades on turbines or compressors the load stiffness effect is significant only for long blades on small wheels, such as the fan blades on modern high bypass jet engines for aircraft: see Hibbitt (1979). The purpose of this example is to illustrate this effect and verify the capability in Abaqus for such vibration studies.

Problem description

The model is a single, flat plate, 328 mm long, 28 mm wide, and 3 mm thick, built into a rigid wheel of 150 mm radius, spinning about its axis. Two versions of the problem are studied. In "Case A" the plate is mounted so that its first vibration mode is in the plane containing the axis of the wheel. Thus, the line of action of the centrifugal load does not change as the blade undergoes small vibrations; hence, the load stiffness effect does not participate in this mode. In "Case B" the plate is mounted so that its first vibration mode is in a plane at right angles to the axis of rotation of the wheel. Thus, the load stiffness effect is important in this mode. Since the plate is relatively long compared to the radius of the wheel, the load stiffness effect is significant: the first mode frequency is substantially lower in Case B than it is in Case A.

Several different element types are used (beams, shells, three-dimensional solid elements). In each case a "reasonable" mesh is chosen—typically six elements along the plate. Since we are comparing only the lowest mode frequency, rather coarse meshing should be adequate.

The plate is made of steel, with Young's modulus 217 GPa and a density of 7850 kg/m^3.

Analysis

The analysis is done in a series of steps. Step 1 extracts the lowest mode of the system at rest (no rotation of the wheel) using the frequency procedure. In this example only the lowest frequency is required: in a practical case several frequencies would probably be needed.

Step 2 is a static procedure in which the centrifugal load, corresponding to a rotational speed of the system of 25 revolutions/second, is applied using a distributed load. This centrifugal load is applied using both the CENT and CENTRIF load types. The distributed load magnitude must be given as $\rho\Omega^2$ with the CENT load type and as Ω^2 with the CENTRIF load type. The CENTRIF load type uses the actual mass matrix of the element in the load calculation for the density, which means that a lumped mass matrix is used for first-order elements and a consistent mass matrix is used for second-order elements. The CENT load type always uses a consistent mass matrix. Geometric nonlinearities are considered in the step, which causes Abaqus to include the initial stress and load stiffness effects and implies a nonlinear analysis.

Step 3 uses the frequency procedure to obtain the lowest frequency at this rotational speed. Step 4 is a static step to increase the centrifugal load to a rotational speed of 50 revolutions/second, Step 5 obtains the lowest eigenmode at this speed, Step 6 increases the speed to 75 revolutions/second, and Step 7 obtains the lowest eigenmode at this speed.

Substructure analysis

This example is suitable for demonstrating the substructure preload capability in Abaqus. With this option it is possible to create a finite element mesh, load it using a nonlinear procedure, and create a substructure using the current stiffness after the loading. If the entire wheel had to be modeled with all the rotating blades, the model could be simplified by using this option. The blade would be modeled as a substructure, the centrifugal force applied, and the stiffness formed including the "load stiffness." The substructure could then be rotated and used for all the blades attached to the wheel.

Preloading is obtained by preceding a substructure generation step with one or several analysis steps. The substructure stiffness is formed from the final loading condition of the preceding general analysis step. Four substructures are generated for each analysis. The first is generated without any preloading. The remaining three substructures are generated after a centrifugal load has been applied so that each includes the load stiffness associated with a different rotational speed. Furthermore, when the substructures are used the load stiffness is included in the substructure stiffness matrix and is, thus, included in the frequency extraction whether or not geometric nonlinearities are considered in the frequency step.

Results and discussion

The frequencies obtained in each case for each geometric model and speed are shown in *Table 1*, where these numerical results are compared to a Rayleigh quotient solution (Lindberg, 1986). The numerical results are very close to the Rayleigh quotient solution. The differences between the results obtained using load type CENT and load type CENTRIF are negligible.

Input files

Case A:

vibrotplate_b21_cent_a.inp	Element type B21 with the CENT loading option.
vibrotplate_b21_centrif_a.inp	Element type B21 with the CENTRIF loading option.
vibrotplate_b23_cent_a.inp	Element type B23 with the CENT loading option.
vibrotplate_b23_centrif_a.inp	Element type B23 with the CENTRIF loading option.
vibrotplate_b31_cent_a.inp	Element type B31 with the CENT loading option.
vibrotplate_b31_centrif_a.inp	Element type B31 with the CENTRIF loading option.
vibrotplate_b33_cent_a.inp	Element type B33 with the CENT loading option.
vibrotplate_b33_centrif_a.inp	Element type B33 with the CENTRIF loading option.
vibrotplate_c3d8i_cent_a.inp	Element type C3D8I with the CENT loading option.
vibrotplate_c3d8i_centrif_a.inp	Element type C3D8I with the CENTRIF loading option.
vibrotplate_c3d10_cent_a.inp	Element type C3D10 with the CENT loading option.
vibrotplate_c3d10_centrif_a.inp	Element type C3D10 with the CENTRIF loading option.
vibrotplate_c3d10hs_cent_a.inp	Element type C3D10HS with the CENT loading option.
vibrotplate_c3d10hs_centrif_a.inp	Element type C3D10HS with the CENTRIF loading option.
vibrotplate_c3d10m_cent_a.inp	Element type C3D10M with the CENT loading option.
vibrotplate_c3d10m_centrif_a.inp	Element type C3D10M with the CENTRIF loading option.
vibrotplate_c3d20_cent_a.inp	Element type C3D20 with the CENT loading option.

vibrotplate_c3d20_centrif_a.inp	Element type C3D20 with the CENTRIF loading option.
vibrotplate_c3d20r_cent_a.inp	Element type C3D20R with the CENT loading option.
vibrotplate_c3d20r_centrif_a.inp	Element type C3D20R with the CENTRIF loading option.
vibrotplate_s8r_cent_a.inp	Element type S8R with the CENT loading option.
vibrotplate_s8r_centrif_a.inp	Element type S8R with the CENTRIF loading option.
vibrotplate_s8r5.inp	Element type S8R5 with the CENT loading option.
vibrotplate_s8r5_centrif_a.inp	Element type S8R5 with the CENTRIF loading option.
vibrotplate_s8r5_substr.inp	Element type S8R5 when the blade is modeled as a substructure with the CENT loading option.
vibrotplate_s8r5_substr_gen1.inp	Substructure generation referenced in the analysis vibrotplate_s8r5_substr.inp.
vibrotplate_s8r5_substr_centrif_a.inp	Element type S8R5 when the blade is modeled as a substructure with the CENTRIF loading option.
vibrotplate_s8r5_substr_centrif_a_gen1.inp	Substructure generation referenced in the analysis vibrotplate_s8r5_substr_centrif_a.inp.

Case B:

vibrotplate_b21_cent_b.inp	Element type B21 with the CENT loading option.
vibrotplate_b21_centrif_b.inp	Element type B21 with the CENTRIF loading option.
vibrotplate_b23_cent_b.inp	Element type B23 with the CENT loading option.
vibrotplate_b23_centrif_b.inp	Element type B23 with the CENTRIF loading option.
vibrotplate_b31_cent_b.inp	Element type B31 with the CENT loading option.
vibrotplate_b31_centrif_b.inp	Element type B31 with the CENTRIF loading option.
vibrotplate_b33_cent_b.inp	Element type B33 with the CENT loading option.
vibrotplate_b33_centrif_b.inp	Element type B33 with the CENTRIF loading option.
vibrotplate_c3d8i_cent_b.inp	Element type C3D8I with the CENT loading option.
vibrotplate_c3d8i_centrif_b.inp	Element type C3D8I with the CENTRIF loading option.
vibrotplate_c3d10_cent_b.inp	Element type C3D10 with the CENT loading option.
vibrotplate_c3d10_centrif_b.inp	Element type C3D10 with the CENTRIF loading option.
vibrotplate_c3d10hs_cent_b.inp	Element type C3D10HS with the CENT loading option.
vibrotplate_c3d10hs_centrif_b.inp	Element type C3D10HS with the CENTRIF loading option.
vibrotplate_c3d10m_cent_b.inp	Element type C3D10M with the CENT loading option.
vibrotplate_c3d10m_centrif_b.inp	Element type C3D10M with the CENTRIF loading option.
vibrotplate_c3d20_cent_b.inp	Element type C3D20 with the CENT loading option.
vibrotplate_c3d20_centrif_b.inp	Element type C3D20 with the CENTRIF loading option.
vibrotplate_c3d20r_cent_b.inp	Element type C3D20R with the CENT loading option.
vibrotplate_c3d20r_centrif_b.inp	Element type C3D20R with the CENTRIF loading option.
vibrotplate_s8r_cent_b.inp	Element type S8R with the CENT loading option.
vibrotplate_s8r_centrif_b.inp	Element type S8R with the CENTRIF loading option.
vibrotplate_s8r5_cent_b.inp	Element type S8R5 with the CENT loading option.
vibrotplate_s8r5_centrif_b.inp	Element type S8R5 with the CENTRIF loading option.
vibrotplate_s8r5_substr_cent_b.inp	Element type S8R5 when the blade is modeled as a substructure with the CENT loading option.
vibrotplate_s8r5_substr_cent_b_gen1.inp	Substructure generation referenced in the analysis vibrotplate_s8r5_substr_cent_b.inp.

vibrotplate_s8r5_substr_centrif_b.inp	Element type S8R5 when the blade is modeled as a substructure with the CENTRIF loading option.
vibrotplate_s8r5_substr_centrif_b_gen1.inp	Substructure generation referenced in the analysis vibrotplate_s8r5_substr_centrif_b.inp.

References

- Hibbitt, H. D., "Some Follower Forces and Load Stiffness," *International Journal for Numerical Methods in Engineering*, vol. 14, pp. 937–941, 1979.
- Lindberg, B., "Berechnung der ersten Eigenfrequenz eines Balkens in Fliehkraftfeld mit Rayleigh Quotient," *Internal report HTGE-ST-0051, Brown Boveri & Cie., Baden, Switzerland*, 1986.

Tables

Table 1: Spinning beam frequencies (Hz).

		Vibration in the plane of the rotation axis (Case A)			Vibration normal to the rotation axis (Case B)		
Rotary speed (cycles/sec)	0	25	50	75	25	50	75
Rayleigh quotient	23.68	41.74	72.10	104.27	33.42	51.95	72.44
B21	23.44	41.20	71.00	102.29	32.94	50.86	70.24
B23	23.68	41.72	71.94	103.73	33.40	51.73	71.66
B31	23.44	41.20	71.00	102.29	32.94	50.86	70.24
B33	23.68	41.83	72.14	103.98	33.54	52.00	72.02
S8R	23.89	41.90	72.13	103.91	33.63	51.99	71.91
S8R5	23.81	41.82	72.05	103.82	33.53	51.87	71.79
Substructure	23.82	41.88	72.33	104.58	33.56	51.98	72.04
C3D8I	24.23	41.90	71.93	103.66	33.82	52.15	72.25
C3D10	25.14	42.70	72.88	104.91	34.62	53.03	73.34
C3D10HS	25.14	42.71	72.89	104.92	34.63	53.04	73.37
C3D10M	24.82	42.40	72.51	104.45	34.32	52.70	72.96
C3D20	24.53	42.45	72.87	105.02	34.30	53.01	73.51
C3D20R	24.28	42.25	72.54	104.38	34.06	52.55	72.60

Figures

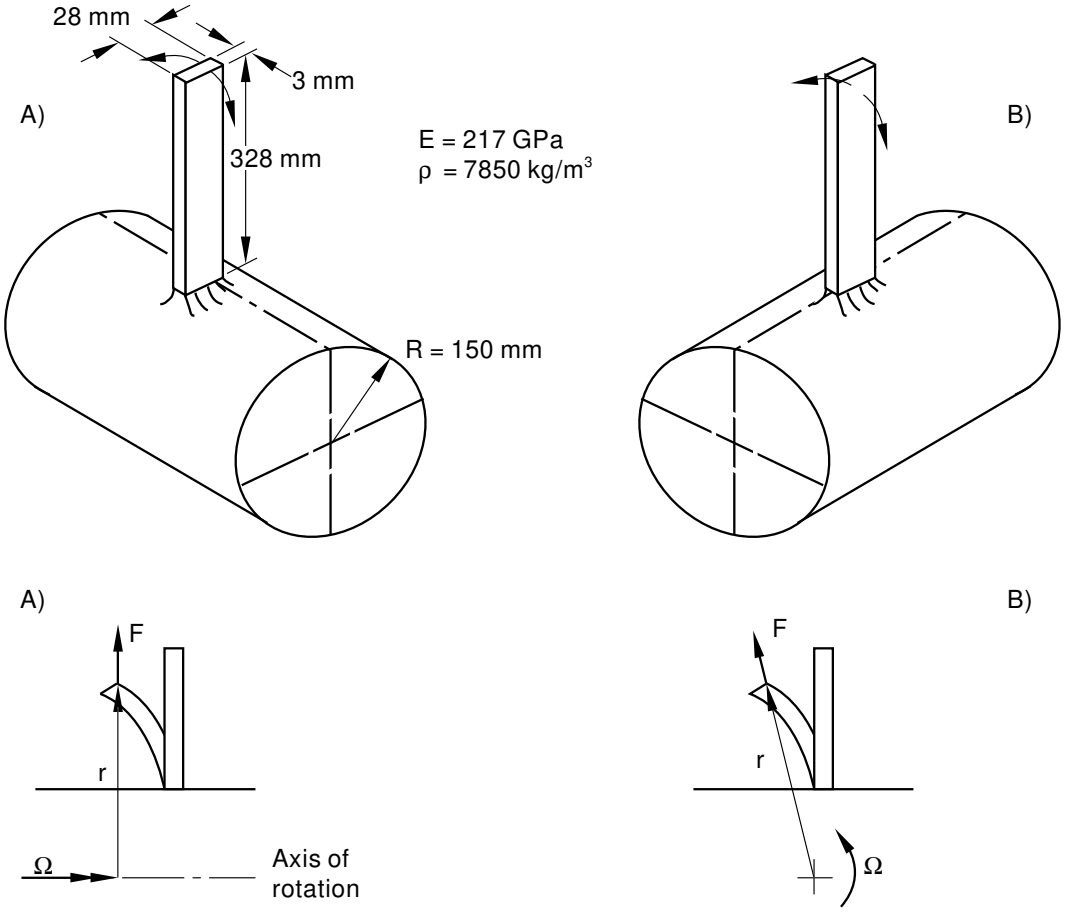


Figure 1: Plate and wheel geometry.

Response spectrum analysis of a simply supported beam

Products: Abaqus/Standard

Overview

This problem verifies the Abaqus capability for response spectrum analysis by comparing the Abaqus results to an exact solution for a simple case.

Problem description

The problem is a simply supported beam analyzed by Biggs (1964) and is shown in *Figure 1*. The beam has a rectangular cross-section of width 37 mm (1.458 in) and depth 355.6 mm (14 in). The mass density of the beam is 1.0473×10^5 kg/m^3 (0.0098 lb-s^2/in^4).

The finite element model is also shown in *Figure 1*. The response spectrum is applied in the vertical direction at both supports, and the response is determined based on the first mode of the model. Analyses are run using element types B21 and B23, with response spectra defined in the following section. Zero damping is specified for the problem. The beam section is defined as a beam section and a general beam section to test both specifications.

Response spectra definition

The response spectrum is defined as the peak response of a single degree of freedom spring-mass system excited by a given acceleration history applied to its base. Biggs (1964) defines the problem as having both supports moving vertically according to an acceleration history that ramps linearly from $+g$ to $-g$ (where g is the acceleration due to gravity) over a time period of 0.1 seconds and is zero after that. With this base acceleration history, the acceleration of the mass in the single degree of freedom spring-mass system is

$$\ddot{u} = g\left(1 - \cos\omega t + \frac{1}{\omega t_d}\sin\omega t - \frac{t}{t_d}\right) \quad \text{for} \quad t \le 2t_d,$$

$$\ddot{u} = -\dot{u}_{t=2t_d}\omega\cos\omega\left(t - 2t_d\right) - u_{t=2t_d}\omega^2\sin\omega\left(t - 2t_d\right) \quad \text{for} \quad t > 2t_d,$$

where ω is the natural frequency and $2t_d$ is the time of the ramp of the acceleration from $+g$ to $-g$.

The solution of these two equations for the maximum acceleration as a function of frequency defines the response spectrum. This has been done for frequencies of 5., 6., 6.098, 7., and 8. Hz. The following table shows the resulting response spectrum:

FREQUENCY (Hz)	ACCELERATION (g's)
5.	2.0000
6.	1.6667
6.098	1.6399
7.	1.4286
8.	1.4530

Abaqus provides options for spectrum input in terms of acceleration, velocity, and displacement. The table above is expanded to these forms using the definitions that $v = \bar{a}/\omega$ and $u = \bar{a}/\omega^2$, where $\bar{a}$ is the peak acceleration (in m/s^2 or in/sec^2), v is the peak velocity, and u is the peak displacement. The response spectra used in the four

runs are shown in the *Table 1*. In the table the acceleration spectrum in m/s^2 (in/sec^2) has been doubled and a compensating scale factor of 0.5 is used in the input.

Results and discussion

Biggs (1964) calculates the exact natural frequency of the first mode as 6.1 Hz, with a modal participation factor of 1.27324. Abaqus gives the first mode frequency as 6.098 Hz for the 10-element model using element type B23 and 6.0808 Hz for the model using element type B21. The corresponding modal participation factors are 1.2733 and 1.2628. Both of the Abaqus results are quite close to Biggs's values, with the cubic beam (B23) results giving better agreement—possibly because the linear beam, B21, allows transverse shear deformation, which adds flexibility to the model and, hence, reduces the stiffness.

Biggs also gives the values of the maximum displacement, bending moment, curvature, and bending stress at the beam midspan using SRSS summation. These values are used in *Table 2* to check the Abaqus calculations (the stress, moment, and curvature values reported from the Abaqus runs are obtained by extrapolation of integration point values to the midspan node). The Abaqus results compare well for all four test cases.

Input files

responsespecbeam.inp	Displacement response spectrum problem.
responsespecbeam_velocity.inp	Velocity response spectrum.
responsespecbeam_acc.inp	*g* response spectrum.
responsespecbeam_absacc.inp	Absolute acceleration spectrum.

References

- Biggs, J. M., *Introduction to Structural Dynamics*, McGraw-Hill, pp. 256–263, 1964.

Tables

Table 1: Response spectra definition.

Frequency		Acceleration			Velocity		Displacement	
Hz	rad/sec	g's	m/s^2	in/sec^2	m/s	in/sec	m	in
5.	31.4159	2.0000	39.258	1545.60	.6248	24.5990	.0199	.7830
6.	37.6991	1.6667	32.716	1288.02	.4339	17.0830	.0115	.4531
6.098	38.3418	1.6399	32.190	1267.32	.4201	16.5382	.0110	.4316
7.	43.9823	1.4286	28.042	1104.02	.3188	12.5507	.0072	.2854
8.	50.2654	1.4530	28.521	1122.88	.2837	11.1695	.0056	.2222

Table 2: Response spectrum analysis results.

Model	Spectrum	Midspan displacement	Midspan stress	Midspan moment	Midspan curvature
		mm	MPa	N-m	rad/m
		(in)	(lb/in^2)	(lb-in)	(rad/in)
Biggs		14.2	140.4	5.479×10^3	3.778×10^{-3}

Model	Spectrum	Midspan displacement	Midspan stress	Midspan moment	Midspan curvature
		mm	MPa	N-m	rad/m
		(in)	(lb/in^2)	(lb-in)	(rad/in)
		(.56)	(20,100)	(9.595×10^5)	(9.595×10^{-5})
B23	Displ.	14.0	n/a	5.420×10^3	3.738×10^{-3}
		(.549)		(9.493×10^5)	(9.493×10^{-5})
B21	Vel.	14.0	n/a	5.282×10^3	3.642×10^{-3}
		(.550)		(9.251×10^5)	(9.251×10^{-5})
B23	g	14.0	139.3	5.420×10^3	n/a
		(.550)	(19,937)	(9.493×10^5)	
B21	Acc.	14.0	135.8	5.282×10^3	n/a
		(.551)	(19,443)	(9.251×10^5)	
n/a in the table above means that this variable is not available in the run, because of the beam section definition used.					

Figures

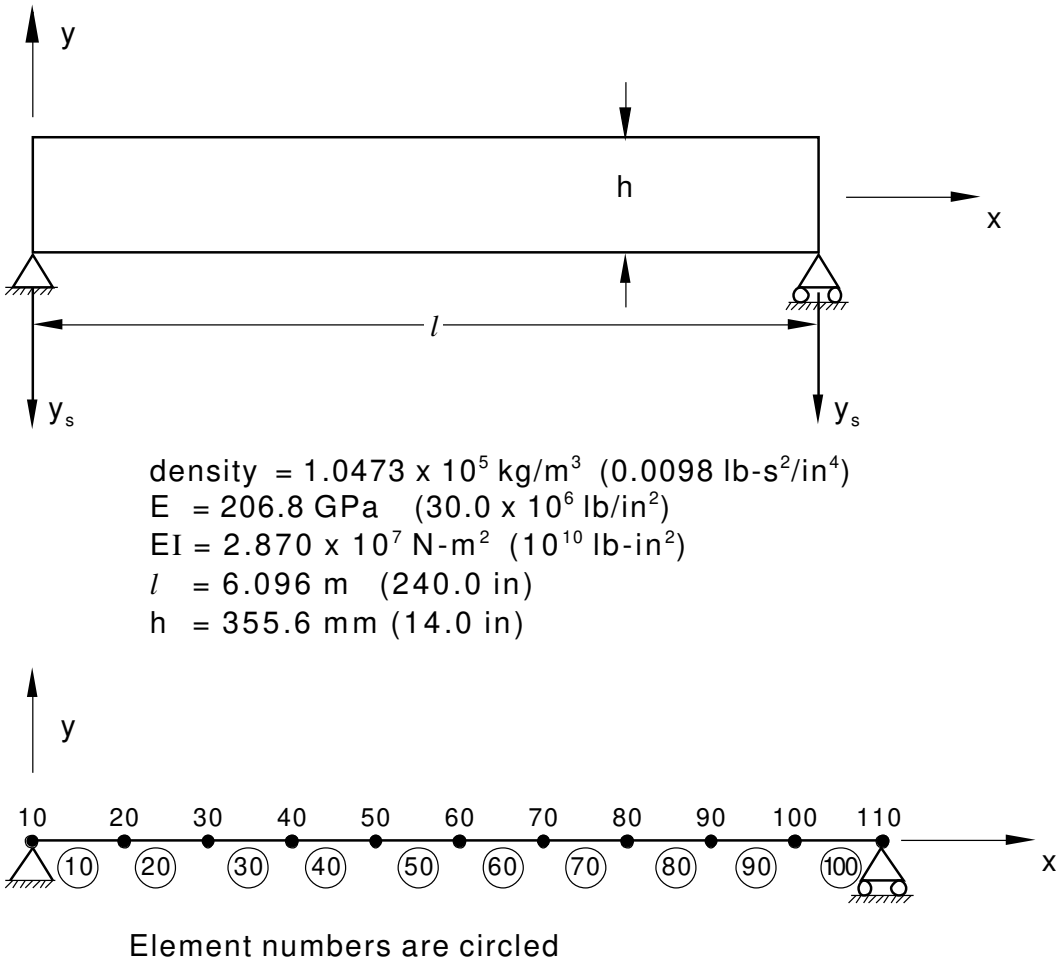


Figure 1: Simply supported beam for response spectrum test.

Linear analysis of a rod under dynamic loading

Products: Abaqus/Standard

References:

- *Modal dynamics*

Overview

This example verifies the linear dynamic procedures in Abaqus by comparing the solutions with exact solutions for a simple system with three degrees of freedom.

Abaqus offers four dynamic analysis procedures for linear problems based on extraction of the eigenmodes of the system: modal dynamic analysis, which provides time history response; response spectrum analysis, in which peak response values are computed for a given response spectrum; steady-state dynamic analysis, which gives the response amplitude and phase when the system is excited continuously with a sinusoidal loading; and random response analysis, which provides statistical measures of a structure's response to nondeterministic loading.

Problem description

The model consists of three truss elements of type T3D2 located along the x-axis, with the y- and z-displacement components restrained, so the problem is one-dimensional. The x-displacement at node 1 is also restrained, leaving three active degrees of freedom. The structure has a total length of 30, cross-sectional area of 2, density of 1/90, and Young's modulus of 5. (All values are given in consistent units.)

Eigenvalue calculations

The first step for all of the linear dynamics procedures is to calculate the eigenvalues and eigenvectors of the system. The mass matrix of element type T3D2 is lumped; therefore, the mass matrix of this three truss system is

$$M = \begin{pmatrix} 1/9 & 0 & 0 & 0 \\ 0 & 2/9 & 0 & 0 \\ 0 & 0 & 2/9 & 0 \\ 0 & 0 & 0 & 1/9 \end{pmatrix}.$$

The stiffness matrix of the system is

$$K = \begin{pmatrix} 1 & -1 & 0 & 0 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & 1 \end{pmatrix}.$$

The three eigenvalues and the corresponding eigenvectors using the default normalization method are given in the following table:

Mode	Eigenvalue	Frequency	Eigenvector magnitude at node			
		(Hz)	1	2	3	4
1	1.2058	0.1748	0	0.5	0.866	1.0

Mode	Eigenvalue	Frequency	Eigenvector magnitude at node			
		(Hz)	1	2	3	4
2	9.0	0.4775	0	1.0	0	–1.0
3	16.794	0.6522	0	0.5	–0.866	1.0

Abaqus also calculates the modal participation factors, Γ_i, the generalized mass, m_i, and the effective mass for each eigenvector (see *Variables associated with the natural modes of a model* for definitions). The values in this case are:

Mode	Participation	Generalized	Effective
	factor	mass	mass
1	1.244	0.333	0.5158
2	0.333	0.333	0.0370
3	0.0893	0.333	0.00266

Alternate normalization

Abaqus allows the eigenvectors to be normalized in one of two ways: such that the largest displacement entry in each eigenvector is unity (default) or such that the generalized mass for each eigenvector is unity. Normalization of eigenvectors is discussed in *Natural Frequency Extraction*. In general, if the default normalization is requested the signs of the eigenvectors obtained using different eigenvalue extraction methods or different platforms are consistent because the largest displacement entry in each eigenvector is scaled to positive unity. For this type of normalization the signs of the eigenvector entries may differ for different methods and different platforms only in the case that the maximum and minimum displacement entries in an eigenvector are of equal magnitude but opposite sign. On the other hand, if mass normalization is requested, the signs of the eigenvectors obtained using different methods or different platforms may vary because, in this case, the eigenvectors are scaled by positive values. The values and signs of the modal participation factors depend on the normalization type and signs of corresponding eigenvectors.

Generalized coordinates for modal dynamic, response spectrum, steady-state, and random response analyses are different depending on the eigenvector normalization. Consequently, for mass normalization the signs of generalized coordinates will change depending on the signs of the eigenvectors. However, the physical values calculated using the summation of the modal values are independent of the eigenvector normalization.

For this example, the corresponding values using mass normalization are given in the following tables:

Mode	Eigenvalue	Frequency	Eigenvector magnitude at node			
		(Hz)	1	2	3	4
1	1.2058	0.1748	0	0.866	1.5	1.732
2	9.0	0.4775	0	–1.732	0	1.732
3	16.794	0.6522	0	–0.866	1.5	–1.732

Mode	Participation	Generalized	Effective
	factor	mass	mass
1	0.718	1.0	0.5158
2	–0.192	1.0	0.0370
3	–0.0516	1.0	0.00266

Modal dynamic analysis

This analysis is performed for three types of systems, described below.

Tip load—damped system

The time history response is obtained for the system when a load of 10 is applied suddenly and held fixed at node 4. Damping of 10% of critical damping in each mode is used. With this excitation the solution for q_i, the amplitude of the i^{th} eigenmode, is

$$\begin{aligned} q_i &= f_i \frac{1}{\omega^2}\left[1-\exp(-\xi\omega t)\left(\cos(\overline{\omega}t)+\frac{\xi\omega}{\overline{\omega}}\sin(\overline{\omega}t)\right)\right], \\ \dot{q}_i &= f_i \frac{1}{\overline{\omega}}\exp(-\xi\omega t)\sin(\overline{\omega}t), \\ \ddot{q}_i &= f_i \frac{1}{\overline{\omega}}\exp(-\xi\omega t)\left(\overline{\omega}\cos(\overline{\omega}t)-\xi\omega\sin(\overline{\omega}t)\right), \end{aligned}$$

where ω is the frequency of vibration, ξ is the fraction of critical damping, $\overline{\omega}=\omega\sqrt{1-\xi^2}$, t is time, and f_i is the projection of the force onto the i^{th} eigenmode. f_i is given by

$$f_i = \frac{1}{m_i}\sum_N \Phi_i^N F^N,$$

where F^N is the force at degree of freedom N ($F^2 = F^3 = 0$, $F^4 = 10$ in this case), Φ_i^N is the component of the i^{th} eigenvector at degree of freedom N, and m_i is the generalized mass for the i^{th} mode.

Base acceleration—damped system

Next, the structure is excited by a constant acceleration of 1.0 at the fixed node (node 1), which is defined using base motion. It can be shown that the equations given above for force excitation can be used for this case when we define the force as

$$f_i = -\Gamma_i \ddot{u}(t),$$

where Γ_i is the modal participation factor (defined in *Variables associated with the natural modes of a model*).

Static preload—undamped system (one mode only)

The modal dynamic step is a linear perturbation procedure and will start from the undeformed configuration by default. However, it is also possible to start the analysis from a deformed configuration by using a static linear perturbation procedure to create the deformed configuration. This step is followed by a modal dynamic step that specifies that the starting position is the linear perturbation solution from the previous step (*General and Perturbation Procedures*). This solution is projected onto the eigenvalues to give the initial modal amplitude:

$$q_i = \frac{1}{m_i}\Phi_i^N M^{NM} u^M \quad \text{(summation over } N \text{ and } M\text{; no summation on } i\text{)}.$$

In general, this projection will preserve all the predeformation only if all of the modes of the system are included in the modal dynamic solution: if only a small number of the modes of the system are used in the modal dynamic analysis—as is the case in practical applications—this projection will only be approximate: that part of the predeformation that is orthogonal to the modes included in the analysis will be lost.

In this analysis an initial displacement of 1.0 is given to node 4 using a boundary condition at this node in a static linear perturbation procedure. The frequency step is then done with the restraint at node 4 removed so that this node is free to vibrate in the subsequent modal dynamic step. (It is essential that the boundary condition be

removed before the eigenvalue problem is solved for the natural modes of the system. Otherwise, incorrect modes—with the boundary condition still in place—will be obtained.) Only one mode is used, so some part of the static response is lost in the projection onto this mode.

At the beginning of the modal dynamic step that carries over initial conditions, Abaqus calculates the initial values of the modal amplitude, using the equation given above, as $q_1 = 0.8293$ for displacement normalization and $q_1 = 0.4779$ for mass normalization. With no damping the response will, therefore, be

$$q_1 = 0.8293 \cos \omega t$$

for displacement normalization and

$$q_1 = 0.4779 \cos \omega t$$

for mass normalization.

Response spectrum analysis

The displacement response spectra shown in *Figure 1* are used in the next analysis. Spectra are defined in the figure for no damping and for 10% of critical damping in each mode. In this example 2% of critical damping is used so that the logarithmic interpolation gives a magnitude of 1.7411 for the maximum displacement for each mode. The analysis is done for two cases: absolute summation of the contributions from each mode and the square root of the sum of the squares (SRSS) summation. Since frequencies are well separated in this case, the use of the ten-percent summation method will give results that are identical to the SRSS method, the complete quadratic combination response will differ only by a small amount from SRSS (because of very small cross-correlation factors between the modes), and the Naval Research Laboratory summation method will calculate results that are very close to the absolute summation. For a comparison of all five summation rules, see *Response spectra of a three-dimensional frame building*. Absolute summation means that the peak displacement response is estimated as

$$u^k = \sum_{i=1}^{\text{modes}} \left| \Phi_i^k q_i^{\max} \right|,$$

where u^k is the displacement at degree of freedom k, Φ_i^k is the i^{th} eigenmode in degree of freedom k, $q_i^{\max}$ is the maximum value for the amplitude in the i^{th} mode, and q_i is found from the appropriate spectrum definition S given in the input. In this case S is represented by displacement spectrum S^D, applied in the global x-direction. SRSS summation estimates the peak displacement response as

$$u^k = \sqrt{\sum_{i=1}^{\text{modes}} \left(\Phi_i^k q_i^{\max} \right)^2}.$$

Steady-state analysis

The steady-state analysis procedure is verified by exciting the model over a range of frequencies. A load of the form

$$F = F_0 \exp(i\Omega t),$$

where Ω is the forcing frequency and $F_0 = 5$, is applied to node 4 in the x-direction.

Two kinds of damping are available for this type of analysis. One is modal damping, which defines the damping term for a mode as

$$c_i = 2\xi_i \omega_i,$$

where ξ_i is the fraction of critical damping. The other is structural damping, for which the damping force is defined as

$$c_j \dot{q}_j = i s_j \omega_j{}^2 m_j q_j,$$

where $i = \sqrt{-1}$ and s_j is the structural damping factor.

Abaqus provides output as the response amplitude, $|q_i|$, and phase angle, ψ_i, for the i^{th} mode. For this example, with only the real loads applied, the exact solution—with both modal and structural damping present—is

$$|q_i| = f_{i0}/m_i \sqrt{\left(\omega_i{}^2 - \Omega^2\right)^2 + \omega_i{}^2 (s_i \omega_i + 2\xi_i \Omega)^2}.$$

and

$$\psi_i = \arctan\left(\frac{\omega_i (s_i \omega_i + 2\xi_i \Omega)}{\omega_i^2 - \Omega^2}\right),$$

where $f_{i0} = \sum_N \Phi_i^N F_0^N$ is the amplitude of the forcing function, F_0, projected onto the i^{th} mode.
The input file *rodlindynamic_ssdynamics.inp* requests a steady-state dynamic analysis for the forcing frequency range from 0.01 to 10 cycles/time. All three mode shapes are extracted with a frequency step and are used throughout the steady-state analysis, as indicated by the modal damping definition, where the damping value is defined to be 10% of critical damping in each mode.

Random response analysis

The same rod model with structural damping present is now exposed to nondeterministic loading. The case we consider is uncorrelated white noise applied to all nodes. The exact solution for the cross-spectral density matrix of the modal amplitudes (the generalized coordinates) as a function of frequency, $S^q_{\alpha\beta}(\Omega)$, for continuously distributed white noise is

$$S^q_{\alpha\beta}(\Omega) = H_\alpha(\Omega) H^*_\beta(\Omega) \int\int \Phi_\alpha(\mathbf{x_1}) \Phi_\beta(\mathbf{x_2}) S^F(\mathbf{x_1}, \mathbf{x_2}, \Omega) \, d\mathbf{x_1} \, d\mathbf{x_2},$$

where

$$H_\alpha(\Omega)^{-1} = m_\alpha \left(\omega_\alpha^2 - \Omega^2 + i\omega_\alpha{}^2 s_\alpha\right)$$

is the complex frequency response function for mode α, with m_α the generalized mass for the mode, ω_α the frequency of the mode, and s_α the structural damping used with the mode; H^*_α is the complex conjugate of H_α; and $S^F(\mathbf{x_1}, \mathbf{x_2}, \Omega)$ is the cross-spectral density matrix of the external loading. Abaqus assumes that the integrated projection of the cross-spectral density matrix onto the eigenmodes can be expressed as a matrix between the loaded nodal degrees of freedom projected onto the eigenmodes, so

$$\int\int \Phi_\alpha(\mathbf{x_1}) \Phi_\beta(\mathbf{x_2}) S^F(\mathbf{x_1}, \mathbf{x_2}, \Omega) \, d\mathbf{x_1} \, d\mathbf{x_2} \simeq \Phi^N_\alpha \Phi^M_\beta S^F_{NM}(\Omega).$$

S^F_{NM} is defined by applying nodal loads, F^I_N (where *N* refers to a degree of freedom in the model and *I* refers to the load case number) and giving a matrix of scaling factors, Ψ^{IJ}_{NM}, and corresponding frequency functions, $P^J(\Omega)$, for each load case. Here *J* refers to the matrix of scaling factors Ψ^{IJ}_{NM} by which to scale P^J in load case *I*. S^F_{NM} is then defined as

$$S^F_{NM}(\Omega) = \sum_J P^J(\Omega) \sum_I \Psi^{IJ}_{NM} F^I_N F^I_M \qquad \text{for } N < M,$$

$$S^F_{NM}(\Omega) = \left[S^F_{MN}(\Omega)\right]^* \qquad \text{for } N > M,$$

$$S^F_{NN}(\Omega) = \Re\left(\sum_J P^J(\Omega) \sum_I \Psi^{IJ}_{NN} F^I_N F^I_N\right).$$

In this case we need only one load case, $I = 1$, and one frequency function and associated matrix of scaling factors, $J = 1$. (See *Random response to jet noise excitation* for a problem in which several frequency functions and scaling factor matrices are needed to define the cross-spectral density matrix of the loading.) Since white noise is assumed to be uncorrelated, Ψ^{11}_{NM} is defined as a diagonal matrix: $\Psi^{11}_{NM} = 0$ for $N \neq M$. Uncorrelated loadings are specified using a correlation definition, where Ψ^{IJ}_{NM} is defined. We choose a unit magnitude for the scaling factors so that Ψ^{11}_{NM} becomes a unit matrix. Since the diagonal terms of the cross-spectral density matrix are the power spectral density functions of the loading, the cross-spectral density matrix will be a real diagonal matrix. Therefore, imaginary frequency functions and scaling factors need not be considered here. As a result, the power spectral definition is a reference power spectral density function (rather than a general frequency function), $P^1(\Omega)$, which is scaled by the product of load magnitudes, $F^1_N F^1_M$ (and by Ψ^{11}_{NM}, but Ψ^{11}_{NM} is a unit matrix). We apply loads F^1_N of 10 to each of nodes 2 and 3 and a load of 5 to node 4, corresponding to a unit load distributed continuously along the rod.

At a frequency of 0.1 cycles/time $S^q_{\alpha\beta}$ is, therefore,

$$S^q_{\alpha\beta} = \begin{bmatrix} 1710. & 32.24 - i0.1421 & -16.91 + i7.825 \times 10^{-2} \\ 32.24 + i0.1421 & 15.19 & 1.594 - i3.476 \times 10^{-4} \\ -16.91 - i7.825 \times 10^{-2} & 1.594 + i3.476 \times 10^{-4} & 4.183 \end{bmatrix}.$$

The cross-spectral density matrices for the displacements, velocities, and accelerations of the nodes can be calculated directly from $S^q_{\alpha\beta}$. For example, the cross-spectral density matrix of the displacements is

$$S^u_{NM} = \Phi^N_\alpha S^q_{\alpha\beta} \Phi^M_\beta.$$

Results and discussion

The results of the various calculations for this example are given in tables in the text below. In all cases the Abaqus results agree with the exact solution.

Modal dynamic analysis: tip load–damped system

Results for the three generalized coordinates in this model at times of 0.1, 0.2, and 0.3 for displacement normalization are:

Time	Mode	q_i	$\dot{q}_i$	$\ddot{q}_i$
0.1	1	0.149	2.96	29.2
	2	–0.146	–2.87	–27.0
	3	0.144	2.80	25.3
0.2	1	0.589	5.82	28.0
	2	–0.560	–5.32	–21.8

Time	Mode	q_i	$\dot{q}_i$	$\ddot{q}_i$
	3	0.538	4.94	16.9
0.3	1	1.31	8.55	26.5
	2	–1.19	–7.17	–15.0
	3	1.10	6.12	6.53

The results for mass normalization are:

Time	Mode	q_i	$\dot{q}_i$	$\ddot{q}_i$
0.1	1	0.0859	1.71	16.8
	2	0.0843	1.66	15.6
	3	–0.0831	–1.62	–14.6
0.2	1	0.340	3.36	16.2
	2	0.323	3.07	12.6
	3	–0.311	–2.85	–9.77
0.3	1	0.756	4.94	15.3
	2	0.687	4.14	8.65
	3	–0.635	–3.53	–3.77

The signs of the generalized coordinates may change depending on the sign of the corresponding eigenvectors.

Physical values are obtained by summation of the modal values at each time:

$$a = \sum_{i=1}^{\text{modes}} q_i a_i,$$

where a is a physical quantity and a_i is the value of this quantity computed for mode i.

For the stress and strain in the elements in this structure this gives the following results:

Time	Element	Stress	Strain
0.1	1	0.000206	0.000041
	2	0.001870	0.000374
	3	0.2173	0.043452
0.2	1	0.001797	0.000359
	2	0.020377	0.004076
	3	0.8210	0.1642
0.3	1	0.007051	0.001410
	2	0.083857	0.016771
	3	1.708	0.3416

The values for nodal variables are calculated using the same summation method, so the displacements, velocities, accelerations, and reaction forces are:

Time	Node	Displacement	Velocity	Acceleration	Reaction force
0.1	1	0.0	0.0	0.0	–0.000412

Time	Node	Displacement	Velocity	Acceleration	Reaction force
	2	0.00041	0.0126	0.2632	
	3	0.00415	0.1394	3.363	
	4	0.4387	8.630	81.42	
0.2	1	0.0	0.0	0.0	–0.003595
	2	0.00359	0.0583	0.6979	
	3	0.0444	0.7689	9.602	
	4	1.686	16.08	66.71	
0.3	1	0.0	0.0	0.0	–0.014102
	2	0.01410	0.1660	1.547	
	3	0.1818	2.110	17.33	
	4	3.598	21.84	48.06	

Modal dynamic analysis: tip load–undamped system

Time history response is also obtained for an undamped system. The results for the generalized coordinates for displacement normalization are:

Time	Mode	q_i	$\dot{q}_i$	$\ddot{q}_i$
0.1	1	0.150	2.99	29.8
	2	–0.149	–2.96	–28.7
	3	0.148	2.92	27.5
0.2	1	0.598	5.95	29.3
	2	–0.582	–5.65	–24.8
	3	0.567	5.35	20.5
0.3	1	1.34	8.84	28.4
	2	–1.26	–7.83	–18.6
	3	1.19	6.90	10.0

The results for mass normalization are:

Time	Mode	q_i	$\dot{q}_i$	$\ddot{q}_i$
0.1	1	0.0865	1.73	17.2
	2	0.0860	1.71	16.6
	3	–0.0854	–1.68	–15.9
0.2	1	0.345	3.44	16.9
	2	0.336	3.26	14.3
	3	–0.327	–3.09	–11.8
0.3	1	0.772	5.10	16.4
	2	0.728	4.52	10.8
	3	–0.686	–3.98	–5.80

Modal dynamic analysis: base acceleration–damped system

With the modal damping set to 10% of critical damping for all three modes, the responses of the three generalized coordinates to this base acceleration for displacement normalization are:

Time	Mode	q_i	$\dot{q}_i$	$\ddot{q}_i$
0.1	1	−0.00617	−0.123	−1.21
	2	−0.00162	−0.0319	−0.30
	3	−0.00043	−0.00834	−0.0753
0.2	1	−0.02442	−0.241	−1.16
	2	−0.00622	−0.05912	−0.242
	3	−0.00160	−0.01469	−0.0504
0.3	1	−0.05428	−0.355	−1.10
	2	−0.01322	−0.07966	−0.167
	3	−0.003272	−0.01821	−0.01944

The results for mass normalization are:

Time	Mode	q_i	$\dot{q}_i$	$\ddot{q}_i$
0.1	1	−0.00356	−0.0709	−0.698
	2	0.000936	0.0184	0.173
	3	0.000247	0.00481	0.0435
0.2	1	−0.0140	−0.139	−0.671
	2	0.00359	0.0341	0.140
	3	0.000924	0.00848	0.0291
0.3	1	−0.0313	−0.205	−0.636
	2	0.00763	0.0460	0.0962
	3	0.00189	0.0105	0.0112

These responses give the following results for the nodal variables. (In this table, as in the Abaqus output, the displacement, velocity, and acceleration values are normally given relative to the base motion: total displacement values are also given.)

Time	Node	Displacement	Velocity	Acceleration	Total displacement
0.1	1	0.0	0.0	0.0	0.0050000
	2	−0.00492	−0.0974	−0.9421	0.0000797
	3	−0.00497	−0.0991	−0.9824	0.0000290
	4	−0.00498	−0.0993	−0.9853	0.0000244
0.2	1	0.0	0.0	0.0	0.0200000
	2	−0.01923	−0.1872	−0.8478	0.0007692
	3	−0.01976	−0.1964	−0.9623	0.0002365
	4	−0.01980	−0.1970	−0.9700	0.0001965
0.3	1	0.0	0.0	0.0	0.0450000

Time	Node	Displacement	Velocity	Acceleration	Total displacement
	2	–0.04200	–0.2661	–0.7266	0.0030027
	3	–0.04417	–0.2914	–0.9364	0.0008259
	4	–0.04433	–0.2932	–0.9536	0.0006692

Modal dynamic analysis: static preload–undamped system (one mode only)

The results for the modal amplitude for displacement normalization are:

Time	q_1
0.06	0.828
1.43	0.0004
2.86	–0.829
5.32	0.750
5.72	0.829

The results for mass normalization are:

Time	q_1
0.06	0.478
1.43	0.0003
2.86	–0.479
5.32	0.433
5.72	0.479

Response spectrum analysis

The response spectrum analysis gives the following results for the nodal displacements:

Node	Displacement	Displacement
	(abs. summation)	(SSRS)
1	0.0	0.0
2	1.741	1.231
3	2.010	1.881
4	2.902	2.248

Steady-state analysis

The results for the amplitude and phase angle of the generalized displacements (the modal amplitudes, q_i) for displacement normalization are shown in the table below:

Forcing frequency	Mode	Amplitude, q_i	Phase, ψ
0.01	1	12.48	–0.66
	2	1.667	179.8

Forcing frequency	Mode	Amplitude, q_i	Phase, ψ
	3	0.8934	–0.1757
0.175	1	62.2	–90.0
	2	1.918	175.2
	3	0.9607	–3.304
0.477	1	1.918	–175.2
	2	8.333	90.0
	3	1.835	–17.51

The results for mass normalization are shown in the table below:

Forcing frequency	Mode	Amplitude, q_i	Phase, ψ
0.01	1	7.705	–0.66
	2	0.9627	179.8
	3	0.5158	–0.1757
0.175	1	35.91	–90.0
	2	1.107	175.2
	3	0.5546	–3.304
0.477	1	1.107	–175.2
	2	4.811	90.0
	3	1.060	–17.51

Stress and strain amplitudes for element 1 and the amplitude of the reaction force at node 1 are:

Forcing frequency	Stress	Strain	Reaction force, node 1
0.01	2.51	0.5019	5.019
0.175	15.50	3.10	31.00
0.477	3.988	0.7977	7.977

Output of the phase angle can be requested for any variable. For example, the stress in element 1 at a forcing frequency of 0.477 cycles/time has an amplitude of 3.988 and a phase angle of 90.58° with respect to the forcing function.

A third step is included in which the steady-state solution is calculated with 10% structural damping. At low frequencies ($\Omega < 0.01$) the results for this step do not differ very much from the results using modal damping, but significant differences appear at forcing frequencies in the range of the eigenfrequencies of the structure.

Random response analysis

Abaqus provides the diagonal terms of the cross-spectral density matrix; i.e., the power spectral densities. The power spectral densities of displacement, velocity, and acceleration at 0.1 cycles/time are:

Node	Displacement	Velocity	Acceleration
2	469.1	185.2	73.12
3	1311.	517.6	204.3
4	1628.	642.7	253.7

Root mean square values are calculated as the square roots of the variances, which are the integrals of the power spectral densities up to the frequency of interest. The root mean square values of the nodal variables at 1 Hz are:

Node	RMS value of displacement	RMS value of velocity	RMS value of acceleration
2	81.51	134.0	353.7
3	129.3	158.0	334.2
4	152.9	207.4	485.6

The power spectral densities and the RMS values of stress and strain throughout the model are likewise calculated from $S^q_{\alpha\beta}$ and the modal vectors of the stress and strain.

Input files

rodlindynamic_modal_subeigen.inp	**MODAL DYNAMIC* analysis with a damping value of 0.1 and the structure excited by a point load applied at node 4.
rodlindynamic_respspec_subeigen.inp	**RESPONSE SPECTRUM* analysis.
rodlindynamic_ssdyn_subeigen.inp	**STEADY STATE DYNAMICS* analysis with modal and structural damping for the given range of forcing frequencies.
rodlindynamic_random_subeigen.inp	**RANDOM RESPONSE* analysis with structural damping.
rodlindynamic_correlationdata.inp	Contains the correlation definition for use inrodlindynamic_random.inp.
rodlindynamic_composite.inp	**MODAL DYNAMIC* analysis with composite modal damping.
rodlindynamic_modal_nodamp.inp	**MODAL DYNAMIC* analysis with damping set to 0.
rodlindynamic_modal_base.inp	**MODAL DYNAMIC* analysis with **BASE MOTION*.
rodlindynamic_modal_preload.inp	**MODAL DYNAMIC* analysis in which the excitation is caused by a static preloading of the structure, with the load removed suddenly to cause the dynamic event.
rodlindynamic_modal_base2.inp	**MODAL DYNAMIC* analysis with **BASE MOTION* using the secondary base motion and composite modal damping.
rodlindynamic_modal.inp	Same as rodlindynamic_modal_subeigen.inp, except that it uses the Lanczos solver and the eigenvectors are normalized with respect to the generalized mass.
rodlindynamic_respspect.inp	Same as rodlindynamic_respspec_subeigen.inp, except that it uses the Lanczos solver and the eigenvectors are normalized with respect to the generalized mass.
rodlindynamic_ssdyn_massnorm.inp	Same as rodlindynamic_ssdyn_subeigen.inp, except that the eigenvectors are normalized with respect to the generalized mass. The subspace iteration solver is used.
rodlindynamic_random_massnorm.inp	Same as rodlindynamic_random_subeigen.inp, except that the eigenvectors are normalized with respect to the generalized mass. The subspace iteration solver is used.

rodlindynamic_ssdynamics.inp	Same as rodlindynamic_ssdyn_subeigen.inp, except that the Lanczos solver is used. The eigenvectors are normalized with respect to the maximum displacement.
rodlindynamic_random.inp	Same as rodlindynamic_random_subeigen.inp, except that the Lanczos solver is used. The eigenvectors are normalized with respect to the maximum displacement.

Figures

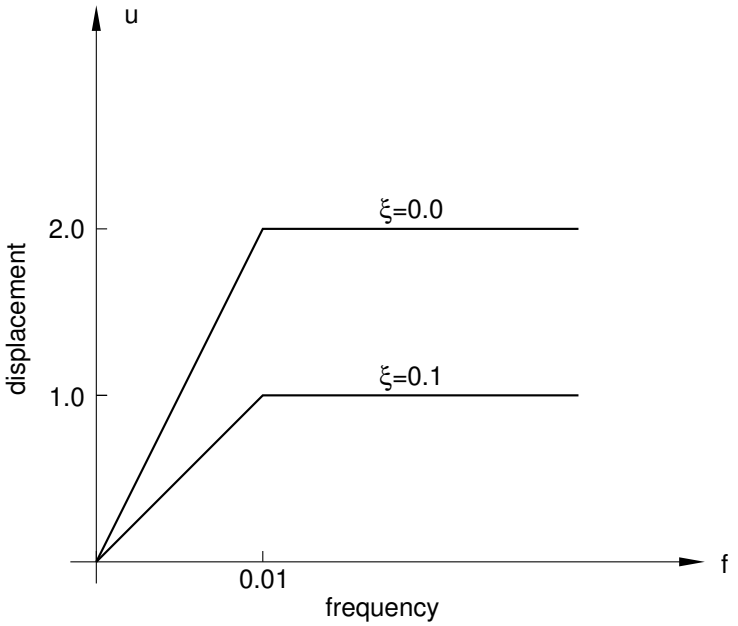


Figure 1: Displacement response spectra.

Random response to jet noise excitation

Products: Abaqus/Standard

Overview

This example illustrates and verifies the random response analysis capability in Abaqus with a simple beam example that was originally studied by Olson (1972).
The problem in this example is a five-span continuous beam exposed to jet noise. The example is solved using the built-in moving noise loading option and, as an illustration, with user subroutines *UPSD* and *UCORR*.

Problem description

Except for the assumption that time is measured in seconds (so that frequencies are expressed in Hz), no specific set of units is used in this example. The units are assumed to be consistent.

The structure is a five-span straight beam, simply supported at its ends and at the four intermediate supports (*Figure 1*). Each span has unit length. The beam is excited in bending. It has unit bending stiffness and mass of 1×10^{-4} per unit length.

Each span is modeled with four elements of type B23 (cubic beam in a plane), as shown in *Figure 1*. No mesh convergence studies have been performed; however, the first 15 natural frequencies agree quite well with the exact values given by Olson, so we assume that the mesh is reasonable. The response analysis is based on 1% of critical damping in each mode, as used by Olson.

Loading

Jet noise is an acoustic excitation that applies random pressure loading to the surface of a structure. The pressure at a point is assumed to have a power spectral density $S_0(\Omega)$, where Ω is frequency, measured in cycles per time. For this case, following Olson, we assume that the excitation is white noise ($S_0 = 1.0$ at all frequencies) and that the acoustic waves are traveling along the structure with a velocity c_0 (where c_0 is taken to be 6.0 in this case). The cross-spectral density of the pressure loading between any two points can then be written as

$$S^F(\Omega, \overline{x}) = S_0(\Omega) \exp\left(-i2\pi\Omega\frac{\overline{x}}{c_0}\right),$$

where $\overline{x}$ is the distance between the two points for which S^F is being given. This type of loading is specified as a moving noise load in the correlation definition for random response analysis. The correlation definition acts between loads applied at the nodes of the model. In this case, since the elements are all of equal length, a load of magnitude 0.25 is applied equally to all nodes to simulate the pressure loading. Thus, $\overline{x}$ has only the discrete values of the distance between any combination of two nodes. Olson points out that this approximation reduces the accuracy of the results unless a rather fine mesh is used. However, the mesh used here provides results that agree well with those of Olson up to relatively high frequencies, suggesting that the approximation is not too coarse.

Loading via user subroutines

For purposes of illustration we also show input data for the case where we apply the loading via user subroutines *UPSD* and *UCORR*. These subroutines allow the user to define a different frequency dependence and magnitude for each entry in the cross spectral density matrix. Any number of frequency functions can be used to define the cross spectral density of the loading as

$$S^F_{(N,i)(M,j)}(\Omega) = \sum_J P_J(\Omega) \sum_I \Psi^{IJ}_{(N,i)(M,j)} F^I_{(N,i)} F^I_{(M,j)},$$

where $P_J(\Omega)$ is a complex frequency function defined in the power spectral density definition and referenced in the Jth correlation definition, $\Psi^{IJ}_{(N,i)(M,j)}$ is the corresponding Jth correlation matrix for load case I for degrees of freedom i at node N and j at node M, and $F^I_{(N,i)}$ is the load applied to degree of freedom i at node N in load case I. Since there are 21 nodes in our model and the elements are all of equal length, the construction of $S^F(\Omega, \bar{x})$ can be accomplished as follows. Concentrated nodal loads $F^1_{(N,i)}$ of 0.25 (corresponding to the length of each element) are applied to all of the nodes. In user subroutine *UPSD* we specify 21 complex frequency functions,

$$P_J(\Omega) = S_0 \exp(-i2\pi\Omega(x_N - x_M)/c_0),$$

each for a particular value of $x_N - x_M$, the distance between two nodes N and M. S_0 is equal to 1.0 (white noise) in this case. These frequency functions are referenced in 21 correlation definitions in the random response step.

For each of these options matrices $\Psi^{IJ}_{(N,i)(M,j)}$ are defined in user subroutine *UCORR*, each with unity in the appropriate (N,i) (M,j) positions and zero elsewhere.

Results and discussion

The first 15 natural frequencies agree closely with the exact values given by Olson, suggesting that the mesh is suitable for frequencies up to at least 110 Hz.

The random response results obtained with the two approaches are identical within numerical accuracy. *Figure 2* illustrates the power spectral density of the transverse displacement at node 2. These results, and similar plots for other nodes and for rotations, are in good agreement with those obtained by Olson (1972).

Figure 3 shows the root mean square (RMS) value of the transverse displacement at node 2. Since the higher modes tend to contribute less and less to the response, we expect the RMS values to level off as the frequency increases. As shown in *Table 1*, the RMS values of rotation and transverse displacement at all nodes along the beam are seen to be in good agreement with Olson's results.

Input files

jetnoise_eigen.inp	Eigenvalue extraction step.
jetnoise_restart.inp	Restart run for the random response analysis.
jetnoise_restart_usr_upsd.inp	Restart run for the random response analysis.
jetnoise_restart_usr_upsd.f	User subroutines *UPSD* and *UCORR* used in jetnoise_restart_usr_upsd.inp.

References

- Olson, M. D., “A Consistent Finite Element Method for Random Response Problems,” *Computers and Structures*, vol. 2, 1972.

Tables

Table 1: Root mean square displacements and rotations.

Node	Displacement		Rotation	
	Olson	Abaqus	Olson	Abaqus
1	0.	0.	0.7988	0.8679
2	0.1719	0.1820	0.5101	0.5289
3	0.2274	0.2349	0.2775	0.3867
4	0.1557	0.1656	0.5319	0.5537
5	0.	0.	0.6308	0.6811
6	0.1225	0.1301	0.3436	0.3619
7	0.1534	0.1589	0.2421	0.3230
8	0.1040	0.1123	0.3662	0.3840
9	0.	0.	0.4378	0.4921
10	0.0904	0.0998	0.2819	0.3044
11	0.1176	0.1245	0.2253	0.3050
12	0.0841	0.0932	0.2902	0.3139
13	0.	0.	0.3801	0.4315
14	0.0889	0.0954	0.3308	0.3469
15	0.1360	0.1400	0.2216	0.2877
16	0.1129	0.1188	0.3005	0.3185
17	0.	0.	0.6113	0.6539
18	0.1585	0.1670	0.5652	0.5911
19	0.2391	0.2478	0.2198	0.3042
20	0.1793	0.1884	0.5378	0.5615
21	0.	0.	0.8235	0.8821

Figures

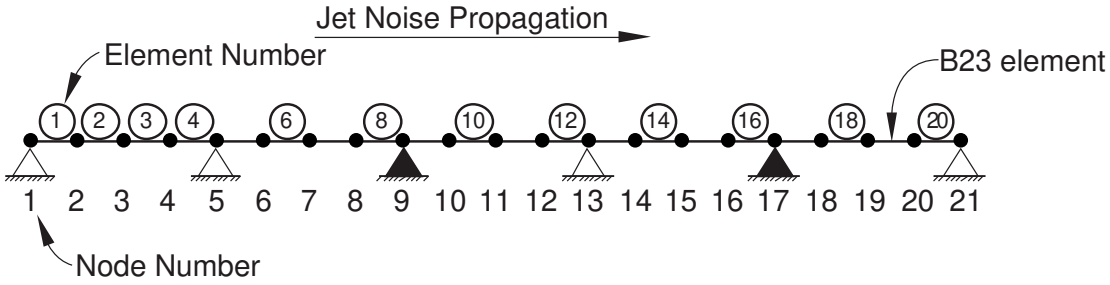


Figure 1: Beam subjected to jet noise.

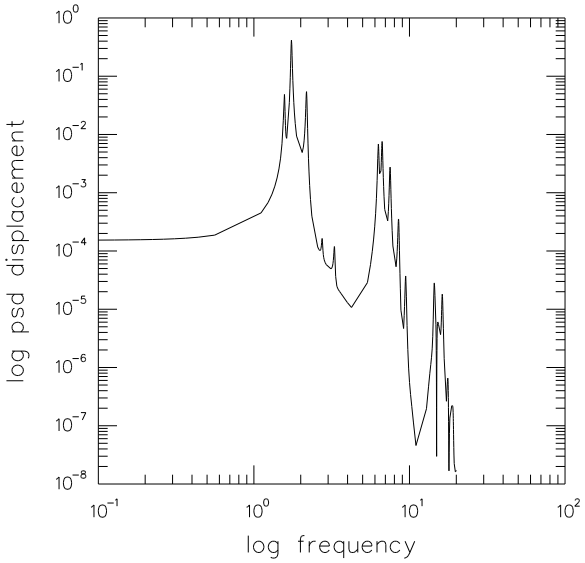


Figure 2: Power spectral density of displacement at node 2.

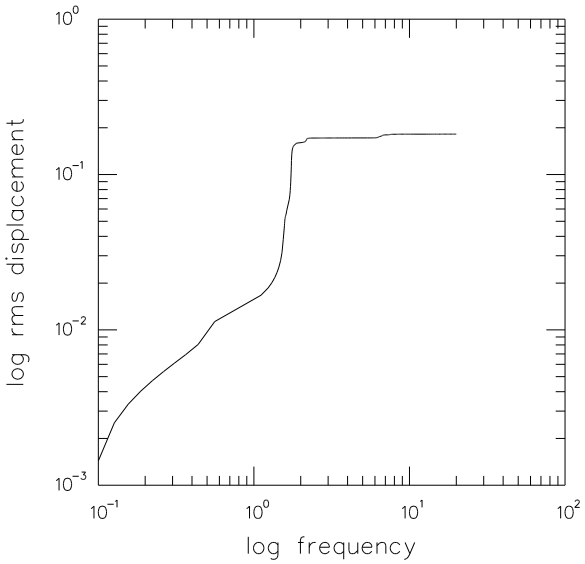


Figure 3: Root mean square of displacement at node 2.

Random response of a cantilever subjected to base motion

Products: Abaqus/Standard

Overview

This example verifies the random response analysis procedure for a case where the structure is excited by base motion. The model used in this example is a steel cantilever attached to a stiff vibrating structure that subjects it to prescribed transverse acceleration with a given power spectral density. The results are compared with the analysis of Johnsen and Dey (1978).

Problem description

The cantilever is 1 m long and has a square cross-section of 100 mm × 100 mm. The steel has a Young's modulus of 210 GPa, a Poisson's ratio of 0.3, and its density is 8000 kg/m^3. A 10% structural damping factor is used for all the modes. The mesh has 10 elements of type B23 (cubic beam in a plane) and is shown in *Figure 1*.

Loading

The base motion is applied as an acceleration with the power spectral density function shown in *Figure 2*. Since the excitation is in one degree of freedom only, the correlation matrix is a unit matrix.

Results and discussion

The first 10 natural frequencies agree within 0.1% with those given by Johnsen and Dey (1978). The power spectral density of the displacement at the tip of the cantilever is shown in *Figure 3*. For all nodes the values at the eigenfrequencies compare well with the results of Johnsen and Dey. For nodes close to the built-in end of the cantilever, discrepancies appear at higher frequencies. These differences are attributed to the use of a beam-column element in Abaqus (element type B23) that uses the axial strain as an internal degree of freedom in the element, so some axial modes appear at higher frequencies. The element used by Johnsen and Dey does not have these same modes. The differences are not important because they could be eliminated by using a finer mesh if the high frequency response close to the base of the cantilever must be predicted accurately.

Input files

randomrespcantilever.inp	Input data for running the random response analysis.

References

- Johnsen, T. L, and S. S. Dey, *ASKA Part II – Linear Dynamic Analysis, Random Response*, ASKA UM 218, ISD, University of Stuttgart, 1978.

Figures

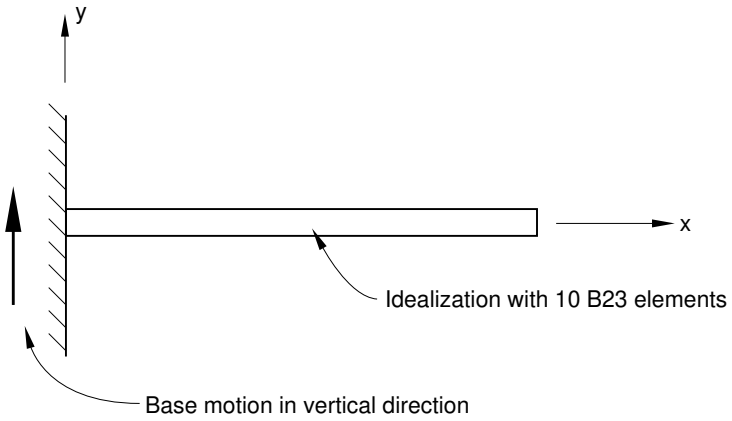


Figure 1: Steel cantilever subjected to base motion.

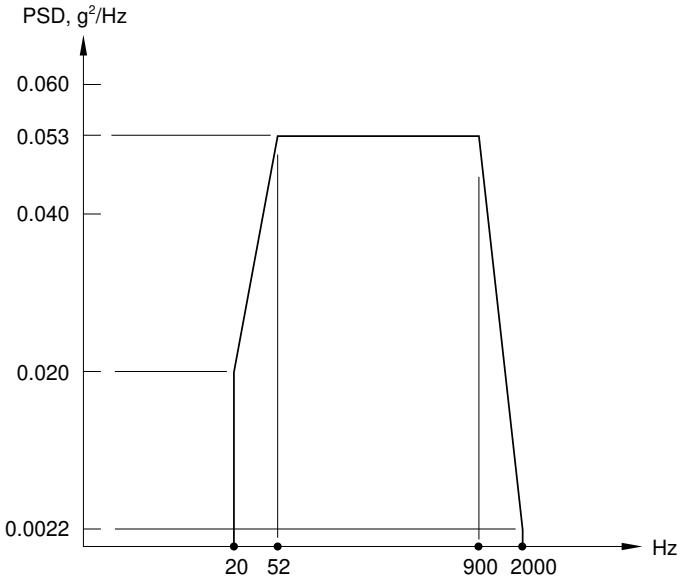


Figure 2: Base acceleration power spectral density.

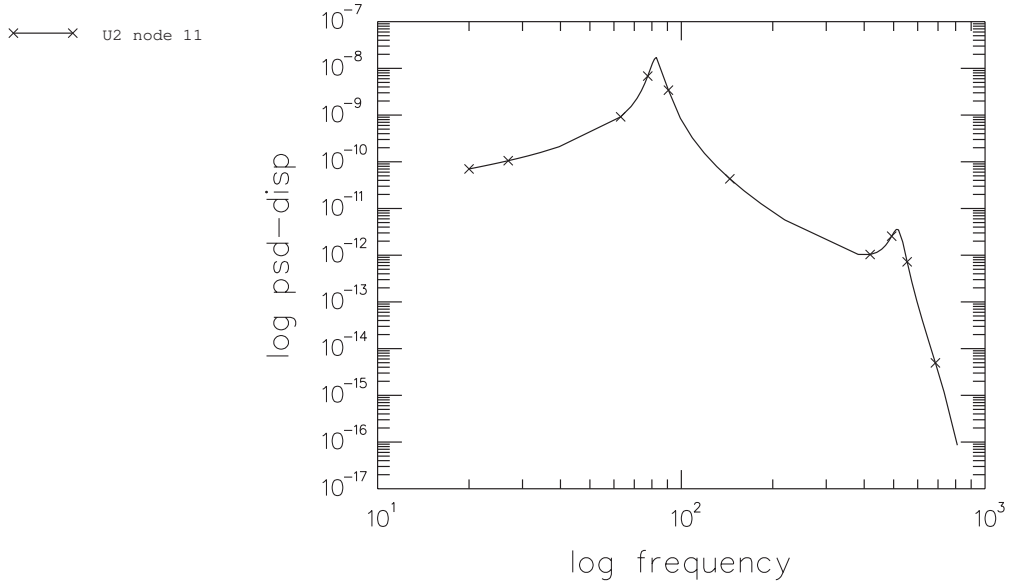


Figure 3: Power spectral density of the displacement response at the tip of the cantilever.

Double cantilever subjected to multiple base motions

Products: Abaqus/Standard

Overview

This example shows the effect of multiple base motions on a double cantilever.

Enforced motion is often the primary source of excitation in vibration problems. Examples include vehicle suspensions responding to road irregularities and civil structures subjected to seismic ground motions. In these problems the forcing functions are given by the time history of motions at the supports of the structure.

For modal-based analyses using the modal dynamic and the steady-state dynamic procedures, the support motions are simulated by prescribed excitations, called base motions. Base motions are applied by constraining groups of degrees of freedom into one or more "bases" by specifying a base name on the boundary condition in the frequency step. Multiple bases are required if base motions cannot be described by a single set of rigid body motions.

Degrees of freedom that are constrained without being assigned to a named base make up the *primary* base. This is the only base if the motion can be described by a single set of rigid body motions. Constrained degrees of freedom that are associated with named boundary conditions make up the *secondary* base or bases. Abaqus uses the modal participation method for primary base motions and the "large mass" method for secondary base motions (see *Base motions in modal-based procedures*).

Problem description

As an illustration we consider a simple model of a bridge whose supports are subjected to seismic excitations. Two cases are analyzed: the first considers identical base excitations at the supports, and the second assumes that the left-end support is subjected to the excitation with a time shift. The forcing function corresponds to the same El Centro N–S earthquake records used in *Analysis of a cantilever subject to earthquake motion*. The model is a double cantilever lying horizontally along the x-direction (see *Figure 1*), analyzed with 20 equal-sized B23 elements. A two-second event is studied. Analyses are also performed using the implicit dynamic procedure to provide a basis for comparison of the results obtained by the modal dynamic procedure. The time incrementation scheme is the same as that used in *Analysis of a cantilever subject to earthquake motion*.

Three models with different "base" organizations are used for the modal dynamic analyses. In the first model the base motion is invoked without referring to a secondary base. In the second model the base motion is invoked without referring to a secondary base for the right-end support—the primary base—and with the base name for the left-end support, which is declared in the frequency step as a secondary base, named `NODE21`. Finally, a model with two secondary bases and no primary base is used. In this model both supports are declared in the frequency step as secondary bases, named `NODE1` and `NODE21`.

Results and discussion

The base acceleration record is shown in *Figure 2*. *Figure 3* shows the (total) displacement response of the midspan node to the unshifted and shifted base excitations predicted using the dynamic procedure.

The modal dynamic analysis results agree closely with the implicit dynamic solution. *Figure 4* shows the total displacement response of the midspan obtained with beam models using different dynamic procedures, as well as various base motions for modal dynamic analyses, for the case with no time shift. *Figure 5* shows the same responses for the case where the motion at the left-end support is delayed by 0.25 second. As a verification exercise the modal dynamic analysis that invokes multiple base motions is repeated using a shell mesh with 10 S8R elements and gives the same results as obtained by the beam model. These solutions are obtained based on superposition of the first six nonzero eigenmodes of the structure. For models with secondary bases the additional

low-frequency modes resulting from the unconstrained degrees of freedom at the bases must be taken into account. Abaqus automatically increases the number of eigenfrequencies to keep the number of relevant frequencies constant. However, the eigenmode range used for the modal damping must be extended by the user. The boundary conditions that make up the primary base normally suppress all rigid body motion. If they do not, as occurs in the third model where the primary base is absent, a suitable (negative) shift point must be used in the frequency procedure to avoid numerical problems.

In modal dynamics the default output gives motion relative to the primary base. The sum of this relative motion and the base motion of the primary base yields the total motion. In the absence of primary base motions the relative and total motions are identical. The plots shown in *Figure 4* and *Figure 5* have been requested appropriately to give total values in all cases.

Input files

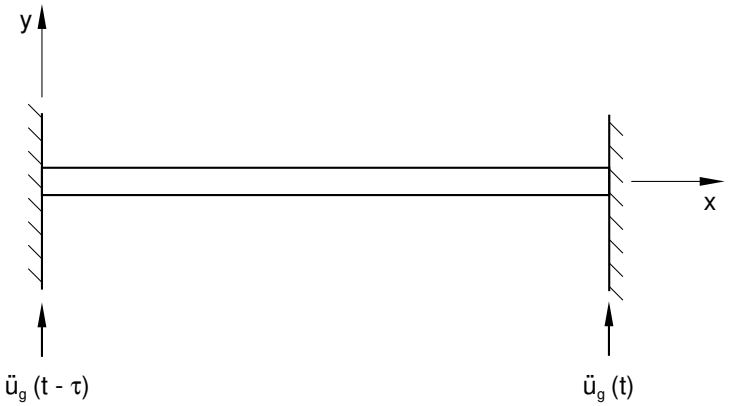
Modal dynamic analysis

multibasemotion_modal1.inp	The only base is the primary base.
multibasemotion_modal12.inp	Both primary and secondary bases. The base acceleration record for the left-end support has a time shift of 0.25 second.
multibasemotion_modal2.inp	Only secondary bases. The base acceleration record for the left-end support has a time shift of 0.25 second.

Other verification problems

multibasemotion_noshift.inp	Same as multibasemotion_modal12.inp but without the time shift.
multibasemotion_direct.inp	Direct integration analysis.
multibasemotion_directdelay.inp	Direct integration analysis. The base motion has a time delay of 0.25 second.
multibasemotion_modal2_noshift.inp	Only secondary bases, no time shift.
multibasemotion_s8r_modal.inp	Modal dynamic analysis using a shell mesh with 10 S8R elements.
multibasemotion_s8r_shift.inp	Modal dynamic analysis using a shell mesh with 10 S8R elements. The base motion has a time delay of 0.25 second.
multibasemotion_quake.inp	Earthquake record.
multibasemotion_quake_shift.inp	Earthquake record, time delay of 0.25 second.
multibasemotion_s4r5.inp	Modal dynamic analysis using a shell mesh with 10 S4R5 elements. The **TRANSFORM* option is also exercised in this analysis.
multibasemotion_s4r5_shift.inp	Modal dynamic analysis using a shell mesh with 10 S4R5 elements. The **TRANSFORM* option is also exercised in this analysis. The base motion has a time delay of 0.25 second.

Figures



Beam cross-section: height 50.8 mm (2.0 in)
width 25.4 mm (1.0 in)

Young's modulus: 206.8 GPa (3.0×10^7 lb/in^2)

Density: 7780 kg/m^3 (0.00078 lb-s^2/in^4)

Figure 1: Double cantilever beam.

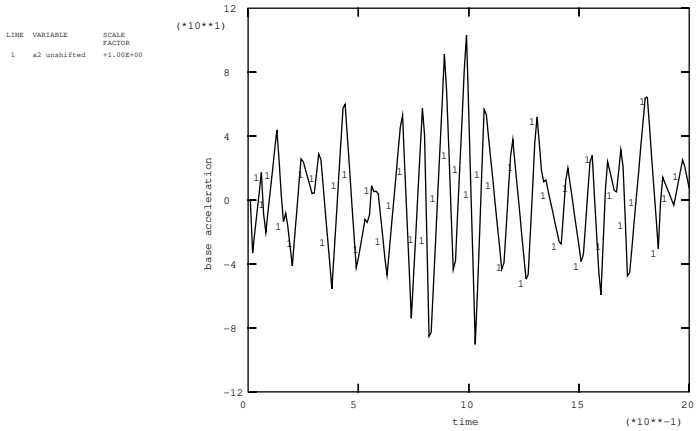


Figure 2: Base acceleration record.

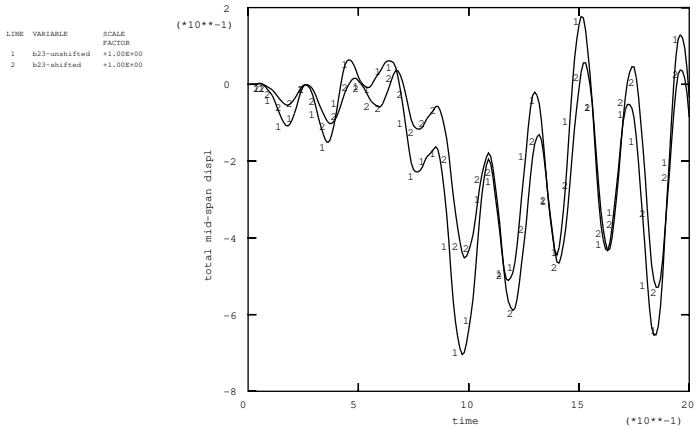


Figure 3: Total transverse displacement response of beam midspan to base excitations with and without the 0.25 second time shift.

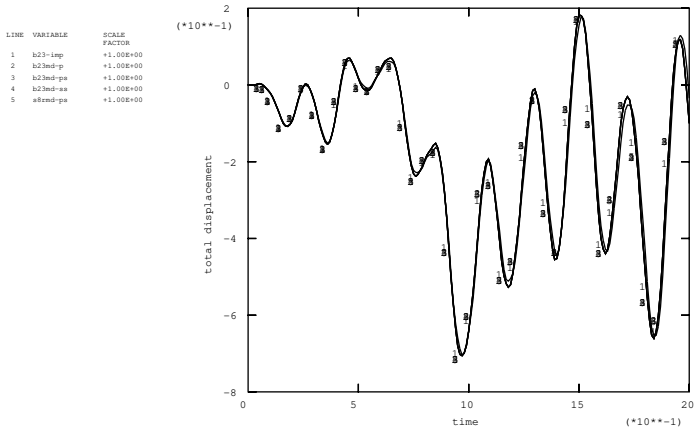


Figure 4: Total transverse displacement responses of beam midspan to base motions without time shift.

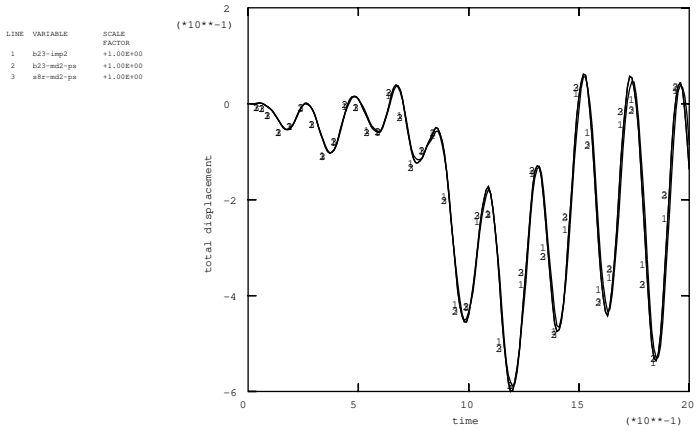


Figure 5: Total transverse displacement responses of beam midspan to base motions with the 0.25 second time shift.

Analysis of a cantilever subject to earthquake motion

Products: Abaqus/Standard

Overview

This example demonstrates the use of Abaqus in a seismic analysis where the forcing function is given by the time history of acceleration at an anchor point of the structure.
In this example three types of analyses are illustrated: modal dynamics in the time domain, direct time integration, and response spectrum analysis.

In problems such as this one the modal dynamic procedure is the analysis method of choice because it is computationally inexpensive and it is very accurate (provided that enough modes are extracted), since the integration of the modal amplitudes (the "generalized coordinates") is exact. Direct time integration is also used in this problem to illustrate the accuracy of the time integration operator. Response spectrum analyses, based on spectra calculated from the same earthquake record, are also performed and compared with the exact solution.

Examples are also included to illustrate the use of baseline correction. Baseline correction is used to modify the acceleration record by adding a correction to the acceleration record to minimize the mean square velocity over the time of the event. The correction to the acceleration record is piecewise quadratic in time. In this example the analyses are first performed without baseline correction. Two different baseline corrections are then applied, and the results with and without baseline correction are compared.

Problem description

The structure chosen for this example is a free standing, vertical cantilevered column. The dimensions of the column, shown in *Figure 1*, have been chosen so that the column will have a number of frequencies in the range that is usually of interest in the seismic analysis of structures. This range of interest is commonly taken to be up to 33 Hz, the rationale being that the spectral content of the acceleration record will not excite the higher frequency modes of the structure.

To choose a mesh for which the geometric discretization error is negligible, it is important to ensure that the modes corresponding to eigenvalues up to 33 Hz are modeled accurately using the chosen mesh. *Table 1* shows that a model with 10 elements of type B23 (cubic beam in a plane) gives the first six frequencies (up to about 60 Hz) very accurately, with an error of about 0.1% in the fourth mode (25 Hz). This mesh is, therefore, chosen for the analysis.

Time domain analysis

The seismic analysis is performed using the El Centro N-S acceleration history, which is discretized every 0.01 second. An exact benchmark solution is readily obtained by integrating the eigenvalues and eigenvectors of the structure exactly in time over the first 10 seconds of the acceleration input (see, for example, Hurty and Rubinstein, 1964). (This solution is calculated using the Fortran program contained in the file *cantilever_exact.f*.) The number of modes included in this solution has been found by trial, which has shown that using the six lowest modes (up to 61.9 Hz) gives displacements that are accurate to 0.01%. The higher modes have a negligible effect since the earthquake acceleration input is discretized every 0.01 second.

The modal dynamic analysis is identical to the benchmark solution, except for the spatial discretization, since Abaqus integrates the response of the generalized coordinates exactly for inputs that vary linearly during each time increment.

The direct integration analysis is run using the Hilber-Hughes operator with the operator parameter α set to 0.0, which gives the standard trapezoidal rule. This operator is unconditionally stable and has no numerical damping,

but it exhibits a phase error. *Figure 2*, taken from Hilber et al. (1977), shows how this error grows with the ratio of the time step to the oscillator period. Automatic time stepping would normally be chosen, with Abaqus adjusting the time step to achieve the accuracy specified by the choice of the half-increment residual tolerance in the dynamic procedure. In this case we choose instead to use a fixed time step of 0.01 seconds so that the integration errors are readily illustrated.

For both of these time history analyses the base motion is read from the given acceleration history by using an amplitude curve. For direct integration this base motion is prescribed by using a boundary condition, whereas for the modal dynamic procedure it must be given using base motion.

Response spectrum analysis

Response spectrum analysis provides an inexpensive technique for estimating the peak (linear) response of a structure to a dynamic excitation. The spectrum is first constructed for the given acceleration history by integrating the equation of motion of a damped single degree of freedom system. This provides the maximum displacement, velocity, and acceleration response of such a system. Plots of these responses as functions of the natural frequency of the single degree of freedom system are known as displacement, velocity, and acceleration spectra. The maximum response of the structure is then estimated from these spectra by the response spectrum procedure.

Results and discussion

The results for each analysis are discussed below.

Modal dynamic

The modal dynamic analysis results agree exactly with the benchmark solution, since the linear variation of the inputs over each increment results in exact integration.

Dynamic

The dynamic analysis is run for 10 sec (1000 increments). The displacement, velocity, and acceleration at the top of the column are plotted as functions of time using the Visualization module in Abaqus/CAE. The response quantities in these plots are all measured relative to the base of the structure. The Fortran program that calculates the benchmark solution writes its results to various files, so that Abaqus/CAE can be used to plot the benchmark solution on the same graphs as the Abaqus results.

Figure 3 shows the displacement of the top of the column, relative to its base, for the first 2 seconds of response. The approximate and benchmark solutions agree well on this plot. The relative velocity and acceleration of the column top for the first 2 sec are shown in *Figure 4* and *Figure 5*. The difference between the benchmark and the approximate solutions is now more apparent, especially in the acceleration trace. *Figure 6* through *Figure 8* show the response from 8 to 10 seconds after the start of the event. The higher mode content of the approximate solution now shows a significant phase error in the relative displacement trace (*Figure 6*), and the acceleration solution is quite seriously in error.

The source of this phase error is the phase error inherent in the time integration operator, shown in *Figure 2*. It is a simple matter to estimate the error and its effect on each mode after 10 seconds of response. Such a calculation is summarized in *Table 2*. As shown in the table, with the 0.01 second time step chosen, the error in the first and second modes is about 4% and 46%, respectively; for all other modes the errors are well in excess of 100%, so the effect shown in *Figure 6* is entirely predictable: with a 0.01 second time step, the errors are very large for all but the first mode response. It is interesting to observe that, to achieve less than a 5% phase error after 10 seconds in Mode 6, the phase error would have to be less than 8×10^{-5} per cycle, implying a time step that is not larger than about 10^{-5} seconds.

Figure 9 shows the displacement of the undamped system during the entire 10-second analysis. Even without damping, the first mode response so dominates the solution that the predicted tip displacement response after 10 seconds is not grossly in error. In reality, there will always be some damping; if the structure is undergoing

large motion, it is likely that the damping will be enough to remove most of the response above the second mode in this period of time. The common design approach is to incorporate all dissipation of energy as equivalent linear viscous damping—typically assumed to be a certain fraction (2–6%) of critical damping in each mode when modal dynamics is used. This approach cannot be used in direct integration analysis since the modes are not extracted. Instead, damping can be used to introduce mass and stiffness proportional damping into models that are integrated directly. We have not used this option here. In calculations for extremely large input motions this linearized approach is usually replaced with a nonlinear analysis in which the damping mechanisms are modeled explicitly.

Response spectrum

The spectra for response spectrum analysis are obtained by integrating 10 seconds of the acceleration record using the Fortran program given in the file *cantilever_spectradata.f*. By varying the frequency range and the damping values, several different response spectra can be obtained. *Figure 10* through *Figure 12* depict the displacement and velocity spectra for the frequency ranges 0.1–30 Hz and 0.01–5.0 Hz with no damping and with damping chosen as 2% and 4% of critical damping.

The response spectrum procedure estimates the response at each frequency either as the sum of the absolute values of the modal responses (the absolute summation, or ABS, method) or as the square root of the sum of the squares of the modal responses (SRSS method). The absolute summation method is always conservative, in the sense that it overpredicts the response.

Since the natural modes of the cantilever are well separated in this case, the ten-percent summation method will give the same results as the SRSS method. The complete quadratic combination method will also give these same results, and the Naval Research Laboratory method will give values close to those provided by ABS summation. A comparison of these methods in a more complex case is provided in *Response spectra of a three-dimensional frame building*.

We can compare the response estimates provided by the response spectrum analysis with the exact values by examining the predictions of response quantities at the top of the column. The exact peak displacement is 59.2 mm (2.33 in), and the peak velocity is 0.508 m/sec (20 in/sec). The comparison is based on the response spectrum values obtained with the assumption of no damping and is shown in *Table 3*. We see that, using the displacement spectrum, the ABS summation method overestimates the peak displacement by 14% and the peak velocity by 28%, whereas the SRSS method underestimates the peak displacement by 3% and the peak velocity by 22%. Using the velocity spectrum, the ABS method overestimates the peak displacement by 20% and the peak velocity by 27%, whereas the SRSS method overpredicts the peak displacement by 4% and underpredicts the peak velocity by 22%. Despite these rather large errors, the method is commonly used because of its simplicity and ready application to design cases. The response spectra results found in *Table 3* can be obtained by executing the Fortran program given in the file *cantilever_spectradata.f* and then running the Abaqus input given in *cantilever_responsespec.inp*. To obtain results using the ABS summation method, two response spectrum steps must be added to sum the directional excitation components algebraically and to sum the absolute values of the responses in each natural mode.

Baseline correction

Baseline correction adds a piecewise quadratic correction to the acceleration record to minimize the mean square velocity of the motion. This correction will change the displacement quite substantially (the corrected base displacement will tend to zero at the end of the motion), but the change in the acceleration record will not be very large. As a result, the relative displacement between the tip and the base of the cantilever will be affected very little, but the absolute displacement will change substantially if significant baseline correction is added.

Baseline correction can be applied in Abaqus as a piecewise quadratic correction through the time domain. In this example we apply two corrections: one done for the entire period of time (here 25 sec) and one done using three intervals: 0.0–8.3 sec, 8.3–16.7 sec, and 16.7–25 sec. *Figure 13* shows the total (not relative) displacement of the tip of the cantilever with and without these corrections, and *Figure 14* shows the base displacement with and without baseline correction. *Figure 14* was produced by running all three analyses for 25 seconds (the

duration of the acceleration record) and then plotting the total displacement of the base of the cantilever. The effect of the correction on displacement is clear from *Figure 14*; as more intervals are used for the correction, the base displacement at the end of the analysis tends more toward zero.

Input files

Direct integration analysis

cantilever_dynamic.inp	Used to obtain the first 5 seconds of response with direct integration (**DYNAMIC*).
cantilever_quakedata.inp	The earthquake record, read as file `QUAKE.AMP`.
cantilever_restart.inp	Restarts the direct integration analysis and completes the 10 seconds of response.

Modal dynamic analysis

cantilever_modal_10s.inp	**MODAL DYNAMIC* analysis.
cantilever_modal_25s.inp	Identical to cantilever_modal_10s.inp, except that the analysis time is 25 seconds instead of 10 seconds.

Response spectrum analysis

cantilever_responsespec.inp	Response spectrum analysis. To run this file, the user must first run cantilever_spectradata.f.
cantilever_spectradata.f	Fortran program that generates displacement and velocity spectra. This program integrates the equation of motion of a single degree of freedom system at given frequencies and, thus, creates the needed spectrum definitions. The response spectra are written to ASCII files `QUAKE`*x*`.DIS`, `QUAKE`*x*`.VEL`, `QUAKER`*x*`.DIS`, and `QUAKER`*x*`.VEL`, where the extension indicates displacement `(.DIS)` or velocity `(.VEL)` data. The *x* in the file name indicates the damping percentage, and the R indicates reduced frequency range results.

Benchmark solution

cantilever_exact.f	Fortran program that will create an “exact” solution to the problem. The program works by first calculating the eigenmodes of the cantilever and then calculating the response using modal superposition. The results file from the direct integration analysis is also read by this program to obtain the response relative to the base of the structure; hence, cantilever_dynamic.inp and cantilever_restart.inp must be run before the Fortran program will work properly. The Fortran program then creates a new results file containing the relative response at the top of the cantilever as degree of freedom 1 (Abaqus solution obtained from input file cantilever_restart.inp) and the results from the modal superposition analysis as degree of freedom 2 (exact solution). Furthermore, ASCII files `exactdisp`, `exactvelo`, and `exactaccl` are also generated. These contain the displacement, velocity, and acceleration results, respectively.

Eigenvalue analysis

cantilever_eig_b21.inp	B21 elements.
cantilever_eig_b21_fine.inp	B21 elements, fine mesh.
cantilever_eig_b23.inp	B23 elements.
cantilever_eig_b23_fine.inp	B23 elements, fine mesh.

Modal dynamic analysis with baseline correction

cantilever_baseline1_10s.inp	Modal dynamic analysis with baseline correction over one interval.
cantilever_baseline3_10s.inp	Modal dynamic analysis with baseline correction over three intervals.
cantilever_baseline1_25s.inp	Identical to cantilever_baseline1_10s.inp, except that the analysis time is 25 seconds instead of 10 seconds.
cantilever_baseline3_25s.inp	Identical to cantilever_baseline3_10s.inp, except that the analysis time is 25 seconds instead of 10 seconds.
cantilever_baseline_eqspaced.inp	Variation of file cantilever_baseline1_10s.inp that uses equally spaced amplitude data to define the earthquake record.
cantilever_data.inp	Equally spaced amplitude data used in the file above.

Other verification problems

cantilever_s8r_dynamic.inp	Tests direct integration using S8R elements.
cantilever_s8r_modal.inp	Tests modal dynamics using S8R elements.

References

- Hilber, H. M., T. J. R. Hughes, and R. L. Taylor, “Improved Numerical Dissipation of Time Integration Algorithms in Structural Dynamics,” *Earthquake Engineering and Structural Dynamics*, vol. 5, pp. 283–292, 1977.
- Hurty, W. C., and M. F. Rubinstein, Dynamics of Structures, Prentice-Hall, New Jersey, 1964.

Tables

Table 1: Natural frequencies in Hertz.

Mode	Exact	Finite Element				
		B23 elements			B21 elements	
		10	20	50	10	20
1	.729	.729	.729	.729	.726	.728
2	4.567	4.567	4.567	4.567	4.519	4.554
3	12.787	12.791	12.787	12.787	12.623	12.740
4	25.058	25.082	25.059	25.058	24.774	24.961
5	41.423	41.529	41.430	41.423	41.222	41.288
6	61.878	62.220	61.901	61.879	62.328	61.767
7	86.425	87.317	86.488	86.426	88.453	86.472
8	115.060	117.040	115.210	115.070	119.210	115.510
9	147.790	151.470	148.100	147.800	151.380	141.010
10	148.610	169.170	169.170	169.170	168.990	169.120

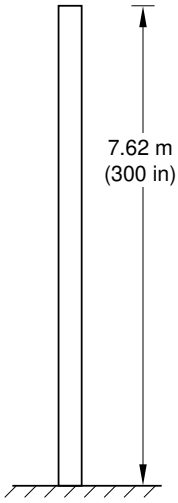
Table 2: Estimated phase errors after 10 seconds of response, using a time step of 0.01 second.

Mode	Period, T, (seconds)	$\Delta t/T$	Phase error per period	Phase error after 10 seconds
1	1.37	.007	.005%	3.6%
2	0.219	.046	.01%	46%
3	0.078	.128	.05%	600%
4	0.040	.251	.17%	4000%
5	0.024	.414	.4%	16000%
6	0.016	.619	.6%	37000%

Table 3: Estimates of maximum displacement and velocity at the top of the column provided by response spectrum analysis.

	Displacement	Velocity
Exact value	59.2 mm (2.33 in)	0.508 m/sec (20 in/sec)
Displacement spectrum:		
ABS summation	67.3 mm (2.65 in)	0.641 m/sec (25.22 in/sec)
SRSS summation	57.1 mm (2.25 in)	0.392 m/sec (15.45 in/sec)
Velocity spectrum:		
ABS summation	70.9 mm (2.79 in)	0.642 m/sec (25.28 in/sec)
SRSS summation	61.0 mm (2.40 in)	0.395 m/sec (15.57 in/sec)

Figures



Beam cross-section: height 50.8 mm (2.0 in)
width 25.4 mm (1.0 in)

Young's modulus: 206.8 GPa (3.0×10^7 lb/in^2)

Density: 7780 kg/m^3 (0.00728 lb-s^2/in^4)

Figure 1: Vertical cantilever beam.

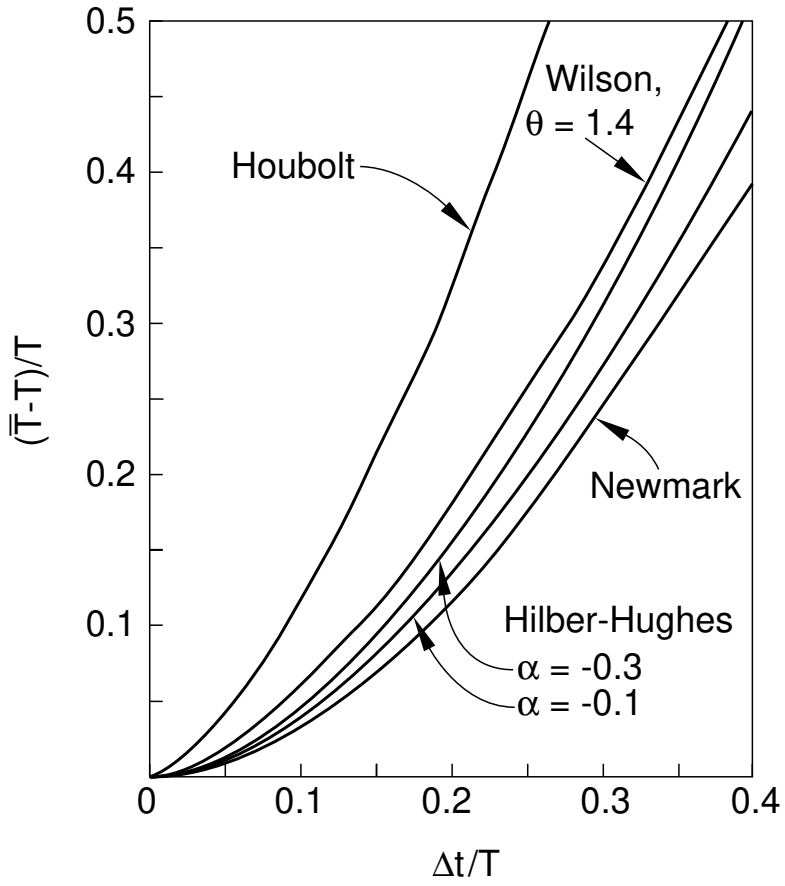


Figure 2: Relative period error (phase error) versus $\Delta t/T$ for Hilber-Hughes, Wilson, Newmark, and Houbolt methods (from Hilber et al., 1977).

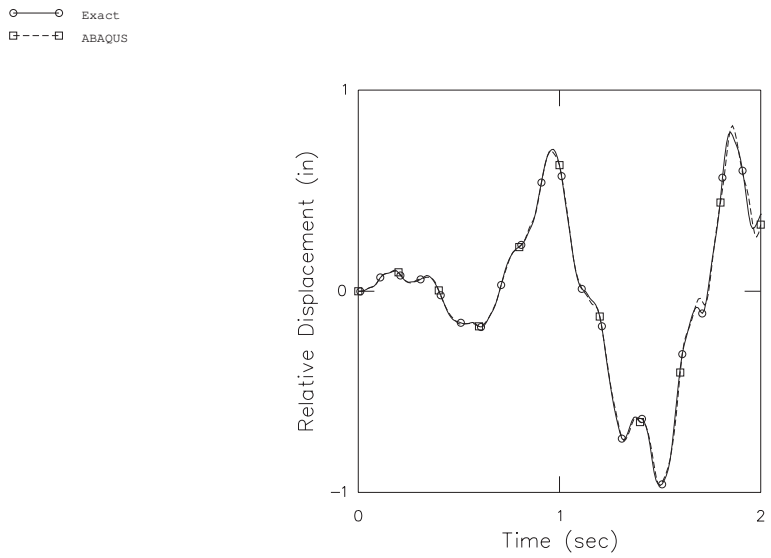


Figure 3: Relative displacement at the top of the column for the first 2 seconds of response.

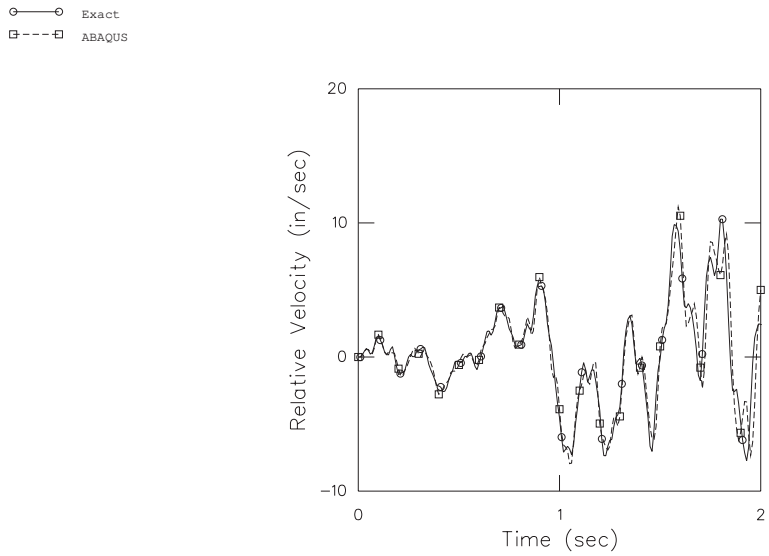


Figure 4: Relative velocity at the top of the column for the first 2 seconds of response.

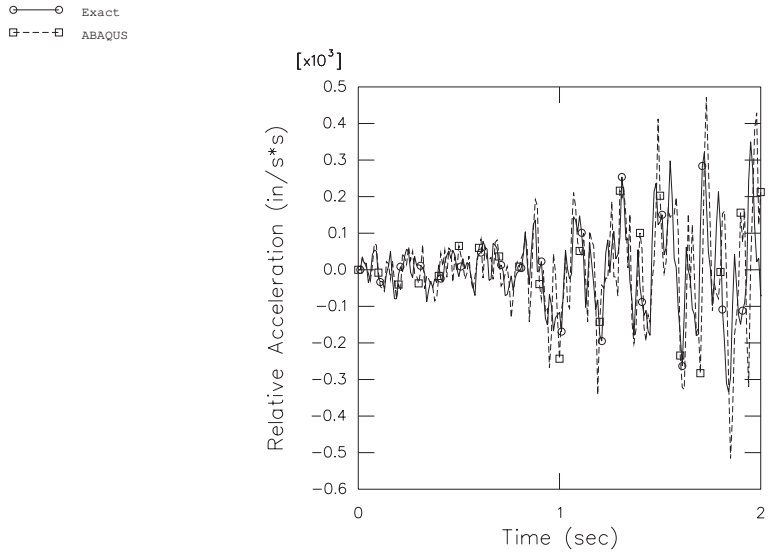


Figure 5: Relative acceleration at the top of the column for the first 2 seconds of response.

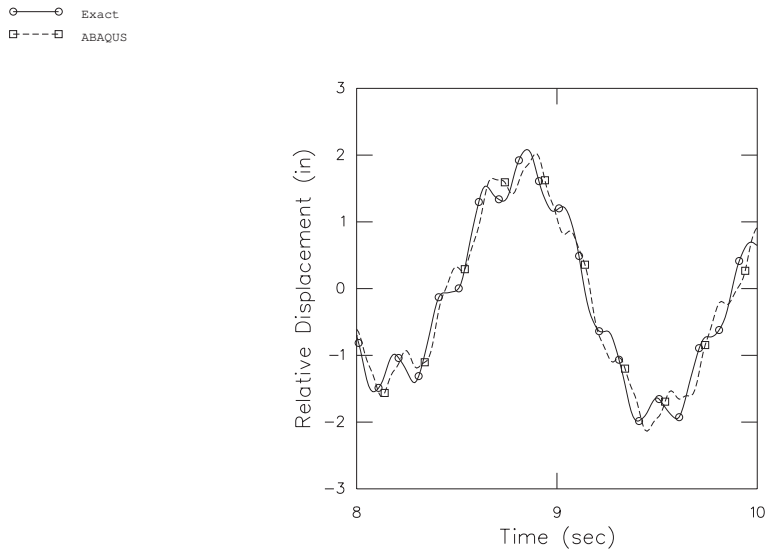


Figure 6: Relative displacement at the top of the column for the period 8–10 seconds.

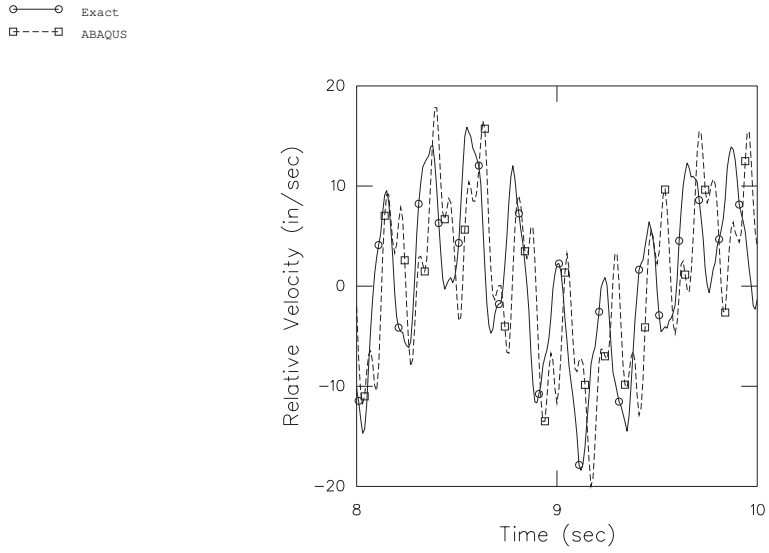


Figure 7: Relative velocity at the top of the column for the time period 8–10 seconds.

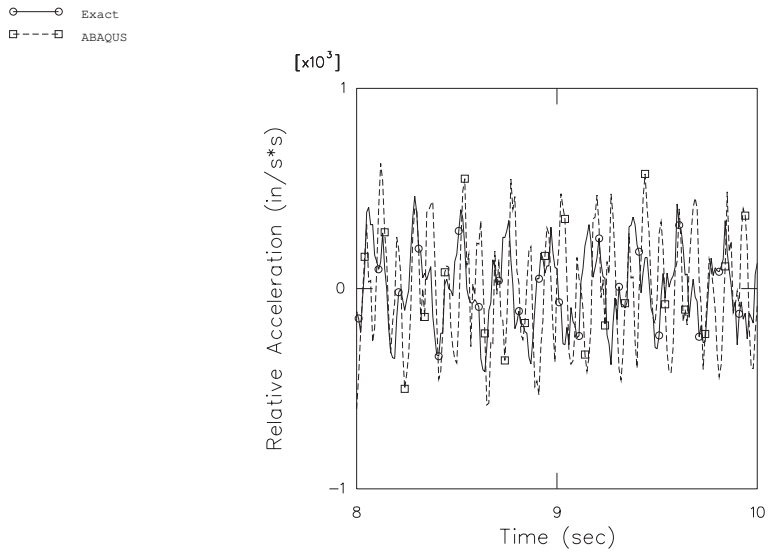


Figure 8: Relative acceleration at the top of the column for the time period 8–10 seconds.

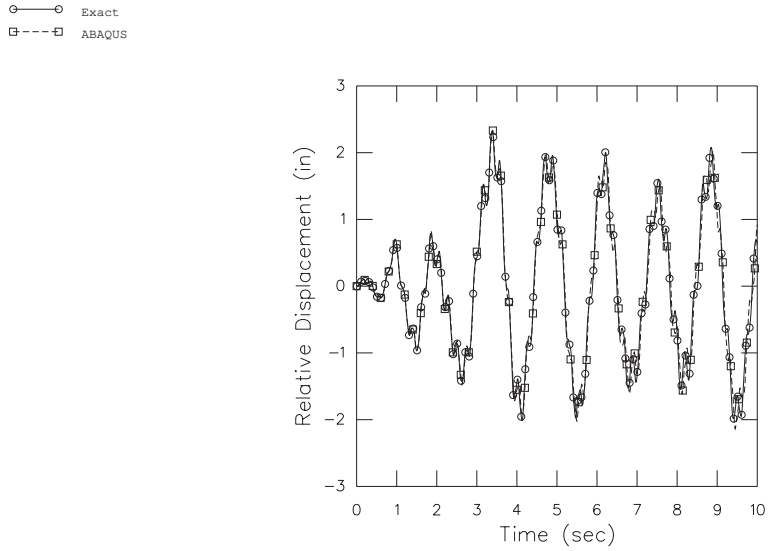


Figure 9: Relative displacement at the top of the column for the time period 1–10 seconds.

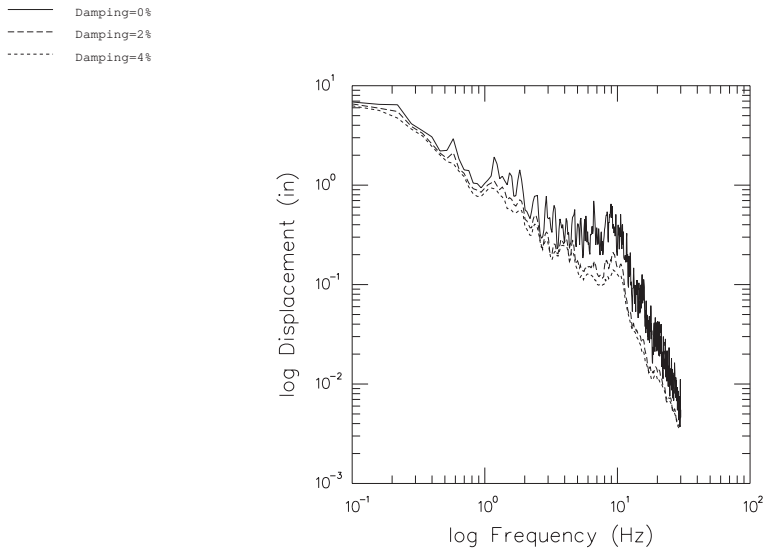


Figure 10: Displacement spectra for the frequency range 0.1–30 Hz.

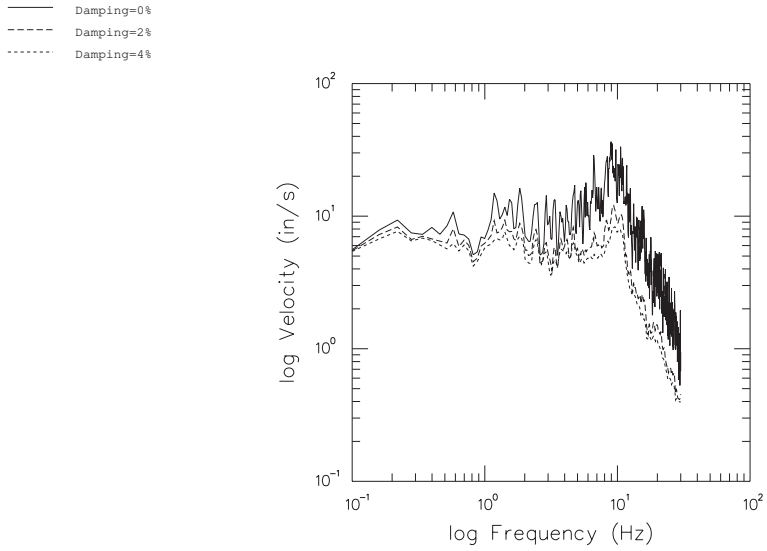


Figure 11: Velocity spectra for the frequency range 0.1–30 Hz.

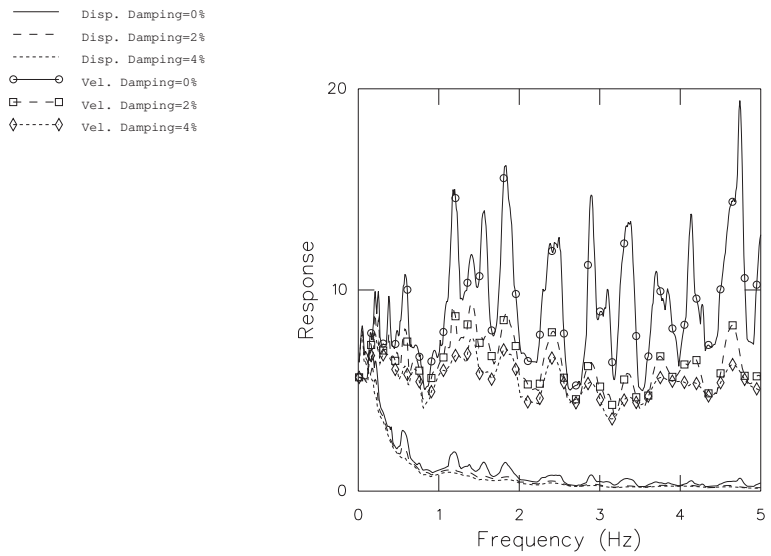


Figure 12: Displacement and velocity spectra for the frequency range 0.01–5.0 Hz.

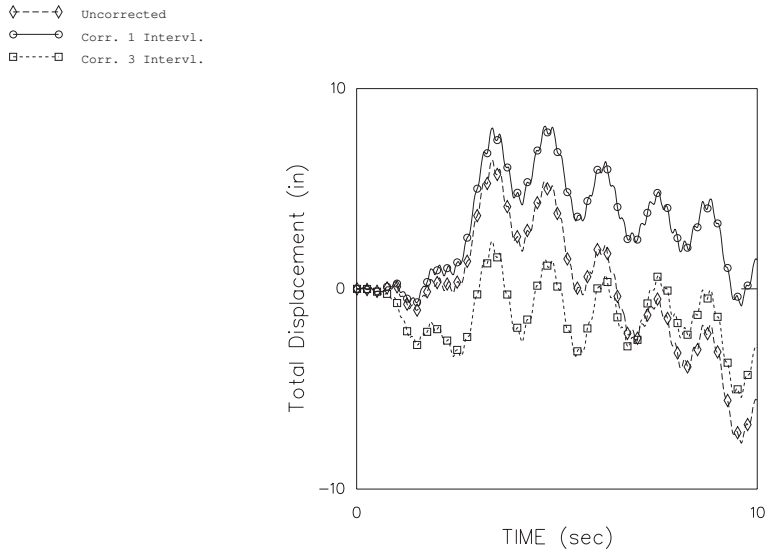


Figure 13: Absolute displacement of the cantilever's tip with and without baseline correction.

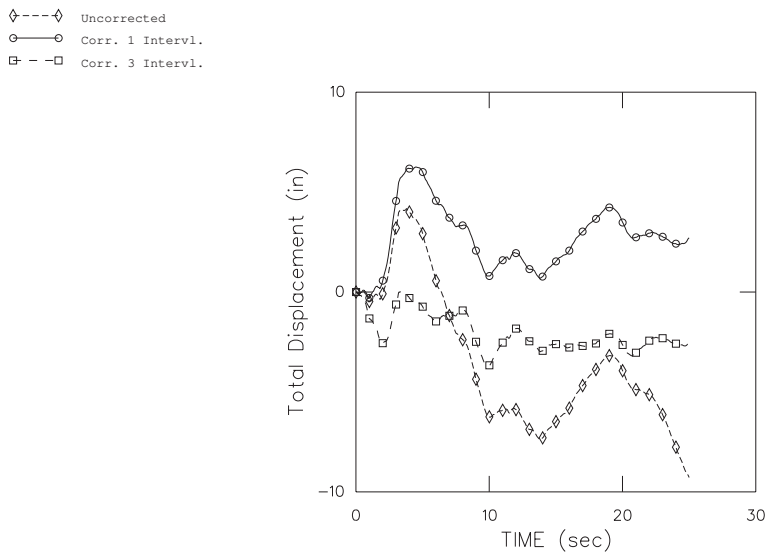


Figure 14: Base displacement with and without baseline correction.

Residual modes for modal response analysis

Products: Abaqus/Standard

Overview

This example illustrates the use of the residual modes capability in Abaqus and verifies the solution accuracy.

In many modal response analyses, simplifying a model by reducing the number of degrees of freedom or extracting only a small subset of eigenmodes is often a common practice. These assumptions are beneficial for cost reductions, but the accuracy of the modal solution may suffer. To improve solution accuracy, the method of residual modes (see *Natural Frequency Extraction*) can be employed. This method extracts an additional set of modes based on loading conditions to help correct for errors introduced by mode truncation. Residual modes are orthogonal to retained eigenmodes and to each other and are computed when specified in the frequency step.

Problem description

A simple multiple-degree-of-freedom spring-mass system is used to demonstrate the capability of using residual modes to obtain high solution accuracy. The model consists of 4 masses and 5 springs, as shown in *Figure 1*. The assembled mass and stiffness matrices are as follows:

$$M = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & .5 \end{pmatrix},$$

$$K = 10,000 \begin{pmatrix} 2 & -1 & 0 & 0 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & 2 \end{pmatrix}.$$

The mass for node 4 is set to half the value of the other three nodes so as to have four distinct modes for the system. A spatial loading of unit force R is applied to node 3 in the y-direction, where

$$R = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}.$$

The eigenfrequencies and corresponding eigenmodes are given in the following table:

Mode No.	Frequency (Hz)	Nodal Eigendisplacements			
		Node 1	Node 2	Node 3	Node 4
1	10.155	0.39948	0.63631	0.61408	0.03418
2	20.222	0.68548	0.26428	–0.58359	–0.48927
3	28.258	0.60461	–0.69678	0.19839	0.46815
4	34.963	0.07056	–0.19939	0.49292	–1.19360

The spatial loading is applied harmonically with an excitation frequency of 3 Hz to verify the steady-state response of the system. The single residual mode corresponding to the excitation load is included in the projected basis. A modal damping factor of 0.02 is applied to all the modes including the residual modes.

Results and discussion

Only one eigenmode is extracted to demonstrate the capability of improving the solution accuracy by extracting residual modes. The residual mode (RM) obtained by Abaqus is identical to that given in the reference.

Mode No.	Frequency (Hz)	Nodal Eigendisplacements			
		Node 1	Node 2	Node 3	Node 4
Published solutions					
1	10.155	0.39948	0.63631	0.61408	0.03418
RM	21.865	0.68548	0.26428	–0.58359	–0.48927
Abaqus					
1	10.155	0.39948	0.63631	0.61408	0.03418
RM	21.865	0.68548	0.26428	–0.58359	–0.48927

For the 3 Hz harmonic response analysis, displacements and accelerations of all the nodes are presented for two cases. The first case uses only the first eigenmode, while the second case uses both the first eigenmode and the residual mode. The percentage error shows very clearly how solution accuracy can be significantly improved by adding the residual modes to the original set of eigenvectors.

	Displacements			
	Node 1	Node 2	Node 3	Node 4
Published results (all modes)	4.52E−5	8.89E−5	1.29E−4	6.53E−5
Abaqus solutions				
Case 1 (mode 1 only)	6.60E−5	1.05E−4	1.01E−4	5.65E−5
Case 2 (mode 1 with RM)	4.53E−5	8.88E−5	1.29E−4	6.51E−5
Percentage error				
Case 1	46.02	18.11	–21.71	–13.63
Case 2	0.22	–0.11	0.00	–0.31

	Accelerations			
	Node 1	Node 2	Node 3	Node 4
Published results (all modes)	1.61E−2	3.16E−2	4.59E−2	2.32E−2
Abaqus solutions				
Case 1 (mode 1 only)	2.34E−2	3.73E−2	3.60E−2	2.00E−2
Case 2 (mode 1 with RM)	1.61E−2	3.16E−2	4.60E−2	2.31E−2
Percentage error				
Case 1	45.34	18.04	–21.57	–13.79
Case 2	0.00	0.00	0.22	–0.43

Input files

dickens_model.inp	Dickens numerical example.

References

- Dickens, J. M., J. M. Nakagawa, and M. M. Wittbrodt, “A Critique of Mode Acceleration and Modal Truncation Augmentation Methods for Modal Response Analysis,” *Computers & Structures*, vol. 62, no. 6, pp. 985–998, 1997.

Figures

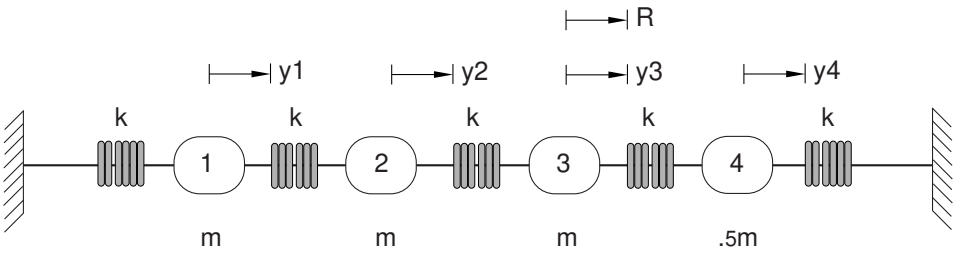


Figure 1: A four-degree-of-freedom spring-mass model.

Steady-state transport analysis

In this section:

- *Steady-state transport analysis*
- *Steady-state spinning of a disk in contact with a foundation*

Steady-state transport analysis

Products: Abaqus/Standard

Overview

This example verifies the steady-state transport analysis capability in Abaqus.
The verification concentrates on frictional effects, inertia effects, and material convection. Frictional effects are verified by comparing results obtained with Abaqus to results published by Faria (1989). Inertia effects are verified by comparing special cases of steady-state transport analyses with results obtained from an Abaqus analysis where centrifugal loads are applied using a distributed load with load type CENT. Material convection is verified by comparison with a transient Lagrangian analysis.

Frictional effects

Problem description

In this series of tests the free rolling angular velocity, ω, of a circular disk in contact with a flat rigid surface is calculated for different disk geometries, contact pressures, friction coefficients, material models, and element types. The ground velocity is specified as either a straight-line translational velocity of $v_0 = 2.0$ or as a cornering angular velocity of $\Omega = 0.02$. By specifying a large cornering radius, $R_c = 100.0$, straight line rolling with velocity $v_0 \approx \Omega R_c = 2.0$ is recovered. The results obtained with Abaqus are compared to numerical results published by Faria (1989).

The model consists of a ring with outer radius $R_o = 2.0$ and variable inner radius R_i. Three different geometries ($R_i = 0.2, 1.0, 1.7$) are considered. The model is fully fixed on the inside, and plane strain boundary conditions are imposed along the axial direction.

Two material models are considered: a linear elastic material with $E = 800.0$ and $\nu = 0.3$ and an incompressible hyperelastic material with $C_1 = 80.0$ and $C_2 = 20.0$. The friction coefficients considered are $\mu = 0.02$ and $\mu = 0.2$. The first analysis step is a static analysis where the rigid surface is displaced a distance $\Delta = 0.05$ or $\Delta = 0.1$ to establish a contact pressure. The friction coefficient during this step is held constant at zero. This step is followed by a steady-state transport analysis where the ground traveling velocity and spinning angular velocity are applied and the friction coefficient is ramped to its final value.

The problem is discretized with different types of three-dimensional elements. The models that are discretized with first-order elements use 34 element divisions along the circumference and 5 element divisions in the radial direction. The second-order and cylindrical element models use 18 elements along the circumference and 3 elements in the radial direction. All the models are discretized with one element in the axial direction. A first-order finite element mesh for the case $R_i = 1.0$ is shown in *Figure 1*.

Results and discussion

Table 1 and *Table 2* compare the free rolling angular velocity, ω, obtained from the Abaqus simulation with the reference solution. The results presented in *Table 2* are obtained using C3D8RH elements.

Additional frictional tests

Additional verification tests are performed to verify contact between a spinning deformable body and a spinning rigid body. In all these tests the deformable body uses the properties and discretization described earlier. The

rotating rigid body is in contact either with the inside surface of the deformable body (such as in the case where a tire is mounted on a rigid rim) or with the outside surface of the deformable body (such as in the case where a tire is in contact with a rotating drum). No reference solutions are available for the case where the rigid body is in contact with the inside surface of the deformable body. By specifying a large radius for the rigid body in the case where a rigid spinning drum is in contact with the outside surface of the deformable body, straight line rolling is recovered. We selected a rigid body radius of $R_r = 1000.0$ and an angular velocity of $\Omega = 0.002$, which corresponds to straight line rolling with a velocity $v_0 \approx \Omega R_r = 2.0$.

Another verification test is performed to verify contact between a rolling gear-like thick cylinder with an outer radius of 8.5 and a flat rigid surface. The model is generated by revolving a single three-dimensional 15° sector about the symmetry axis. The gear-like cylinder travels at a ground velocity of 2.7778 with an angular velocity varying from 0.2 to 0.5. The results are compared to those obtained from a transient Lagrangian analysis.

Input files

pstca4shhfa.inp — Axisymmetric model using CAX4H elements and a hyperelastic material.

pstc38shhfs.inp — C3D8H elements, hyperelastic material, $R_i = 1.0$, $\Delta = 0.1$, $\mu = 0.02$, straight line rolling with $v_0 = 2.0$ (requires two-dimensional input file pstca4shhfa.inp).

pstca4syhfa.inp — Axisymmetric model using CAX4RH elements and a hyperelastic material.

pstc38syhfs.inp — C3D8RH elements, hyperelastic material, $R_i = 1.0$, $\Delta = 0.05$, $\mu = 0.2$, straight line rolling with $v_0 = 2.0$ (requires two-dimensional input file pstca4syhfa.inp).

pstca8sfefa.inp — Axisymmetric model using CAX8H elements and an elastic material.

pstc3ksfefs.inp — C3D20 elements, elastic material, $R_i = 0.2$, $\Delta = 0.10$, $\mu = 0.02$, straight line rolling with $v_0 = 2.0$ (requires two-dimensional input file pstca8sfefa.inp).

pstca8srefa.inp — Axisymmetric model using CAX8R elements and an elastic material.

pstc3ksrefs.inp — C3D20R elements, elastic material, $R_i = 1.7$, $\Delta = 0.05$, $\mu = 0.02$, straight line rolling with $v_0 = 2.0$ (requires two-dimensional input file pstca8srefa.inp).

pstca4siefa.inp — Axisymmetric model using CAX4I elements and an elastic material.

pstc38siefc.inp — C3D8I elements, elastic material, $R_i = 10.2$, $\Delta = 0.05$, $\mu = 0.02$, cornering with $\Omega = 0.02$ and $R_c = 100.0$ (requires two-dimensional input file pstca4siefa.inp).

pstca3shhfa.inp — Axisymmetric model using CAX3H elements and a hyperelastic material.

pstc36shhfc.inp — C3D6H elements, hyperelastic material, $R_i = 1.0$, $\Delta = 0.10$, $\mu = 0.02$, cornering with $\Omega = 0.02$ and $R_c = 100.0$ (requires two-dimensional input file pstca3shhfa.inp).

Additional frictional tests:

pstc38shhfd.inp — Contact between a rigid drum and the outside surface of a deformable body, C3D8H elements; similar to pstc38shhfs.inp (requires two-dimensional input file pstca4shhfa.inp).

pstc38syhfd.inp — Contact between a rigid drum and the outside surface of a deformable body, C3D8RH elements; similar to pstc38syhfs.inp (requires two-dimensional input file pstca4syhfa.inp).

pstca3shhfr.inp — Axisymmetric model using CAX3H elements and a hyperelastic material.

pstc36shhfr.inp — Contact between a rigid “rim” and the inside surface of a deformable body and contact between a flat rigid foundation and the outside surface of a deformable

	body, C3D6H elements; similar to pstc36shhfc.inp (requires two-dimensional input file pstca3shhfr.inp).
pstcc12shhfs.inp	CCL12H elements, hyperelastic material, R_i = 1.0, Δ = 0.1, μ = 0.02, straight line rolling with v_0 = 2.0 (requires two-dimensional input file pstca4shhfa.inp).
sstransp_per_hyper_preload.inp	3D sector model using C3D8H elements and a hyperelastic material with viscosity. Preloaded with an internal pressure.
sstransp_per_hyper_rolling.inp	Contact between a flat rigid surface and a gear-like deformable cylinder (requires three-dimensional input file sstransp_per_hyper_preload.inp).

References

- Faria, L. O., Tire Modeling by Finite Elements, Ph.D. dissertation, The University of Texas at Austin, 1989.

Tables

Table 1: Comparison of Abaqus results with reference solutions for the free rolling angular velocity.

Input file	Reference solution	Abaqus	% Difference
pstc38shhfs.inp	0.95009	0.94635	0.39
pstc38syhfs.inp	0.98006	0.98213	0.21
pstc3ksfefs.inp	1.02970	1.02726	0.24
pstc3ksrefs.inp	1.00297	1.00283	0.01
pstc38siefc.inp	1.02180	1.00674	1.47
pstc36shhfc.inp	0.95195	0.94568	0.66

Table 2: C3D8RH results.

Material type	Δ	R_i	μ	Reference solution	Abaqus	% Difference
Hyperelastic	0.05	0.2	0.02	0.99349	0.99219	0.13
		1.0	0.02	0.97977	0.98053	0.08
		1.7	0.02	0.87183	0.84974	2.53
		1.0	0.20	0.98066	0.98212	0.15
	0.10	0.2	0.02	0.98558	0.98422	0.14
		1.0	0.02	0.95009	0.95059	0.05
		1.7	0.02	0.73057	0.65790	9.95
		1.0	0.20	0.95195	0.95100	0.10
Linear elastic	0.05	0.2	0.02	1.02180	1.02332	0.15
		1.0	0.02	1.02415	1.02574	0.16
		1.7	0.02	1.00297	1.00263	0.03
	0.10	0.2	0.02	1.02970	1.03410	0.43
		1.0	0.02	1.02810	1.02872	0.06
		1.7	0.02	0.99156	0.99542	0.39

Figures

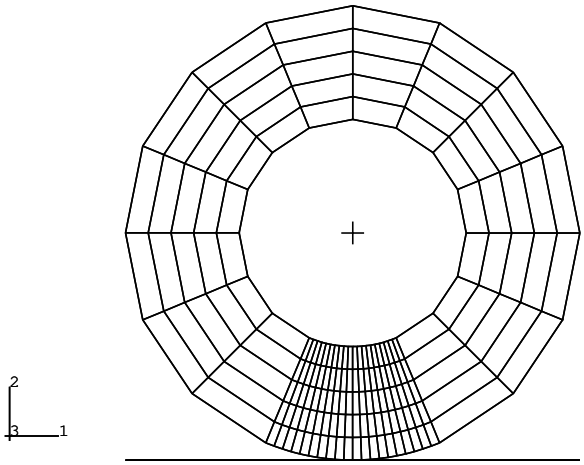


Figure 1: Finite element mesh for R_i = 1.0.

Inertia effects

Problem description

In this series of tests the effects of inertia on a free spinning and/or cornering structure are verified for different element types and angular velocities. An incompressible hyperelastic material with C_1 = 80.0, C_2 = 20.0, and ρ = 0.036 is used. The model consists of a ring with outer radius R_o = 2.0 and inner radius R_i = 1.0. Each input file contains two models with identical geometry: one model is loaded using a distributed centrifugal load (load type CENT) and serves as the reference solution; the loading on the other model is caused by steady-state rolling inertia effects.

The problem is discretized with different types of three-dimensional elements. The models that are discretized with first-order elements use 24 element divisions along the circumference and 2 element divisions in the radial direction. The second-order and cylindrical element models use 12 elements along the circumference and 1 element in the radial direction. All the models are discretized with 1 element in the axial direction.

The models are fully fixed on the inside. Plane strain boundary conditions are imposed along the axial direction in the first step. Since the material is incompressible, the loading does not give rise to any deformation.

Results and discussion

For the straight line rolling tests, the results match the reference solution. For the cornering tests without free spinning, the results match the reference solution. For the cornering tests with free spinning, the results do not match the reference solution because the steady-state inertia loading includes Coriolis effects due to a spinning wheel in a rotating reference frame; these effects are not accounted for in the reference solution.

Input files

pstc36shhic.inp	C3D6H elements; requires two-dimensional input file pstca3shhia.inp. Cornering only; no free spinning.
pstca4syhia.inp	Axisymmetric model using CAX4RH elements and a hyperelastic material.
pstc38syhic.inp	C3D8RH elements; requires two-dimensional input file pstca4syhia.inp. Cornering only; no free spinning.

pstc38shhic.inp	C3D8RH elements; requires two-dimensional input file pstca4shhia.inp. Cornering and free spinning.
pstc3kshhic.inp	C3D20H elements; requires two-dimensional input file pstca8shhia.inp. Cornering and free spinning.
pstca8yhhia.inp	Axisymmetric model using CAX8RH elements and a hyperelastic material.
pstc3ksyhic.inp	C3D20RH elements; requires two-dimensional input file pstca8yhhia.inp. Cornering and free spinning.
pstm34srhic.inp	M3D4R elements; requires two-dimensional input file pstma2srhia.inp. Cornering and free spinning.
pstm38srhic.inp	M3D8R elements; requires two-dimensional input file pstma3srhia.inp. Cornering only; no free spinning.
pstsf4shhic.inp	S4R elements; requires two-dimensional input file pstsa2shhia.inp. Cornering only; no free spinning.
psts68sheic.inp	S8R elements; requires two-dimensional input file pstsa3sheia.inp. Cornering and free spinning.

Straight line rolling tests:

pstca3shhia.inp	Axisymmetric model using CAX3H elements and a hyperelastic material.
pstc36shhis.inp	C3D6H elements; requires two-dimensional input file pstca3shhia.inp.
pstca4shhia.inp	Axisymmetric model using CAX4H elements and a hyperelastic material.
pstc38shhis.inp	C3D8H elements; requires two-dimensional input file pstca4shhia.inp.
pstca6shhia.inp	Axisymmetric model using CAX6H elements and a hyperelastic material.
pstc3fshhis.inp	C3D15H elements; requires two-dimensional input file pstca6shhia.inp.
pstca8shhia.inp	Axisymmetric model using CAX8H elements and a hyperelastic material.
pstc3kshhis.inp	C3D20H elements; requires two-dimensional input file pstca8shhia.inp.
pstma2srhia.inp	Axisymmetric model using CAX4H elements, MAX1 elements, and a hyperelastic material.
pstm34srhis.inp	M3D4R elements; requires two-dimensional input file pstma2srhia.inp.
pstma3srhia.inp	Axisymmetric model using CAX8H elements, MAX2 elements, and a hyperelastic material.
pstm38srhis.inp	M3D8R elements; requires two-dimensional input file pstma3srhia.inp.
pstsa2shhia.inp	Axisymmetric model using CAX4H elements, SAX1 elements, and a hyperelastic material.
pstsf4shhis.inp	S4R elements; requires two-dimensional input file pstsa2shhia.inp.
pstsf4shhis_po.inp	Postprocesssing analysis.
pstsa3sheia.inp	Axisymmetric model using CAX8RH elements, MAX2 elements, and an elastic material.
psts68sheis.inp	S8R elements; requires two-dimensional input file pstsa3sheia.inp.
pstcc9shhis.inp	CCL9H elements; requires two-dimensional input file pstca3shhia.inp.
pstcc12shhis.inp	CCL12H elements; requires two-dimensional input file pstca4shhia.inp.
pstcc18shhis.inp	CCL18H elements; requires two-dimensional input file pstca6shhia.inp.
pstcc24shhis.inp	CCL24RH elements; requires two-dimensional input file pstca8shhia.inp.

Cornering tests:

Material convection effects

Problem description

In this series of tests the effect of material convection with a viscoelastic material model is verified for different element types. The model consists of a ring with outer radius R_o = 2.0 and inner radius R_i = 1.0. The model is fully fixed on the inside, and plane strain boundary conditions are imposed along the axial direction.

An incompressible hyperelastic material is used, with long-term moduli C_1^∞ = 80.0, C_2^∞ = 20.0, shear relaxation coefficient of g_1 = 0.2, and relaxation time τ = 0.1.

The first analysis step is a static step where the long-term response is requested and where the rigid surface is displaced a distance Δ = 0.2 to establish a contact pressure. This step is followed by a steady-state transport analysis step where viscoelastic material effects are considered. No frictional stresses are transmitted so that the disk spins without translating along the foundation.

The problem is discretized with different types of three-dimensional elements. The models that are discretized with first-order elements use 30 element divisions along the circumference and 5 element divisions in the radial direction. The second-order and cylindrical element models use 20 elements along the circumference and 3 elements in the radial direction. All the models are discretized with one element in the axial direction.

Some other tests were performed to verify the effects of material convection when the material response is slightly compressible and allows for relaxation of the pressure stress; when viscoelastic effects take place in plane stress elements; when a hyperfoam material with relaxation is used; when the model contains viscoelastic rebars embedded in an elastic or viscoelastic material; and when an incompressible hyperelastic material with two terms defining the Prony series is used.

Results and discussion

The reaction force normal to the foundation and the torque around the axle are compared to results obtained from a transient Lagrangian analysis using the quasi-static analysis procedure that is run until steady-state conditions are achieved. A model with fine meshing (C3D8RH elements) along the entire circumference is used to obtain this reference solution.

Table 3 compares the solutions obtained using different element types with the reference solution.

Additional material convection tests

Additional tests were performed to verify the effects of material convection with an elastic-plastic or a viscoplastic material model. Both the Mises metal plasticity model with kinematic hardening and 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 temperature, have been used. The model consists of a disc with outer radius of 4.0 mm, inner radius of 1.0 mm, and thickness of 3.0 mm. The disc is generated either by revolving the cross-section of an axisymmetric mesh about the symmetry axis or by revolving a single three-dimensional repetitive sector of the model about the symmetry axis. The bottom surface of the disc is fixed. The top surface is subjected either to a nonuniform distributed load or a nonuniform contact pressure and frictional stress due to a pad being applied to the top surface of the disc. For each of these tests the disc is assumed to rotate at an angular velocity of 87.2 rad/sec or 5 rad/sec, respectively.

Each model has been analyzed using both a quasi steady-state transport solution technique through the use of the steady-state transport pass-by-pass analysis technique and a directly sought steady-state solution technique. For each of the tests the circumferential stress and circumferential plastic strain are compared to results obtained from a transient Lagrangian analysis.

Input files

pstca3shhma.inp	Axisymmetric model using CAX3H elements and a hyperelastic material with viscosity.
pstc36shhvs.inp	C3D6H elements; requires two-dimensional input file pstca3shhma.inp.
pstca4syhma.inp	Axisymmetric model using CAX4RH elements and a hyperelastic material with viscosity.
pstc38syhvs.inp	C3D8RH elements; requires two-dimensional input file pstca4syhma.inp.
pstca4shhma.inp	Axisymmetric model using CAX4H elements and a hyperelastic material with viscosity.
pstc38shhvs.inp	C3D8H elements; requires two-dimensional input file pstca4shhma.inp.
pstca4sjhma.inp	Axisymmetric model using CAX4IH elements and a hyperelastic material with viscosity.
pstc38sjhvs.inp	C3D8IH elements; requires two-dimensional input file pstca4sjhma.inp.
pstca6shhma.inp	Axisymmetric model using CAX6H elements and a hyperelastic material with viscosity.
pstc3fshhvs.inp	C3D15H elements; requires two-dimensional input file pstca6shhma.inp.
pstca8srhma.inp	Axisymmetric model using CAX8RH elements and a hyperelastic material with viscosity.
pstc3ksrhvs.inp	C3D20RH elements; requires two-dimensional input file pstca8srhma.inp.
pstca8sfhma.inp	Axisymmetric model using CAX8H elements and a hyperelastic material with viscosity.
pstc3ksfhvs.inp	C3D20H elements; requires two-dimensional input file pstca8sfhma.inp.
pstcc9shhvs.inp	CCL9H elements; requires two-dimensional input file pstca3shhma.inp.
pstcc12shhvs.inp	CCL12H elements; requires two-dimensional input file pstca4shhma.inp.
pstcc18shhvs.inp	CCL18H elements; requires two-dimensional input file pstca6shhma.inp.
pstcc24shhvs.inp	CCL24H elements; requires two-dimensional input file pstca8sfhma.inp.
pstma2srhma.inp	Axisymmetric model using CAX4RH elements, MAX1 elements, and a hyperelastic material with viscosity.
pstm34srhvs.inp	M3D4R elements; requires two-dimensional input file pstma2srhma.inp.
pstma3srhma.inp	Axisymmetric model using CAX8RH elements, MAX2 elements, and a hyperelastic material with viscosity.
pstm38srhvs.inp	M3D8R elements; requires two-dimensional input file pstma3srhma.inp.
pstma2rbema.inp	Axisymmetric model using CAX4RH elements, MAX1 elements (with rebar layers), and a hyperelastic material with viscosity.
pstm34rbevs.inp	M3D4R elements with viscoelastic rebar; requires two-dimensional input file pstma2rbema.inp.
pstca8rbema.inp	Axisymmetric model using CAX8RH elements (with rebar layers) and a hyperelastic material, viscosity, and rebar.
pstc3krbevs.inp	C3D20RH elements with viscoelastic rebar; requires two-dimensional input file pstca8rbema.inp.
pstrebara.inp	Axisymmetric model using CAX4RH elements (with rebar layers) and a hyperelastic material, viscosity, and rebar.
pstrebar.inp	Viscoelastic continuum and viscoelastic rebar; requires two-dimensional input file pstrebara.inp.

pstpressa.inp	Axisymmetric model using RAX2 elements and CAX4RH elements with a hyperelastic material with viscosity.
pstpress.inp	Pressure stress relaxation; requires two-dimensional input file pstpressa.inp.
psthfoama.inp	Axisymmetric model using RAX2 elements and CAX4R elements with a hyperfoam material with viscosity.
psthfoam.inp	Hyperfoam material; requires two-dimensional input file psthfoama.inp.
pst2pronya.inp	Axisymmetric model using RAX2 elements and CAX4RH elements with a hyperelastic material with viscosity.
pst2prony.inp	Two-term Prony series; requires two-dimensional input file pst2pronya.inp.

Additional material convection tests:

sstransp_axi_pls_preload.inp	Axisymmetric model using CAX4 elements and an elastic-plastic material.
sstransp_axi_pls_dload_dir.f	User subroutine to apply distributed load.
sstransp_axi_pls_dload_dir.inp	Steady-state transport analysis with linear kinematic hardening plasticity model subjected to nonuniform distributed loads; requires two-dimensional input file sstransp_axi_pls_preload.inp and user subroutine sstransp_axi_pls_dload_dir.f.
sstransp_axi_pls_dload_pbp.inp	Pass-by-pass steady-state transport analysis with linear kinematic hardening plasticity model subjected to nonuniform distributed loads; requires two-dimensional input file sstransp_axi_pls_preload.inp and user subroutine sstransp_axi_pls_dload_dir.f.
sstransp_axi_pls_surf_dir.inp	Steady-state transport analysis with linear kinematic hardening plasticity model subjected to nonuniform contact pressure and frictional stress; requires two-dimensional input file sstransp_axi_pls_preload.inp and user subroutine sstransp_axi_pls_dload_dir.f.
sstransp_axi_pls_surf_pbp.inp	Pass-by-pass steady-state transport analysis with linear kinematic hardening plasticity model subjected to nonuniform contact pressure and frictional stress; requires two-dimensional input file sstransp_axi_pls_preload.inp and user subroutine sstransp_axi_pls_dload_dir.f.
sstransp_axi_visp_preload.inp	Axisymmetric model using CAX4 elements and a viscoplastic material.
sstransp_axi_visp_dload_dir.inp	Steady-state transport analysis with two-layer viscoplasticity model subjected to nonuniform distributed loads; requires two-dimensional input file sstransp_axi_visp_preload.inp and user subroutine sstransp_axi_pls_dload_dir.f.
sstransp_axi_visp_dload_pbp.inp	Pass-by-pass steady-state transport analysis with two-layer viscoplasticity model subjected to nonuniform distributed loads; requires two-dimensional input file sstransp_axi_visp_preload.inp and user subroutine sstransp_axi_pls_dload_dir.f.
sstransp_axi_visp_surf_dir.inp	Steady-state transport analysis with two-layer viscoplasticity model subjected to nonuniform contact pressure and frictional stress; requires two-dimensional input file sstransp_axi_visp_preload.inp and user subroutine sstransp_axi_pls_dload_dir.f.
sstransp_axi_visp_surf_pbp.inp	Pass-by-pass steady-state transport analysis with two-layer viscoplasticity model subjected to nonuniform contact pressure and frictional stress; requires two-dimensional input file sstransp_axi_visp_preload.inp and user subroutine sstransp_axi_pls_dload_dir.f.
sstransp_per_pls_preload.inp	3D sector model using C3D8H elements and an elastic-plastic material. Preloaded with an internal pressure.
sstransp_per_pls_dload_dir.inp	Steady-state transport analysis with linear kinematic hardening plasticity model subjected to nonuniform distributed loads for a periodic disc; requires another three-dimensional input file sstransp_per_pls_preload.inp and user subroutine sstransp_axi_pls_dload_dir.f.

sstransp_per_pls_dload_pbp.inp	Pass-by-pass steady-state transport analysis with linear kinematic hardening plasticity model subjected to nonuniform distributed loads for a periodic disc; requires another three-dimensional input file sstransp_per_pls_preload.inp and user subroutine sstransp_axi_pls_dload_dir.f.
sstransp_per_pls_surf_dir.inp	Steady-state transport analysis with linear kinematic hardening plasticity model subjected to nonuniform contact pressure and frictional stress for a periodic disc; requires another three-dimensional input file sstransp_per_pls_preload.inp and user subroutine sstransp_axi_pls_dload_dir.f.
sstransp_per_pls_surf_pbp.inp	Pass-by-pass steady-state transport analysis with linear kinematic hardening plasticity model subjected to nonuniform contact pressure and frictional stress for a periodic disc; requires another three-dimensional input file sstransp_per_pls_preload.inp and user subroutine sstransp_axi_pls_dload_dir.f.
sstransp_per_visp_preload.inp	3D sector model using C3D8 elements and a viscoplastic material. Preloaded with an internal pressure.
sstransp_per_visp_dload_dir.inp	Steady-state transport analysis with two-layer viscoplasticity model subjected to nonuniform distributed loads for a periodic disc; requires another three-dimensional input file sstransp_per_visp_preload.inp and user subroutine sstransp_axi_pls_dload_dir.f.
sstransp_per_visp_dload_pbp.inp	Pass-by-pass steady-state transport analysis with two-layer viscoplasticity model subjected to nonuniform distributed loads for a periodic disc; requires another three-dimensional input file sstransp_per_visp_preload.inp and user subroutine sstransp_axi_pls_dload_dir.f.
sstransp_per_visp_surf_dir.inp	Steady-state transport analysis with two-layer viscoplasticity model subjected to nonuniform contact pressure and frictional stress for a periodic disc; requires another three-dimensional input file sstransp_per_visp_preload.inp and user subroutine sstransp_axi_pls_dload_dir.f.
sstransp_per_visp_surf_pbp.inp	Pass-by-pass steady-state transport analysis with two-layer viscoplasticity model subjected to nonuniform contact pressure and frictional stress for a periodic disc; requires another three-dimensional input file sstransp_per_visp_preload.inp and user subroutine sstransp_axi_pls_dload_dir.f.

Tables

Table 3: Reaction forces and torques for analyses using different elements.

Element Type	Force	Torque
C3D6H	373.12	–9.55
C3D8RH	370.46	–9.49
C3D8H	373.98	–9.55
C3D8IH	376.92	–9.62
C3D15H	372.74	–9.66
C3D20RH	372.53	–9.91
C3D20H	373.43	–9.97
CCL9H	357.3	–10.02
CCL12H	375.5	–10.33
CCL18H	371.3	–10.04
CCL24H	373.3	–9.97

Element Type	Force	Torque
Reference solution	374.47	–9.77

Steady-state spinning of a disk in contact with a foundation

Products: Abaqus/Standard

Overview

This example illustrates the nature of viscoelastic material effects in steady-state rolling problems and serves as a validation test for the material convection algorithm used in the steady-state transport procedure.

Since the steady-state transport capability uses a kinematic description that implies flow of material through the mesh, convective effects must be considered for history-dependent material response. Abaqus provides material convection in a steady-state transport analysis for viscoelastic materials. An overview of the capability is provided in *Steady-State Transport Analysis*.

We use an independent transient Lagrangian analysis to obtain a reference solution for the validation of the steady-state transport material convection algorithm. A finite element analysis of a similar problem, together with numerical results, has also been published by Oden et al. (1986).

Problem description

The model consists of a circular disk with an inner radius of 1 and an outer radius of 2. No particular unit system is used, but it is assumed that the units are consistent. The disk is in contact with a flat rigid surface and spins at a constant angular velocity. Friction is neglected so that the disk spins without translating along the surface. Inertia effects are also neglected. The material is incompressible hyperelastic with instantaneous elastic moduli $C_{10}^{\infty} = 100$ and $C_{01}^{\infty} = 25$, shear relaxation coefficient $\overline{g}_1^P = 0.2$, and relaxation time $\tau = 0.1$s. Plane strain boundary conditions are applied in the axial direction.

The steady-state transport analysis capability requires a finite element mesh of the cross-section of the body as a starting point. The cross-section is discretized with axisymmetric CAX4RH elements. The inside of the disk is assumed to be in contact with a rigid rim. We model this by a kinematic coupling constraint that couples all the nodes on the inside surface to a reference node placed on the center of the axle. This node is used to prescribe the motion of the disk in the subsequent three-dimensional Lagrangian reference analysis. A kinematic coupling constraint is specified.

A **datacheck** analysis is performed to write the axisymmetric model information to a restart file. The restart file is then read in a subsequent run, and a three-dimensional model is generated by Abaqus by revolving the cross-section about the symmetry axis using symmetric model generation. This method of generating the finite element model is required by Abaqus to define the streamlines in the model. The axisymmetric CAX4RH elements are converted to C3D8RH elements during the model generation. Since the foundation is not axisymmetric, it is defined in the three-dimensional model as a rigid surface. The three-dimensional finite element mesh is shown in *Figure 1*. To obtain a reference solution, a similar mesh is used for a Lagrangian analysis except that the entire circumference is finely discretized to accommodate the changing contact conditions during the spinning motion.

We also include a model using cylindrical (CCL12) elements.

Loading

The loading is applied over two analysis steps. In the first step the disk is brought in contact with the foundation by applying a prescribed displacement of 0.3 units to the rigid body reference node on the foundation (*Figure 1*). A static load step that provides the fully relaxed long-term viscoelastic solution is used for this analysis. The long-term solution ensures a smooth transition between the static and slow rolling solutions.

The second analysis step is a steady-state transport analysis. Steady-state solutions at various angular velocities (ranging from $\omega =$ 0.001 rad/s to $\omega =$ 1000 rad/s) are obtained by specifying transport velocity.

The reference Lagrangian solution is obtained using the quasi-static procedure. The file *spinningdisk_visco.inp* contains the input data for this analysis.

Results and discussion

When the stress in the material of a spinning body is influenced by the rate of strain, such as in a viscoelastic material, the deformation depends on the angular velocity of the body. During the spinning motion, material entering the contact area (leading edge) is compressed by the sudden increase in contact pressure, while material leaving the foundation relaxes. For a perfectly elastic material the deformation is reversible, so the contact area (and stress state) is symmetrical about a plane normal to the foundation and containing the axle. A viscoelastic material, on the other hand, responds instantaneously to the sudden increase in contact pressure but requires a finite time to relax after leaving the contact area. During such a loading/unloading stress-strain cycle some strain energy is dissipated. In other words, in contrast to a perfectly elastic material, the deformation is not reversible, and the loading and unloading stress-strain paths do not coincide. Consequently, the point at which material leaves the foundation is closer to the center plane than the point at which material enters the contact zone. Furthermore, since the contact pressure is asymmetrical, rolling is resisted by a moment around the axle.

The nature of viscoelastic material effects in this problem is illustrated in *Figure 2* through *Figure 4*. *Figure 2* shows the reaction force normal to the foundation; *Figure 3* shows the moment around the axle as a function of the angular spinning velocity. The bullet points in the two figures represent the reference transient Lagrangian solution. *Figure 4* shows the contact pressure at different angular velocities. These figures indicate that at low angular velocities, when the time that a material point is in contact with the foundation is long compared to the relaxation time of the material, the behavior of the disk corresponds to the fully relaxed long-term elastic solution. The vertical reaction force (*Figure 2*) is at a minimum, and the stress state is symmetrical about the midplane (*Figure 4*), so the moment around the axle is zero (*Figure 3*). At high angular velocities the solution corresponds to the instantaneous (or dynamic) elastic solution with the vertical reaction force reaching a limiting value. The stress state is still symmetrical about the midplane, so the moment around the axle is zero. The viscoelastic effects become important when the time that a material point is in contact with the foundation is of the same order of magnitude as the relaxation time of the material. Under these conditions energy is dissipated in each loading/unloading cycle, so the contact area becomes asymmetrical (*Figure 4*) and rolling is resisted by a moment around the axle (*Figure 3*).

Figure 5 through *Figure 7* compare the radial, circumferential, and shear stress between the two analysis methods for the case where the viscoelastic effects are a maximum ($\omega =$ 2.5 rad/s). The solid lines represent the steady-state transport solution; the broken lines represent the reference transient Lagrangian solution. The figures plot the stress near the outer surface along a streamline—the angle is measured about the z-axis along the direction of material flow, with the x-axis defining $\theta = 0°$. The reference solution is obtained by monitoring the stress at one integration point on the streamline during the analysis history. Since the solution is steady, the time variation of stress can be converted to a variation along the streamline. The figures show very good agreement between the two solution methods.

The steady-state transport solution obtained with cylindrical elements also agrees closely with the reference solution. The results of this simulation are not reported here.

Input files

spinningdisk_axi.inp	Reference axisymmetric model for the Lagrangian analysis and the steady-state rolling analysis using C3D8RH elements.
spinningdisk_3d.inp	Steady-state rolling analysis using C3D8RH elements.
spinningdisk_visco.inp	Transient Lagrangian analysis using the **VISCO* procedure.

spinningdisk_axi_ccl.inp	Reference axisymmetric model for the steady-state rolling analysis using CCL12H elements.
spinningdisk_3d_ccl.inp	Steady-state rolling analysis using CCL12H elements.

References

- Oden, J. T., and T. L. Lin, "On the General Rolling Contact Problem for Finite Deformations of a Viscoelastic Cylinder," *Computer Methods in Applied Mechanics and Engineering*, vol. 57, pp. 297–367, 1986.

Figures

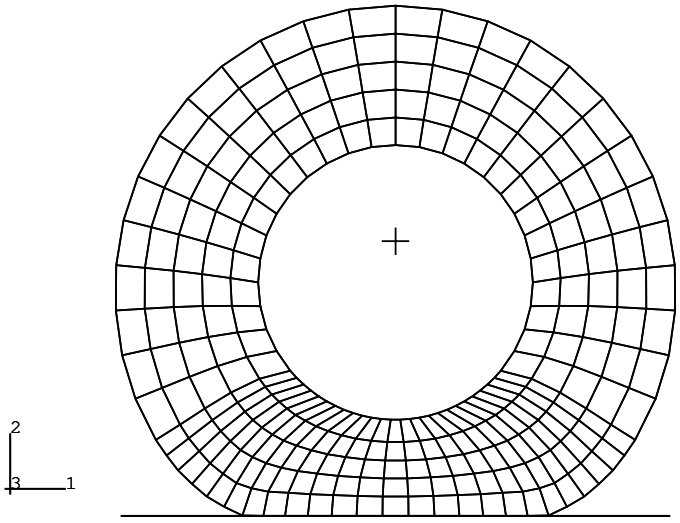


Figure 1: Displaced shape of disk ($\omega = 0.0$ rad/s).

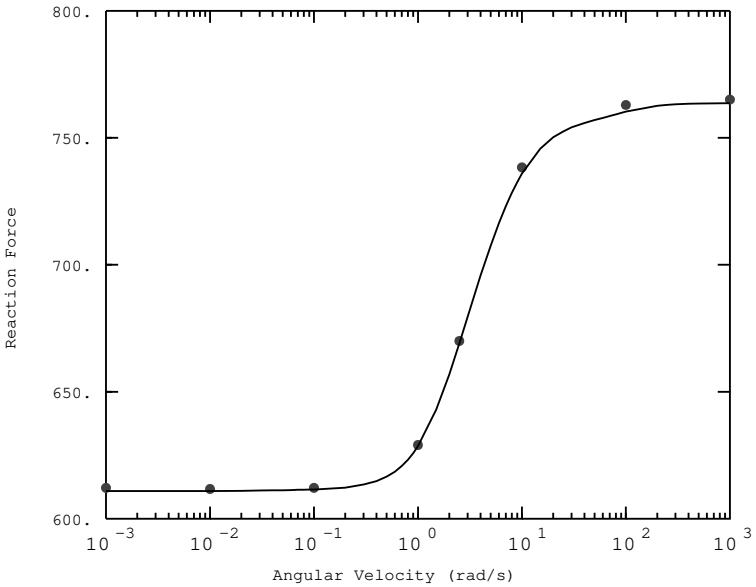


Figure 2: Reaction force normal to the foundation. The bullet points are the transient Lagrangian solution.

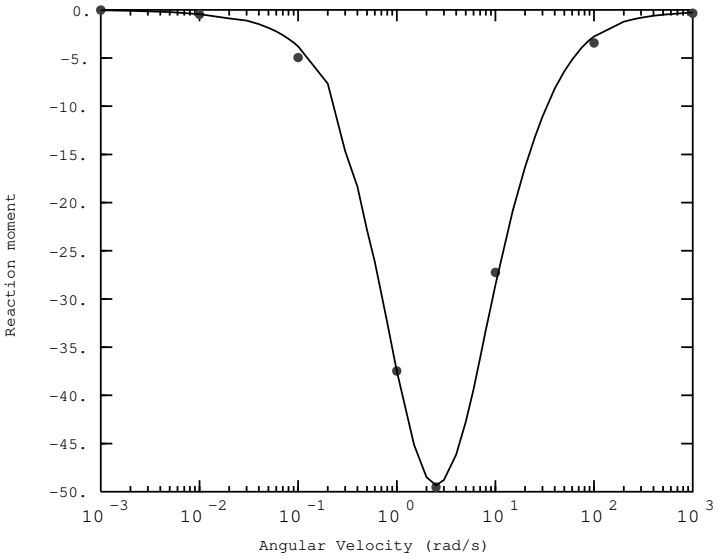


Figure 3: Moment around the axle. The bullet points are the transient Lagrangian solution.

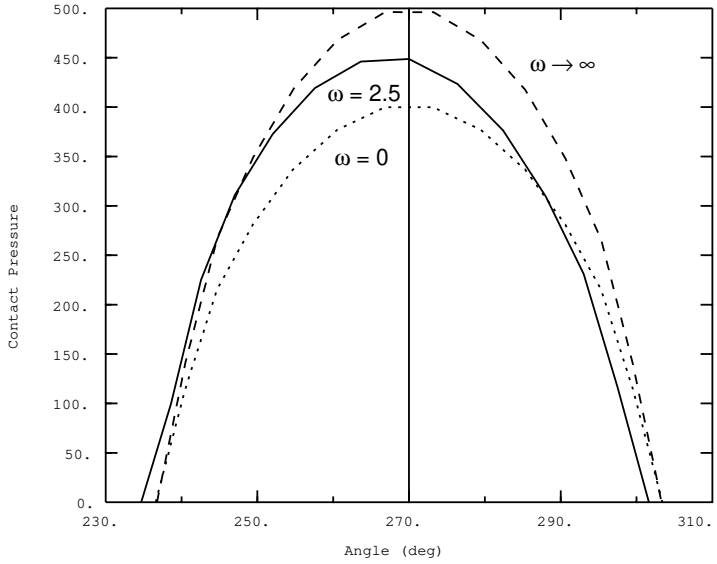


Figure 4: Contact pressure.

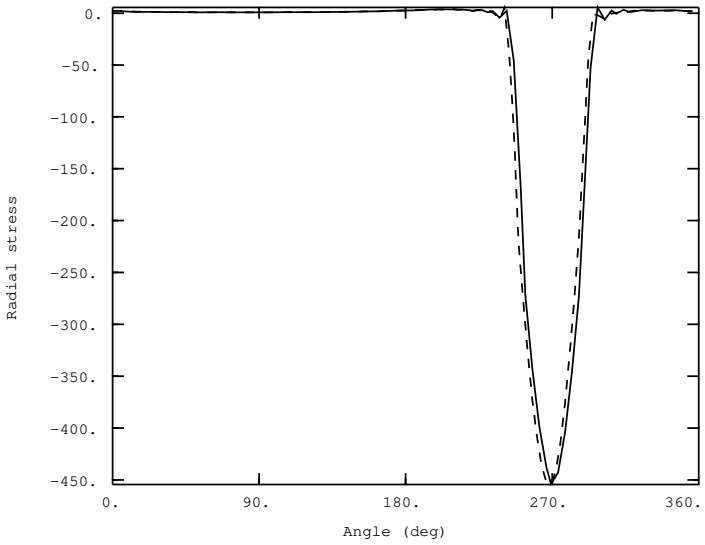


Figure 5: Radial stress variation along a streamline. Comparison with transient Lagrangian solution (broken line).

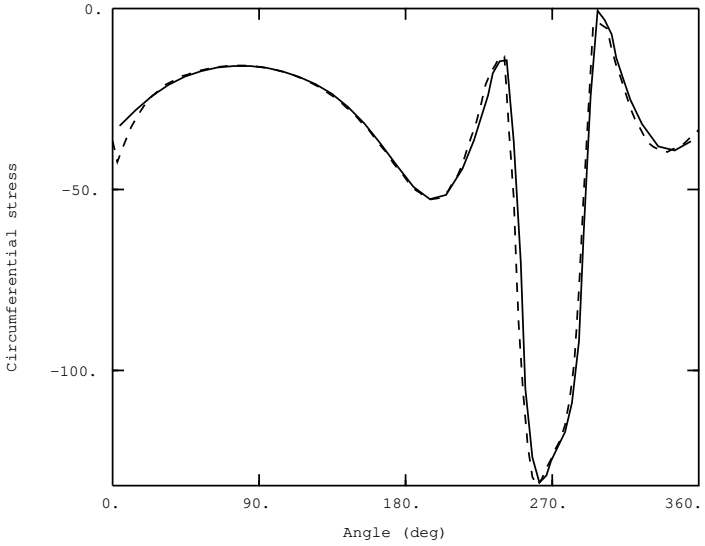


Figure 6: Circumferential stress variation along a streamline. Comparison with transient Lagrangian solution (broken line).

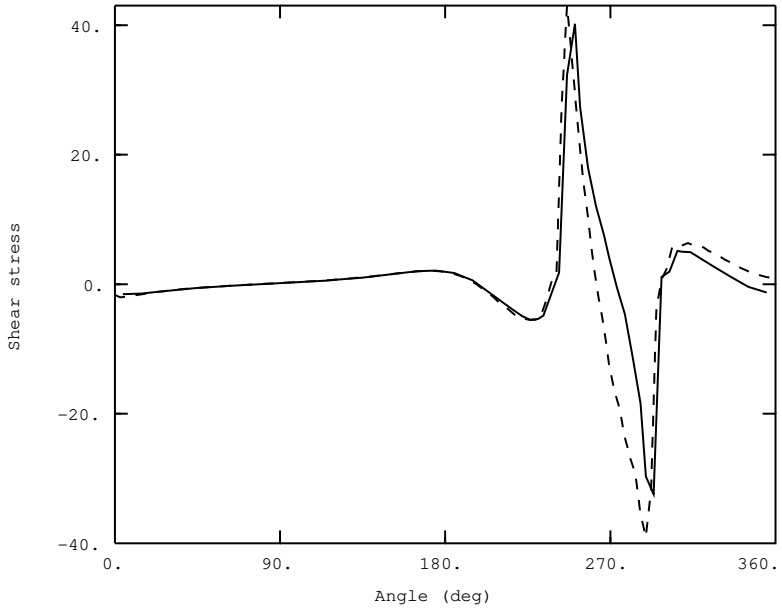


Figure 7: Shear stress variation along a streamline. Comparison with transient Lagrangian solution (broken line).

Heat transfer and thermal-stress analysis

In this section:

- *Convection and diffusion of a temperature pulse*
- *Freezing of a square solid: the two-dimensional Stefan problem*
- *Coupled temperature-displacement analysis: one-dimensional gap conductance and radiation*
- *Quenching of an infinite plate*
- *Two-dimensional elemental cavity radiation view factor calculations*
- *Axisymmetric elemental cavity radiation view factor calculations*
- *Three-dimensional elemental cavity radiation view factor calculations*
- *Radiation analysis of a plane finned surface*

Convection and diffusion of a temperature pulse

Products: Abaqus/Standard

Overview

This example verifies the use of convective/diffusive heat transfer elements for the transport and diffusion of a temperature pulse in the form of a Gaussian wave.

Variations of the problem are done in one and two dimensions. The problems are taken from the papers by Yu and Heinrich (1986, 1987).

The convective/diffusive heat transfer elements in Abaqus are intended for use in thermal problems involving heat transfer in a flowing fluid so that heat is transported (convected) by the velocity of the fluid and, at the same time, is diffused by conduction through the fluid and its surroundings. The elements utilize a Petrov-Galerkin finite element formulation (an "upwinding" method) and can also include numerical dispersion control. The techniques used in these elements are described in *Convection/diffusion*. The elements are typically used in conjunction with purely diffusive heat transfer elements, connected directly, or through thermal interfaces used to represent boundary layer effects (film coefficients) between the fluid and the solid surface. They can also be used alone. The problems in this example involve the convective/diffusive elements alone and are used to illustrate the characteristics of these types of elements.

Problem description

The geometry and models for each analysis are described in the following sections.

One-dimensional case

No particular set of physical units is used in this case: we assume that the units are consistent. The problem consists of the one-dimensional domain from $x = 0$ to $x = 2$, through which fluid is flowing at a velocity $v = 0.25$. Abaqus requires definition of the fluid mass flow rate, $\dot{m} = \rho v A$, at the nodes of the convective elements, where ρ is the fluid density and A is the cross-sectional area of the convective/diffusive element. At the start of the problem there is a temperature pulse in the form of a Gaussian wave centered at $x = v$ with peak amplitude of unity, defined by

$$\theta(x, t = 0) = \exp\left(-(x - v)^2/4K\right),$$

where K is the thermal diffusivity of the fluid, defined as $K = k/\rho c$, in which k is the conductivity of the fluid and c is its specific heat.

Yu and Heinrich show that the solution to this problem is the temperature distribution at any time, t, given by

$$\theta(x, t) = \frac{1}{\sqrt{1+t}} \exp\left[-(x - v(1+t))^2/4K(1+t)\right].$$

We use a uniform mesh of 64 elements of type DCC1D2 or DCC1D2D in the one-dimensional domain from $x = 0$ to $x = 2$. The DCC1D2D elements include numerical dispersion control; the DCC1D2 elements do not. The rather fine mesh is necessary to model the convection/diffusion of the temperature field with reasonable accuracy.

The mesh has been chosen to provide a Peclet number of 20. The Peclet number, γ, is defined as

$$\gamma = v\frac{\Delta h}{K},$$

where Δh is the length of an element. γ provides an indication of the extent to which convection dominates the heat transport in an element: $\gamma = 0$ implies no convection (zero velocity), and as $\gamma \to \infty$, the problem becomes purely convective—there is no time for diffusion. The value used in this case, $\gamma = 20$, makes the problem strongly convective but, nevertheless, leaves sufficient diffusion in the system to make it important in the solution.

The problem is transient. We use fixed time increments chosen to provide a Courant number C of 0.8. The Courant number is defined by

$$C = v\frac{\Delta t}{\Delta h},$$

where Δt is the time increment. C measures how quickly energy can be convected across an element compared to the time increment. If $C > 1$ energy can convect across more than a single element in a time increment. The convective/diffusive elements used in Abaqus cannot provide accurate transient solutions for $C > 1$, and for those elements that include numerical dispersion control (which is desirable for such transient cases) $C = 1$ is a stability limit in the sense that the solution can become numerically unstable if this value is exceeded. Therefore, we choose $C = 0.8$, which requires a time increment of $\Delta t = 0.1$ with the mesh chosen.

In a separate run we also evaluate the behavior of these elements as the wave leaves the domain of the mesh. All of the parameters here are the same as above except that the one-dimensional domain now extends from $x = 0$ to $x = 1$ (32 elements are used). The boundary condition at the edge of the mesh, $x = 1$, is the natural boundary condition:

$$k\frac{d\theta}{dx}\Big|_{x=1} = 0.$$

This boundary condition prevents conduction of heat out of the mesh but allows energy to convect through the boundary, which is convenient for practical applications. Since it is the natural boundary condition in the formulation, it requires no specification in the input data.

Two-dimensional case

Again, no particular set of physical units is used in this case: we assume that the units are consistent.

The problem consists of a two-dimensional rectangular domain defined as $0.0 \le x \le 1.0$, $0.0 \le y \le 0.5$. There is no heat generation in the region, and the boundary conditions are

$$\theta(0, y, t) = \theta(1, y, t) = 0\,, \quad 0.0 \le y \le 0.5,$$

$$k\frac{d\theta}{dy}\Big|_{y=0} = k\frac{d\theta}{dy}\Big|_{y=0.5} = 0\,, \quad 0.0 \le x \le 1.0.$$

We consider unidirectional flow that is skewed to the mesh at an angle of 25° to the x-axis and is given as $v_x = 0.25$ and $v_y = 0.1166$, where v_x is the velocity in the x-direction and v_y is the velocity in the y-direction. The initial temperature pulse is centered at $x = y = 0.175$ and is defined by

$$\theta(x, y, t) = \exp\left[-\left((x - 0.175)^2 + (y - 0.175)^2\right) / (0.00125)\right].$$

We consider the pure convection case where $K = 0$, so that there should be no diffusion of the temperature pulse.

We use a uniform rectangular 40 × 20 mesh of type DCC2D4 or DCC2D4D elements. The DCC2D4D elements include numerical dispersion control; the DCC2D4 elements do not. We use fixed time increments chosen to provide a Courant number, C, of 0.73. The Courant number in a two-dimensional rectangular mesh is defined by

$$C = \left(\frac{|v_x|}{\Delta x} + \frac{|v_y|}{\Delta y}\right)\Delta t,$$

where Δt is the time increment. The chosen mesh and Courant number define a fixed time increment of 0.05.

Results and discussion

The results for each case are presented here.

One-dimensional case

The value of the upwinding and numerical dispersion control techniques is illustrated by the three numerical solutions shown in *Figure 1*, *Figure 2*, and *Figure 3*. In each of the plots the temperature pulse is shown at three time points: at the start of the problem ($t = 0$), at $t = 2$, and at $t = 4$. Each plot shows the exact solution and a numerical solution.

The plot in *Figure 1* shows a solution generated with a standard Galerkin finite element method. (This solution cannot be generated by any standard element in Abaqus since all the convective elements include upwinding.) Spurious oscillation of the temperature on the trailing (upstream) side of the pulse is evident in the numerical solution. The plot in *Figure 2* is generated with element type DCC1D2, which includes upwinding only. There is significantly less oscillation following the trailing end of the pulse, but the peak temperature is not well predicted. (This formulation can be shown to be optimal for steady-state convection/diffusion: see Yu and Heinrich.) The plot in *Figure 3* includes upwinding and numerical dispersion control (element type DCC1D2D). The results in this case show almost no oscillation trailing the pulse. The peak temperature is slightly underestimated, but the solution is clearly superior. Further improvements in accuracy require a finer mesh.

The series of plots in *Figure 4* illustrate the wave leaving the mesh as time progresses. Element type DCC1D2D was used to generate these results at times $t = 2.4$, 2.7, 3.0, and 3.4. The exact solutions are plotted also for comparative purposes. The traveling wave exhibits no undesirable reflections as it leaves the mesh. This reflection-free response could not be obtained with a Galerkin formulation element.

Two-dimensional case

The value of the upwinding control technique is illustrated by the two numerical solutions shown in *Figure 5* and *Figure 6*. In each of the plots the temperature pulse is shown in its initial state and at a time of 1.3. The exact solution of the problem is transport of the initial wave in the direction of flow with zero dissipation. All calculations presented here contain some inherent numerical dissipation.

The plot in *Figure 5* shows a solution generated with a standard Galerkin finite element method. The peak temperature with this method is $\theta_{max} = 0.82$, but dispersive oscillations as large as 44% of θ_{max} are observed. The plot in *Figure 6* illustrates the advantages gained by the Petrov-Galerkin formulation implemented in Abaqus. *Figure 6*, which was generated with element type DCC2D4, shows $\theta_{max} = 0.51$. Here the dispersion present is only about 11% of θ_{max}. Further improvements in accuracy can be obtained with a finer mesh.

Axisymmetric and three-dimensional element tests

The two-dimensional problem is also modeled with axisymmetric and three-dimensional elements. The axisymmetric model, consisting of DCCAX4 or DCCAX4D elements, uses a mesh of the same size as the two-dimensional problem. The mesh is located at a very large radius, so the problem definition is approximately the same. The three-dimensional model, consisting of DCC3D8 or DCC3D8D elements, uses a single layer mesh of the same size as the two-dimensional model. The results for both models are the same as for the two-dimensional model.

Input files

convectdifftemppulse_dcc1d2.inp	One-dimensional case of upwinding only (element type DCC1D2). Numerical dispersion control is added by changing the element type to DCC1D2D.
convectdifftemppulse_mass.inp	Contains the mass flow rate data used in the file convectdifftemppulse_dcc1d2.inp.
convectdifftemppulse_dcc1d2d.inp	One-dimensional case of the wave leaving the mesh.
convectdifftemppulse_exact.f	A program used to create the one-dimensional analytical solution.
convectdifftemppulse_dcc2d4.inp	Two-dimensional skewed transport case of upwinding only (element type DCC2D4). Numerical dispersion control is added by changing the element type to DCC2D4D.
convectdifftemppulse_2dtemp0.f	A program used to create the two-dimensional initial temperature conditions.
convectdifftemppulse_dccax4.inp	Axisymmetric skewed transport case of upwinding only (element type DCCAX4). Numerical dispersion control is added by changing the element type to DCCAX4D.
convectdifftemppulse_dcc3d8.inp	Three-dimensional skewed transport case of upwinding only (element type DCC3D8). Numerical dispersion control is added by changing the element type to DCC3D8D.
convectdifftemppulse_3dtemp0.f	A program used to create the three-dimensional initial temperature conditions.

References

- Yu, C. C., and J. C. Heinrich, “Petrov-Galerkin Methods for the Time-Dependent Convective Transport Equation,” *International Journal for Numerical Methods in Engineering*, vol. 23, pp. 883–901, 1986.
- Yu, C. C., and J. C. Heinrich, “Petrov-Galerkin Method for Multidimensional, Time-Dependent, Convective-Diffusion Equations,” *International Journal for Numerical Methods in Engineering*, vol. 24, pp. 2201–2215, 1987.

Figures

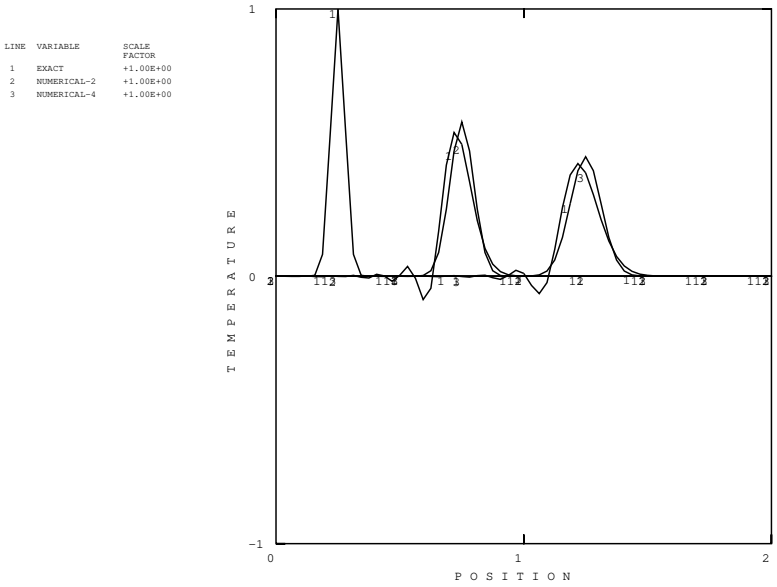


Figure 1: One-dimensional convection/diffusion model problem (no upwinding).

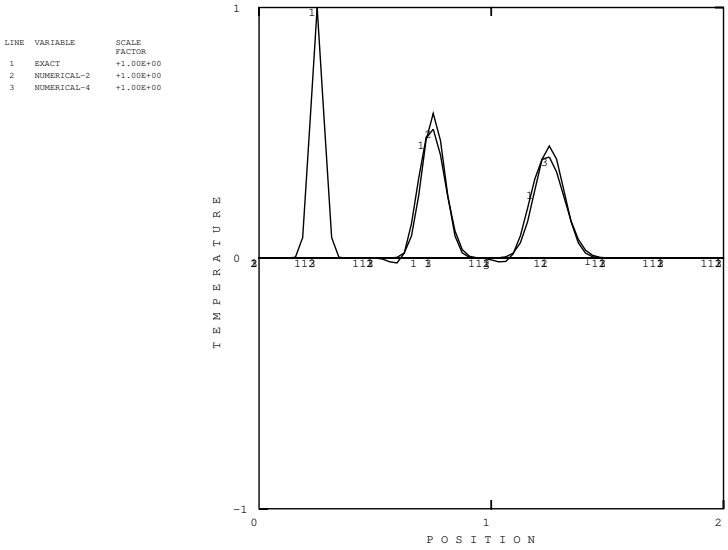


Figure 2: One-dimensional convection/diffusion model problem (with upwinding).

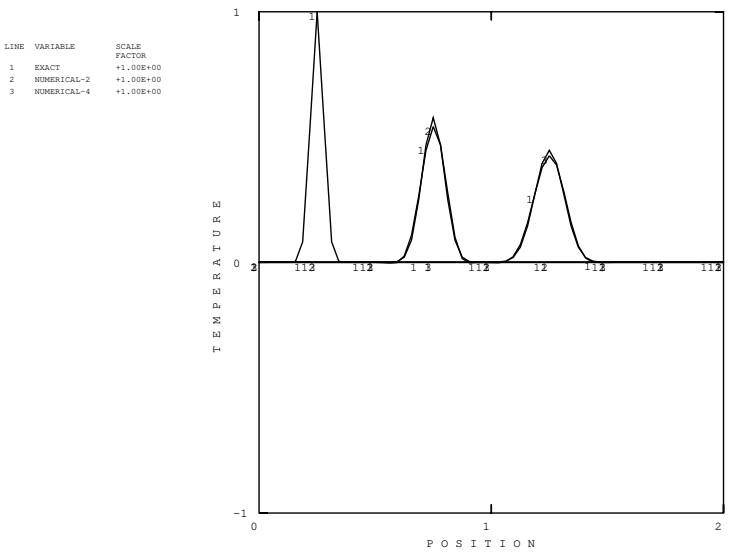


Figure 3: One-dimensional convection/diffusion model problem (with upwinding and numerical dispersion control).

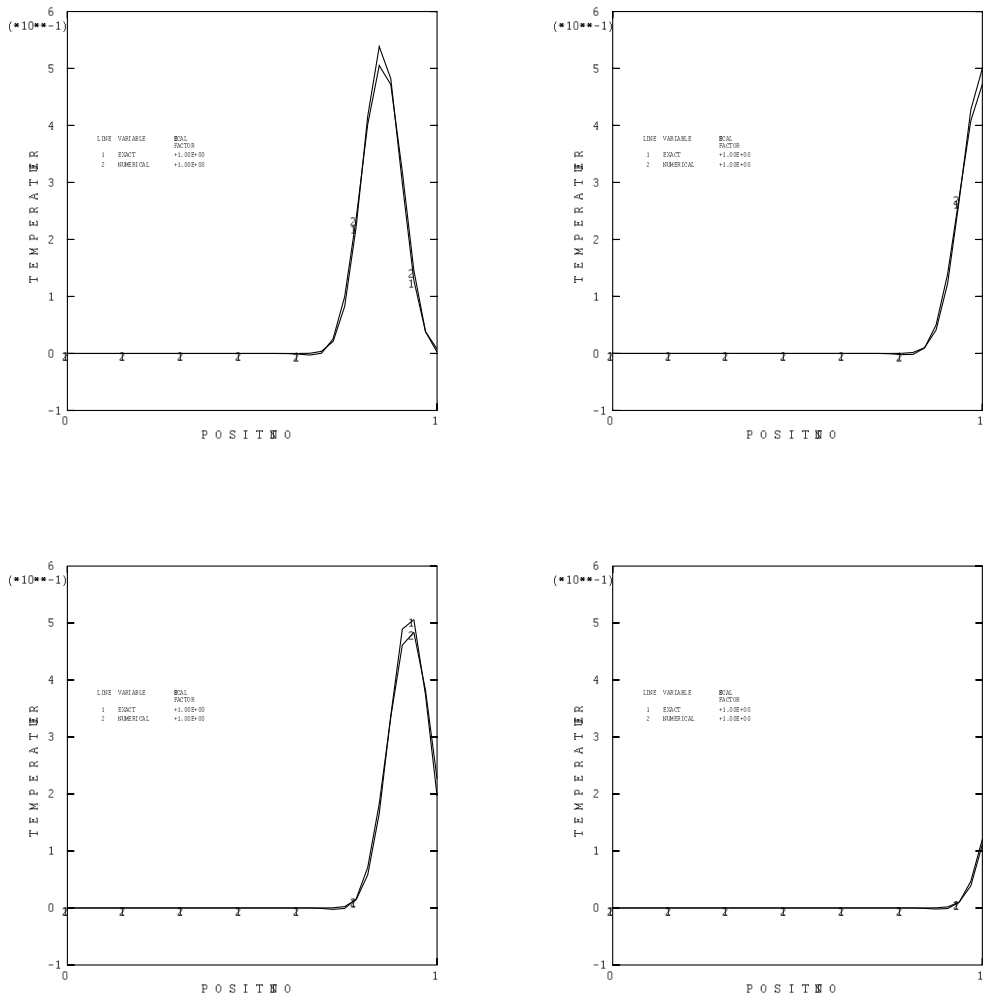


Figure 4: One-dimensional convection/diffusion model problem: wave leaving mesh.

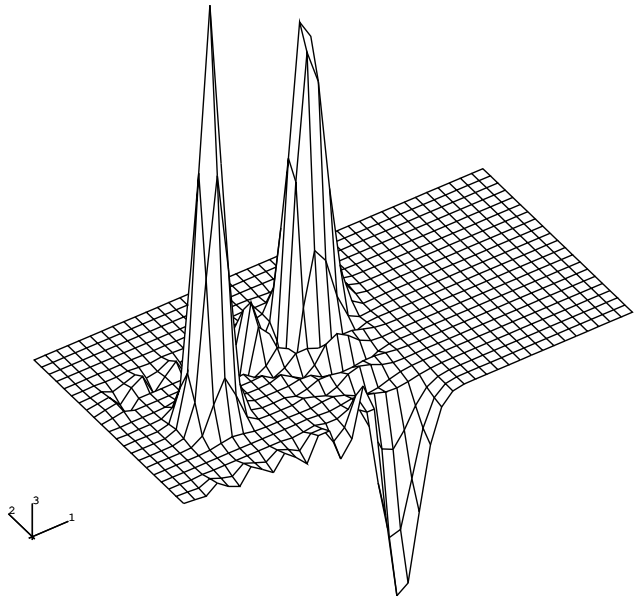


Figure 5: Two-dimensional skewed transport model problem (no upwinding).

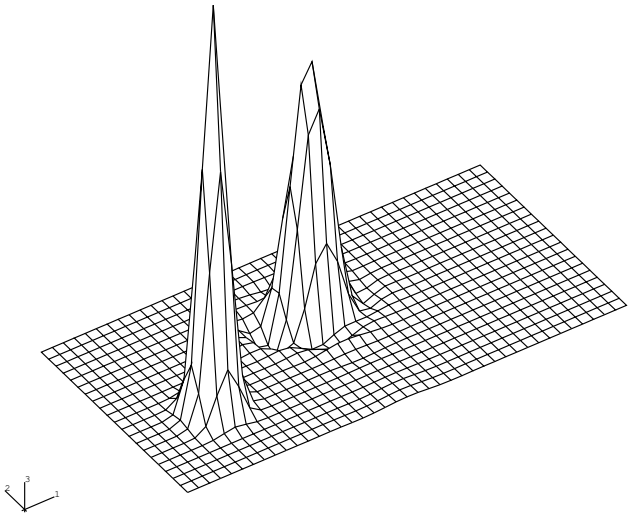


Figure 6: Two-dimensional skewed transport model problem (with upwinding).

Freezing of a square solid: the two-dimensional Stefan problem

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example verifies the two-dimensional Stefan problem, which consists of freezing a square block of material.

Heat conduction problems involving latent heat effects occur often in practice (examples are metal casting and permafrost meltout) but are not simple to solve. In some cases the phase change occurs with little latent heat effect and rapid temperature changes can partially suppress the change, as in the case of the amorphous/crystalline polymer phase change. For such cases Abaqus/Standard provides a user subroutine, `HETVAL`, in which the user can program the kinetics of the phase change and the consequent latent heat exchange in terms of solution-dependent state variables. In contrast, a liquid/solid phase change is usually fairly abrupt and is accompanied by a strong latent heat effect. This case is the one considered in this example.

The problem is the two-dimensional Stefan problem (*Figure 1*): a square block of material is initially liquid, just above the freezing temperature. The temperature of its outside perimeter is reduced suddenly by a large value, so that the block starts to freeze from the outside toward the core. The freezing has a very large latent heat effect associated with it that dominates the solution. The problem has no exact solution, but a number of researchers have provided approximate solutions. Probably the most accurate of these is the numerical solution of Lazaridis (1970), who considers the problem as a moving boundary condition problem. Lazaridis's solution is used here as verification of the Abaqus modeling of such cases.

Problem description

The block is a square of dimension 8×8 length units. Because of symmetry we need to consider only an octant, but we model a quarter for simplicity in generating the mesh.

Severe latent heat effects involve moving boundary conditions (the freezing front), across which the spatial gradient of temperature, $\partial\theta/\partial\mathbf{x}$, is discontinuous. Simple finite elements, such as the linear and quadratic elements used in Abaqus, do not allow gradient discontinuities within an element, although they do allow such discontinuities between elements in the direction of the normal to their sides. Since the actual problem involves discontinuities along surfaces moving through the mesh, the best we can do with a fixed grid of simple elements is to use a fine mesh of lowest-order elements, thus providing a high number of gradient discontinuity surfaces. In Abaqus/Standard two-dimensional heat transfer elements (DS3 and DS4) and first-order coupled temperature-displacement elements (CPE4T and CPEG4T) are used to model the plate. The meshes used are coarse for the problem; but they suffice to give reasonable solutions and, thus, verify the capability. In practical cases a more refined model is recommended. In Abaqus/Explicit two- and three-dimensional, first-order coupled temperature-displacement elements (CPE4RT, C3D8RT, and SC8RT) are used to model the plate.

Material

The material properties (in consistent units) are

Density	1.0
Specific heat	1.0
Latent heat of freezing	70.26
Freezing temperature	0°
Thermal conductivity	1.08

This set of values includes a latent heat effect that is far more severe than that in any material of practical importance. This value is deliberately chosen to provide a stringent test of the accuracy of the algorithm.

The latent heat must be specified in Abaqus over a temperature range. For this purpose we give the solidus and liquidus temperatures as −0.25° and −0.15°, respectively.

In the simulations involving Abaqus/Explicit dummy mechanical properties are used to complete the material definition.

Boundary conditions

The symmetry lines are insulated; this is the default surface boundary condition and, so, need not be specified. The outside surfaces must be reduced at time zero to −45°. This value can be specified directly; however, we ramp the temperature down to −45° over a time of 0.05 to prevent the automatic time incrementation scheme in Abaqus/Standard from choosing very small time increments at the beginning of the simulation.

Time increment controls

Automatic time incrementation is chosen, which is the usual option for transient heat conduction problems. In Abaqus/Standard a maximum temperature change of 4° is allowed per time increment to allow the time increment to increase to large values at later times as the solution smoothes out.

Results and discussion

Temperature-time plots for points *A* and *B* of *Figure 1* are shown in *Figure 2*, where they are compared to Lazaridis's (1970) numerical solution. The numerical results shown in this figure are based on the solution obtained with Abaqus/Standard. The Abaqus results are quite accurate considering the coarseness of the mesh used and the extreme severity of the latent heat effect in this example. The solution oscillates about Lazaridis's results because the finite element mesh allows temperature gradient discontinuities only at element boundaries, so that the fusion fronts effectively jump between these locations. This effect is also the reason for the delay in the start of the temperature drop.

Figure 3 and *Figure 4* show isotherm contour plots at different times. The form of the solution is very clear from these plots.

The results obtained with Abaqus/Explicit compare well with those obtained with Abaqus/Standard, as illustrated in *Figure 5*. This figure compares the results obtained with Abaqus/Explicit for the temperature history of points A and B against the same results obtained with Abaqus/Standard.

Input files

Abaqus/Standard input files

freezingofsolid_2d.inp	Input data for the two-dimensional problem.
freezingofsolid_3d.inp	Similar model in three dimensions.
freezingofsolid_postoutput.inp	**POST OUTPUT* analysis.
freezingofsolid_2d_usr_umatht.inp	Two-dimensional simulation of the problem with the material behavior defined in user subroutine `UMATHT` to illustrate the coding of this subroutine.
freezingofsolid_2d_usr_umatht.f	User subroutine `UMATHT` used in freezingofsolid_2d_usr_umatht.inp.
freezingofsolid_ds3.inp	Two-dimensional analysis with DS3 elements.
freezingofsolid_ds4.inp	Two-dimensional analysis with DS4 elements.

freezingofsolid_deftorigid.inp	Two-dimensional analysis with CPE4T elements declared as rigid.
freezingofsolid_cpeg4t.inp	Two-dimensional analysis with CPEG4T elements.

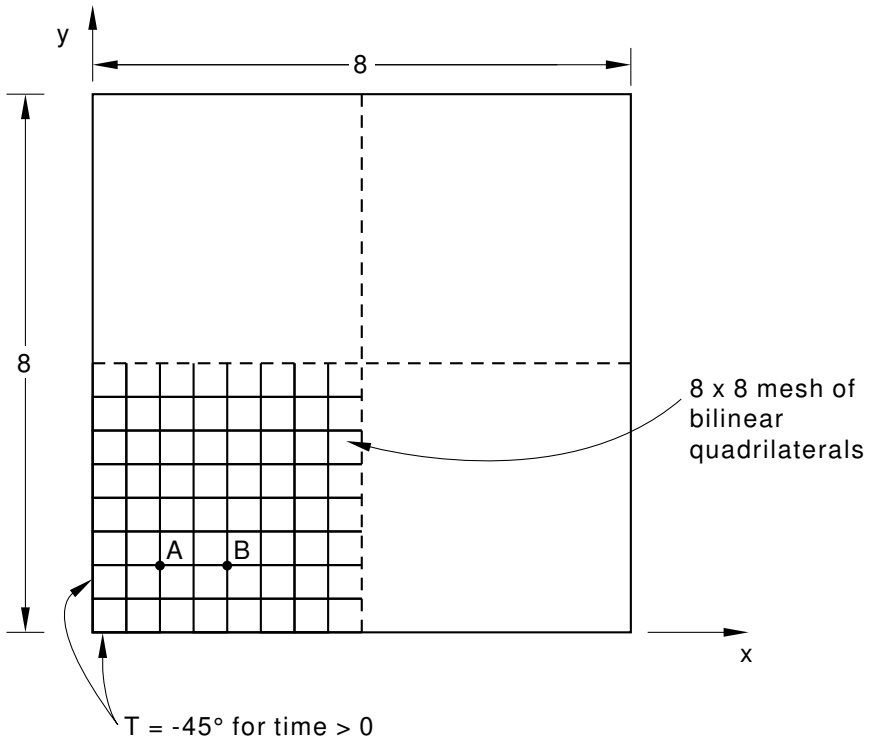
Abaqus/Explicit input files

freezingofsolid_xpl_cpe4rt.inp	Two-dimensional analysis with CPE4RT elements.
freezingofsolid_xpl_c3d8rt.inp	Three-dimensional analysis with C3D8RT elements.
freezingofsolid_xpl_c3d8rt_vumatht.inp	Three-dimensional analysis with C3D8RT elements with the material behavior defined in user subroutine *VUMATHT* to illustrate the coding of this subroutine.
freezingofsolid_xpl_c3d8rt_vumatht.f	User subroutine *VUMATHT* used in freezingofsolid_xpl_c3d8rt_vumatht.inp.
freezingofsolid_xpl_sc8rt.inp	Three-dimensional analysis with SC8RT elements.

References

- Lazaridis, A., "A Numerical Solution of the Multidimensional Solidification (or Melting) Problem," *International Journal of Heat and Mass Transfer*, vol. 13, 1970.

Figures



Material:
density = 1.0
specific heat = 1.0
latent heat = 70.26
solidus temperature= -0.25°
liquidus temperature = -0.15°
conductivity = 1.08

Figure 1: Square plate freezing example.

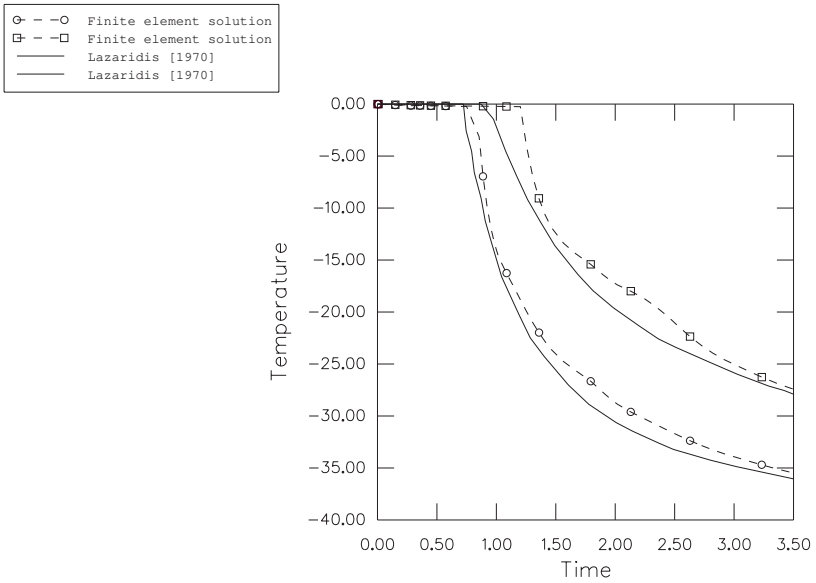


Figure 2: Square plate fusion—temperature versus time at nodes A and B of figure above.

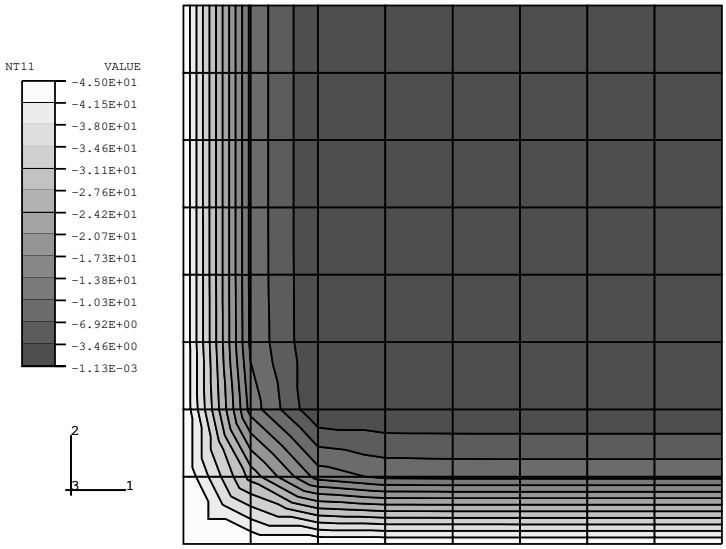


Figure 3: Square plate fusion—isotherms at $t = 0.450$ (Abaqus/Standard).

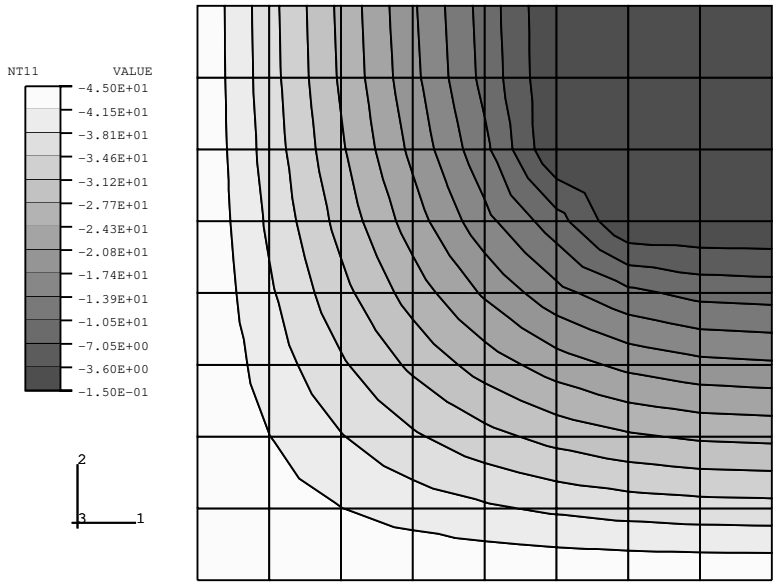


Figure 4: Square plate fusion—isotherms at $t = 5.0$ (Abaqus/Standard).

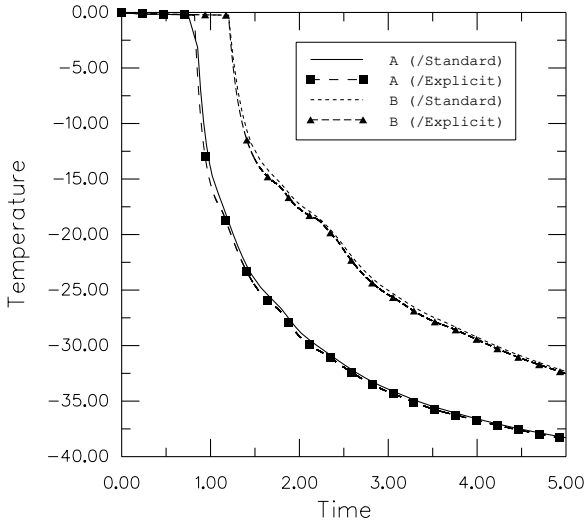


Figure 5: Comparison of results obtained with Abaqus/Explicit and Abaqus/Standard.

Coupled temperature-displacement analysis: one-dimensional gap conductance and radiation

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example illustrates two elementary nonlinear cases of one-dimensional, fully coupled, heat transfer and stress analysis.
The problems in this example are simple enough that exact solutions are obtained easily, thus providing verification of the numerical solutions obtained with Abaqus.

Problem description

The model is shown in *Figure 1*. A conductive rod of unit area is fixed at one end, *A*, and free at the other end, B. Between the free end and an adjacent fixed wall, *C*, there is a gap across which heat will be conducted or radiated.

In case 1 two forms of clearance-dependent heat transfer are considered: in the first, the conductivity for the gap drops linearly as the clearance increases; in the second, the gap radiation view factor drops linearly as the clearance increases. The fixed ends of the rod, *A*, and the wall, *C*, are both held at fixed temperatures, θ_A and θ_C. Initially the gap is open, so the distance between *B* and *C* is d_0 ($d_0 > 0$). The objective is to predict the steady-state displacement, u_B, and temperature, θ_B, of the free end of the rod. We assume that the strains are small and that the behavior of the rod is linear elastic, with constant modulus and thermal expansion coefficient. In this case the gap never closes, so the rod is always stress-free.

In case 2 it is assumed that the conductivity across the closed gap increases linearly as the pressure transmitted through the gap, *p*, increases. The fixed end of the rod, *A*, and the wall, *C* (which is also fixed in position), are both held at fixed temperatures, θ_A and θ_C. Since in this case the gap never opens, the axial stress in the rod will be nonzero. We solve for the pressure across the gap, *p*, and the temperature, θ_B, of the end of the rod, assuming that the strains are small and the behavior of the rod is linear elastic with constant modulus and thermal expansion coefficient.

In Abaqus/Standard the bar is modeled with either two- or three-dimensional elements; the contact between the end of the bar and the wall is modeled in one of three ways: as a gap element (GAPUNIT) or as an element-based rigid surface made of T2D2T, S4RT, S4T, or S8RT elements. In Abaqus/Explicit the bar is modeled with either two- or three-dimensional elements; the wall is modeled one of two ways: either as an analytical rigid surface or as an element-based rigid surface. Surface-based contact is employed between the bar and the wall; both kinematic and penalty mechanical contact are considered.

Solution

Mechanical equilibrium along the rod requires that

$$\frac{d\sigma}{dS} = 0,$$

where *S* is the distance along the rod measured from the fixed end, A. Integrating along the rod, the stress is

$$\sigma(S) = -p,$$

where *p* is the pressure transmitted by contact between the end of the rod, *B*, and the adjacent fixed point, C.

Thermal equilibrium requires that the heat flux along the rod, q, has no gradient:

$$\frac{dq}{dS} = 0.$$

Integrating this along the rod and imposing the boundary condition that the flux at B is the same as the flux transmitted from B to C through the gap, q_g, then gives the thermal equilibrium equation

$$q(S) = q_g.$$

Since we assume that the strains are small, the strain at any point in the rod is

$$\varepsilon(S) = \frac{du}{dS},$$

and the displacement is

$$u(S) = \int \varepsilon(S)\, dS.$$

The rod is assumed to be made of a linear elastic material, so the stress constitutive equation is

$$\sigma = E(\varepsilon - \alpha\theta),$$

where the modulus, E, and the thermal expansion coefficient, α, are constants (they are not temperature dependent).

Heat conduction in the rod is assumed to be governed by Fourier's law, which states that the heat flux is determined by

$$q = -k_r \frac{d\theta}{dS},$$

where k_r is the thermal conductivity of the rod and is also assumed to be constant. Combining thermal equilibrium with the Fourier law in the rod shows that $d\theta/dS$ is constant in the rod, so the temperature, θ, varies linearly along the rod:

$$\theta = \theta_A + \frac{S}{L}(\theta_B - \theta_A),$$

where L is the length of the rod.

The heat flux in the gap, q_g, between the end of the rod, B, and the fixed point C is assumed to be proportional to the difference in temperature between B and C:

$$q_g = k_g(\theta_B - \theta_C), \quad k_g \geq 0.$$

Case 1

First we assume that the gap is open and that the gap thermal conductivity, k_g, increases linearly as the gap reduces, so

$$k_g = \max(k_o + k'_u u_B, 0),$$

where k_0 and k'_u are nonnegative constants and u_B is the displacement of point B. Gap radiation is neglected in these calculations.

The thermal boundary conditions are that the temperatures at A and C, θ_A and θ_C, are held constant. The mechanical boundary conditions are that points A and C are fixed. Since in this case the end of the rod never touches C, force equilibrium requires that

$$\sigma = -p = 0.$$

The equations given above define the problem. Their solution is readily developed as follows. Combining integrated force equilibrium with the linear elastic constitutive equation and the displacement relationship, $u(S)$, gives

$$u_B = \frac{\alpha L}{2}(\theta_A + \theta_B).$$

Thermal equilibrium combined with Fourier's law and the gap heat flux equation then gives

$$\frac{k_r}{L}(\theta_A - \theta_B) = k_g(\theta_B - \theta_C).$$

With the assumed form of the gap thermal conductivity, k_g, and assuming $k_g > 0$, this is

$$\frac{k_r}{L}(\theta_A - \theta_B) = (k_0 + k_u' u_B)(\theta_B - \theta_C), \quad k_g > 0.$$

Substituting for u_B then gives a quadratic equation for θ_B:

$$\theta_B^2 + 2\left(\frac{1}{k_u' \alpha L}\left(\frac{k_r}{L} + k_0\right) + \frac{1}{2}(\theta_A - \theta_C)\right) - \theta_A\theta_C - \frac{2}{k_u' \alpha L}\left(k_0\theta_C + \frac{k_r}{L}\theta_A\right) = 0, \quad k_g > 0.$$

The roots of this quadratic equation provide two solutions for θ_B. The solutions for u_B are available once θ_B is determined. Only one of the two solutions gives a value of u_B for which $k_g > 0$; hence, this is the only physically acceptable solution.

As a numerical example the parameters are chosen in consistent units as $k_r = k_0 = 1.0$; $\alpha = 10^{-5}$, $L = 1.0$; $k_u' = 100$; $\theta_A = 400°$; and $\theta_C = 200°$.

These values give $\theta_B = 285.4°$ or $-4485°$, so $u_B = 3.427 \times 10^{-3}$ or -2.043×10^{-2}. The second solution must be rejected as it gives $k_g < 0$. The first solution is valid so long as $d_0 > u_B$.

Next we assume that the gap is open and that the gap radiation view factor, F_g, increases linearly as the gap reduces; so

$$F_g = \max(F_o + F_u' u_B, 0),$$

where F_0 and F_u' are nonnegative constants and u_B is the displacement of point B. In this case gap conduction is neglected.

The thermal and mechanical boundary conditions are the same as the gap conduction problem considered above. Force equilibrium requires that

$$\sigma = -p = 0.$$

Following a procedure similar to that used in the gap conduction problem and combining integrated force equilibrium with the linear elastic constitutive equation and the displacement relationship, $u(S)$, gives

$$u_B = \frac{\alpha L}{2}(\theta_A + \theta_B).$$

Combining thermal equilibrium with Fourier's law and the gap heat flux equation then gives

$$\frac{k_r}{L}(\theta_A - \theta_B) = q_g.$$

With the assumed form of the gap radiation, q_g, and assuming $F_g > 0$, this is

$$\frac{k_r}{L}(\theta_A - \theta_B) = \sigma\epsilon(F_0 + F_u' u_B)\left(\theta_B^4 - \theta_C^4\right), \quad F_g > 0.$$

Substituting for u_B then gives the following equation for θ_B:

$$\theta_B^5 + \left(\theta_A + \frac{2F_0}{F_u'\alpha L}\right)\theta_B^4 - \left(\theta_C^4 - \frac{2K_r}{F_u'\alpha\sigma\epsilon L^2}\right)\theta_B - \left(\theta_A\theta_C^4 + \frac{2K_r}{F_u'\alpha\sigma\epsilon L^2}\theta_A + \frac{2F_0}{F_u'\alpha L}\theta_C^4\right) = 0$$

θ_B is obtained by solving the above equation numerically. The solutions for u_B are available once θ_B is determined.

As a numerical example the parameters are chosen in consistent units as $F_0 =$ 1.0; $\alpha = 10^{-5}$; $L =$ 1.0; $F_u' =$ 50; $\sigma =$ 1.E–8; $\epsilon =$ 1.0; $\theta_A =$ 400°; $\theta_C =$ 200°; and absolute zero = –460.

These values give $\theta_B =$ 222.4°, so $u_B = 3.112 \times 10^{-3}$. This solution is valid so long as $d_0 > u_B$. All other solutions must be rejected since they give $F_g < 0$ or $d_0 < u_B$.

Case 2

In this case the rod is always in contact with *C*. Combining the integrated equilibrium equation with the mechanical constitutive model gives

$$p = -E(\varepsilon - \alpha\theta).$$

Combining this with the temperature solution, θ, gives

$$p = -E\left(\varepsilon - \alpha\left(\theta_A + \frac{S}{L}(\theta_B - \theta_A)\right)\right).$$

Integrating this along the rod using the displacement relationship, $u(S)$, gives the pressure as a function of the temperature at points *A* and *B*,

$$p = \frac{E\alpha}{2}(\theta_A + \theta_B)$$

since $u_B = 0$.

In this case the conductivity of the closed gap is proportional to the contact pressure:

$$k_g = k_a + k_p' p,$$

where k_a and k_p' are nonnegative constants. Since $k_r(\theta_A - \theta_B)/L = k_g(\theta_B - \theta_C)$, with this behavior for k_g we have

$$\frac{k_r}{L}(\theta_A - \theta_B) = (k_a + k_p' p)(\theta_B - \theta_C), \quad p > 0.$$

Combining this with the equation for the pressure provides a quadratic equation for θ_B:

$$\theta_B^2 + 2\left(\frac{1}{k_p' E\alpha}\left(\frac{k_r}{L} + k_a\right) + \frac{1}{2}(\theta_A - \theta_C)\right)\theta_B - \theta_A\theta_C - \frac{2}{k_p' E\alpha}\left(k_a\theta_C + \frac{k_r}{L}\theta_A\right) = 0, \quad p > 0.$$

The roots of this equation provide two solutions for θ_B, and the corresponding values of *p* are then defined by $p = E\alpha(\theta_A + \theta_B)/2$. Only one solution gives a positive value for *p*; the other must be rejected because it is inconsistent with the assumption that the gap is closed.

As a numerical example the parameters are chosen in consistent units as $k_r =$ 10; $k_a =$ 2; $k_p' =$ 0.2; $\alpha = 10^{-5}$; $E = 10^5$; $L =$ 1.0; $\theta_A =$ 200°; and $\theta_C =$ 100°.

These values give $\theta_B =$ 122.6° or –342.6°, so $p =$ 161.3 or –71.3. The second solution must be rejected as it gives $p < 0$.

Results and discussion

In both cases Abaqus/Standard uses a full Newton method and obtains the solution in one or two increments requiring two or three iterations per increment. The values for $\boldsymbol{u_B}$ and $\boldsymbol{\theta_B}$ in case 1 and p and $\boldsymbol{\theta_B}$ in case 2 agree with the exact solutions obtained above.

The results obtained with Abaqus/Explicit also agree with the analytical solutions.

Input files

Abaqus/Standard input files

Clearance-dependent problem with gap conduction:

coupledtempdisp_clearance.inp	T2D3T elements and GAPUNIT elements.
coupledtempdisp_std_c_cpe3t.inp	CPE3T elements for the rod and an element-based rigid surface made of T2D2T elements for the wall.
coupledtempdisp_std_c_cps3t.inp	CPS3T elements for the rod and an element-based rigid surface made of T2D2T elements for the wall.
coupledtempdisp_std_c_cpe4t.inp	CPE4T elements for the rod and an element-based rigid surface made of T2D2T elements for the wall.
coupledtempdisp_std_c_cpe4rt.inp	CPE4RT elements for the rod and an element-based rigid surface made of T2D2T elements for the wall.
coupledtempdisp_std_c_cpe4rht.inp	CPE4RHT elements for the rod and an element-based rigid surface made of T2D2T elements for the wall.
coupledtempdisp_std_c_cpeg4t.inp	CPEG4T elements for the rod and an element-based rigid surface made of T2D2T elements for the wall.
coupledtempdisp_std_c_c3d4t.inp	C3D4T elements for the rod and an element-based rigid surface made of S4RT elements for the wall.
coupledtempdisp_std_c_c3d6t.inp	C3D6T elements for the rod and an element-based rigid surface made of S4RT elements for the wall.
coupledtempdisp_std_c_c3d8ht.inp	C3D8HT elements for the rod and an element-based rigid surface made of S8RT elements for the wall.

Clearance-dependent problem with gap radiation:

coupledtempdisp_clearancerad.inp	T2D3T elements and GAPUNIT elements.
coupledtempdisp_std_r_cpe3t.inp	CPE3T elements for the rod and an element-based rigid surface made of T2D2T elements for the wall.
coupledtempdisp_std_r_cps3t.inp	CPS3T elements for the rod and an element-based rigid surface made of T2D2T elements for the wall.
coupledtempdisp_std_r_cpe4t.inp	CPE4T elements for the rod and an element-based rigid surface made of T2D2T elements for the wall.
coupledtempdisp_std_r_cpeg4t.inp	CPEG4T elements for the rod and an element-based rigid surface made of T2D2T elements for the wall.
coupledtempdisp_std_r_c3d4t.inp	C3D4T elements for the rod and an element-based rigid surface made of S4T elements for the wall.
coupledtempdisp_std_r_c3d6t.inp	C3D6T elements for the rod and an element-based rigid surface made of S4T elements for the wall.

coupledtempdisp_std_r_c3d8ht.inp	C3D8HT elements for the rod and an element-based rigid surface made of S8RT elements for the wall.

Pressure-dependent problem with gap conduction:

coupledtempdisp_pressure.inp	T2D3T elements and GAPUNIT elements.
coupledtempdisp_pressure_post.inp	**POST OUTPUT* analysis.
coupledtempdisp_std_p_cpe3t.inp	CPE3T elements for the rod and an element-based rigid surface made of T2D2T elements for the wall.
coupledtempdisp_std_p_cps3t.inp	CPS3T elements for the rod and an element-based rigid surface made of T2D2T elements for the wall.
coupledtempdisp_std_p_cps4t.inp	CPS4T elements for the rod and an element-based rigid surface made of T2D2T elements for the wall.
coupledtempdisp_std_p_cpeg4t.inp	CPEG4T elements for the rod and an element-based rigid surface made of T2D2T elements for the wall.
coupledtempdisp_std_p_c3d4t.inp	C3D4T elements for the rod and an element-based rigid surface made of S4RT elements for the wall.
coupledtempdisp_std_p_c3d6t.inp	C3D6T elements for the rod and an element-based rigid surface made of S4RT elements for the wall.
coupledtempdisp_std_p_c3d8t.inp	C3D8T elements for the rod and an element-based rigid surface made of S8RT elements for the wall.

Abaqus/Explicit input files

Clearance-dependent gap conduction problem, kinematic mechanical contact between analytical rigid and deformable surfaces:

coupledtempdisp_xa_c_cpe3t.inp	CPE3T elements.
coupledtempdisp_xa_c_cpe4rt.inp	CPE4RT elements.
coupledtempdisp_xa_c_cps3t.inp	CPS3T elements.
coupledtempdisp_xa_c_cps4rt.inp	CPS4RT elements.
coupledtempdisp_xa_c_c3d4t.inp	C3D4T elements.
coupledtempdisp_xa_c_c3d6t.inp	C3D6T elements.
coupledtempdisp_xa_c_c3d8rt.inp	C3D8RT elements.
coupledtempdisp_xa_c_c3d8t.inp	C3D8T elements.
coupledtempdisp_xa_c_sc6rt.inp	SC6RT elements.
coupledtempdisp_xa_c_sc8rt.inp	SC8RT elements.

Clearance-dependent gap conduction problem, penalty mechanical contact between analytical rigid and deformable surfaces:

coupledtempdisp_xap_c_cpe4rt.inp	CPE4RT elements.
coupledtempdisp_xap_c_c3d4t.inp	C3D4T elements.

Clearance-dependent gap radiation problem, kinematic mechanical contact between analytical rigid and deformable surfaces:

coupledtempdisp_xa_r_cpe3t.inp	CPE3T elements.
coupledtempdisp_xa_r_cpe4rt.inp	CPE4RT elements.
coupledtempdisp_xa_r_cps3t.inp	CPS3T elements.
coupledtempdisp_xa_r_cps4rt.inp	CPS4RT elements.
coupledtempdisp_xa_r_c3d4t.inp	C3D4T elements.

coupledtempdisp_xa_r_c3d6t.inp	C3D6T elements.
coupledtempdisp_xa_r_c3d8rt.inp	C3D8RT elements.
coupledtempdisp_xa_r_c3d8t.inp	C3D8T elements.
coupledtempdisp_xa_r_sc6rt.inp	SC6RT elements.
coupledtempdisp_xa_r_sc8rt.inp	SC8RT elements.

Clearance-dependent gap radiation problem, penalty mechanical contact between analytical rigid and deformable surfaces:

coupledtempdisp_xap_r_cpe4rt.inp	CPE4RT elements.
coupledtempdisp_xap_r_c3d4t.inp	C3D4T elements.

Pressure-dependent gap conduction problem, kinematic mechanical contact between analytical rigid and deformable surfaces:

coupledtempdisp_xa_p_cpe3t.inp	CPE3T elements.
coupledtempdisp_xa_p_cpe4rt.inp	CPE4RT elements.
coupledtempdisp_xa_p_cps3t.inp	CPS3T elements.
coupledtempdisp_xa_p_cps4rt.inp	CPS4RT elements.
coupledtempdisp_xa_p_c3d4t.inp	C3D4T elements.
coupledtempdisp_xa_p_c3d6t.inp	C3D6T elements.
coupledtempdisp_xa_p_c3d8rt.inp	C3D8RT elements.
coupledtempdisp_xa_p_c3d8t.inp	C3D8T elements.
coupledtempdisp_xa_p_sc8rt.inp	SC8RT elements.

Pressure-dependent gap conduction problem, penalty mechanical contact between analytical rigid and deformable surfaces:

coupledtempdisp_xap_p_cps4rt.inp	CPS4RT elements.
coupledtempdisp_xap_p_c3d6t.inp	C3D6T elements.

Clearance-dependent gap conduction problem, kinematic mechanical contact between element-based rigid and deformable surfaces:

coupledtempdisp_xd_c_cpe3t.inp	CPE3T elements.
coupledtempdisp_xd_c_cpe4rt.inp	CPE4RT elements.
coupledtempdisp_xd_c_cps3t.inp	CPS3T elements.
coupledtempdisp_xd_c_cps4rt.inp	CPS4RT elements.
coupledtempdisp_xd_c_c3d4t.inp	C3D4T elements.
coupledtempdisp_xd_c_c3d6t.inp	C3D6T elements.
coupledtempdisp_xd_c_c3d8rt.inp	C3D8RT elements.
coupledtempdisp_xd_c_c3d8t.inp	C3D8T elements.
coupledtempdisp_xd_c_sc8rt.inp	SC8RT elements.

Clearance-dependent gap conduction problem, penalty mechanical contact between element-based rigid and deformable surfaces:

coupledtempdisp_xdp_c_cpe3t.inp	CPE3T elements.
coupledtempdisp_xdp_c_c3d8rt.inp	C3D8RT elements.
coupledtempdisp_xdp_c_c3d8t.inp	C3D8T elements.
coupledtempdisp_xdp_c_sc8rt.inp	SC8RT elements.

Clearance-dependent gap radiation problem, kinematic mechanical contact between element-based rigid and deformable surfaces:

coupledtempdisp_xd_r_cpe3t.inp	CPE3T elements.
coupledtempdisp_xd_r_cpe4rt.inp	CPE4RT elements.
coupledtempdisp_xd_r_cps3t.inp	CPS3T elements.
coupledtempdisp_xd_r_cps4rt.inp	CPS4RT elements.
coupledtempdisp_xd_r_c3d4t.inp	C3D4T elements.
coupledtempdisp_xd_r_c3d6t.inp	C3D6T elements.
coupledtempdisp_xd_r_c3d8rt.inp	C3D8RT elements.
coupledtempdisp_xd_r_c3d8t.inp	C3D8T elements.
coupledtempdisp_xd_r_sc8rt.inp	SC8RT elements.

Clearance-dependent gap radiation problem, penalty mechanical contact between element-based rigid and deformable surfaces:

coupledtempdisp_xdp_r_cpe3t.inp	CPE3T elements.
coupledtempdisp_xdp_r_c3d8rt.inp	C3D8RT elements.
coupledtempdisp_xdp_r_c3d8t.inp	C3D8T elements.
coupledtempdisp_xdp_r_sc8rt.inp	SC8RT elements.

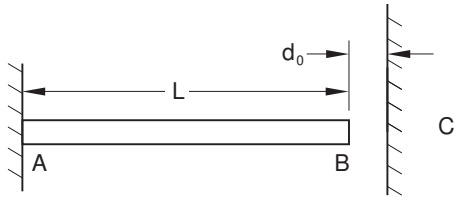
Pressure-dependent gap conduction problem, kinematic mechanical contact between element-based rigid and deformable surfaces:

coupledtempdisp_xd_p_cpe3t.inp	CPE3T elements.
coupledtempdisp_xd_p_cpe4rt.inp	CPE4RT elements.
coupledtempdisp_xd_p_cps3t.inp	CPS3T elements.
coupledtempdisp_xd_p_cps4rt.inp	CPS4RT elements.
coupledtempdisp_xd_p_c3d4t.inp	C3D4T elements.
coupledtempdisp_xd_p_c3d6t.inp	C3D6T elements.
coupledtempdisp_xd_p_c3d8rt.inp	C3D8RT elements.
coupledtempdisp_xd_p_c3d8t.inp	C3D8T elements.
coupledtempdisp_xd_p_sc8rt.inp	SC8RT elements.

Pressure-dependent gap conduction problem, penalty mechanical contact between element-based rigid and deformable surfaces:

coupledtempdisp_xdp_p_cps3t.inp	CPS3T elements.
coupledtempdisp_xdp_p_c3d8rt.inp	C3D8RT elements.
coupledtempdisp_xdp_p_c3d8t.inp	C3D8T elements.
coupledtempdisp_xdp_p_sc8rt.inp	SC8RT elements.

Figures



All specifications in consistent units.

Geometry:

L = rod length = 1.0
A = rod area = 1.0
d_0 = gap clearance
Case 1: $d_0 = 0.01$ Case 2: $d_0 = 0.0$

Material:

E = Young's modulus = 1.0×10^5
□ = expansion coefficient = 1.0×10^{-5}
k_r = rod conductivity
□= emissivity
Case 1a: $k_r = 1.0$, □= 0 Case 1b: $k_r = 0$, □= 1
Case 2: $k_r = 10.0$, □= 0

Gap Conditions:

Case 1a: Gap conductance = 2.0 at clearance = 0.0
Gap conductance = 0.0 at clearance = 0.02
Case 1b: Gap radiation view factor = 1.0 at clearance = 0.0
Gap radiation view factor = 0.0 at clearance = 0.02
Case 2: Gap conductance = 2.0 at pressure = 0.0
Gap conductance = 1002.0 at pressure = 5000.0

Boundary conditions:

Case 1: □A = 400 □C = 200
Case 2: □A = 200 □C = 100

Figure 1: Coupled temperature-displacement analysis specifications.

Quenching of an infinite plate

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example verifies the use of uncoupled heat transfer and subsequent thermal-stress analysis in Abaqus.

A semi-analytic solution is available for the case (see Landau et al., 1960) used in this example.

The purpose of the analysis is to predict the residual stresses caused by the quenching of a large homogeneous plate in regions away from the edges of the plate so that it can be treated as a plate of infinite extent in all but the thickness direction. The plate is made of an elastic, perfectly plastic material, with a yield stress that drops linearly with temperature above 121°C (250°F). The problem is one-dimensional since the plate is assumed to be of infinite extent: the only gradients occur through the thickness. The plate is initially at a uniform temperature, near its melting point (when its yield stress is small). It is assumed to be stress-free in this condition. The surface is then quenched in a medium at room temperature. Cooling is allowed to continue until all of the plate reaches room temperature.

The analyses performed in Abaqus/Standard consist of both sequential thermal-stress and fully coupled solution procedures. In the sequential analyses the transient heat transfer analysis is followed by the thermal stress analysis. During the heat transfer analysis the temperature distributions are recorded in the Abaqus results file. This temperature-time history is then used as input to the thermal stress analysis. The transient stresses are large enough to cause significant plastic flow, so residual stresses will remain after the plate reaches room temperature. In the fully coupled procedures the sequentially coupled problems are simulated by setting the fraction of inelastic dissipation that is converted into heat to zero. In this problem this uncouples the thermal response from the mechanical response.

A fully coupled solution procedure is used in Abaqus/Explicit; the sequentially coupled problem described above is again simulated by setting the fraction of inelastic dissipation that is converted into heat to zero. For completeness, another analysis is performed in Abaqus/Explicit, this time using the *VUMAT* user subroutine to define the material response and assuming that a 0.2 fraction of the inelastic dissipation is converted into heat. This last analysis illustrates the use of the *VUMAT* user subroutine in conjunction with the inelastic heat fraction, specific heat, and conductivity; the heat flux due to inelastic energy dissipation is calculated automatically by Abaqus/Explicit.

Problem description

The plate is shown in *Figure 1*. It is 914.4 mm (36 in) thick and has the following properties:

Young's modulus	206.8 GPa (30×10^6 lb/in^2)
Poisson's ratio	0.3
Yield stress	248.2 MPa for $\theta \leq$ 121°C (36000 lb/in^2, $\theta \leq$ 250°F)
	248.2(1 – (θ – 121)/1111.1) MPa, $\theta >$ 121°C
	(36000(1 – (θ – 250)/2000) lb/in^2, $\theta >$ 250°F)
Density	7832 kg/m^3 (0.283 lb/in^3)
Specific heat	0.6 kJ/kg°C (0.1431 BTU/lb°F)
Thermal conductivity	58.8 W/m°C (7.872×10^{-4} BTU/in sec°F)

The film coefficient on the surface of the plate is 193.1 W/m^2°C (6.559×10^{-5}BTU/in^2 sec°F).

The finite element mesh used in the Abaqus/Standard simulations is shown in *Figure 1*. Ten elements are used through the half-thickness of the plate. Only one row of elements is needed since the problem is one-dimensional.

No mesh convergence studies have been done: it is assumed that this mesh should give reasonably accurate results. This assumption is confirmed by the agreement with the results described by Landau et al. (1960). For the sequential thermal-stress analyses the heat transfer mesh uses elements of type DC2D8 (8-node quadrilaterals) and DC2D4 (4-node quadrilaterals). For the stress analysis the boundary conditions correspond to generalized plane strain in all directions that are normal to the surface of the plate; that is, any straight line that is initially perpendicular to the surface of the plate remains straight and perpendicular to the surface, but the distance between such lines varies as the plate cools. In the sequential thermal-stress analyses this condition is modeled using four element types: axisymmetric elements CAX8R and CAX4I and generalized plane strain elements CPEG4I and CPEG8R. To verify the interpolation technique between dissimilar meshes, the stress analysis using CAX8R elements is driven by the temperature field from the analysis using DC2D4 elements. In the fully coupled Abaqus/Standard simulations CPEG4HT, CPEG8RHT, CAX4T, and CAX4RT elements are used. The generalized plane strain elements have zero relative rotations prescribed between the planes that define the limits of the model in the thickness direction. (This condition is imposed by introducing boundary conditions at the reference node of these elements.) The generalized plane strain condition in the plane of the model is imposed in both models by using a symmetry condition on the left-hand edge of the mesh and an equation constraint to impose equal displacements at all nodes on the right-hand edge of the mesh.

Reduced-integration elements are used in all second-order models. Reduced integration is attractive because it decreases the analysis cost and, at the same time, provides more accurate stress predictions. Reduced integration is generally recommended when second-order elements are chosen.

In the Abaqus/Explicit simulations the axisymmetric plate is modeled with either CAX3T or CAX4RT elements; for the case where the plate is assumed to be in a state of generalized plane strain, C3D8RT elements are used with appropriate constraints to ensure that plane sections remain plane. In each case 20 elements are used through the half-thickness of the plate. Mass scaling is used to reduce the computational cost of the analyses.

Analysis sequence

The Abaqus/Standard sequential thermal-stress simulation consists of a transient heat transfer analysis, followed by a thermal-stress analysis in which the temperatures predicted by the heat transfer analysis are used as the loading of the problem. Abaqus makes it very simple to transfer temperature data in this way. In the heat transfer analysis the temperatures at the nodes are written to a file. Then, in the stress analysis these temperatures are read back into the stress model. This mode of transferring the temperatures is based on node numbers: the temperature at node *N* from the heat transfer analysis is applied at node *N* in the stress mesh. Thus, the node numbers must remain the same from the heat transfer model to the stress model. Abaqus does not check that the nodes are in the same location. In some cases nonstructural components (such as insulation) are modeled in the heat transfer analysis but not in the stress analysis. This situation does not present a problem; if the output from the heat transfer analysis includes temperatures at nodes that do not exist in the stress analysis model, those temperatures are ignored in the stress analysis.

In the Abaqus/Standard and Abaqus/Explicit fully coupled analyses the thermal and mechanical responses of the plate are determined simultaneously.

Controls

The following discussion is relevant only for the Abaqus/Standard simulations.

You can limit the maximum temperature change that may occur in an increment and, thus, determine the accuracy with which the transient temperature solution is integrated in time. Setting this value implies the use of automatic time incrementation, which is desirable in a case such as this where we wish to carry the analysis through to steady-state conditions, so that large time increments are used toward the end of the solution. In this example the maximum temperature change is set to 5.56°C (10°F). This choice should provide sufficient accuracy in the heat transfer solution to define the residual stresses correctly.

The initial time increment is suggested to be 20 seconds, and the time period is suggested to be 4×10^6 seconds. Since the solution is to reach steady state, the time period specification is rather arbitrary: it has to be long enough

to reach steady state. Controls are set for the heat transfer analysis so that the analysis should terminate when steady-state conditions are reached. Steady-state conditions are defined for the purpose of this parameter by the time rate of change of temperature at all nodes falling below a given value. In this analysis this value is set to 0.556×10^{-6}°C per second (10^{-6}°F per second). When a heat transfer analysis specifies to end the analysis when steady state is reached, controls are set for the heat transfer analysis so that the step terminates either when steady-state conditions have been reached or when the time period specified for the step has been completed, whichever comes first. Therefore, a very large time period is generally used in such cases.

It is usually desirable to specify a minimum time increment to cover the possibility that a data error or unforeseen event in the solution causes the automatic time increment scheme to choose very small increments. In this case a value of 0.5 seconds is used for this purpose. *Uncoupled Heat Transfer Analysis* recommends a minimum time increment for transient heat transfer analysis when there is a rapid change in temperature of

$$\Delta t > \frac{\rho c}{6k} \Delta \ell^2.$$

In this case $\Delta \ell$ is 0.9 in, so this formula suggests a minimum time increment of at least 6.9 sec. In the case where the surface temperature is changed suddenly, time increments that are smaller than this can cause initial oscillations in the solution. However, the physics of this problem do not produce sufficiently large temperature gradients to cause such oscillations with the time increment that satisfies the maximum temperature change specified.

Results and discussion

Two cases are considered: one where the initial temperature is 1038°C (1900°F), and one where the initial temperature is 816°C (1500°F). The residual stresses are shown in *Figure 2*, where they are compared to the values given by Landau et al. (1960). The numerical results shown in this figure are based on the solution obtained with Abaqus/Standard. The close agreement between the Abaqus results and those of this reference verifies this class of thermal-stress analysis.

Time histories of the stress at the integration point next to the surface and at the integration point next to the center of the plate are shown in *Figure 3*. The stress reversals that occur early in the analysis are readily observed in this plot. The excellent agreement between the results obtained with Abaqus/Explicit and Abaqus/Standard is also clear from this plot.

The last Abaqus/Explicit analysis shows that an inelastic heat fraction can be used together with the *VUMAT* user subroutine such that the inelastic dissipation computed within the *VUMAT* subroutine is converted into heat generation in a dynamic fully coupled thermal-stress analysis.

Input files

Abaqus/Standard input files

quenchplate_dc2d8.inp	1038°C (1900°F) heat transfer analysis data.
quenchplate_cax8r_quadheat.inp	Stress analysis data with CAX8R elements.
quenchplate_cpeg8r.inp	Stress analysis data with CPEG8R elements.
quenchplate_dc2d4.inp	Heat transfer data using DC2D4 elements.
quenchplate_cax4i.inp	Corresponding stress analysis data for CAX4I elements.
quenchplate_cpeg4i.inp	Corresponding stress analysis data for CPEG4I elements.
quenchplate_postoutput.inp	**POST OUTPUT* analysis.
quenchplate_cax8r_linheat.inp	Stress analysis data for CAX8R elements. The temperature data are read from the results file of quenchplate_dc2d4.inp.
quenchplate_cpeg4ht.inp	Analysis data for CPEG4HT elements.
quenchplate_cpeg8rht.inp	Analysis data for CPEG8RHT elements.

quenchplate_std_cax4t.inp	Analysis data for CAX4T elements.
quenchplate_std_cax4rt.inp	Analysis data for CAX4RT elements.
quenchplate_std_cax3t.inp	Analysis data for CAX3T elements.
quenchplate_cax8r_interpolate.inp	Analysis data for testing temperature interpolation for CAX8R elements. The temperature data are read from the output database file of quenchplate_dc2d4.inp.

Abaqus/Explicit input files

quenchplate_xpl_cax3t.inp	Analysis data for CAX3T elements.
quenchplate_xpl_cax4rt.inp	Analysis data for CAX4RT elements.
quenchplate_xpl_c3d8rt.inp	Analysis data for C3D8RT elements.
quenchplate_xpl_vumat.inp	Analysis data for CAX4RT elements using the user material subroutine *VUMAT*.
quenchplate_xpl_vumat.f	User material subroutine *VUMAT* to be used with quenchplate_xpl_vumat.inp.

To run the problem with an initial temperature of 816°C (1500°F), simply change the initial temperatures in both the heat transfer and stress analysis input data files to 1500.

References

- Landau, H. G., J. H. Weiner, and E. E. Zwicky, Jr., “Thermal Stress in a Viscoelastic-Plastic Plate with Temperature Dependent Yield Stress,” *Journal of Applied Mechanics*, vol. 27, pp. 297–302, 1960.

Figures

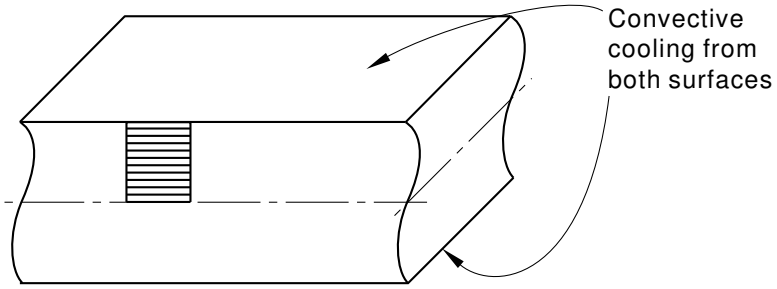


Figure 1: Infinite plate quenching problem and finite element mesh.

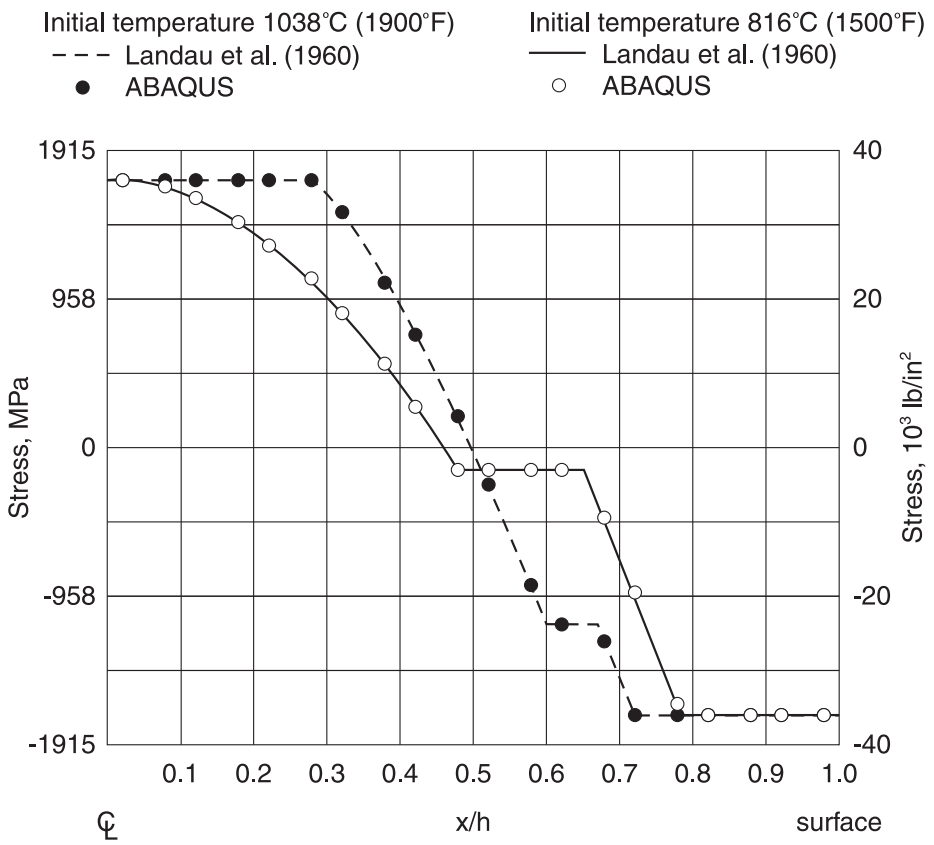


Figure 2: Residual stresses through the half-plate (Abaqus/Standard).

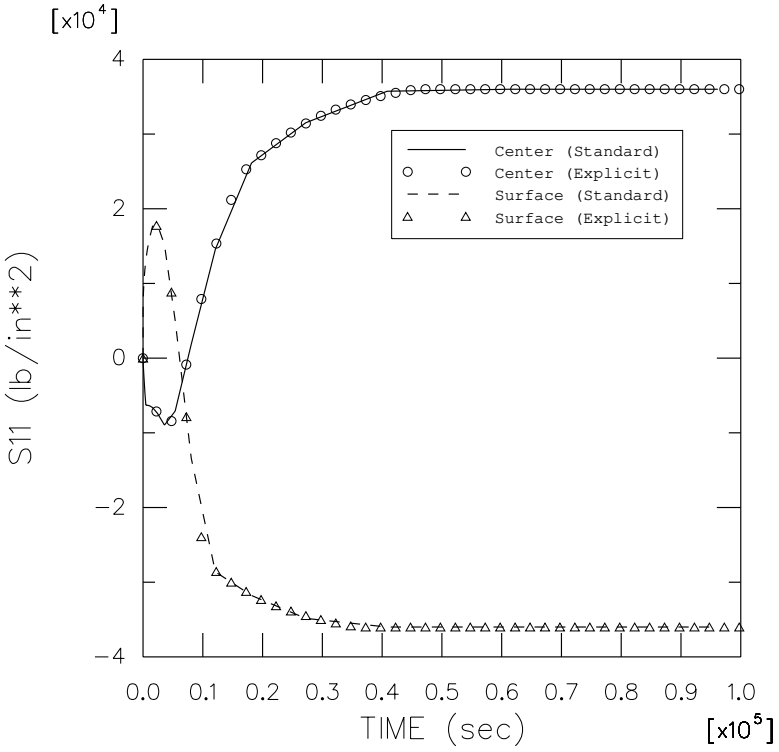


Figure 3: Stress history for the plate surface and center.

Two-dimensional elemental cavity radiation view factor calculations

Products: Abaqus/Standard

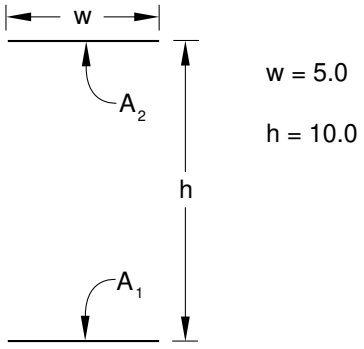
Overview

These examples verify the use of two-dimensional elemental cavity radiation view factor calculations in Abaqus.

Relatively simple configurations were selected for these verification problems to ensure that analytical solutions or tabulated results could be found. In some cases certain parameters such as the distance between two surfaces or the number of elements on a surface were varied to illustrate the effects of these parameters on view factor calculations within Abaqus. To duplicate the tabulated results for the cases where parameters were varied, the user can modify the input files provided with the Abaqus release.

Two infinitely long, directly opposed parallel plates of the same finite width

Problem description



Analytical solution

$$\mathbf{F_{1-2}} = \mathbf{F_{2-1}} = \sqrt{1 + H^2} - H,$$

where $H = h/w$.

Results and discussion

F_{1-2}	
Abaqus	**Analytical**
0.2361	0.2361

Input files

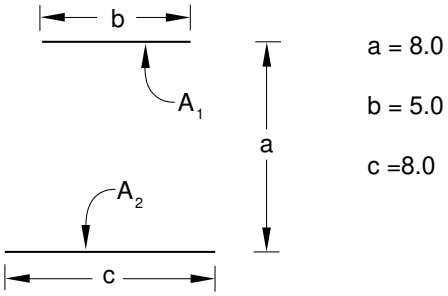
xrvd24n1.inp One DC2D4 element is used to discretize each surface of the cavity.

References

- Howell, J. R., A Catalog of Radiation Configuration Factors, McGraw-Hill Book Company, New York, 1982.

Two infinitely long parallel plates of different widths; the centerlines of each plate are connected by the perpendicular between the plates

Problem description



Analytical solution

$$\mathrm{F}_{1-2} = \frac{1}{2B}\left[\sqrt{(B+C)^2+4} - \sqrt{(C-B)^2+4}\right],$$

where $B = b/a$ and $C = c/a$.

Results and discussion

F_{1-2}	
Abaqus	**Analytical**
0.4337	0.4337

Input files

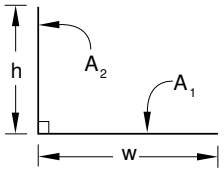
xrvd24n2.inp One DC2D4 element is used to discretize each surface of the cavity.

References

- Howell, J. R., A Catalog of Radiation Configuration Factors, McGraw-Hill Book Company, New York, 1982.

Two infinitely long plates of unequal widths h and w, having one common edge and at an angle of 90° to each other

Problem description



Analytical solution

$$\mathbf{F}_{1-2} = \frac{1}{2}\left(1 + H - \sqrt{1+H^2}\right),$$

where $H = h/w$.

Results and discussion

F_{1-2}	
Abaqus	**Analytical**
0.2229	0.2229

Input files

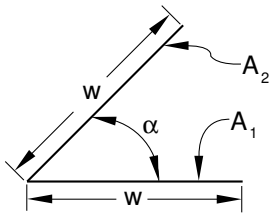
xrvd24n3.inp One DC2D4 element is used to discretize each surface of the cavity.

References

- Siegel, R., and J. R. Howell, Thermal Radiation Heat Transfer, Hemisphere Publishing Corporation, Washington, 3rd, 1992.

Two infinitely long plates of equal finite width w, having one common edge and having an included angle of to each other

Problem description



Analytical solution

$$\mathbf{F}_{1-2} = \mathbf{F}_{2-1} = 1 - \sin\frac{\alpha}{2}.$$

Results and discussion

In *xrvd24n4.inp*α can be varied to obtain the following results (in *xrvd24m4.inp* the angle is varied using prescribed rotational motion):

α	F_{1-2}	
	Abaqus	**Analytical**
10°	0.9128	0.9128
20°	0.8264	0.8264
30°	0.7412	0.7412
40°	0.6580	0.6580
50°	0.5774	0.5774
60°	0.5000	0.5000
70°	0.4264	0.4264
80°	0.3572	0.3572
90°	0.2929	0.2929

Input files

xrvd24n4.inp One DC2D4 element is used to discretize each surface of the cavity; $\alpha = 60°$.

xrvd24m4.inp One DC2D4 element is used to discretize each surface of the cavity; the **MOTION*, ROTATION option is used to vary the angle between the plates.

References

- Siegel, R., and J. R. Howell, Thermal Radiation Heat Transfer, Hemisphere Publishing Corporation, Washington, 3rd, 1992.

Axisymmetric elemental cavity radiation view factor calculations

Products: Abaqus/Standard

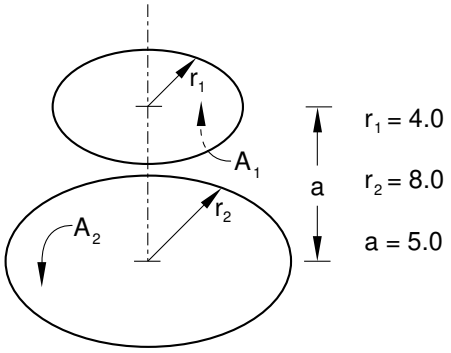
Overview

These examples verify the use of axisymmetric elemental cavity radiation view factor calculations in Abaqus.

Relatively simple configurations were selected for these verification problems to ensure that analytical solutions or tabulated results could be found. In some cases certain parameters such as the distance between two surfaces or the number of elements on a surface were varied to illustrate the effects of these parameters on view factor calculations within Abaqus. To duplicate the tabulated results for the cases where parameters were varied, the user can modify the input files provided with the Abaqus release.

Parallel circular disks with centers along the same normal

Problem description



Analytical solution

$$\mathbf{F}_{1-2} = \frac{1}{2}\left(X - \sqrt{X^2 - 4\left(\frac{R_2}{R_1}\right)^2}\right)$$

and

$$\mathbf{F}_{2-1} = \frac{1}{4}\mathbf{F}_{1-2},$$

where $R_1 = r_1/a$, $R_2 = r_2/a$, and $X = 1 + \frac{1+R_2^2}{R_1^2}$.

Results and discussion

The number of elements along the bottom area can be varied to obtain the following results:

# of elements on bottom plane	$\mathbf{F}_{1-2}$		$\mathbf{F}_{2-1}$	
	Abaqus	**Analytical**	**Abaqus**	**Analytical**
1	0.6853	0.6800	0.1713	0.1700

# of elements on bottom plane	F_{1-2}		F_{2-1}	
	Abaqus	Analytical	Abaqus	Analytical
2	0.6836	0.6800	0.1709	0.1700
4	0.6820	0.6800	0.1705	0.1700

Input files

xrvda4n1_1elem.inp DCAX4 elements are used to discretize the surfaces of the cavity; one element for the top surface and one element for the bottom surface.

xrvda4n1.inp DCAX4 elements are used to discretize the surfaces of the cavity; one element for the top surface and two elements for the bottom surface.

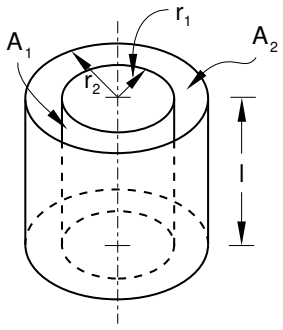
xrvda4n1_4elem.inp DCAX4 elements are used to discretize the surfaces of the cavity; one element for the top surface and four elements for the bottom surface.

References

- Siegel, R., and J. R. Howell, Thermal Radiation Heat Transfer, Hemisphere Publishing Corporation, Washington, 3rd, 1992.

Two concentric cylinders of same finite length

Problem description



$r_1 = 8.0$

$r_2 = 16.0$

$l = 5.0$

Analytical solution

$$\mathbf{F_{2-1}} = \frac{1}{R} - \frac{1}{\pi R}\left(\arccos\frac{B}{A} - \frac{1}{2L}\left[\sqrt{(A+2)^2-(2R)^2}\,\arccos\frac{B}{RA} + B\arcsin\frac{1}{R} - \frac{\pi A}{2}\right]\right),$$

and

$$\mathrm{F}_{2-2} = 1 - \frac{1}{R} + \frac{2}{\pi R}\arctan\frac{2\sqrt{R^2-1}}{L} - \frac{L}{2\pi R}\left(\frac{\sqrt{(2R)^2+L^2}}{L}\arcsin\frac{4\left(R^2-1\right)+\left(L^2/R^2\right)\left(R^2-2\right)}{L^2+4\left(R^2-1\right)} - \arcsin\frac{R^2-2}{R^2} + \frac{\pi}{2}\left[\frac{\sqrt{(2R)^2+L^2}}{L}-1\right]\right),$$

where for any argument θ, $-\frac{\pi}{2} \le \arcsin\theta \le \frac{\pi}{2}$, and $0 \le \arccos\theta \le \pi$; and where $R = r_2/r_1$, $L = l/r_1$, $A = L^2 + R^2 - 1$, and $B = L^2 - R^2 + 1$.

Results and discussion

F_{2-1}		F_{2-2}	
Abaqus	**Analytical**	**Abaqus**	**Analytical**
0.1645	0.1626	0.0954	0.0925

Input files

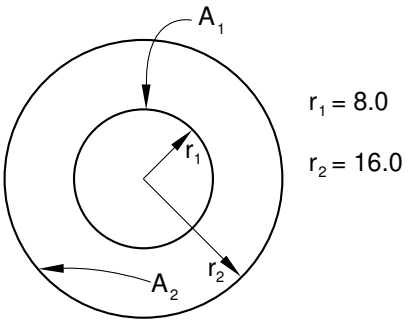
xrvda4n2.inp One DCAX4 element is used to discretize each surface of the cavity.

References

- Siegel, R., and J. R. Howell, Thermal Radiation Heat Transfer, Hemisphere Publishing Corporation, Washington, 3rd, 1992.

Concentric cylinders of infinite length

Problem description



Analytical solution

$$\mathbf{F_{1-2}} = 1.0,$$
$$\mathbf{F_{2-1}} = \frac{r_1}{r_2},$$
and
$$\mathbf{F_{2-2}} = 1 - \frac{r_1}{r_2}.$$

Results and discussion

The number of elements on each face can be increased to obtain the additional results:

# of elements	F_{1-2}		F_{2-1}		F_{2-2}	
	Abaqus	Analytical	Abaqus	Analytical	Abaqus	Analytical
4	0.9979	1.0000	0.4990	0.5000	0.4895	0.5000
8	0.9998	1.0000	0.4999	0.5000	0.4998	0.5000

Input files

xrvda4p3.inp — Four DCAX4 elements are used to discretize each surface of the cavity. The infinite extent of the cavity is modeled by repeating the elements in the *z*-direction using periodic symmetry (NR = 10).

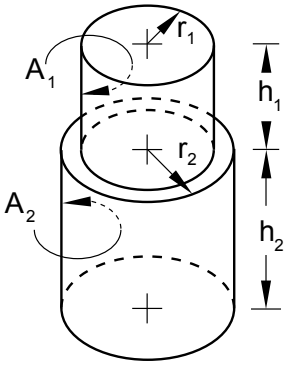
xrvda4p3_8elem.inp — Eight DCAX4 elements are used to discretize each surface of the cavity. The infinite extent of the cavity is modeled by repeating the elements in the *z*-direction using periodic symmetry (NR = 10).

References

- Siegel, R., and J. R. Howell, Thermal Radiation Heat Transfer, Hemisphere Publishing Corporation, Washington, 3rd, 1992.

Coaxial right circular cylinders of different radii, one on top of the other

Problem description



$r_1 = 8.0$

$r_2 = 16.0$

$h_1 = 10.0$

$h_2 = 5.0$

Analytical solution

If $\frac{h_2}{h_1} \le \frac{1}{2}\left(\frac{r_2}{r_1} - 1\right)$ and $r_2 > r_1$,

$$\mathbf{F}_{1-2} = \frac{1}{4H_1}\left(1 - L_2 + \sqrt{(1+L_2)^2 - 4R^2}\right),$$

where $R = r_2/r_1$, $H_n = h_n/r_1$, and $L_n = \frac{r_2^2 + h_n^2}{r_1^2} = R^2 + H_n^2$.

Results and discussion

F_{1-2}	
Abaqus	**Analytical**
0.0502	0.0446

Input files

xrvda4n4.inp DCAX4 elements are used to discretize the surfaces of the cavity; two elements for the top area, and one element for the bottom area.

References

- Howell, J. R., A Catalog of Radiation Configuration Factors, McGraw-Hill Book Company, New York, 1982.

Three-dimensional elemental cavity radiation view factor calculations

Products: Abaqus/Standard

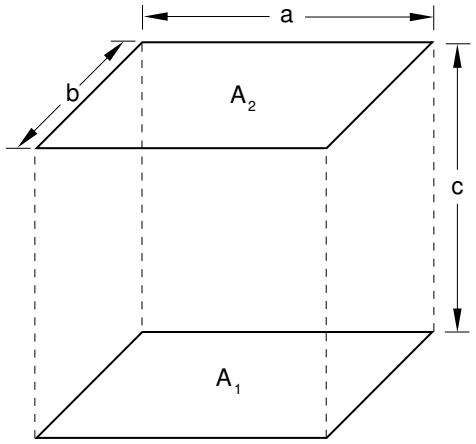
Overview

These examples verify the use of three-dimensional elemental cavity radiation view factor calculations in Abaqus.

Relatively simple configurations were selected for these verification problems to ensure that analytical solutions or tabulated results could be found. In some cases certain parameters such as the distance between two surfaces or the number of elements on a surface were varied to illustrate the effects of these parameters on view factor calculations within Abaqus. To duplicate the tabulated results for the cases where parameters were varied, the user can modify the input files provided with the Abaqus release.

Identical, directly opposed parallel rectangles

Problem description



a = 5.0

b = 8.0

Analytical solution

$$\mathrm{F}_{1-2} = \frac{2}{\pi XY}\left(\ln\left[\frac{\left(1+X^2\right)\left(1+Y^2\right)}{1+X^2+Y^2}\right]^{\frac{1}{2}} + X\sqrt{1+Y^2}\,\arctan\frac{X}{\sqrt{1+Y^2}} \right.$$

$$\left. +Y\sqrt{1+X^2}\,\arctan\frac{Y}{\sqrt{1+X^2}} - X\arctan X - Y\arctan Y \right),$$

where $X = a/c$ and $Y = b/c$.

Results and discussion

- One element per area (*xrvd38n1.inp*, *xrvd38m1.inp*, *xrvds4n1.inp* and *xrvds8n1.inp*); *c* can be varied to obtain the following results:

c	F_{1-2}	
	Abaqus	**Analytical**
1	0.7370	0.7374
3	0.4236	0.4237
6	0.2090	0.2090
10	0.1001	0.1001
15	0.0502	0.0502
25	0.0195	0.0195
35	0.0102	0.0102
40	0.0078	0.0078

- Two elements per area (*xrvd38n2.inp*); c can be varied to obtain the following results:

c	F_{1-2}	
	Abaqus	**Analytical**
1	0.7370	0.7374
3	0.4236	0.4237
6	0.2090	0.2090
10	0.1011	0.1001
15	0.0502	0.0502
25	0.0197	0.0195
35	0.0102	0.0102
40	0.0078	0.0078

- The Abaqus results for c = 15 are 0.0502 (*xrvds3n1.inp* and *xrvds6n1.inp*).

Input files

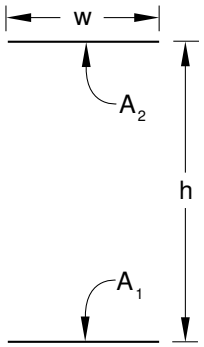
xrvd38n1.inp	One DC3D8 element is used to discretize each surface of the cavity; c = 15.
xrvd38m1.inp	One DC3D8 element is used to discretize each surface of the cavity; the **MOTION* option is used to vary the distance between the rectangles.
xrvd38m1.f	User subroutine *UMOTION* used in xrvd38m1.inp.
xrvd38n2.inp	Two DC3D8 elements are used to discretize each surface of the cavity; c = 6.
xrvds3n1.inp	Two DS3 elements are used to discretize each surface of the cavity; c = 15.
xrvds4n1.inp	One DS4 element is used to discretize each surface of the cavity; c = 15.
xrvds6n1.inp	Two DS6 elements are used to discretize each surface of the cavity; c = 15.
xrvds8n1.inp	One DS8 element is used to discretize each surface of the cavity; c = 15.

References

- Howell, J. R., A Catalog of Radiation Configuration Factors, McGraw-Hill Book Company, New York, 1982.

Two infinitely long, directly opposed parallel plates of the same finite width

Problem description



w = 5.0

h = 10.0

Analytical solution

$$\mathbf{F}_{1-2} = \mathbf{F}_{2-1} = \sqrt{1+H^2} - H,$$

where $H = h/w$.

Results and discussion

F_{1-2}	
Abaqus	**Analytical**
0.2356	0.2361

Input files

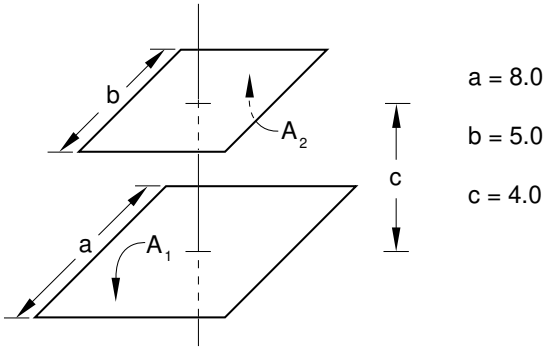
xrvd38p3.inp One DC3D8 element is used to discretize each surface of the cavity. The infinite extent of the cavity is modeled with three-dimensional periodic symmetry (NR = 15).

References

- Howell, J. R., A Catalog of Radiation Configuration Factors, McGraw-Hill Book Company, New York, 1982.

Coaxial parallel squares of different sizes

Problem description



Analytical solution

$$\mathrm{F}_{1-2} = \frac{(AB)^2}{\pi} \quad \text{for} \quad A < 0.2$$

and

$$\mathrm{F}_{1-2} = \frac{1}{\pi A^2}\left(\ln \frac{\left[A^2\left(1+B^2\right)+2\right]^2}{\left(Y^2+2\right)\left(X^2+2\right)} + \sqrt{Y^2+4}\left[Y \arctan \frac{Y}{\sqrt{Y^2+4}} - X \arctan \frac{X}{\sqrt{Y^2+4}}\right] \right.$$

$$\left. + \sqrt{X^2+4}\left[X \arctan \frac{X}{\sqrt{X^2+4}} - Y \arctan \frac{Y}{\sqrt{X^2+4}}\right] \right) \quad \text{for} \quad A \geq 0.2,$$

where $A = a/c$, $B = b/c$, $X = A\left(1+B\right)$, and $Y = A\left(1-B\right)$. Reference solution: F_{2-1} = 0.4974.

Results and discussion

Abaqus results for F_{2-1}: 0.4974 (*xrvd38n4.inp*); 0.4974 (*xrvds3n4.inp*).

Input files

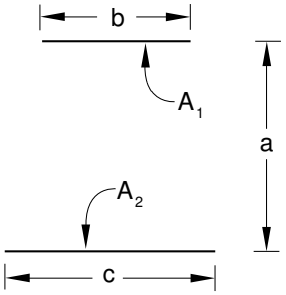
xrvd38n4.inp	One DC3D8 element is used to discretize each surface of the cavity.
xrvds3n4.inp	Two DS3 elements are used to discretize each surface of the cavity.

References

- Howell, J. R., A Catalog of Radiation Configuration Factors, McGraw-Hill Book Company, New York, 1982.

Two infinitely long parallel plates of different widths; the centerlines of each plate are connected by the perpendicular between the plates

Problem description



a = 8.0

b = 5.0

c =8.0

Analytical solution

$$\mathbf{F_{1-2}} = \frac{1}{2B}\left[\sqrt{(B+C)^2+4} - \sqrt{(C-B)^2+4}\right],$$

where $B = b/a$ and $C = c/a$.

Results and discussion

F_{1-2}	
Abaqus	**Analytical**
0.4335	0.4337

Input files

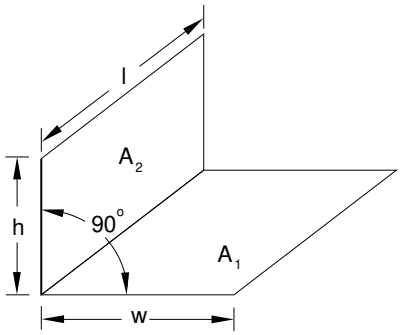
xrvd38p5.inp One DC3D8 element is used to discretize each surface of the cavity. The infinite extent of the cavity is modeled with three-dimensional periodic symmetry (NR = 15).

References

- Howell, J. R., A Catalog of Radiation Configuration Factors, McGraw-Hill Book Company, New York, 1982.

Two finite rectangles of the same length, having one common edge and at an angle of 90° to each other

Problem description



h = 5.0

l = 10.0

w = 8.0

Analytical solution

$$\mathrm{F}_{1-2} = \frac{1}{\pi W}\left(W \arctan\frac{1}{W} + H \arctan\frac{1}{H} - \sqrt{H^2+W^2}\,\arctan\frac{1}{\sqrt{H^2+W^2}} \right.$$
$$\left. + \frac{1}{4}\ln\left[\frac{\left(1+W^2\right)\left(1+H^2\right)}{1+W^2+H^2} \left\{ \frac{W^2\left(1+W^2+H^2\right)}{\left(1+W^2\right)\left(W^2+H^2\right)} \right\}^{W^2} \left\{ \frac{H^2\left(1+H^2+W^2\right)}{\left(1+H^2\right)\left(H^2+W^2\right)} \right\}^{H^2} \right] \right),$$

where $H = h/l$ and $W = w/l$. Reference solution: F_{1-2} = 0.1746.

Results and discussion

Abaqus results for F_{1-2}: 0.1746 (*xrvd38n6.inp*); 0.1746 (*xrvds6n6.inp*).

Input files

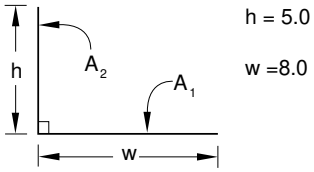
xrvd38n6.inp	One DC3D8 element is used to discretize each surface of the cavity.
xrvds6n6.inp	Two DS6 elements are used to discretize each surface of the cavity.

References

- Siegel, R., and J. R. Howell, Thermal Radiation Heat Transfer, Hemisphere Publishing Corporation, Washington, 3rd, 1992.

Two infinitely long plates of unequal widths *h* and *w*, having one common edge and at an angle of 90° to each other

Problem description



Analytical solution

$$\mathrm{F}_{1-2} = \tfrac{1}{2}\left(1 + H - \sqrt{1+H^2}\right),$$

where $H = h/w$.

Results and discussion

F_{1-2}	
Abaqus	**Analytical**
0.2221	0.2229

Input files

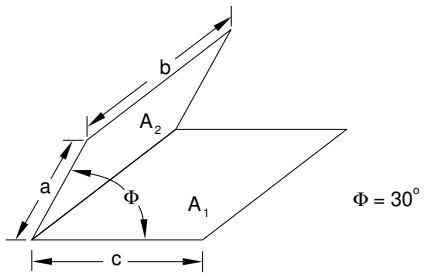
xrvd38p7.inp One DC3D8 element is used to discretize each surface of the cavity. The infinite extent of the cavity is modeled with three-dimensional periodic symmetry (NR = 15).

References

- Siegel, R., and J. R. Howell, Thermal Radiation Heat Transfer, Hemisphere Publishing Corporation, Washington, 3rd, 1992.

Two rectangles with one common edge and an included angle of

Problem description



Analytical solution

Definitions: $A = a/b$; $C = c/b$.

C	F_{1-2}		
	A = 0.6	A = 1.0	A = 2.0
0.1	0.894753	0.898003	0.899505
0.2	0.859340	0.868201	0.871800
0.4	0.777610	0.812110	0.822722
0.6	0.665734	0.754703	0.778772
1.0	0.452822	0.619028	0.700100
2.0	0.233632	0.350050	0.521308
4.0	0.117384	0.177461	0.286713
6.0	0.078311	0.118499	0.192535
10.0	0.047002	0.071148	0.115803
20.0	0.023504	0.035583	0.057945

Results and discussion

F_{1-2}	
Abaqus	**Analytical**
0.6195	0.6190

Input files

xrvd38n8.inp — One DC3D8 element is used to discretize each surface of the cavity; $a = b = c =$ 8.0.

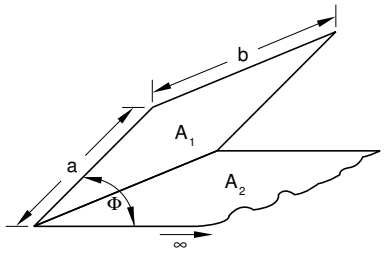
xrvds4n8.inp — One DS4 element is used to discretize each surface of the cavity; $a = b = c =$ 8.0.

References

- Howell, J. R., A Catalog of Radiation Configuration Factors, McGraw-Hill Book Company, New York, 1982.

Rectangles having a common edge and forming an arbitrary angle; one rectangle is infinitely long

Problem description



Analytical solution

Definition: $A = a/b$.

A	F_{1-2}		
	$\Phi = 30°$	$\Phi = 45°$	$\Phi = 60°$
0.1	0.900022	0.804838	0.690483
0.2	0.872918	0.767740	0.648105
0.4	0.825360	0.706295	0.581494
0.6	0.783499	0.655351	0.529168
1.0	0.711717	0.573951	0.450407
2.0	0.579571	0.441004	0.332686
4.0	0.426592	0.307875	0.225049
6.0	0.341612	0.240643	0.173501
10.0	0.249219	0.171450	0.121970
20.0	0.154976	0.104189	0.073151

Results and discussion

F_{1-2}	
Abaqus	**Analytical**
0.5732	0.5740

Input files

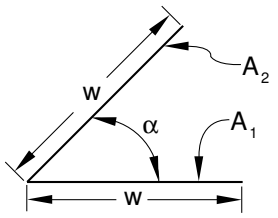
xrvd38n9.inp DC3D8 elements are used to discretize the surfaces of the cavity; one element for the finite surface and nine elements with an edge length of eight units for the infinite surface; $a = b =$ 10.0; $\Phi =$ 45°.

References

- Howell, J. R., A Catalog of Radiation Configuration Factors, McGraw-Hill Book Company, New York, 1982.

Two infinitely long plates of equal finite width *w*, having one common edge and having an included angle of _ to each other

Problem description



w = 8.0

Analytical solution

$$F_{1-2} = F_{2-1} = 1 - \sin\frac{\alpha}{2}.$$

Results and discussion

For this test three parameters can be varied: the angle, the number of reflections, and the number of elements used to model the bottom plate. All of the variations shown in the following tables can be verified by modifying input file *xrvd38p0.inp*.

- One element per plate, $\alpha =$ 60°:

NR	F_{1-2}	
	Abaqus	**Analytical**
2	0.4969	0.5000
4	0.4900	0.5000
8	0.4993	0.5000
12	0.4994	0.5000

NR	F_{1-2}	
	Abaqus	Analytical
16	0.4994	0.5000
20	0.4994	0.5000

- One element per plate, NR= 12:

α	F_{1-2}	
	Abaqus	Analytical
10°	0.9123	0.9128
20°	0.8258	0.8264
30°	0.7406	0.7412
40°	0.6574	0.6580
50°	0.5768	0.5774
60°	0.4994	0.5000
70°	0.4258	0.4264
80°	0.3566	0.3572
90°	0.2923	0.2929

- NR= 2, $\alpha =$ 60°:

# of elements on bottom plate	F_{1-2}	
	Abaqus	Analytical
1	0.4969	0.5000
3	0.4969	0.5000
6	0.4969	0.5000
9	0.4969	0.5000
12	0.4969	0.5000
15	0.4969	0.5000

Input files

xrvd38p0.inp One DC3D8 element is used to discretize each surface of the cavity. The infinite extent of the cavity is modeled with three-dimensional periodic symmetry (NR = 12); $\alpha =$ 60°.

References

- Siegel, R., and J. R. Howell, Thermal Radiation Heat Transfer, Hemisphere Publishing Corporation, Washington, 3rd, 1992.

Radiation analysis of a plane finned surface

Products: Abaqus/Standard

Overview

This example illustrates the Abaqus capability to solve heat transfer problems including cavity radiation.
In this example we simulate the effects of a fire condition on a plane finned surface. This problem was proposed by Glass et al. (1989) as a benchmark for thermal radiation. We compare their results with those obtained using Abaqus.

The configuration shown in *Figure 1* represents a plane wall with a uniform array of parallel rectangular fins attached. The problem represents three phases in a fire test. The first is the pretest, a steady-state condition where heat is transferred by natural convection from an internal fluid at a fixed temperature of 100°C to the plane inside wall. Heat is conducted through the wall and dissipated by radiation and natural convection from the outside wall and fin surfaces to the surrounding medium, which is at a temperature of 38°C. The second phase is a 30-minute fire transient, where heat is supplied by radiation and forced convection from a hot external fluid at 800°C. After conduction through the fins and wall, heat is rejected by natural convection to the internal fluid. Finally, the third phase is a 60-minute cool down period, where heat absorbed during the fire transient is rejected to the surroundings by the same process as that used to establish the initial steady-state condition.

Problem description

The finite element mesh used for the wall and fins is shown in *Figure 2*. By making use of the radiation periodic symmetry capability in Abaqus, we are able to represent the array of fins while meshing only one fin and corresponding wall section.

The outside ambient is modeled with a single horizontal row of elements at some distance above the top of the fin (not shown in the figure). The varying ambient temperature is simulated by prescribing temperatures to the nodes of these elements. The elements representing the outside ambient are also assigned a surface emissivity of 1.0.

Material and boundary conditions

The thermal conductivity of the wall and fins is 50 W/m°C (k), their specific heat is 500 J/kg°C (c), and the density is 7800 kg/m^3 (ρ). The surface emissivity of the wall and fins is 0.8, the Stefan-Boltzmann radiation constant is 5.6697×10^{-8} W/m^2°K^4, and the temperature of absolute zero is −273°C.

The natural convection between the internal fluid and the inside of the wall is modeled with a film boundary condition where the film coefficient is given as $500(\theta_w - \theta_f)^{1/3}$ W/m^2°C, where θ_w is the inside wall temperature and θ_f is the temperature of the internal fluid. The film boundary condition user subroutine is used for this purpose since the film condition is temperature dependent.

The natural convection between the outside finned surface and its surroundings is modeled with a film boundary condition where the film coefficient is given as $2(\theta_s - \theta_a)^{1/3}$ W/m^2°C, where θ_s is the temperature of the finned surface and θ_a is the outside ambient temperature. Again, the film boundary condition user subroutine is employed. The forced convection between the hot surroundings and the finned surface is modeled with a constant film coefficient of 10 W/m^2°C.

Loading

The first simulation step is a steady-state heat transfer analysis to establish the initial pretest conditions. This is followed by a 30-minute transient heat transfer analysis during which time the ambient fire temperature is 800°C. Finally, a second transient heat transfer step is performed to simulate the 60-minute cool down period.

The integration procedure used in Abaqus for transient heat transfer analysis procedures introduces a relationship between the minimum usable time increment and the element size and material properties. The guideline given in *Uncoupled Heat Transfer Analysis* is

$$\Delta t > \frac{\rho c}{6k}\Delta l^2,$$

where Δl is the element size. This suggests that an initial time increment of 10 seconds is appropriate for the transient steps of this problem. Automatic time incrementation is chosen for the transient steps by setting DELTMX to 5°C. DELTMX controls the time integration by limiting the temperature change allowed at any point during an increment.

Results and discussion

The Glass et al. reference summarizes numerical results for this model for many heat transfer codes (all of which give similar results) along with the mean and standard deviation among the different codes. *Table 1* shows a comparison of the results obtained by Abaqus with the corresponding mean values reported by Glass et al. Table 5.1.5–1 also indicates the standard deviation reported by Glass et al. among the codes considered in that reference; the Abaqus results are within one standard deviation of the mean values reported in Glass et al. in all cases.

Figure 3 shows the history of the temperature at the top of the fin (point 1 in *Figure 1*). *Figure 4* shows the histories of the temperature at the root of the fin (point 2 in *Figure 1*) and on the wall inside surface (point 3). In all cases the results obtained with Abaqus match the TAU results quite well. In *Figure 5* we show the temperature distribution around the fin perimeter (starting at point 1 and ending at point 2) at the end of the fire transient. Again, the Abaqus and TAU results match closely. Finally, temperature contours at the end of the fire transient are shown in *Figure 6*.

Input files

radiationfinnedsurf.inp	Fire transient problem.
radiationfinnedsurf.f	User subroutine *FILM* used in radiationfinnedsurf.inp.

References

- Glass, R. E., et al., “Standard Thermal Problem Set,” *Proceedings of the Ninth International Symposium on the Packaging of Radioactive Materials*, pp. 275–282, June 1989.
- Johnson, D., “Surface to Surface Radiation in the Program TAU, Taking Account of Multiple Reflection,” *United Kingdom Atomic Energy Authority Report ND-R-1444(R)*, 1987.

Tables

Table 1: Comparison of the results obtained by Abaqus with those published by Glass et al.

Step	Locations	Glass et al. (°C)		Abaqus (°C)
		Mean	Standard deviation	
Initial (t=0 s)	Fin tip (point 1)	75.8	0.2	75.7
	Fin root (point 2)	93	0.3	92.9
	Inside surface (point 3)	97	0.1	96.9
End of fire (t=1800 s)	Fin tip (point 1)	652.2	4.9	649.9
	Fin root (point 2)	238.6	6.6	237.2
	Inside surface (point 3)	133.7	1.1	133.6
End of cooldown (t=5400 s)	Fin tip (point 1)	80.4	0.7	80.9
	Fin root (point 2)	95.7	0.5	96.1
	Inside surface (point 3)	98.4	0.2	98.5

Figures

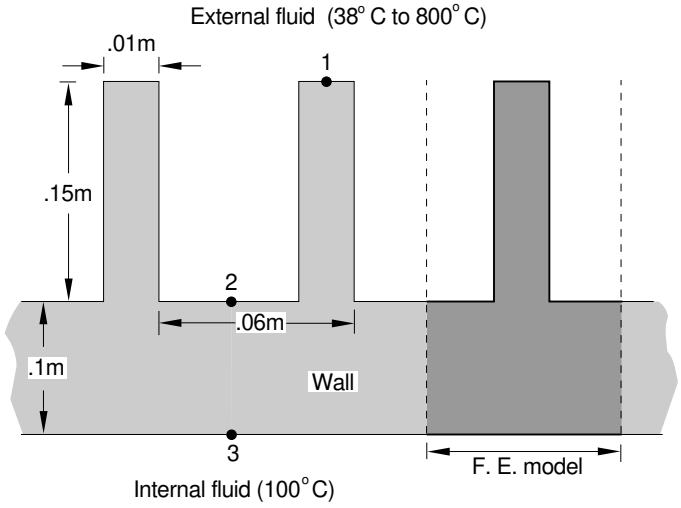


Figure 1: Plane finned surface.

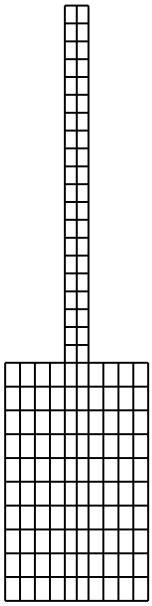


Figure 2: Finite element mesh of fin and inner wall.

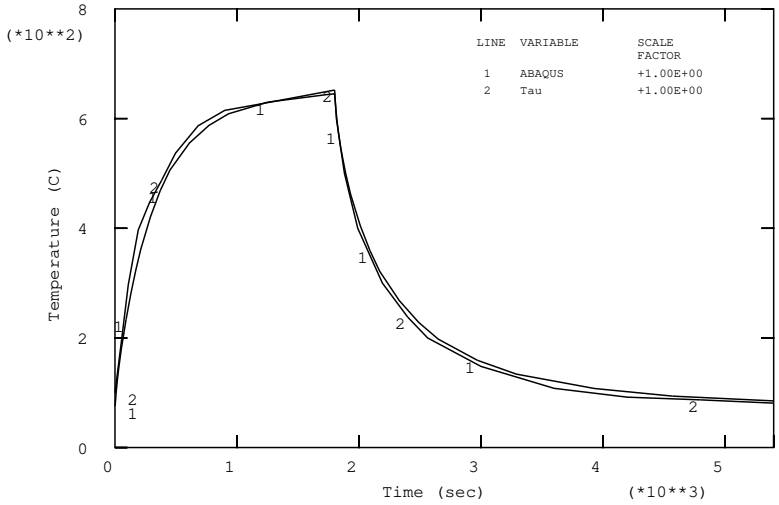


Figure 3: Temperature history at top of fin.

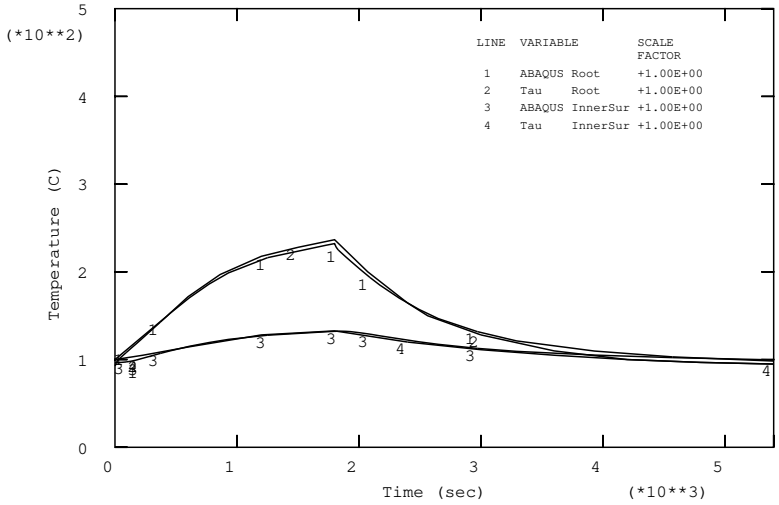


Figure 4: Temperature history at root of fin and inside wall surface.

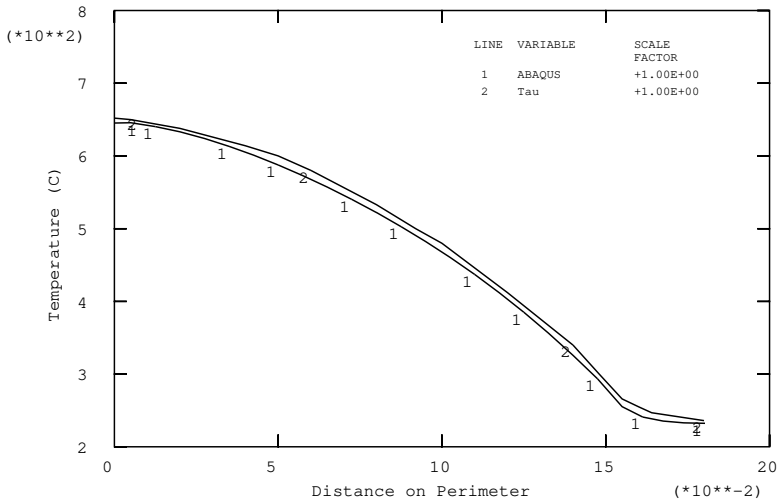


Figure 5: Temperature distribution along fin perimeter at end of fire transient.

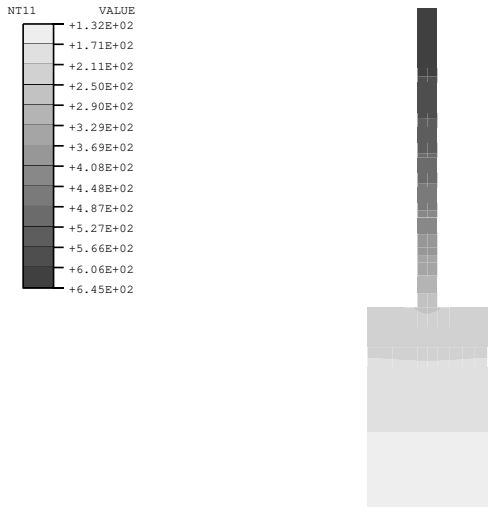


Figure 6: Temperature contours at end of fire transient.

Eulerian analysis

In this section:

- *Eulerian analysis of a collapsing water column*
- *Deflection of an elastic dam under water pressure*

Eulerian analysis of a collapsing water column

Products: Abaqus/Explicit Abaqus/CAE

Overview

This example utilizes the pure Eulerian analysis technique to model a dynamic fluid flow event involving large deformation.
A column of water is subjected to a gravity load, causing the column to collapse and flow along a flat, rigid floor. The analysis results can be compared to experimental results from Martin and Moyce (1952), demonstrating the efficacy of the Eulerian technique and equation of state material models for simulating fluid dynamics in Abaqus/Explicit.

Problem description

The model is created in Abaqus/CAEusing a simple rectangular Eulerian domain measuring 10 × 5 × 0.05 m. Because Eulerian analyses must be conducted in three-dimensional space, it is common to approximate two-dimensional problems using a thin domain with a single Eulerian element through its thickness. Cubic elements provide the best accuracy and performance in Eulerian analyses, so the thickness is chosen to correspond to the height and width of each element in the eventual mesh.

Zero-velocity boundary conditions normal to all the domain faces prevent the flow of material into or out of the domain. The domain is partitioned, and the Eulerian material (water) is assigned to a 2.25 × 4.5 m region along the left edge of the domain (see *Figure 1*).

The water is modeled as a nearly incompressible, viscous Newtonian fluid. The linear $U_s - U_p$ Hugoniot form of the Mie-Grüneisen equation of state is used in the material model. The parameters used to define the material, based on a bulk modulus of approximately 2.246 GPa, are listed in *Table 1*.

In addition to a gravity load applied to the entire Eulerian domain, initial geostatic stresses are defined in the water to model the hydrostatic pressure in the column. Since geostatic stresses cannot be defined directly in Abaqus/CAE, they are added to the model using the **Keywords Editor**.

The Eulerian domain is finely meshed with a grid of 222 × 111 Eulerian EC3D8R elements.

Results and discussion

In the Visualization module of Abaqus/CAE, an isosurface view cut based on the Eulerian volume fraction for water (output variable EVF_WATER) is used to visualize the progression of the column collapse within the Eulerian mesh, as shown in *Figure 2*. The results can be compared to experimental data from a similar physical model by Martin and Moyce (1952). The trends of the surge front in the experimental case and in the Abaqus case are similar (see *Figure 3*). Given the potential inaccuracies associated with the experimental measurement techniques, as documented by Martin and Moyce, the two cases agree reasonably well.

The surge front can be tracked in the Visualization module of Abaqus/CAE by investigating the Eulerian volume fraction for water in the elements along the bottom of the mesh. The tracking can be done manually or by creating *X*–*Y* data objects from output variable EVF_WATER and using mathematical operations to convert the volume fraction to an associated distance.

Figure 4 illustrates the dynamics of the water by contouring the velocity in the horizontal direction (V1) and the velocity in the vertical direction (V3) during the collapse. The water in the column moves downward under the gravity load (with the free edge of the column falling faster than the edge along the wall), forcing an accelerating surge in the horizontal direction.

Python scripts

eulerian_column_model.py	Script to generate the model in Abaqus/CAE.

Input files

eulerian_column.inp	Input file for the model.

References

- Martin, J. C., and W. J. Moyce, "An Experimental Study of the Collapse of Liquid Columns on a Rigid Horizontal Plane," *Philosophical Transactions of the Royal Society of London, Series A, Mathematical and Physical Sciences*, vol. 244, no. 882, pp. 312–324, 1952.

Tables

Table 1: Material parameters for water.

Parameter	Value
Density (ρ)	998.2 kg/m^3
Viscosity (η)	0.001003 N s/m^2
c_0	1500 m/s
s	0
Γ_0	0

Figures

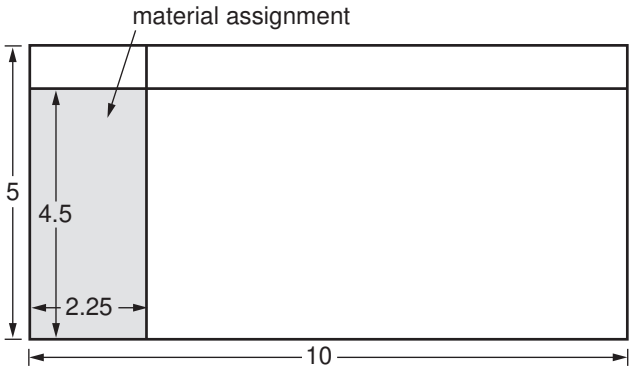


Figure 1: Geometry of the Eulerian domain. All dimensions are in meters.

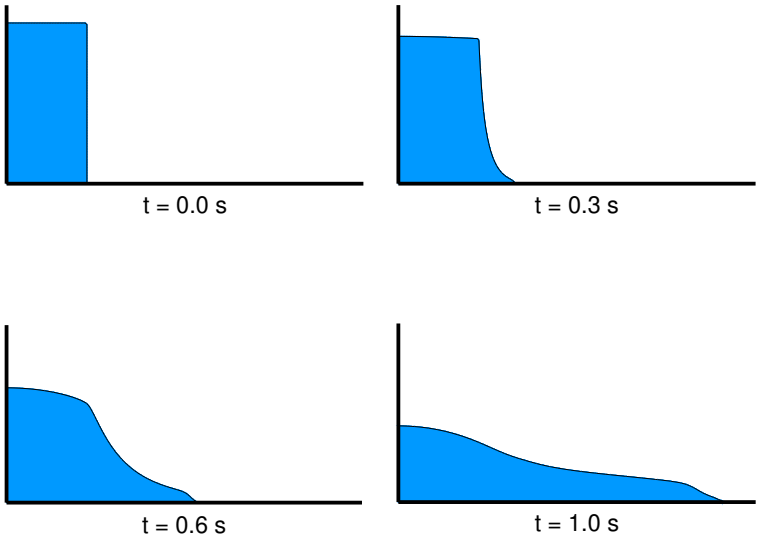


Figure 2: Deformation of the water column under gravity loading.

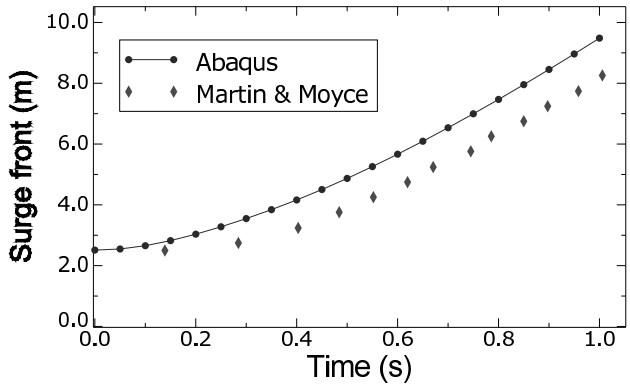


Figure 3: Results in Abaqus compared to experimental results.

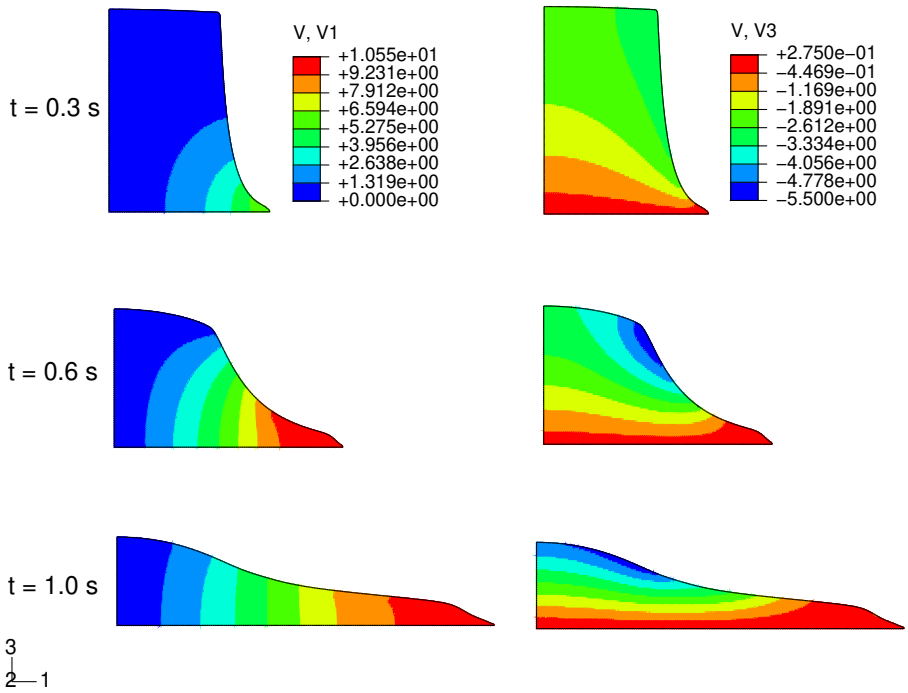


Figure 4: Water velocity in the horizontal (V1) and vertical (V3) directions.

Deflection of an elastic dam under water pressure

Products: Abaqus/Explicit Abaqus/CAE

References:

- *Eulerian Analysis*
- *Defining Adaptive Mesh Refinement in the Eulerian Domain*

Overview

This problem investigates the response of fluid under gravity loading passing through a flexible dam gate.
In this example the fluid undergoes extreme deformation during the simulation; therefore, it is modeled as an Eulerian fluid. The significantly stiffer dam is modeled with Lagrangian elements. The coupled Eulerian-Lagrangian (CEL) analysis technique is used to account for the interaction between the fluid and the dam. The response of the dam is compared to experimental results.

Problem description

In the initial configuration of the model, water is located in a rectangular reservoir. The floor and right side of the reservoir are fixed. On the left side the water is contained by an elastic dam wall; the top half of the dam is fixed, but the bottom half is unconstrained. Under a gravity load the water pushes against the dam, deflects the bottom portion of the dam, and flows freely out of the reservoir.

The model is created in Abaqus/CAEusing two parts. An Eulerian part represents the domain within which the water will flow. A Lagrangian part represents the dam. The problem is essentially two-dimensional with horizontal (*X*-direction) and vertical (*Z*-direction) components; but because Eulerian elements must be three-dimensional, all parts are modeled with a thickness in the *Y*-direction equivalent to one Eulerian element.

The Eulerian part is shown in *Figure 1*. *Figure 2* shows the distribution of material within the part: the region on the right is filled with water, the region on the left is the anticipated outflow region, and the middle region contains the Lagrangian dam. Zero-velocity boundary conditions are applied normal to the floor and right side of the Eulerian part to prevent water from flowing out of these boundaries. No boundary conditions are applied along the left side of the dam; water is free to flow out of the part at this interface (which results in a corresponding decrease in total mass for the model). Zero-velocity boundary conditions are applied in the horizontal and vertical directions on the upper half of the dam, but the bottom half is free to deflect (see *Figure 2*). Another set of zero-velocity boundary conditions in the *Y*-direction are applied to each part to prevent movement out of the two-dimensional plane. A frictionless general contact definition enforces contact between the water and the dam.

The dam is modeled as an elastic material with Young's modulus of 1.2×10^7 N/m^2, Poisson's ratio of 0.4, and density of 1100 kg/m^3. The water is defined using the linear $U_s - U_p$ Hugoniot form of the Mie-Grüneisen equation of state with the parameters listed in *Table 1*.

A gravitational load is applied to the entire model. In addition, initial geostatic stresses are defined to model the hydrostatic pressure in the water. In this problem, the geostatic stresses are added to the model in Abaqus/CAE using the **Keywords Editor**.

The Eulerian part is meshed with EC3D8R elements using a global mesh seed of 5 mm; this global mesh seed allows a uniform distribution of cube-shaped elements throughout the part, which greatly improves the accuracy of the Eulerian analysis. The dam is meshed with C3D8R elements in a grid measuring 75 × 4; there are three elements through the thickness of the part. Multiple elements through the width and thickness of the dam are necessary to ensure that its bending behavior is captured adequately.

Results and discussion

The pressure of the water under gravity deflects the bottom half of the dam, allowing water to flow out of the reservoir. *Figure 3* uses an isosurface view cut based on output variable EVF_WATER to show the position of the water at six points during the analysis. The displacement in the horizontal and vertical directions of the lower-right corner of the dam can be compared to experimental results from Antoci et al. (2007), as shown in *Figure 4*. The Abaqus results agree well with the experimental results. Discrepancies are likely due to idealizations in the dam material model, rendering it slightly more flexible than the experimental dam. Antoci et al. also note some minor flaws in the experimental setup that could result in decreased water pressure on the dam, which in turn would lead to a lower overall deflection.

Python scripts

dam_deflection_cel.py	Script to generate the model in Abaqus/CAE.

Input files

dam_deflection_cel.inp	Input file for the model.
dam_deflection_cel_enhcont.inp	Input file using the enhanced contact formulation.

References

- Antoci, C., M. Gallati, and S. Sibilla, “Numerical Simulation of Fluid-Structure Interaction by SPH,” *Computer and Structures*, vol. 85, pp. 879–890, 2007.

Tables

Table 1: Material parameters for water.

Parameter	Value
Density (ρ)	1000 kg/m^3
Viscosity (η)	0.001 N s/m^2
c_0	1500 m/s
s	0
Γ_0	0

Figures

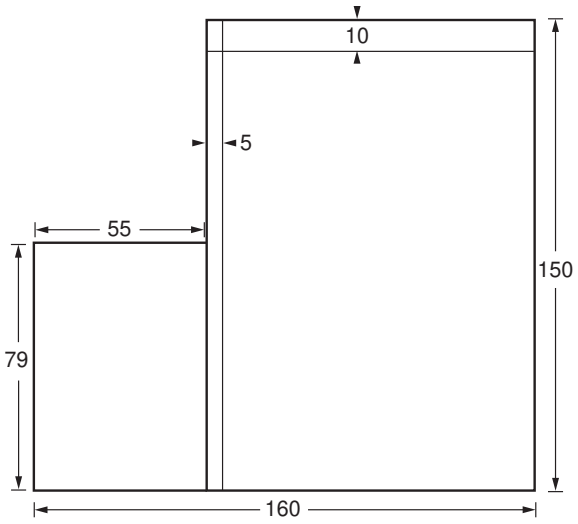


Figure 1: Geometry for the Eulerian part. All dimensions are in millimeters.

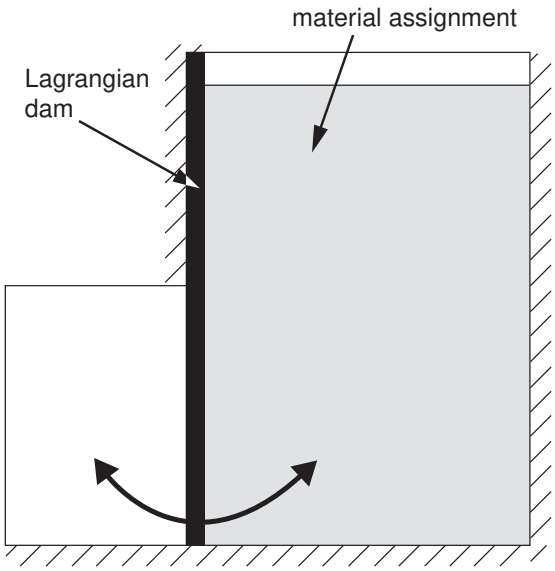


Figure 2: Material assignments and boundary conditions in the assembled model.

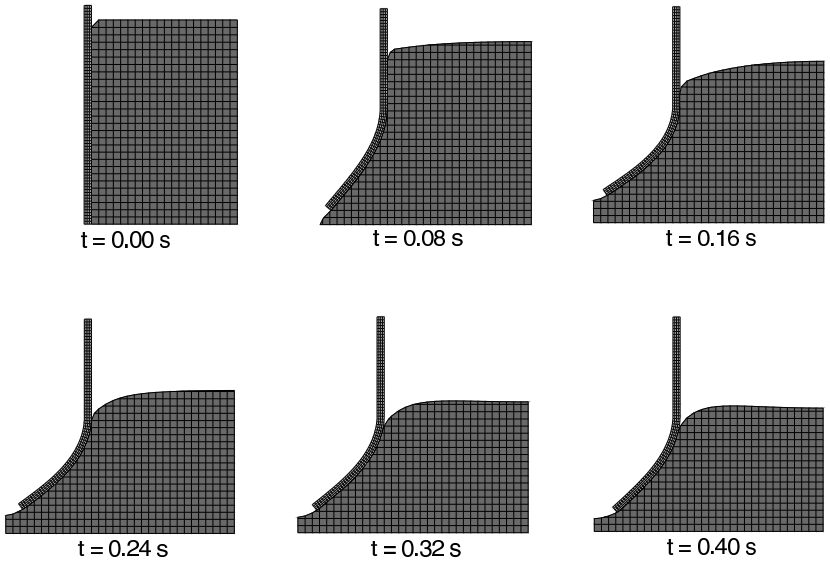


Figure 3: Flow of the water and resulting deformation of the dam.

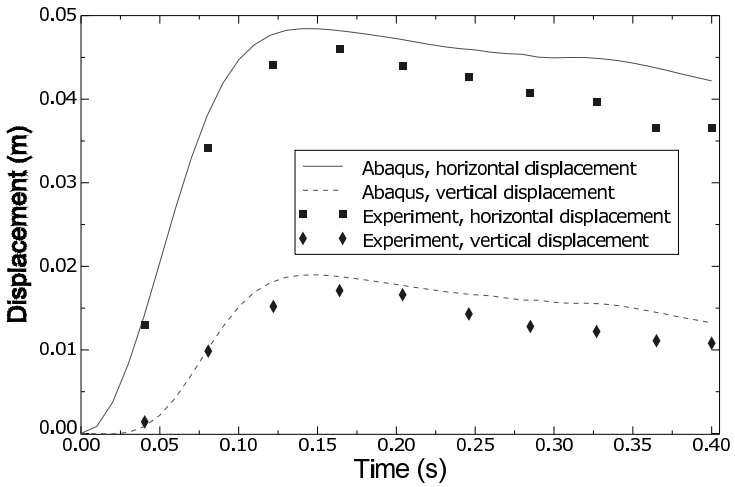


Figure 4: Displacement of the lower-right corner of the dam.

Electromagnetic analysis

In this section:

- *Eigenvalue analysis of a piezoelectric cube with various electrode configurations*
- *Modal dynamic analysis for piezoelectric materials*
- *Steady-state dynamic analysis for piezoelectric materials*
- *TEAM 2: Eddy current simulations of long cylindrical conductors in an oscillating magnetic field*
- *TEAM 4: Eddy current simulation of a conducting brick in a decaying magnetic field*
- *TEAM 6: Eddy current simulations for spherical conductors in an oscillating magnetic field*
- *TEAM 13: Three-dimensional nonlinear magnetostatic analysis*
- *Induction heating of a cylindrical rod by an encircling coil carrying time-harmonic current*

Eigenvalue analysis of a piezoelectric cube with various electrode configurations

Products: Abaqus/Standard

References:

- *Piezoelectric analysis*

Overview

This problem examines the vibrational breathing modes of a piezoelectric cube of PZT4 material with multiple configurations of electroded surfaces.
One analysis in this example has two ends of the cube fully electroded, while the second analysis has the two ends only partially electroded. Both the resonant (close-circuited) and antiresonant (open-circuited) frequencies are extracted for both electrode patterns. The elements used are the 8-node and 20-node three-dimensional brick elements.

Problem description

This problem has been used as a basis for verification for finite element piezoelectric capabilities in several references: Boucher et al. (1981), Lerch (1990), and Ostergaard et al. (1986). The structure is a cube consisting of the piezoelectric material PZT4. Each side length of the cube is 20 mm. For the first case the top and bottom surfaces, which are orthogonal to the axis of polarization, are considered to be completely covered with electrodes. In the second case only a portion of the surfaces are covered with electrodes. The portion covered consists of a centered square section with an edge length of 10 mm.

The properties for the materials in the transducer are available in Boucher et al. (1981). These properties for the PZT4 are given as

Elasticity Matrix:

$$\begin{bmatrix} 132 & 71 & 73 & 0 & 0 & 0 \\ 71 & 132 & 73 & 0 & 0 & 0 \\ 73 & 73 & 115 & 0 & 0 & 0 \\ 0 & 0 & 0 & 30 & 0 & 0 \\ 0 & 0 & 0 & 0 & 26 & 0 \\ 0 & 0 & 0 & 0 & 0 & 26 \end{bmatrix} \text{GPa}$$

Piezoelectric Coupling Matrix (Stress Coefficients):

$$\begin{bmatrix} 0 & 0 & -4.1 \\ 0 & 0 & -4.1 \\ 0 & 0 & 14.1 \\ 0 & 0 & 0 \\ 10.5 & 0 & 0 \\ 0 & 10.5 & 0 \end{bmatrix} \text{coulomb/m}^2$$

Dielectric Matrix:

$$\begin{bmatrix} 7.12 & 0 & 0 \\ 0 & 7.12 & 0 \\ 0 & 0 & 5.84 \end{bmatrix} 10^{-9} \quad \text{farad/m}$$

The poling direction is the 3-direction. The electrodes are placed on the faces that are orthogonal to the 3-axis.

See *Eigenvalue analysis of a piezoelectric transducer* for a note on the ordering of the stress components.

Models

If we wished to extract all the natural frequencies of the cube, symmetry could not be utilized in the discretizations. However, in the references used for comparison, only the breathing-type modes are given. This allows the use of some symmetry in the models. An eighth of the cube cannot be used for the distribution of the electrical potentials because they may not be symmetrical about the x–y plane. Therefore, a quarter of the cube is modeled with symmetry about the x–z and y–z planes. The piezoelectric cube is modeled with both the 8-node and 20-node three-dimensional brick elements each with two levels of refinement. The discretizations used are shown in *Figure 1*.

In each analysis constraints are used to ensure that only the modes of interest, the breathing-type modes, are extracted. These constraints are applied as both boundary conditions and equations.

Each level of discretization for each element is analyzed with the two configurations of electrodes. The first has the electrodes fully covering the top and bottom surfaces where these surfaces are those orthogonal to the poling direction. The second configuration has the electrodes partially covering the top and the bottom surfaces. The analyses are performed considering the electrodes to be both closed-and open-circuited. The closed-circuited cases are specified by setting the potentials on both electrodes to zero. This situation yields the resonant frequencies. The open-circuited cases are specified by setting the potentials on only one surface electrode to zero, which allows a different potential to exist on each electrode. This situation yields the antiresonant frequencies.

Results and discussion

The solutions obtained with the Abaqus models, along with the results available from other references, are given in *Table 1* and *Table 2*. In *Table 1* both the resonant and antiresonant frequencies are given for the case with fully covered electrodes. The corresponding results are given for the analyses with the partially covered electrodes in *Table 2*. The two modes of interest are the breathing-type modes described by Boucher et al. (1981). The corresponding mode shapes obtained from Abaqus for the resonant frequencies for the case of completely covered electrodes are shown in *Figure 2*. The model for these mode shapes used the 8-node brick elements in the refined discretization.

The results from Abaqus compare well with the results from the other references. Even the coarser meshes are seen to give reasonable results.

Input files

piezocube_c3d8e_coarse_reson.inp	Coarse mesh with 8-node three-dimensional brick elements for the closed-circuited case for resonant frequencies.
piezocube_c3d8e_coarse_anti.inp	Coarse mesh with 8-node three-dimensional brick elements for the open-circuited case for antiresonant frequencies.
piezocube_c3d8e_fine_reson.inp	Refined mesh with 8-node three-dimensional brick elements for the closed-circuited case for resonant frequencies.
piezocube_c3d8e_fine_anti.inp	Refined mesh with 8-node three-dimensional brick elements for the open-circuited case for antiresonant frequencies.

piezocube_c3d20e_coarse_reson.inp	Coarse mesh with 20-node three-dimensional brick elements for the closed-circuited case for resonant frequencies.
piezocube_c3d20e_coarse_anti.inp	Coarse mesh with 20-node three-dimensional brick elements for the open-circuited case for antiresonant frequencies.
piezocube_c3d20e_fine_reson.inp	Refined mesh with 20-node three-dimensional brick elements for the closed-circuited case for resonant frequencies.
piezocube_c3d20e_fine_anti.inp	Refined mesh with 20-node three-dimensional brick elements for the open-circuited case for antiresonant frequencies.

The input files are currently set up for the situation of fully covered electrodes. Commented data lines exist in each input file for the partially covered electroded cases.

References

- Boucher, D., M. Lagier, and C. Maerfeld, "Computation of the Vibrational Modes for Piezoelectric Array Transducers using a Mixed Finite Element-Perturbation Method," *IEEE Transactions on Sonics and Ultrasonics*, vol. SU-28, no. 8, pp. 318–330, September 1981.
- Lerch, R., "Simulation of Piezoelectric Devices by Two- and Three-Dimensional Finite Elements," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 37, no. 2, pp. 233–247, 1990.
- Ostergaard, D., and T. Pawlak, "Three-Dimensional Finite Elements for Analyzing Piezoelectric Structures," *Proceedings IEEE Ultrasonics Symposium, Williamsburg, VA*, pp. 639–642, 1986.

Tables

Table 1: Eigenvalue estimates for breathing modes in piezoelectric cube with fully covered electrodes.

Model		Resonant freq. (kHz)		Anti-resonant freq. (kHz)	
Element	**# in Model**	**Mode 1**	**Mode 2**	**Mode 1**	**Mode 2**
C3D8E	128	64.3	82.1	76.9	90.1
C3D8E	1024	64.9	86.6	79.4	92.7
C3D20E	16	65.1	88.4	80.2	94.0
C3D20E	128	65.1	88.2	80.1	93.7
Boucher et al.–FEA		67.0	91.9	83.1	96.8
Ostergaard et al.–FEA		65.7	86.5	81.8	95.2
Lerch–FEA		66.0	87.3	80.5	94.9
Boucher et al.–Measured		66.6	88.0	81.6	93.4

Table 2: Eigenvalue estimates for breathing modes in piezoelectric cube with partially covered electrodes.

Model		Resonant freq. (kHz)		Anti-resonant freq. (kHz)	
Element	**# in Model**	**Mode 1**	**Mode 2**	**Mode 1**	**Mode 2**
C3D8E	128	67.1	84.0	77.1	90.2
C3D8E	1024	68.3	88.6	79.6	92.9
C3D20E	16	68.2	90.3	80.4	94.2

Model		Resonant freq. (kHz)		Anti-resonant freq. (kHz)	
Element	**# in Model**	**Mode 1**	**Mode 2**	**Mode 1**	**Mode 2**
C3D20E	128	68.6	90.1	80.3	93.9
Boucher et al.–FEA		70.7	92.9	84.1	97.1
Lerch–FEA		69.5	88.5	80.5	92.9
Boucher et al.–Measured		70.4	90.1	82.5	93.6

Figures

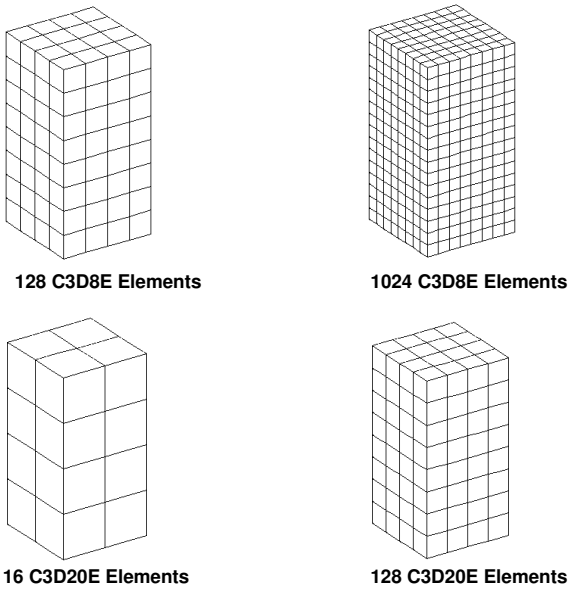


Figure 1: Discretizations used with 8-node and 20-node three-dimensional elements.

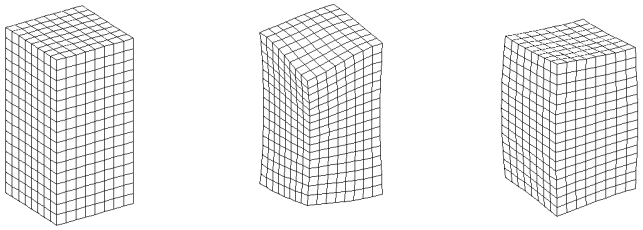


Figure 2: Undeformed mesh and first two breathing modes.

Modal dynamic analysis for piezoelectric materials

Products: Abaqus/Standard

Overview

This example verifies the modal dynamic analysis capability for materials that include piezoelectric coupling.

Problem description

The model is the cylinder described in *Static analysis for piezoelectric materials*. Three analyses are performed with two models. One model has 16 CAX4E elements, and the other has four CAX8E elements. The modal dynamic analyses use the eigendata from the restart files generated in *Frequency extraction analysis for piezoelectric materials*. The first two problems have no damping and are intended for comparison with the results from Mercer, Reddy, and Eve (1987). For these problems a pressure load is applied sinusoidally on the top surface at a frequency of 100000 rad/sec (15.9 kHz). The third problem introduces Rayleigh modal damping terms for the previous problem to illustrate the effect of damping.

Results and discussion

The deflection at the center of the top surface for the sinusoidally applied pressure load for the two models without damping is shown in *Figure 1*. These results closely resemble the normalized results given in Mercer, Reddy, and Eve (1987) for similar models. The potential at the center of the top surface for this problem is shown in *Figure 2*. In *Figure 3* the deflection at the center of the top surface for the CAX4E model with Rayleigh modal damping is shown, along with the result for the case without damping. The reduction in the response and the phase shift are obvious from the figure. The total force on the restrained bottom edge of the CAX8E model is output; the total force in the vertical direction matches the sum of the reaction forces on the edge.

The steady-state response for these problems is illustrated in *Steady-state dynamic analysis for piezoelectric materials*.

Input files

ppzomod1.inp	Modal dynamic analysis, CAX4E elements, no damping.
ppzomod2.inp	Modal dynamic analysis, CAX8E elements, no damping.
ppzomod3.inp	Modal dynamic analysis, CAX4E elements, damping included.

References

- Mercer, C. D., B. D. Reddy, and R. A. Eve, “Finite Element Method for Piezoelectric Media,” *UCT/CSIR Applied Mechanics Research Unit Technical Report No. 92*, vol. April, 1987.

Figures

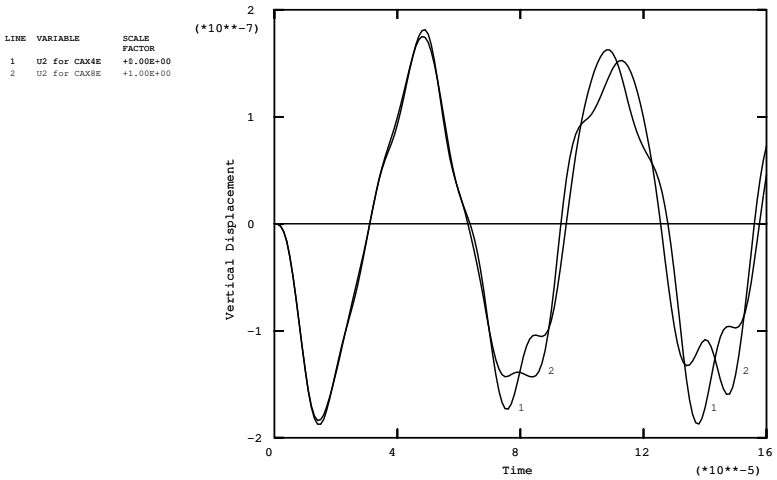


Figure 1: Vertical displacement at center of top of cylinder for pressure load with no damping.

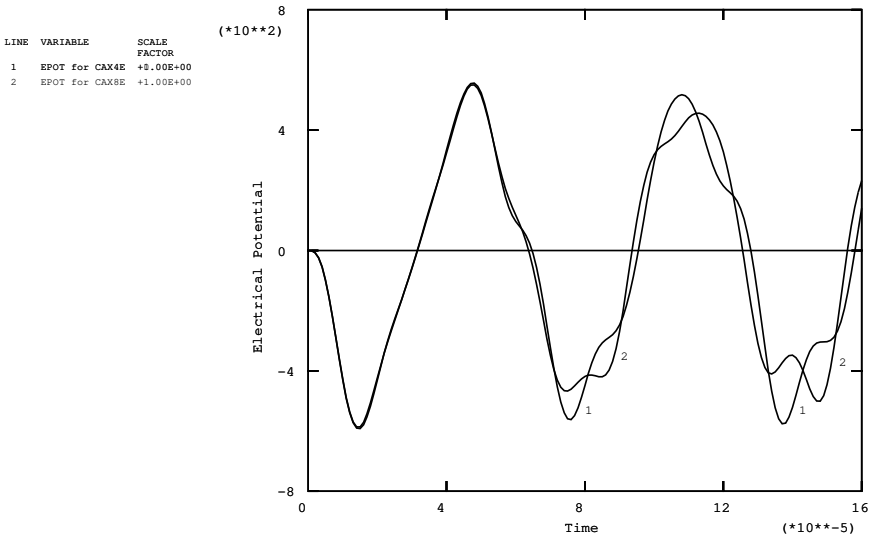


Figure 2: Potential at top of cylinder for pressure load with no damping.

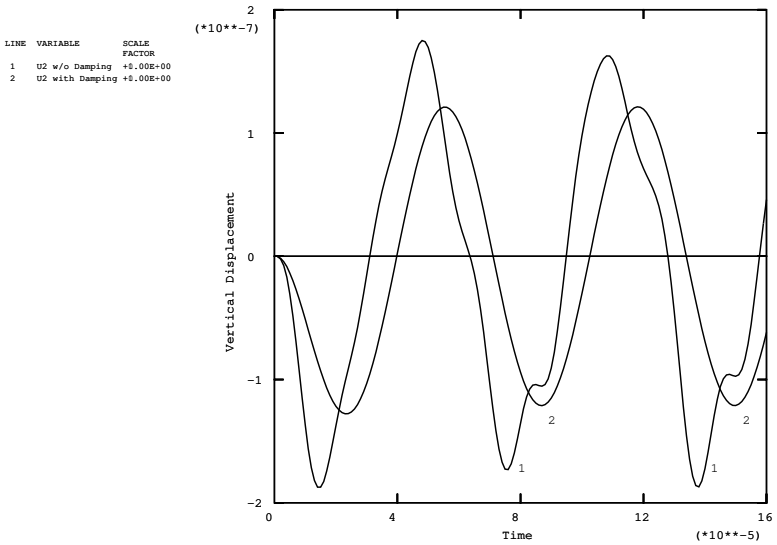


Figure 3: Vertical displacement at center of top of cylinder with and without damping.

Steady-state dynamic analysis for piezoelectric materials

Products: Abaqus/Standard

Overview

This example verifies the modal-based steady-state dynamic analysis capability for materials that include piezoelectric coupling.

Problem description

The model is the cylinder described in *Static analysis for piezoelectric materials*. There are nine input files. The input files *ppzossd1.inp*, *ppzossd3.inp*, *ppzossd4.inp*, and *ppzossd4a.inp* have 16 CAX4E elements. The input file *ppzossd2.inp* has 4 CAX8E elements. The input files *ppzossd7.inp*, *ppzossd8.inp*, and *ppzossd9.inp* have 32 CAX3E, 8 CAX6E, and 4 CAX8RE elements, respectively.

The modal-based steady-state dynamic analyses use the eigendata from the restart files generated in *Frequency extraction analysis for piezoelectric materials*. The input files *ppzossd1.inp* and *ppzossd2.inp* illustrate a steady-state dynamic analysis with no damping and are intended for comparison with the results from Mercer et al. (1987). In this analysis a pressure load is applied on the top surface. In input files *ppzossd3.inp*, *ppzossd7.inp*, *ppzossd8.inp*, and *ppzossd9.inp* modal damping terms are introduced to the steady-state dynamic analysis mentioned previously to illustrate the effect of damping. Both modal and direct calculation steady-state analyses are performed in input files *ppzossd1.inp*, *ppzossd2.inp*, *ppzossd3.inp*, *ppzossd7.inp*, *ppzossd8.inp*, and *ppzossd9.inp*. The input file *ppzossd4.inp* illustrates steady-state analysis with a distributed electrical charge, while the input file *ppzossd4a.inp* performs the steady-state analysis with a concentrated electrical charge instead of the pressure load. Only the direct calculation option is used because the modal-based procedures do not adequately transform the charge loads into modal loads.

In addition to the modal-based and direct-solution analyses, subspace-based steady-state dynamics analyses are performed in the input files *ppzossd1.inp*, *ppzossd3.inp*, *ppzossd7.inp*, *ppzossd8.inp*, and *ppzossd9.inp*. An additional frequency step extracts all eigenmodes available, which are then used in the subspace-based steady-state dynamic steps to compute the response. Since all the eigenmodes are used, the results are identical to the ones obtained in the direct-solution analysis.

For all these analyses a single sinusoidal frequency of 100000 rad/sec (15.9 kHz) is chosen to compare to the modal dynamics results from *Modal dynamic analysis for piezoelectric materials*.

Results and discussion

In the steady-state dynamics procedure the frequency of the sinusoidal load is user-defined, with a complex-valued solution for various quantities such as stresses and displacements that occur when steady state is reached. The accuracy of the modal procedure depends on the representation contained in the eigendata extracted previously, whereas the direct procedure utilizes all the available degrees of freedom. For the modal procedure the vertical deflection at the center of the top surface for the sinusoidally applied pressure load for the two models with no damping is 1.56×10^{-7} for the model that uses CAX4E elements and 1.55×10^{-7} for the model with CAX8E elements. For the direct procedure the deflection is 1.80×10^{-7} for the model with CAX4E elements and 1.79×10^{-7} for the model with CAX8E elements. If all the possible modes are used in the frequency extraction, the modal procedures give the same results as the direct procedure.

These results are in the same range as that reported in Mercer et al. (1987) for similar models. Since no damping is included in the model, the phase angles for all variables are either 0 or 180°. The phase angle for the 2-displacement at the center of the top surface is 180°. This phase shift is evident from *Figure 1*. The total force

on the restrained bottom edge of the CAX8E model is output; the total force in the vertical direction matches the sum of the reaction forces on the edge.

The deflection at the center of the top surface for the analysis including Rayleigh damping using CAX4E elements (input file *ppzossd3.inp*) is 1.21×10^{-7} with a phase angle of 133°. The reduction in magnitude and the phase shift are evident from *Figure 2*. The problem including Rayleigh damping is also analyzed using CAX3E, CAX6E, and CAX8RE elements in input files *ppzossd7.inp*, *ppzossd8.inp*, and *ppzossd9.inp*, respectively. In these analyses the deflection and phase angle at the center of the top surface match with those obtained using CAX4E elements. In addition, the results from direct and modal dynamic solutions in each case are in good agreement.

The analyses using input files *ppzossd4.inp* and *ppzossd4a.inp* have a sinusoidally varying surface charge. Because of the inability of the modal procedures to transform these loads adequately into modal loads, only the direct procedure is used. The first step uses the REAL ONLY feature of the direct calculation analysis procedure. The second step includes both real and imaginary terms, with the load applied in the imaginary plane using the IMAGINARY parameter. The vertical deflection at the center of the top surface is 5.5×10^{-8}. The phase shift for the vertical displacement at this location is 180° (no damping involved) for the case when the charges are applied in the real plane and −90.0° when the charges are applied in the imaginary plane. The potential at that nodal location is 158., with a phase shift of 0° for the first step and a phase shift of 90° in the second step.

Input files

ppzossd1.inp	CAX4E elements.
ppzossd2.inp	CAX8E elements.
ppzossd3.inp	CAX4E elements including damping.
ppzossd4.inp	CAX4E elements with distributed charges (direct calculations only).
ppzossd4a.inp	CAX4E elements with concentrated charges (direct calculations only).
ppzossd7.inp	CAX3E elements including damping.
ppzossd8.inp	CAX6E elements including damping.
ppzossd9.inp	CAX8RE elements including damping.

References

- Mercer, C. D., B. D. Reddy, and R. A. Eve, “Finite Element Method for Piezoelectric Media,” *UCT/CSIR Applied Mechanics Research Unit Technical Report No. 92*, vol. April, 1987.

Figures

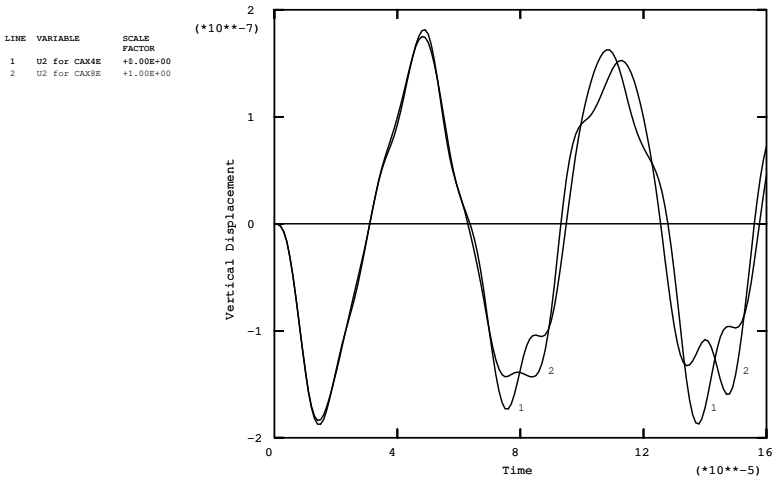


Figure 1: Vertical displacement at center of top of cylinder for pressure load with no damping.

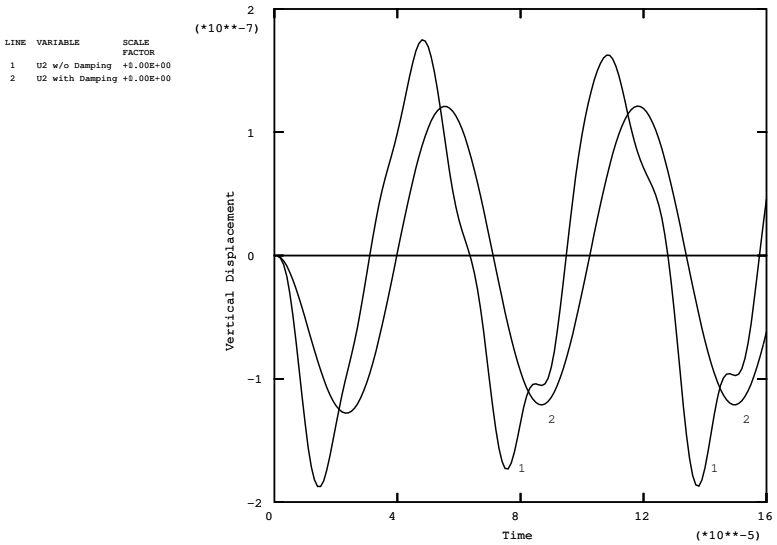


Figure 2: Vertical displacement at center of top of cylinder with and without damping.

TEAM 2: Eddy current simulations of long cylindrical conductors in an oscillating magnetic field

Products: Abaqus/Standard

Overview

This benchmark problem verifies the case of an infinite conducting cylindrical shell immersed in a time-harmonic uniform magnetic field.
It is part of the standard suite of problems designed for Testing Electromagnetic Analysis Methods (TEAM). The objective is to compute the eddy currents induced in the cylindrical shell by the magnetic field that is varying in time. Lorentz force and Joule heating in the conductor are also of interest.

Problem description

The problem setup is shown in *Figure 1*. It depicts an infinite conducting cylindrical shell immersed in a time-harmonic uniform magnetic field. The inner and outer radius of the conducting cylindrical shell are $a = 0.05715$ m and $b = 0.06985$ m. Its resistivity and relative magnetic permeability are assumed to be $1/\sigma = 3.94 \times 10^{-8}$ -m and $\mu_r = 1.0$. The magnetic flux density is assumed to have a magnitude of $B_0 = 0.1$ T and is oscillating with a frequency of $f = 60$ Hz. The magnetic field is assumed to be oriented along the y-direction. We will assume that the medium in which the cylindrical shell is immersed has properties similar to that of a vacuum. For these parameters, the skin depth of the conductor is about $\delta = \sqrt{(2/\omega\mu_0\mu_r\sigma)} = 12.9$ mm, which is comparable to the shell thickness of 12.7 mm.

Model and boundary conditions

The magnetic vector potential formulation is used to solve this problem. Due to the invariance of the geometry along the z-direction and the fact that magnetic flux density lies in the x–y plane, only the z-component of magnetic vector potential is nonzero. Although the geometry is two-dimensional in nature, the magnetic vector potential formulation in two dimensions can only represent the x- and y- components of the magnetic vector potential. Hence, a three-dimensional geometry that contains one element along the z-direction is used.

Due to the symmetry of the problem, it is sufficient to model the first quadrant of the problem domain in the x–y plane. Appropriate boundary conditions are imposed on the symmetry planes $x = 0$ and $y = 0$. Since the magnetic vector potential, $\mathbf{A}$, is oriented along the z-direction, symmetry arguments require that it be identically zero on the $x = 0$ plane. As a result, a homogeneous Dirichlet boundary condition $\mathbf{A} \times \mathbf{n} = \mathbf{0}$ is applied on the symmetry plane $x = 0$. Here $\mathbf{n}$ represents the unit normal perpendicular to the boundary surface. Similarly, on the plane $y = 0$, symmetry arguments require that the magnetic field $\mathbf{H}$ be perpendicular to the plane. Hence, a homogeneous Neumann boundary condition $\mathbf{H} \times \mathbf{n} = \mathbf{0}$ is applied on the symmetry plane $y = 0$.

Since the problem domain is unbounded, it must be truncated in some way. Abaqus does not support absorbing boundary conditions; therefore, the truncation boundary should be chosen far away from the conductor. Boundary surfaces far away from the conductor are chosen such that they are parallel to either the $x = 0$ or $y = 0$ plane. Magnetic vector potential and magnetic flux density far away from the conductor are given by $\mathbf{A} \approx -B_0 x \mathbf{e_3}$ and $\mathbf{B} \approx B_0 \mathbf{e_2}$, where $\mathbf{e_2}$ and $\mathbf{e_3}$ are the unit vectors along the y- and z- coordinate axes. Since the projection of the magnetic vector potential onto the far boundary surface that is parallel to the $x = 0$ plane is constant, an

inhomogeneous Dirichlet boundary condition $\mathbf{A} \times \mathbf{n} = \mathbf{constant}$ is applied on this boundary surface. Since the magnetic flux density is perpendicular to the far boundary surface that is parallel to the $y = 0$ plane, a homogeneous Neumann boundary condition is applied on this boundary surface.

Finally, since the magnetic vector potential is oriented along the z-axis, a homogeneous Dirichlet boundary condition is applied on the boundary surfaces that are parallel to the $z = 0$ plane.

Analytical solution

The magnetic vector potential in various regions of the problem can be expressed as follows:

$$\mathbf{A}(r,\theta) = \begin{cases} B_0 C_1 r \cos\theta \, \mathbf{e}_3, & 0 < r < a, \\ B_0 \left(C_2 J_1(kr) + C_3 Y_1(kr) \right) \cos\theta \, \mathbf{e}_3, & a < r < b, \\ B_0 \left(-C_5 r + C_4 \dfrac{1}{r} \right) \cos\theta \, \mathbf{e}_3, & b < r < c, \end{cases}$$

where C_i, $i = 1, \cdots, 5$ are the constants to be determined; $J_\nu(x)$ and $Y_\nu(x)$ are the cylindrical Bessel functions of the first and second kind, respectively; and $k = \sqrt{-i\omega\mu_0\mu_r\sigma}$ is the complex wave number of the conductor. Enforcing continuity of the normal component of magnetic flux density and the tangential component of magnetic field intensity on the inner and outer surfaces of the cylindrical shell and applying an inhomogeneous Dirichlet boundary condition on an outer cylindrical surface of radius c leads to the following set of relations between the constants:

$$\begin{aligned} C_1 a &= C_2 J_1(ka) + C_3 Y_1(ka) \\ C_1 &= \frac{1}{\mu_r} k \left(C_2 J_1'(ka) + C_3 Y_1'(ka) \right) \\ C_2 J_1(kb) + C_3 Y_1(kb) &= -C_5 b + C_4 \frac{1}{b} \\ \frac{1}{\mu_r} k \left(C_2 J_1'(kb) + C_3 Y_1'(kb) \right) &= -\left(C_5 + C_4 \frac{1}{b^2} \right) \\ -C_5 c + C_4 \frac{1}{c} &= -c \end{aligned}$$

where the primed function $Z_\nu'(x)$ denotes the first derivative of the function with respect to its argument. In the limit of $c \to \infty$ we obtain the true solution to the problem. For comparison with the simulation results, truncated analytical results are generated by choosing the value of c to be the distance from the origin to the outer boundary of the problem domain.

Results and discussion

Figure 2 shows the comparison of the amplitude of the y-component of the magnetic flux density computed using Abaqus/Standard with that of the analytical solution. The labels `EMC3D8' and `EMC3D4' in the legend correspond to the analyses performed with these elements. The labels `Analytical Truncated' and `Analytical True' in the legend correspond to the analytical solution computed by assuming that a Dirichlet boundary condition is applied on an outer cylindrical boundary surface at a finite distance and at infinity, respectively, as described in the previous section. The figure clearly indicates that the analysis results compare very well with the analytical results and that the outer boundary surface is far enough from the cylindrical shell that the error introduced by truncation is small.

Figure 3 shows a contour plot of the amplitude of the electric field. For a time-harmonic analysis the amplitude of the electric field is the same as that of the amplitude of the magnetic vector potential scaled by the radian

frequency. The figure clearly shows that the presence of the conductor distorts the field near its vicinity. Finally, *Figure 4* depicts the induced current density in the conductor due to the magnetic field. The figure shows that the current density in the conductor is larger along the x-axis and vanishes along the y-axis. Consequently, the Joule heat generated in the conductor is large along the x-axis.

Input files

team2_symm_sqr_emc3d8.inp	Eddy current analysis of a conducting cylindrical shell immersed in a time-harmonic uniform magnetic field using element type EMC3D8 and symmetry boundary conditions.
team2_symm_sqr_emc3d4.inp	Eddy current analysis of a conducting cylindrical shell immersed in a time-harmonic uniform magnetic field using element type EMC3D4 and symmetry boundary conditions.

References

- Ida, N., “Infinite Cylinder in a Uniform Sinusoidal Field (Comparison of Results, Problem 2),” *The International Journal for Computation and Mathematics in Electrical and Electronic Engineering*, vol. 7, pp. 29–45, 1988.

Figures

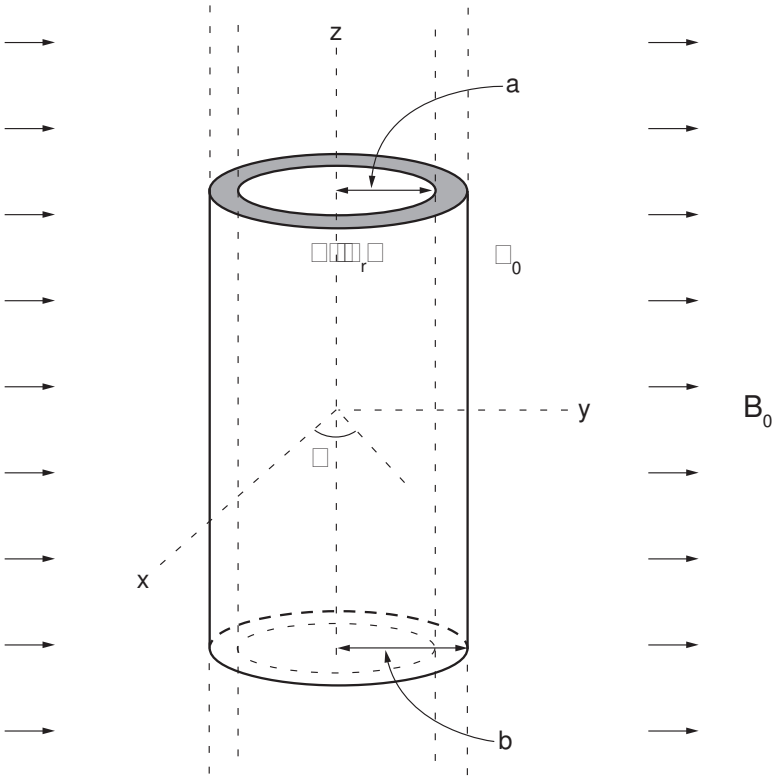


Figure 1: Geometry of an infinite conducting cylindrical shell immersed in a time-harmonic uniform magnetic field.

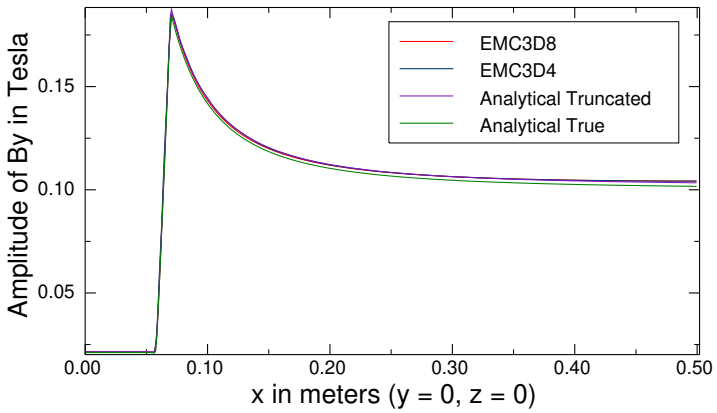


Figure 2: Amplitude of the y-component of magnetic flux density.

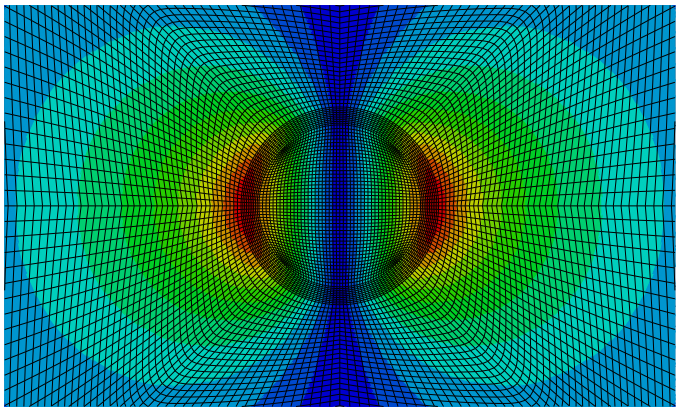


Figure 3: Amplitude of the real part of the electric field.

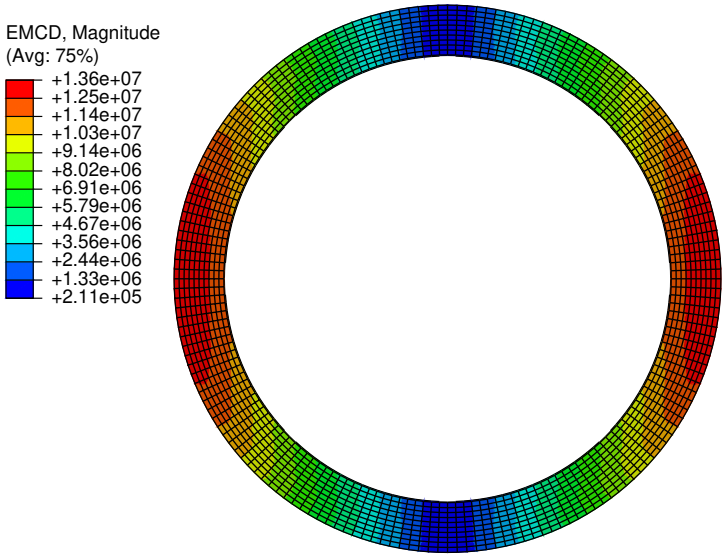


Figure 4: Amplitude of the eddy current induced in the conductor.

TEAM 4: Eddy current simulation of a conducting brick in a decaying magnetic field

Products: Abaqus/Standard

Overview

This benchmark problem verifies the case of a rectangular brick with a rectangular hole through the center placed in a uniform magnetic field that is decaying in time.
It is part of the standard suite of problems designed for Testing Electromagnetic Analysis Methods (TEAM). The objective is to compute the circulating eddy currents induced in the brick and the ensuing Joule heat dissipated due to the magnetic field that is decaying in time.

Problem description

The problem setup is shown in *Figure 1*, which depicts a conductive rectangular brick with a rectangular hole placed in a uniform magnetic field that is decaying in time. The dimensions of the brick are $a = 0.1524$ m, $b = 0.1016$ m, and $c = 0.0508$ m. The brick is assumed to be made of an aluminum alloy with a resistivity of $\rho = 3.94 \times 10^{-8} \Omega$m and a relative magnetic permeability of $\mu_r = 1.0$. The rectangular hole is centered in and penetrates through the brick. The dimensions of the hole are assumed to be $l = 0.0889$ m and $w = 0.0381$ m. The orientation of the uniform magnetic field is parallel to the direction of penetration of the hole and is assumed to be decaying as $B_z = B_0 \exp(-t/\tau)$, where $B_0 = 0.1$ T and $\tau = 0.0119$ s. The medium surrounding the brick is assumed to have properties similar to that of a vacuum.

Model and boundary conditions

The magnetic vector potential formulation is used to solve this problem. Due to the symmetry of the problem, it is sufficient to model the first octant of the problem domain. Appropriate boundary conditions are imposed on the symmetry planes $x = 0$, $y = 0$, and $z = 0$. Due to the symmetry of the external magnetic field with respect to the planes $x = 0$ and $y = 0$, the magnetic vector potential is normal to these symmetry planes, which is enforced by a homogeneous Dirichlet boundary condition. Similarly, due to the asymmetry of the external magnetic field with respect to the plane $z = 0$, the magnetic flux density is normal to the symmetry plane, which is enforced by a homogeneous Neumann boundary condition.

The presence of the conducting brick in a time-varying magnetic field generates eddy currents in the brick, which in turn generate their own magnetic field in the vicinity of the brick. The magnetic field far away from the brick is not changed significantly by these eddy currents from that of the external magnetic field. Since the external magnetic field is pointing in the z-direction, it is easier to specify the far-field boundary conditions if we choose planar truncation boundaries that are parallel to the planes $x = 0$, $y = 0$, and $z = 0$. The external magnetic field on the truncation boundaries is specified as a surface current density load.

Results and discussion

Figure 2 shows the z-component of the magnetic flux density, B_z, computed using an Abaqus/Standard low-frequency transient electromagnetic analysis performed over a period of 20 ms. The magnetic field is plotted along the positive z-axis of the model starting from the center of the hole. The curves in the plot correspond to the field values at different analysis times, as indicated in the legend. The axes are scaled to facilitate the comparison of the results against the published results in Kameari (1988). The horizontal axis is multiplied by 6.35 to convert the true distance along the z-axis to millimeters, and the vertical axis is multiplied by 0.01 to convert the flux density to Tesla. The plot indicates that far away from the conducting brick the magnetic field

is the same as that of the external magnetic field, which is decaying in time. However, at the center of the brick the magnetic field is larger due to the eddy currents induced in the brick; these currents try to compensate for the magnetic field that is reducing in time. The results compare very well with those published in Kameari.

Figure 3 shows the total induced current that is flowing across the cross-section of the conducting brick plotted against the analysis time. The total current is obtained by integrating the current density output across the cross-section. The results compare very well to those published in Kameari.

Input files

team4_emc3d8.inp	Low-frequency transient eddy current analysis of an aluminum brick with a hole placed in a uniform magnetic field that is decaying in time; modeled using EMC3D8 elements and symmetry boundary conditions.

References

- Kameari, A., "Results for Benchmark Calculations of Problem 4 (the FELIX Brick)," *COMPEL*, vol. 7, pp. 65–80, 1988.

Figures

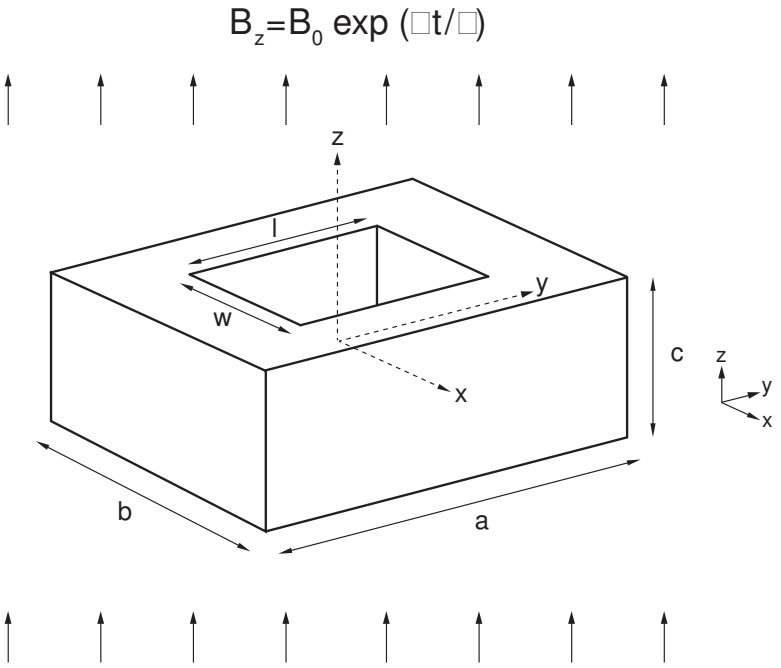


Figure 1: Geometry of a rectangular brick with a hole placed in a decaying magnetic field.

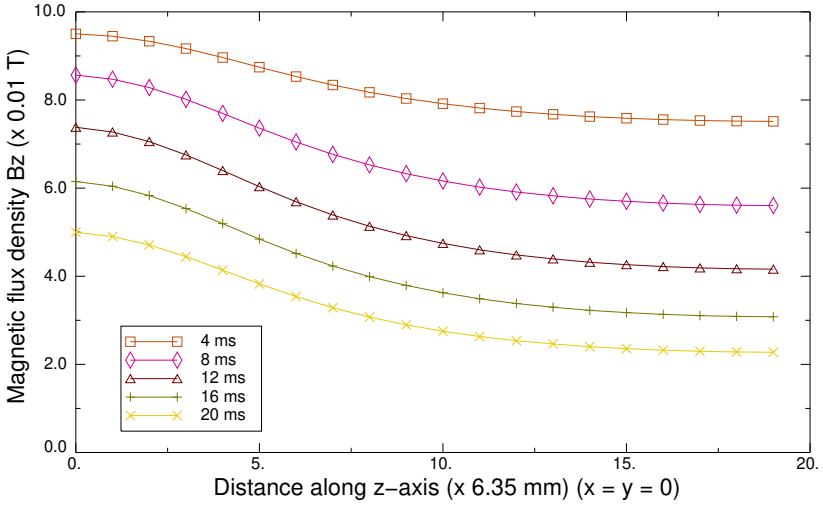


Figure 2: Magnetic flux density along the z-axis.

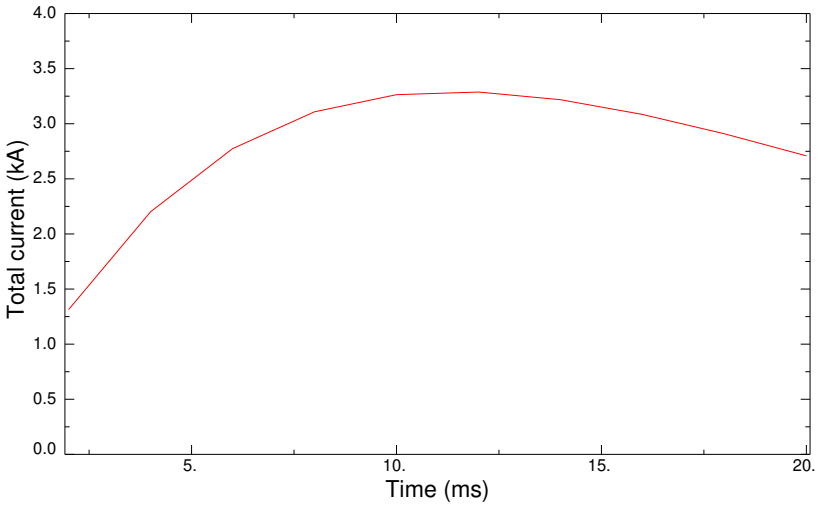


Figure 3: Total induced current flowing across the cross-section of the conducting brick.

TEAM 6: Eddy current simulations for spherical conductors in an oscillating magnetic field

Products: Abaqus/Standard

Overview

This benchmark problem verifies the case of a conducting spherical shell immersed in a time-harmonic uniform magnetic field.
It is part of the standard suite of problems designed for Testing Electromagnetic Analysis Methods (TEAM). The objective is to compute the eddy currents induced in the spherical shell by the magnetic field that is varying in time. Lorentz force and Joule heating in the conductor are also of interest.

Problem description

The problem setup is shown in *Figure 1*. It depicts a conducting spherical shell immersed in a time-harmonic uniform magnetic field. For visual clarity, the figure depicts a spherical shell with a section of it removed. The inner and outer radius of the conducting spherical shell are $a = 0.05$ m and $b = 0.055$ m. Its conductivity and relative magnetic permeability are assumed to be $\sigma = 5 \times 10^8$ S/m and $\mu_r = 1.0$. The magnetic flux density is assumed to have a magnitude of $B_0 = 1.0$ T and is oscillating with a frequency of $f = 50$ Hz. Without loss of generality, we can assume that the magnetic field is oriented along the z-direction. We will assume that the medium in which the spherical shell is immersed has properties similar to that of a vacuum. For these parameters, the skin depth of the conductor is about $\delta = \sqrt{(2/\omega\mu_0\mu_r\sigma)} = 3.18$ mm, which is smaller than the shell thickness of 5 mm.

Model and boundary conditions

The magnetic vector potential formulation is used to solve this problem. Due to the symmetry of the problem, it is sufficient to model the first octant of the problem domain. Appropriate boundary conditions are imposed on the symmetry planes $x = 0$, $y = 0$, and $z = 0$. Since the magnetic flux density is oriented along the z-direction, azimuthal symmetry of the geometry requires that the total magnetic vector potential $\mathbf{A}$ is nonzero only in the azimuthal direction. As a result, the magnetic vector potential on the planes $x = 0$ and $y = 0$ is perpendicular to each of these planes. Hence, a homogeneous Dirichlet boundary condition $\mathbf{A} \times \mathbf{n} = \mathbf{0}$ is imposed on the symmetry planes $x = 0$ and $y = 0$. Symmetry of the problem also requires that the total magnetic field $\mathbf{H}$ on the symmetry plane $z = 0$ be perpendicular to this plane. Hence, a homogeneous Neumann boundary condition $\mathbf{H} \times \mathbf{n} = \mathbf{0}$ is applied on the symmetry plane $z = 0$.

Since the problem domain is unbounded, it must be truncated in some way. Abaqus does not support absorbing boundary conditions; therefore, the truncation boundary should be chosen far away from the conductor. Boundary truncation surfaces are chosen such that they are parallel to one of the $x = 0$, $y = 0$, or $z = 0$ planes. To demonstrate various boundary conditions that can be applied in an Abaqus/Standard analysis, a spherical boundary surface of radius c is chosen to truncate the problem domain. Magnetic vector potential and magnetic flux density far away from the conductor are given by $\mathbf{A} \approx 0.5 B_0 r \cos\phi \mathbf{e}_\theta$ and $\mathbf{B} \approx B_0 \mathbf{e}_3$, where $\mathbf{e}_3$ and $\mathbf{e}_\theta$ are the unit vectors along the z-coordinate axis and along the azimuthal direction. Clearly neither the projection of magnetic vector potential nor that of magnetic field onto this surface is constant. They vary nonuniformly over the boundary surface. In this problem a nonuniform Dirichlet boundary condition is applied on the spherical boundary surface

by supplying a user subroutine *UDEMPOTENTIAL* that computes the magnetic vector potential on the boundary surface.

Analytical solution

The total magnetic vector potential in various regions can be expressed as follows:

$$\mathbf{A}(r,\theta) = \begin{cases} \frac{1}{2}B_0 \cos\phi C_1 r \mathbf{e}_\theta, & 0 < r < a, \\ \frac{1}{2}B_0 \cos\phi \frac{1}{\sqrt{r}}\left(C_2 J_{3/2}(kr) + C_3 Y_{3/2}(kr)\right)\mathbf{e}_\theta, & a < r < b, \\ \frac{1}{2}B_0 \cos\phi \left(C_5 r + C_4 \frac{1}{r^2}\right)\mathbf{e}_\theta, & b < r < c, \end{cases}$$

where C_i, $i = 1, \cdots, 5$ are the constants to be determined; $J_\nu(x)$ and $Y_\nu(x)$ are the cylindrical Bessel functions of the first and second kind, respectively; and $k = \sqrt{-i\omega\mu_0\mu_r\sigma}$ is the complex wave number in the conductor. Enforcing continuity of the normal component of magnetic flux density and the tangential component of magnetic field intensity on the inner and outer surfaces of the spherical shell and applying an inhomogeneous Dirichlet boundary condition on an outer spherical surface of radius c leads to the following set of relations between the constants:

$$\begin{aligned} C_1 a &= \frac{1}{\sqrt{a}}\left(C_2 J_{3/2}(ka) + C_3 Y_{3/2}(ka)\right), \\ \frac{1}{b\sqrt{b}}\left(C_2 J_{3/2}(kb) + C_3 Y_{3/2}(kb)\right) &= C_5 + C_4 \frac{1}{b^3}, \\ 2C_1 &= \frac{1}{a\sqrt{a}}\frac{1}{\mu_r}\left(C_2 \tilde{J}_{3/2}(ka) + C_3 \tilde{Y}_{3/2}(ka)\right), \\ \frac{1}{b\sqrt{b}}\frac{1}{\mu_r}\left(C_2 \tilde{J}_{3/2}(kb) + C_3 \tilde{Y}_{3/2}(kb)\right) &= \left(2C_5 - C_4 \frac{1}{b^3}\right), \\ C_5 + C_4 \frac{1}{c^3} &= 1, \end{aligned}$$

where, for notational simplicity, the functions $\tilde{Z}_\nu(x) = xZ_{\nu-1}(x) - Z_\nu(x)$ are introduced. In the limit of $c \to \infty$ we obtain the true solution to the problem. For comparison with the simulation results, truncated analytical results are generated by choosing the value of c to be the distance from the origin to the outer boundary of the problem domain.

Results and discussion

Figure 2 shows the comparison of the amplitude of the z-component of the magnetic flux density computed using Abaqus/Standard analysis with that of the analytical solution. The labels `EMC3D8' and `EMC3D4' in the legend correspond to the analyses performed with these elements. The labels `Analytical Truncated' and `Analytical True' in the legend correspond to the analytical solution computed by assuming that a Dirichlet boundary condition is applied on an outer spherical boundary surface at a finite distance and at infinity, respectively, as described in the previous section. The figure clearly indicates that the analysis results compare very well with the analytical results and that the outer boundary surface is far enough from the spherical shell that the error introduced by truncation is small.

Figure 3 shows the contour plot of the amplitude of the electric field. Only the first octant of the problem domain is shown in the figure. The view is oriented such that the origin is closer to the reader. For a time-harmonic

analysis the amplitude of the electric field is the same as that of the amplitude of the magnetic vector potential scaled by the radian frequency. Finally, *Figure 4* depicts the induced current density in the conductor due to the magnetic field. In the figure a portion of the spherical shell is removed to expose the interior of the shell. The figure shows that the current density in the conductor is larger along the *x*–*y* plane and decreases toward the poles. Consequently, the Joule heat generated in the conductor is maximum along the *x*–*y* plane.

Input files

team6_symm_nuori_emc3d8.inp	Eddy current analysis of a conducting spherical shell immersed in a time-harmonic uniform magnetic field using element type EMC3D8, symmetry boundary conditions, and user subroutine *UDEMPOTENTIAL*.
team6_symm_nuori_emc3d4.inp	Eddy current analysis of a conducting spherical shell immersed in a time-harmonic uniform magnetic field using element type EMC3D4, symmetry boundary conditions, and user subroutine *UDEMPOTENTIAL*.
team6_symm_nuori_emc3d8.f	User subroutine *UDEMPOTENTIAL* used in the analysis with EMC3D8 elements.
team6_symm_nuori_emc3d4.f	User subroutine *UDEMPOTENTIAL* used in the analysis with EMC3D4 elements.

References

- Emson, C. R. I., "Results for a Hollow Sphere in Uniform Field (Benchmark Problem 6)," *The International Journal for Computation and Mathematics in Electrical and Electronic Engineering*, vol. 7, pp. 89–101, 1988.

Figures

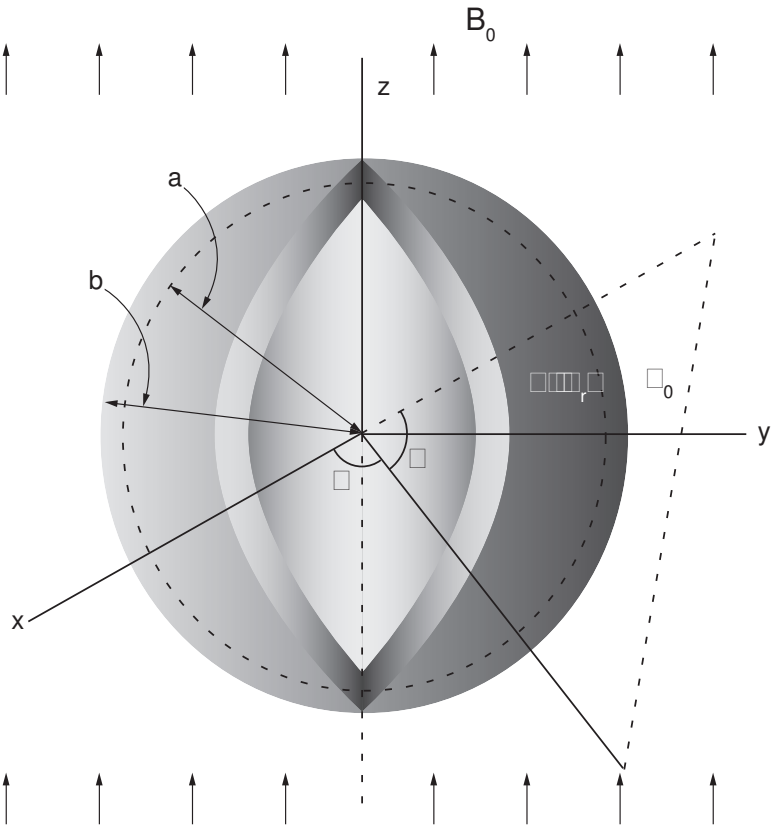


Figure 1: Geometry of a conducting spherical shell immersed in a time-harmonic uniform magnetic field.

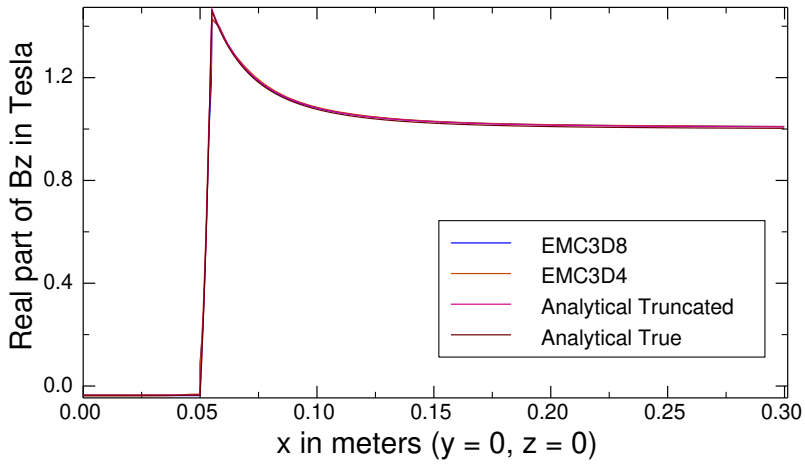


Figure 2: Amplitude of the z-component of magnetic flux density.

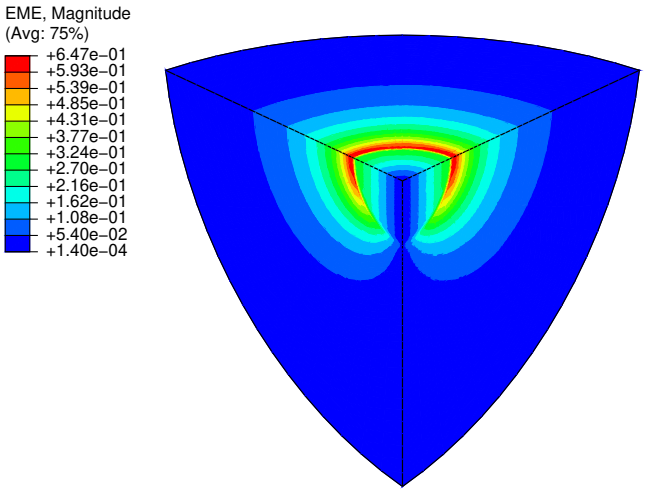


Figure 3: Amplitude of the real part of the electric field.

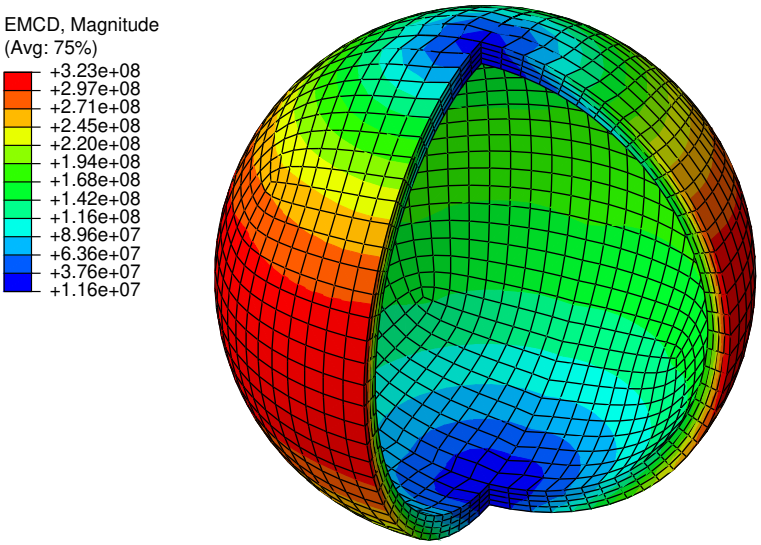


Figure 4: Amplitude of the eddy current induced in the conductor.

TEAM 13: Three-dimensional nonlinear magnetostatic analysis

Products: Abaqus/Standard

Overview

This benchmark problem verifies the case of a magnetostatic analysis with magnetic vector potential formulation to compute the magnetic field in two steel channels and a steel plate.
It is part of the standard suite of problems designed for Testing Electromagnetic Analysis Methods (TEAM). The problem consists of two steel channels and a steel plate excited by a coil carrying direct current. The objective is to compute the magnetic field in the channels and the plate, taking into account the nonlinear magnetic material response of steel.

Problem description

The plan and top views of the problem setup are shown in *Figure 1* and *Figure 2*, respectively. They depict two U-shaped channels and a planar plate excited by a coil carrying direct current. The dimensions of various parts are marked in the figures. The rest of the parameters are as follows. Both the channels and plate are assumed to be made of steel. The nonlinear magnetic properties of the steel are specified as a B–H curve; the data are obtained from Figure 3 of *Team Problem 13: 3-D Non-Linear Magnetostatic Model*. The current in the coil is assumed to be either 1000 A-turns or 3000 A-turns. The medium surrounding the brick is assumed to have properties similar to that of a vacuum.

Model and boundary conditions

A magnetostatic analysis with magnetic vector potential formulation is performed. Due to the symmetry of the problem, it is sufficient to model only one-half of the problem domain. A homogeneous Neumann boundary condition is applied on the symmetry plane $z = 0$ since the magnetic flux density on this plane is expected to be perpendicular to the plane due to symmetry. Homogeneous Dirichlet boundary conditions are assumed on the outer boundaries of the domain.

Results and discussion

Figure 3 shows plots of the average magnetic flux density in the steel plate and channels along the paths A–B, C–D, and E–F, as shown in *Figure 1*. For simplicity, three different plots are combined into one with the horizontal axis split into three segments, each representing the distance along the paths A–B, C–D, and E–F from the beginning of the path. The experimental results are presented together with the simulation results that are obtained using Abaqus/Standard for two different specifications of current; namely, 1000 A-turns and 3000 A-turns. The plots indicate that the agreement between simulation and experimental results is good. *Figure 4* shows plots of the magnetic flux density in the air along the path 1–2, as shown in *Figure 1*, for two different specifications of current. Again, the simulation results compare very well with the experimental results.

Input files

team_13_half_d2_bias2.inp	Nonlinear magnetostatic analysis of a steel plate and two steel channels excited by a coil carrying direct current of 1000 A-turns.
team_13_half_d2_bias2_3000.inp	Nonlinear magnetostatic analysis of a steel plate and two steel channels excited by a coil carrying direct current of 3000 A-turns.

References

- Preis, K., et.al., "Different Finite Element Formulations for 3D Magnetostatic Fields," *IEEE Transactions on Magnetics*, vol. 28, pp. 1056–59, 1992.
- "Team Problem 13: 3-D Non-Linear Magnetostatic Model," accessed December 19, 2012, http://www.compumag.org/jsite/images/stories/TEAM/problem13.pdf.

Figures

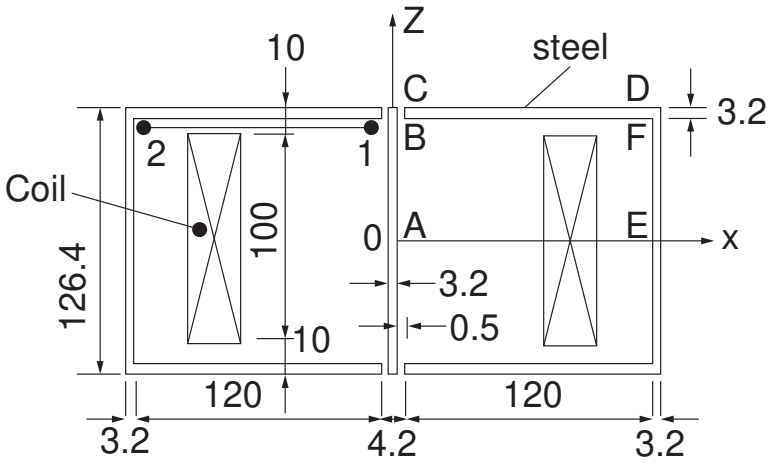


Figure 1: Plan view of the problem setup.

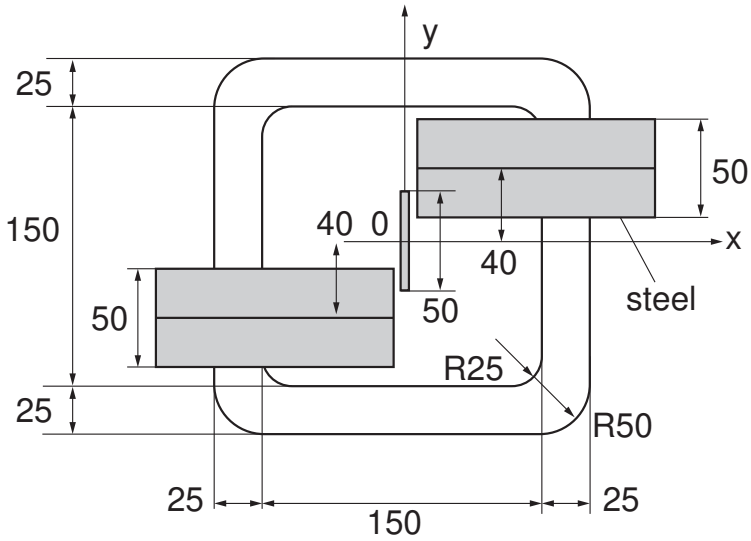


Figure 2: Top view of the problem setup.

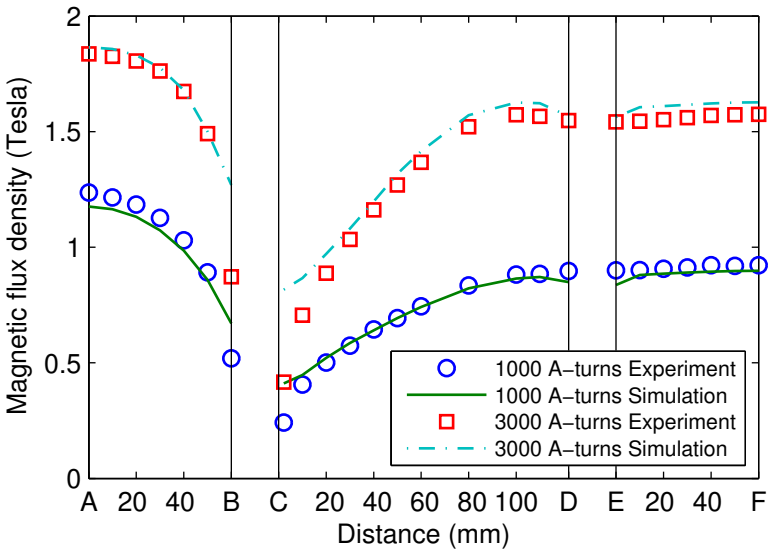


Figure 3: Magnetic flux density along the plate and the channels.

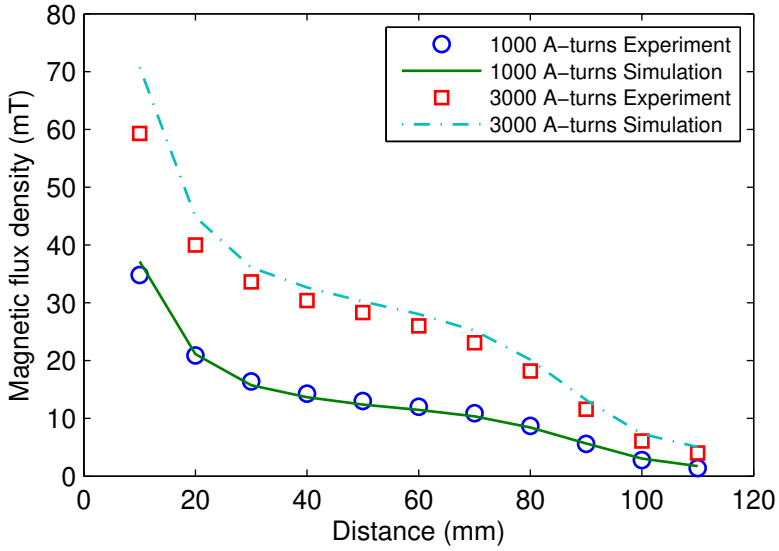


Figure 4: Magnetic flux density in the air.

Induction heating of a cylindrical rod by an encircling coil carrying time-harmonic current

Products: Abaqus/Standard

Overview

This benchmark problem verifies the case of computing eddy currents induced in a conducting rod due to a circulating time-harmonic current.
This problem is frequently encountered in induction heating applications. The circulating time-harmonic current in the coil produces a time-harmonic magnetic field, which in turn induces eddy currents in a conductor in its vicinity. The resistance of the conductor to the flow of the induced currents manifests as heat, and computing this Joule heat is the primary objective of this benchmark problem.

Problem description

The problem setup is shown in *Figure 1*. It depicts a conductive cylindrical rod with an encircling current-carrying coil of rectangular cross-section centered along the length of the rod. The conductive cylindrical rod has a radius of $a = 0.05$ m and a length of $L = 0.50$ m. Its conductivity and relative magnetic permeability are assumed to be $\sigma = 1.0 \times 10^7$ S/m and $\mu_r = 1.0$. The inner and outer radius and the thickness of the encircling coil are $b = 0.09$ m, $c = 0.11$ m, and $h = 0.02$ m. The current density in the coil is assumed to have a magnitude of $J_0 = 1.0 \times 10^7$ A/m^2 and is oscillating with a frequency of $f = 50$ Hz. The current is assumed to be flowing along the azimuthal direction in a clockwise sense when looking toward the negative z-direction. The medium surrounding the rod and coil setup is assumed to have properties similar to that of a vacuum. For these parameters, the skin depth of the conductor is about $\delta = \sqrt{(2/\omega\mu_0\mu_r\sigma)} = 22.5$ mm, which is smaller than the conductor radius of 50 mm.

Model and boundary conditions

The magnetic vector potential formulation is used to solve this problem. Due to the symmetry of the problem, it is sufficient to model the first octant of the problem domain. Appropriate boundary conditions are imposed on the symmetry planes $x = 0$, $y = 0$, and $z = 0$. Due to the asymmetry of the current with respect to the planes $x = 0$ and $y = 0$, the magnetic vector potential is normal to these symmetry planes, which is enforced by a homogeneous Dirichlet boundary condition. Similarly, due to the symmetry of the current with respect to the plane $z = 0$, the magnetic flux density is normal to the symmetry plane, which is enforced by a homogeneous Neumann boundary condition.

Since the problem domain is unbounded, it must be truncated in some way. Abaqus does not support absorbing boundary conditions; hence, the truncation boundary should be chosen far away from the conductor. An outer cylindrical boundary surface and a planar surface that is parallel to the $z = 0$ plane are chosen to truncate the domain. The magnetic vector potential decays away from the coil and can be approximated to have zero magnitude far away from it. Hence, a homogeneous Dirichlet boundary condition is applied on all outer boundary surfaces.

Analytical solution

The analytical solution to this problem has been studied by various authors, but the study that is of particular interest is presented by Bowler and Theodoulidis (2005). They consider the problem of computing eddy currents induced in a cylindrical rod due to an encircling current loop that may be positioned at an arbitrary height along

the length of the rod. Expressions for the magnetic vector potential in various regions can be found in this reference paper.

Results and discussion

Figure 2 shows a comparison of the real and imaginary parts of the circumferential component of the electric field along the *x*-axis computed using an Abaqus/Standard analysis to those of the analytical solution. The figure clearly indicates that the analysis results compare very well with the analytical results and that the outer boundary surface is far enough away that the error introduced by truncation is small. For a time-harmonic analysis the amplitude of the electric field is the same as that of the amplitude of the magnetic vector potential scaled by the radian frequency.

Figure 3 shows a contour plot of the Joule heat produced in the conducting rod due to the induced eddy currents. The figure clearly shows that the Joule heat produced is larger near the surface of the conductor compared to its interior due to the skin effect. The figure also indicates that the Joule heat produced is larger near the center of the rod compared to its ends due to the closer proximity to the current coil.

Input files

src_rod_emc3d8.inp	Eddy current analysis of a conductive cylindrical rod encircled by a coaxial coil carrying a time-harmonic current using element type EMC3D8 and symmetry boundary conditions.
src_rod_emc3d4.inp	Eddy current analysis of a conductive cylindrical rod encircled by a coaxial coil carrying a time-harmonic current using element type EMC3D4 and symmetry boundary conditions.

References

- Bowler, J. R., and T. P. Theodoulidis, “Eddy Current Induced in a Conducting Rod of Finite Length by a Coaxial Encircling Coil,” *Journal of Physics D: Applied Physics*, vol. 38, pp. 2861–2868, 2005.

Figures

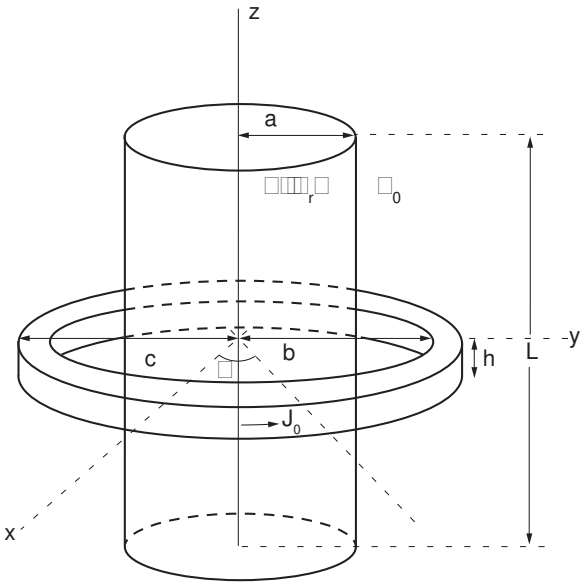


Figure 1: Geometry of a cylindrical rod heated by a circular coil.

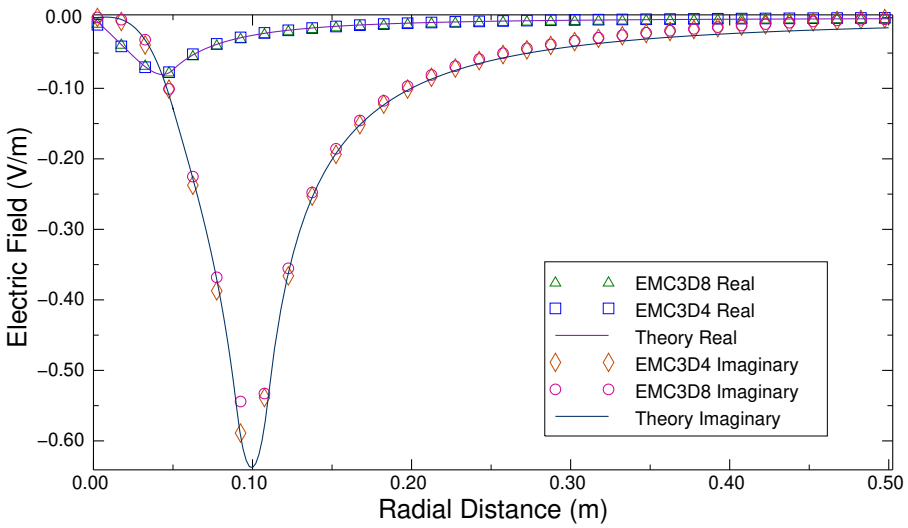


Figure 2: Circumferential component of the electric field on the x-axis.

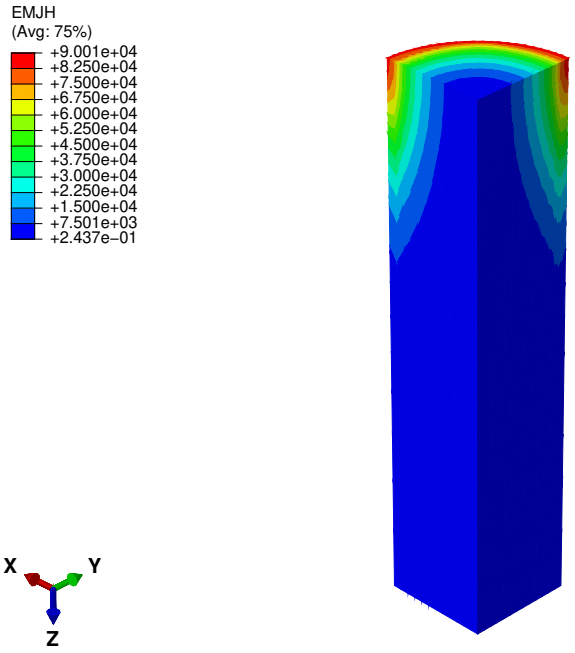


Figure 3: Joule heat generated in the rod.

Coupled pore fluid flow and stress analysis

In this section:

- *Partially saturated flow in a porous medium*
- *Demand wettability of a porous medium: coupled analysis*
- *Wicking in a partially saturated porous medium*
- *Desaturation in a column of porous material*

Partially saturated flow in a porous medium

Products: Abaqus/Standard

Overview

This example illustrates the Abaqus capability to solve problems involving partially saturated flow in porous media.

Abaqus is capable of solving the stress equilibrium/fluid flow coupled problem (see *Demand wettability of a porous medium: coupled analysis*), but in this example we are primarily concerned with the fluid flow part of the problem. For two-dimensional models this uncoupled fluid flow case is obtained by constraining all the displacement degrees of freedom in the problem. In three-dimensional models we allow the model to expand in the global 3-direction.

We consider a "constrained demand wettability" test. The demand wettability test is a common way of measuring the absorption properties of porous materials. In such a test fluid is made available to the material at a certain location, and the material is allowed to absorb as much fluid as it can. In this example we consider a square specimen of material and allow it to absorb fluid at its center. We investigate two cases: one in which the material contains a large number of gel particles that entrap fluid and, as a result, enhance the fluid retention capability of the material; and the other in which the material does not contain gel. We also study the cyclic wetting behavior in the case of the sample containing gel particles.

Problem description

The square specimen is 101.6 mm on a side, and its thickness is 20 mm.

Figure 1 shows one-quarter of the problem, modeled with a uniform 10×10 mesh of CPE8RP plane strain elements. The problem is also solved with a 5×5 mesh, a 15×15 mesh, and meshes using CPE4P, CPE4RP, and CPE6MP elements without significant changes in the results. When second-order elements are used where partially saturated flow is of concern, the use of reduced-integration elements is recommended since the fully integrated elements may lead to spurious oscillations during the initial stages of the transient.

Three-dimensional analyses are also performed using C3D4P, C3D6P, and C3D8P elements.

Material

The permeability of the fully saturated material is 3.7×10^{-4} m/sec. The partially saturated permeability is the default model, which assumes that the permeability varies as a cubic function of saturation. The specific weight of the material is 10^5 N/m^3. The initial void ratio is 5.0 throughout the sample. The capillary action in the porous medium is defined by the absorption/exsorption curves shown in *Figure 2*. These curves give the (negative) pore pressure versus saturation relationship for absorption and exsorption behavior. The transition between absorption and exsorption and vice-versa takes place along a scanning slope that is set by default to 1.05 times the largest slope of any branch in the absorption/exsorption curves. The initial conditions for pore pressure and saturation are assumed to be those at the beginning of the absorption curve, so the initial saturation is 0.05 and the initial pore pressure is −10 kPa.

The gel particles have a radius of 0.5 mm when completely dry and are capable of swelling to a maximum radius of 1.5 mm when fully exposed to fluid. There are 1.0×10^8 gel particles in each cubic meter of the porous material. The relaxation time constant for swelling of the gel particles is 500 sec. A dummy elastic modulus of 10^4 N/m^2 is prescribed to complete the material definition.

Loading and controls

The "loading" consists of prescribing a zero pore pressure (corresponding to full saturation) at the center of the sample, node 1 in *Figure 1*. This is based on the assumption that, in the demand wettability test, the sample has available to it as much fluid as necessary to cause saturation at that point. This boundary condition is held fixed for 600 seconds to model the fluid acquisition process. Then a draining period of 600 seconds is modeled by prescribing a pore pressure of −10 kPa at node 1; this corresponds to a saturation of 10%, which is the least saturation the sample can achieve after it has been wetted (see *Figure 2*). The draining procedure we use is not physically realistic, since the free fluid saturation cannot drop to 10% instantaneously as we assume. Nevertheless, it serves to illustrate the behavior of the model. Finally, we model the rewetting process over a time period of 800 seconds in the third step of analysis by once again prescribing zero pressure at the center of the sample.

The analysis is performed with a transient soils consolidation procedure (*Coupled Pore Fluid Diffusion and Stress Analysis*) using automatic time incrementation. The pore pressure tolerance that controls the automatic incrementation is set to a large value since we expect the nonlinearity of the material to restrict the size of the time increments during the transient stages of the analysis and we do not wish to impose any further control on the accuracy of the time integration. The volume flux tolerance that controls the accuracy of the solution of the flow continuity equations is set to 1.0×10^{-8} m^3/sec (*Convergence Criteria for Nonlinear Problems*). This is less than 1% of the flux that occurs during the initial wetting stage of the problem. The problem can also be run with the default tolerance. The results obtained are unchanged. However, in this problem the default tolerances calculated by Abaqus are extremely tight, resulting in additional iterations without benefit to the solution. For this reason we define a less stringent tolerance.

An important issue in these transient, partially saturated flow 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 give 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. As discussed in *Coupled Pore Fluid Diffusion and Stress Analysis*, the criterion is

$$\Delta t > \frac{\gamma n^0}{6 k_s k} \frac{ds}{du} (\Delta \ell)^2,$$

where γ is the specific weight of the wetting liquid, n^0 is the initial porosity of the material, k is the fully saturated permeability of the material, k_s is the permeability-saturation relationship, ds/du is the rate of change of saturation with respect to pore pressure as defined in the absorption/exsorption material behavior (*Sorption*), and $\Delta \ell$ is a typical element dimension. For our model we have $\Delta \ell =$ 5.08 mm (the size of an element side), $\gamma =$ 1.0×10^4 N/m^3, $k =$ 3.7×10^{-4} m/sec, $k_s = s^3$, and $n^0 =$ 5/6. Near node 1, where we apply the boundary condition, we will approach full saturation conditions early in the transient. If we choose a saturation of 0.9 and the corresponding ds/du in the absorption curve, we can calculate a Δt of about 0.1 sec. We choose to start the analysis with a time increment of 1 sec.

If time increments smaller than the critical value are used, spurious oscillations may appear in the solution (except when reduced-integration, linear, or modified triangular elements are used, in which case Abaqus uses a special integration scheme for the wetting liquid storage term to avoid this problem). If the problem requires analysis with smaller time increments than the critical value, a finer mesh is required. Generally there is no upper limit on the time step, except accuracy, since the integration procedure is unconditionally stable.

Results and discussion

Since the volume occupied by the sample is fixed (all displacements have been constrained) we must expect the volume of fluid absorbed to be the same in the cases of the sample with and without gel particles; the difference will be in the proportions of the volume of the sample that will be occupied by free fluid and fluid trapped in the gel particles. *Figure 3* shows the time history of the pore pressures at six nodes along the diagonal of the

sample during the first wetting stage of the problem; these histories are identical for the two samples. *Figure 4* shows the history of the volume of fluid absorbed by the samples at node 1. In *Figure 5* we show the histories of free fluid saturation at the six integration points closest to the six nodes for which the pore pressure histories are plotted; again these histories are identical for the two samples. The void ratio for the sample without gel remains constant at 5 throughout the test, whereas in the sample with gel it decreases to values below 1, as shown in the time histories of *Figure 6*: as the gel particles grow in a confined volume, the void space available for free fluid flow has to decrease. The growth of the gel particles in the case of the sample with gel is shown in *Figure 7*, where we plot the time histories of the ratio of the volume of gel to the total volume. The proportion of the total volume occupied by the different phases of the porous medium at the beginning and at the end of the wetting stage is shown in *Figure 8* and *Figure 9* for the case of the samples with and without gel.

The cyclic wetting behavior of the sample containing gel particles is given in *Figure 10* to *Figure 14*, where we show histories of pore pressure, volume of fluid absorbed, free fluid saturation, gel volume ratio, and void ratio. The trends observed are as expected, with pore pressure and saturation increasing and decreasing as the volume of fluid in the sample increases and decreases. In addition, once the sample with gel has been wetted, it can never dry out to the original state because some fluid remains trapped in the gel particles. During the draining stage the gel particles stop growing when the saturation of the surrounding free fluid falls below the value required to keep the gel growing. At the same time the void ratio also has to remain constant (*Figure 14*). Finally, in the rewetting stage, full saturation is achieved more quickly than in the first wetting stage since the sample starts off with a higher saturation and has less capacity to absorb fluid.

Input files

partsatflow_cpe8rp.inp	Cyclic demand wettability test in the case of the sample containing gel particles, element type CPE8RP.
partsatflow_cpe4h.inp	Element type CPE4P.
partsatflow_cpe4rp.inp	Element type CPE4RP.
partsatflow_cpe6mp.inp	Element type CPE6MP.
partsatflow_c3d4p.inp	Element type C3D4P.
partsatflow_c3d6p.inp	Element type C3D6P.
partsatflow_c3d8p.inp	Element type C3D8P.
partsatflow_postoutput.inp	Postprocessing analysis.

To run the first test in the case of a sample without gel, the **GEL* material option needs to be removed.

Figures

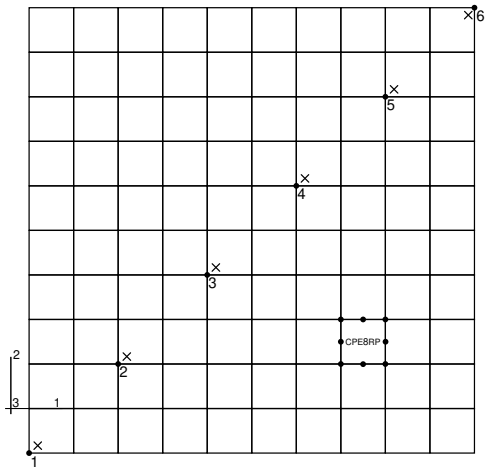


Figure 1: Finite element model for constrained demand wettability example.

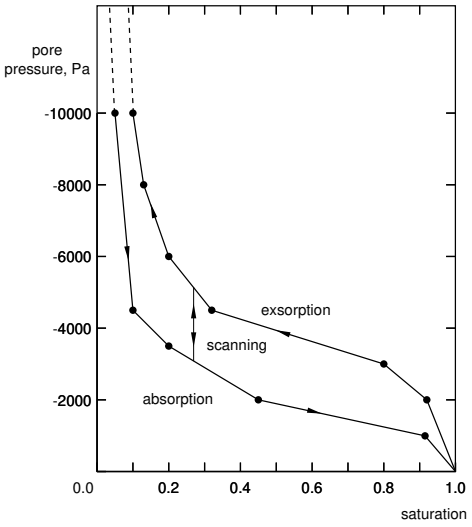


Figure 2: Absorption/exsorption curves for the porous material.

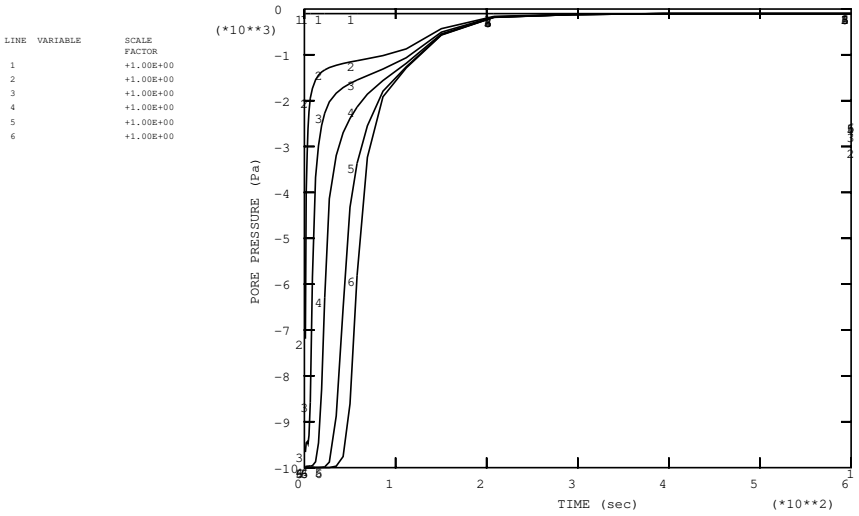


Figure 3: Pore pressure histories for both samples (with and without gel).

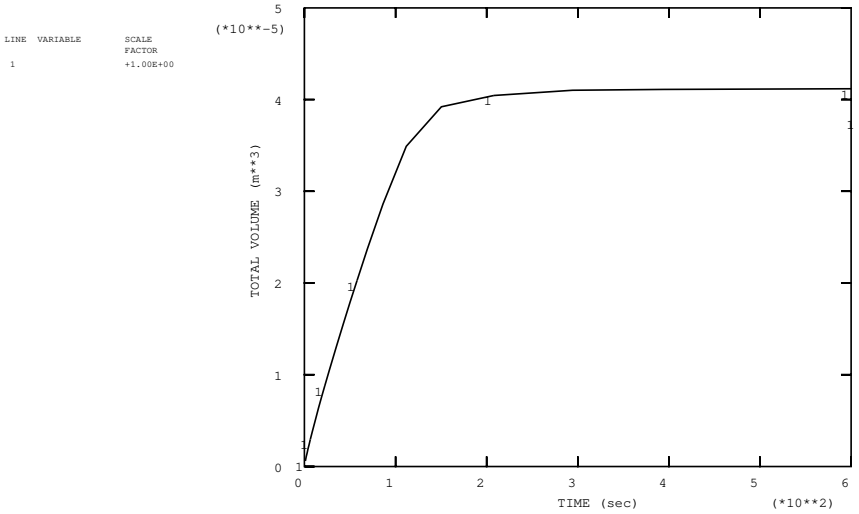


Figure 4: History of fluid volume absorbed at node 1 for both samples (with and without gel).

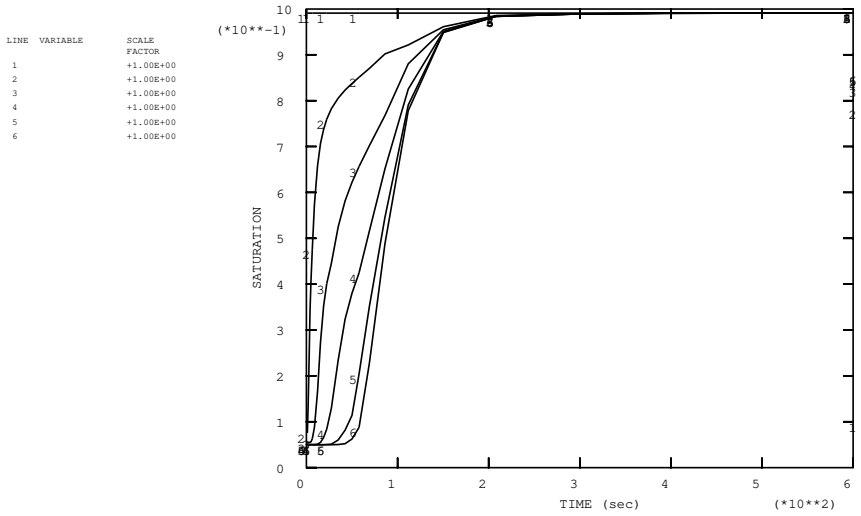


Figure 5: Saturation histories for both samples (with and without gel).

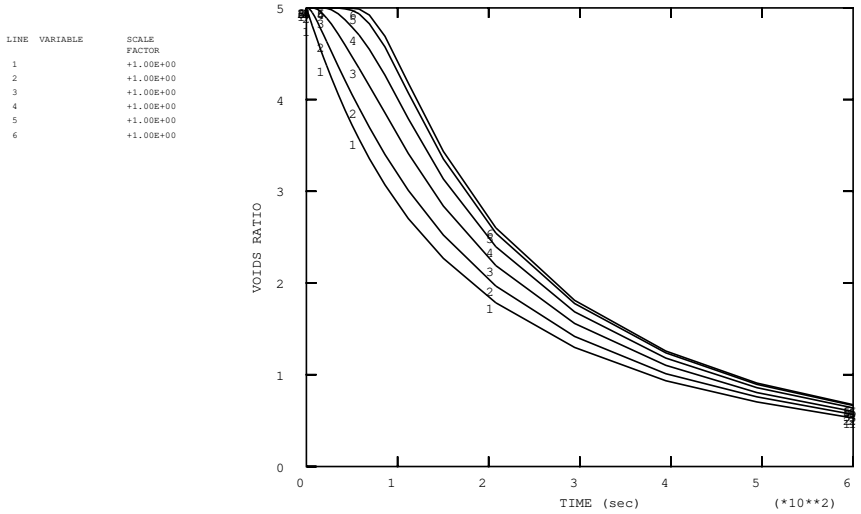


Figure 6: Void ratio histories for sample with gel.

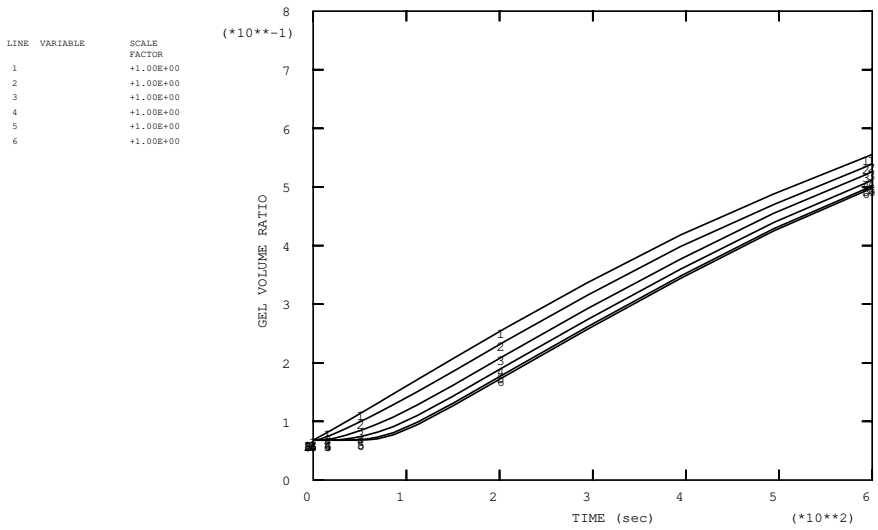


Figure 7: Gel volume ratio histories for sample with gel.

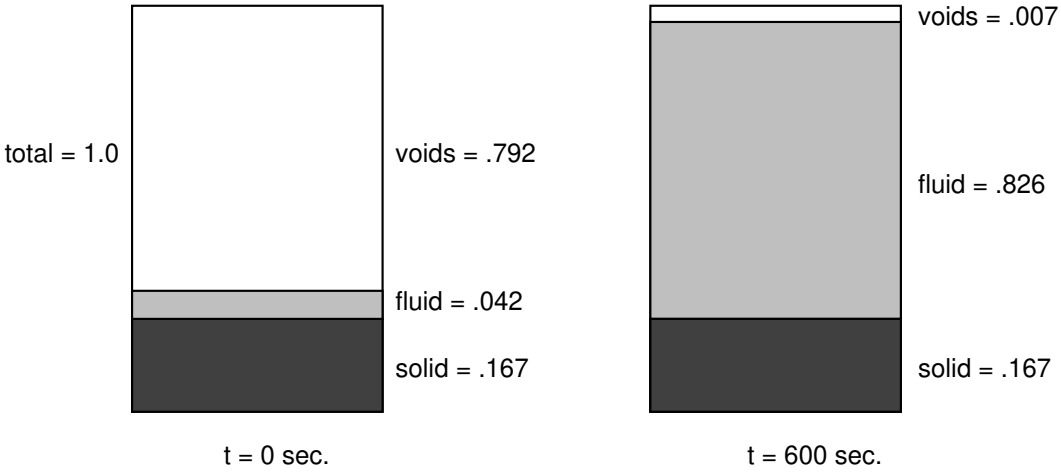


Figure 8: Volume of different phases of porous medium—sample without gel.

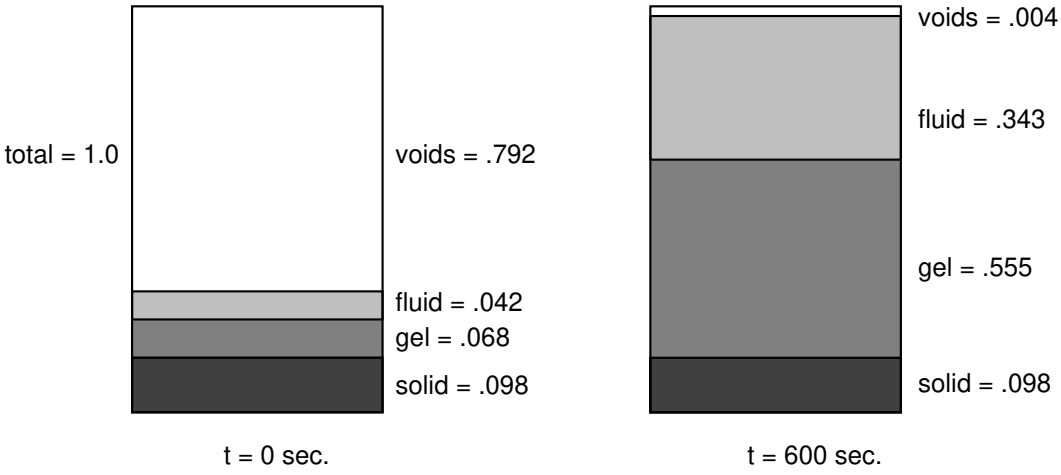


Figure 9: Volume of different phases of porous medium—sample with gel.

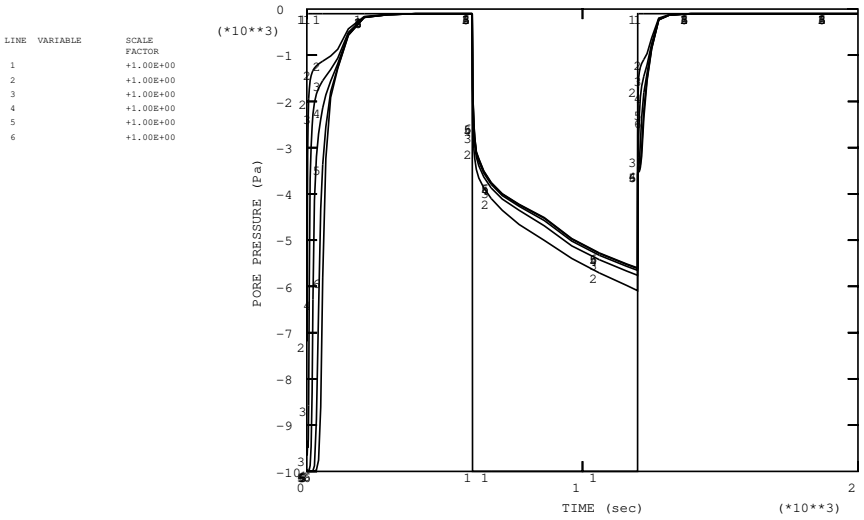


Figure 10: Pore pressure histories for cyclic demand wettability test.

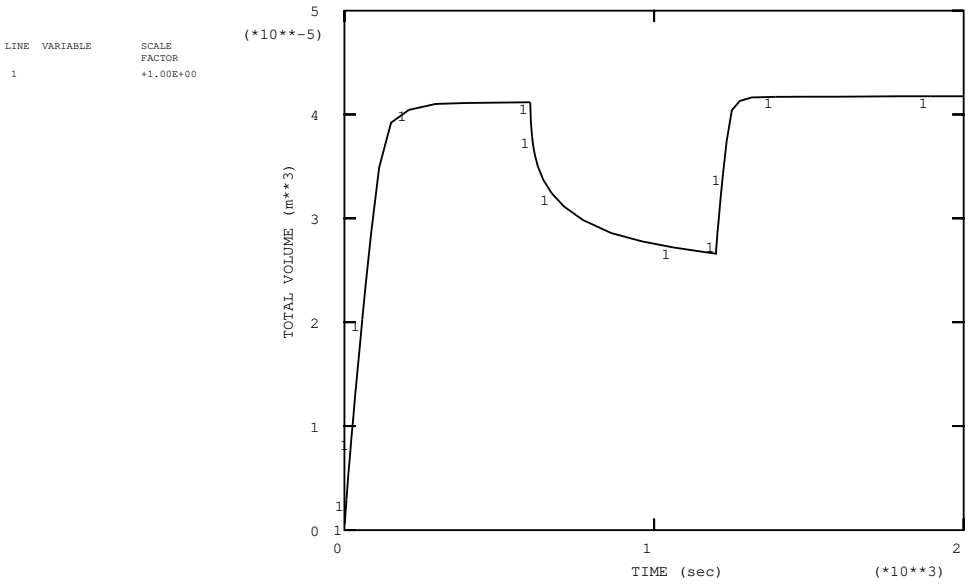


Figure 11: History of fluid volume absorbed at node 1 for cyclic demand wettability test.

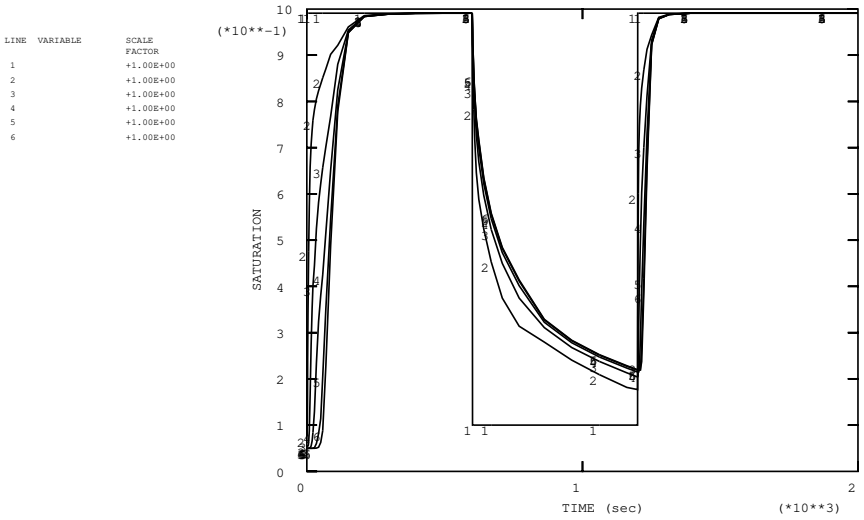


Figure 12: Saturation histories for cyclic demand wettability test.

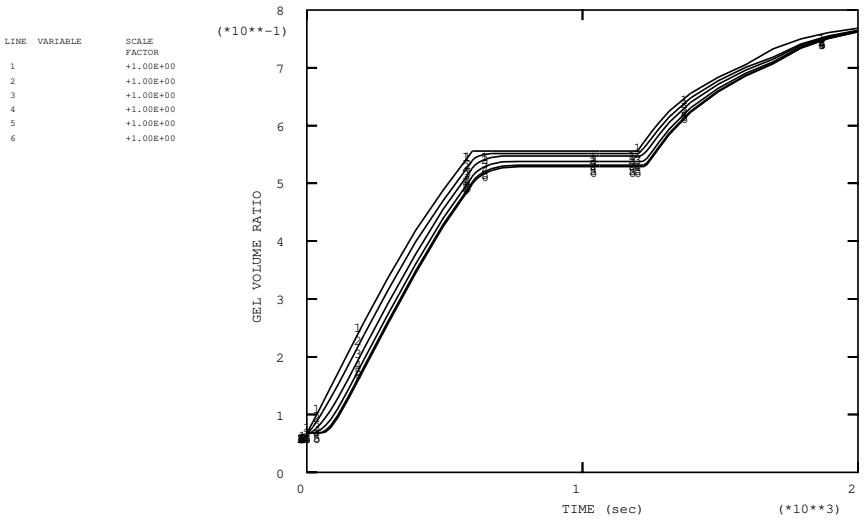


Figure 13: Gel volume ratio histories for cyclic demand wettability test.

LINE	VARIABLE	SCALE FACTOR
1		+1.00E+00
2		+1.00E+00
3		+1.00E+00
4		+1.00E+00
5		+1.00E+00
6		+1.00E+00

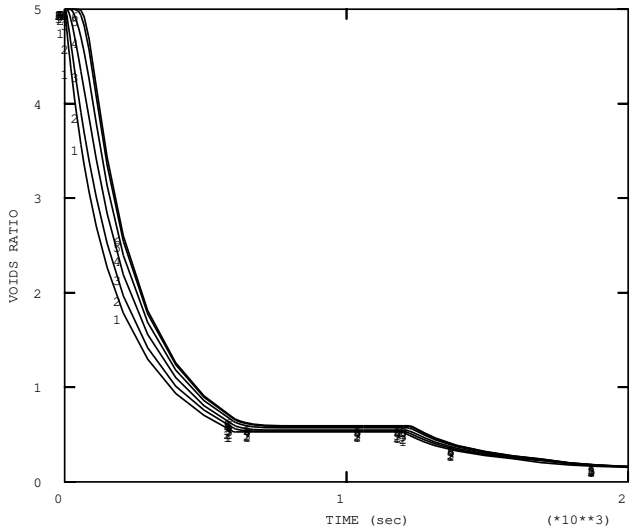


Figure 14: Void ratio histories for cyclic demand wettability test.

Demand wettability of a porous medium: coupled analysis

Products: Abaqus/Standard

Overview

This example illustrates the Abaqus capability to solve coupled problems involving stress equilibrium and partially saturated flow in porous media.

In this example we consider a one-dimensional "demand wettability" test, in which fluid is made available to the material at a certain location and the material is allowed to absorb as much fluid as it can. In this example we consider a column of material and allow it to absorb fluid at the bottom. The column is kinematically constrained in the horizontal direction so that all deformation will be in the vertical direction; in this sense the problem is one-dimensional. We investigate two cases: one in which the material contains a large number of gel particles that entrap fluid and, as a result, enhance the fluid retention capability of the material; and the other in which the material does not contain gel. Additional tests are provided that illustrate the use of solution mapping along with modeling of gel particles.

Problem description

The column of material is 50.8 mm high. We model the problem with 10 CPE8RP plane strain elements. In addition, input files containing different element types are included for verification purposes. The mesh is shown in *Figure 1*. We constrain all horizontal displacements and the vertical displacements at the bottom of the column.

Material

The properties pertaining to the partially saturated flow behavior of the material are the same as those used in *Partially saturated flow in a porous medium*. For the mechanical properties we assume the material is elastic, with Young's modulus 10000 Pa and Poisson's ratio 0.0. The mechanical properties of the gel particles are assumed to be similar to those of a fluid since they are mostly made up of absorbed fluid. Therefore, a bulk modulus of 2.0×10^9 Pa is specified for the gel.

The initial conditions for pore pressure and saturation are assumed to be those at the beginning of the absorption curve, so the initial saturation is 0.05 and the initial pore pressure is −10000 Pa.

Loading and controls

In the first step of the analysis we establish stress equilibrium in the original configuration of the column of material. A stress of 500 Pa is applied to the mesh to balance the initial pore pressure and saturation conditions.

The effective stress principle $\overline{\boldsymbol{\sigma}} = \boldsymbol{\sigma} + s u \mathbf{I}$ (s is the saturation and u is the pore pressure) then gives zero effective stresses, $\overline{\boldsymbol{\sigma}}$, for the undeformed configuration.

The "loading" consists of prescribing essentially zero pore pressure (corresponding to full saturation) at the bottom of the column. This is based on the assumption that, in the demand wettability test, the sample has available to it as much fluid as necessary to cause saturation at that point. This boundary condition is held fixed for 3000 seconds to model the fluid acquisition process.

The analysis is performed with the transient soils consolidation procedure using automatic time incrementation. The pore pressure tolerance that controls the automatic incrementation is set to a large value since we expect the nonlinearity of the material to restrict the size of the time increments during the transient stages of the analysis and we do not wish to impose any further control on the accuracy of the time integration.

The choice of initial time increment in these transient partially saturated flow problems is important for some element types, to avoid spurious solution oscillations. This is discussed in *Partially saturated flow in a porous medium*. As discussed in *Coupled Pore Fluid Diffusion and Stress Analysis*, the criterion for a minimum usable time increment in partial-saturation conditions is

$$\Delta t > \frac{\gamma n^0}{6 k_s k} \frac{ds}{du} (\Delta \ell)^2,$$

where γ is the specific weight of the wetting liquid, n^0 is the initial porosity of the material, k is the fully saturated permeability of the material, k_s is the permeability-saturation relationship, ds/du is the rate of change of saturation with respect to pore pressure as defined in the absorption/exsorption material behavior (*Sorption*), and $\Delta \ell$ is a typical element dimension. For our model we have $\Delta \ell =$ 5.08 mm (the size of an element side), $\gamma =$ 1.0×10^4 N/m^3, $k =$ 3.7×10^{-4} m/sec, $k_s = s^3$, and $n^0 =$ 5/6. Adjacent to where we apply the fully saturated boundary condition, elements will span a region from initial to full saturation early in the transient. A conservative estimate of the minimum time increment is found by choosing the initial saturation of 0.05. From this, we compute k_s, ds/du, and a value of Δt of about 70 sec. We find, in practice, that an initial increment of 50 sec is adequate to avoid oscillations in this problem. For the remaining input files the initial time increment is chosen as discussed in *Partially saturated flow in a porous medium* since we have the same material properties and spatial discretization.

In this analysis the prevailing pore pressure in the medium approaches the magnitude of the stiffness of the material skeleton elastic modulus. When reduced-integration elements are used in such cases, the default choice for the hourglass stiffness control, which is based on a scaling of skeleton material constitutive parameters, may not be adequate to control hourglassing in the presence of the relatively large pore pressure fields. An appropriate hourglass control setting in these cases should scale with the expected magnitude of pore pressure changes over an element and must be defined explicitly by the user.

Geometric nonlinearities are considered in the analysis since we expect large deformations due to the growth of the gel particles.

Results and discussion

In the case of the specimen without gel we expect the material to absorb fluid until it is fully saturated, without any significant change in volume. However, in the case of the specimen containing gel particles, we expect a significant volume increase associated with the swelling of the gel particles as they entrap fluid. *Figure 2* shows the time history of the pore pressures at six nodes along the height of the column of material; this is identical for the specimens with and without gel. The time history of the volume of fluid absorbed by the two specimens is shown in *Figure 3* for the case of the material with gel and *Figure 4* for the material without gel: the specimen containing gel absorbs roughly twice as much fluid (which is consistent with the fact that it roughly doubles in volume as the gel expands during absorption). Time histories of vertical displacement for the material with gel are shown in *Figure 5*. The material without gel does not show a significant change in volume. The growth of the gel particles in the case of the sample with gel is shown in *Figure 6*, where we plot the time histories of the ratio of the volume of gel in the specimen to the total volume of the specimen at the six integration points closest to the six nodes for which the pore pressure histories are plotted. In *Figure 7* we show time histories of free fluid

saturation. These histories are identical for the two samples. The void ratio for the sample without gel remains close to its initial value of 5.0 throughout the test, whereas in the sample with gel the void ratio decreases to values below 1.0, as shown in the time histories of *Figure 8*. This is a result of the void space available for free fluid flow decreasing as the gel particles swell.

Moisture swelling of the soil skeleton is added to the material behavior for verification purposes. With the addition of the moisture swelling more fluid volume is absorbed into the specimen, the specimen becomes longer when compared to the model without moisture swelling, and it takes more time to saturate the specimen. These observations are consistent with the added swelling of the soil skeleton in the presence of the moisture.

Input files

demandwetpormed_c3d4p_gel.inp	Element type C3D4P.
demandwetpormed_c3d4ph_gel.inp	Element type C3D4PH.
demandwetpormed_c3d4pt_gel.inp	Element type C3D4PT.
demandwetpormed_c3d4pht_gel.inp	Element type C3D4PHT.
demandwetpormed_c3d6p_gel.inp	Element type C3D6P.
demandwetpormed_c3d6ph_gel.inp	Element type C3D6PH.
demandwetpormed_c3d6pt_gel.inp	Element type C3D6PT.
demandwetpormed_c3d6pht_gel.inp	Element type C3D6PHT.
demandwetpormed_c3d8p_gel.inp	Element type C3D8P.
demandwetpormed_c3d8p_gel_post.inp	**POST OUTPUT* analysis.
demandwetpormed_c3d8rp_gel.inp	Element type C3D8RP.
demandwetpormed_cax4p_gel.inp	Element type CAX4P.
demandwetpormed_cpe4ph_gel.inp	Element type CPE4PH.
demandwetpormed_cpe4ph_gel_post.inp	**POST OUTPUT* analysis.
demandwetpormed_cpe8rp_gel.inp	Sample containing gel particles (element type CPE8RP).
demandwetpormed_cpe8rp_swell.inp	Adds moisture swelling to the material behavior in demandwetpormed_cpe8rp_gel.inp.
demandwetpormed_cpe8rp_nogel.inp	Same as demandwetpormed_cpe8rp_gel.inp, except that **GEL* is removed.
demandwetpormed_cpe8rp_gel_anc.inp	**MAP SOLUTION* ancestor analysis.
demandwetpormed_cpe8rp_gel_des.inp	**MAP SOLUTION* descendant analysis.

Figures

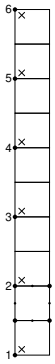


Figure 1: Finite element model for coupled demand wettability example.

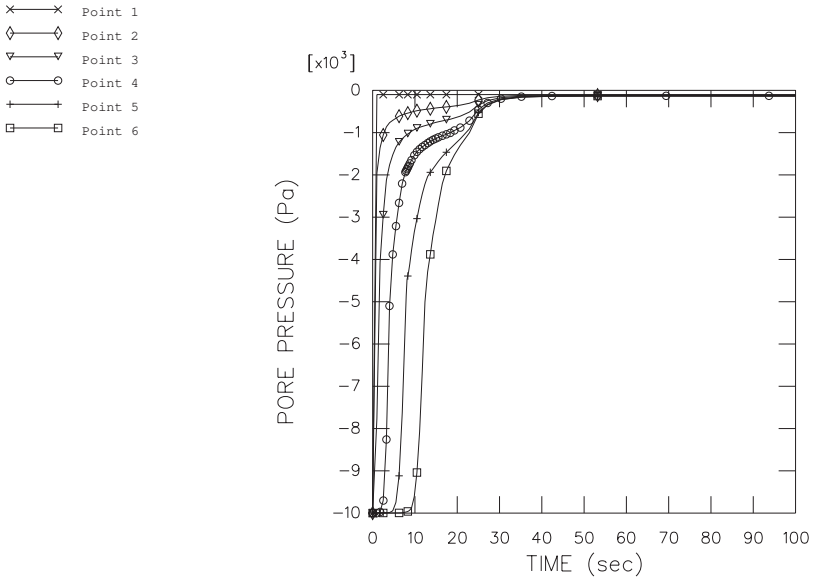


Figure 2: Pore pressure histories for both samples (with and without gel).

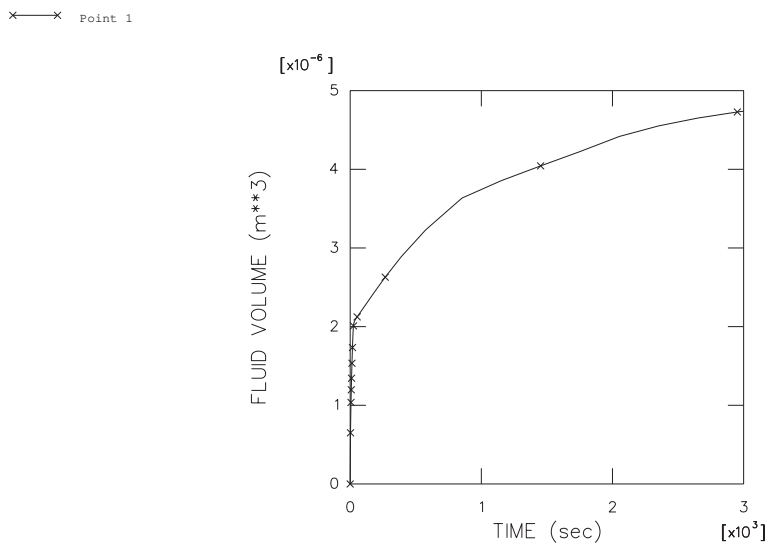


Figure 3: History of fluid volume absorbed at node 1 of sample with gel.

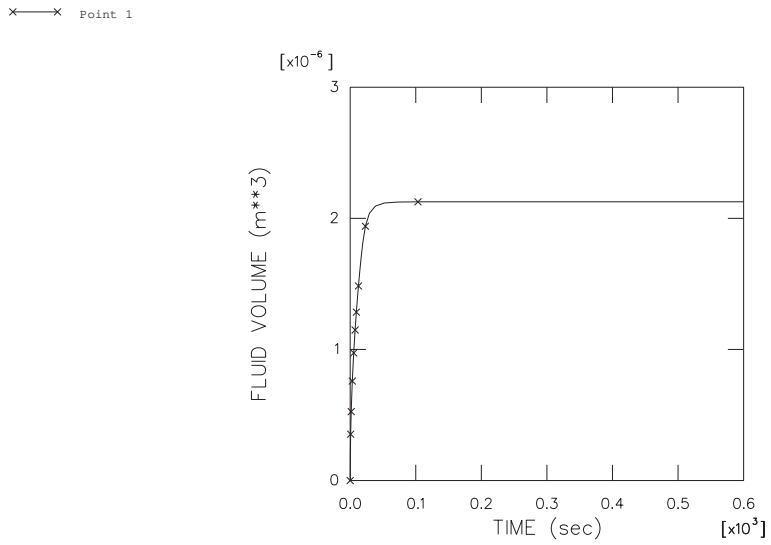


Figure 4: History of fluid volume absorbed at node 1 of sample without gel.

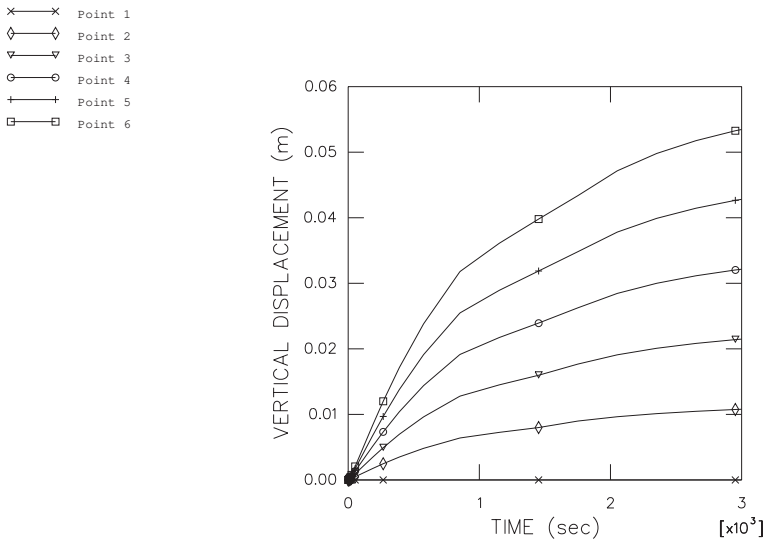


Figure 5: Vertical displacement histories for sample with gel.

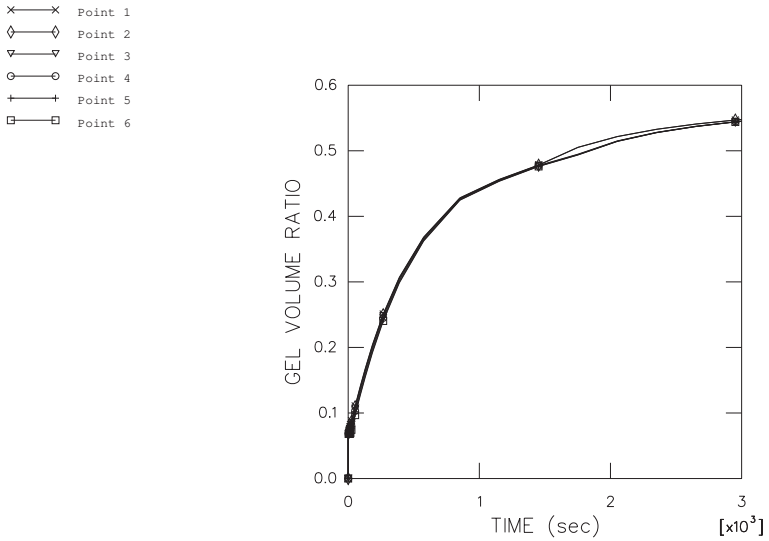


Figure 6: Gel volume ratio histories for sample with gel.

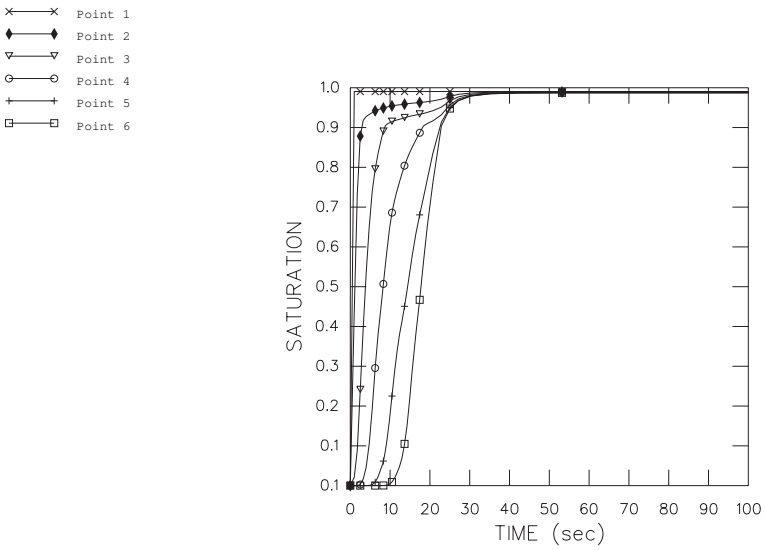


Figure 7: Saturation histories for both samples (with and without gel).

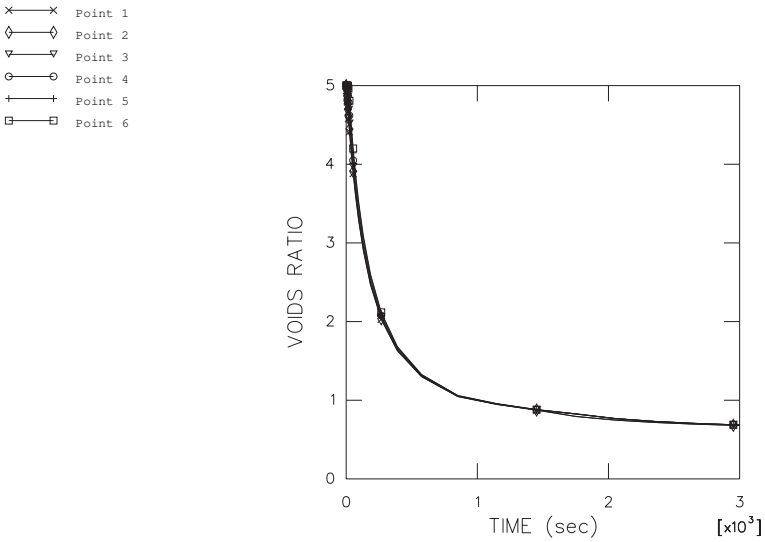


Figure 8: Void ratio histories for sample with gel.

Wicking in a partially saturated porous medium

Products: Abaqus/Standard

Overview

This example illustrates the Abaqus capability to solve fluid flow problems in partially saturated porous media where the effects of gravity are important.

In this example we consider a one-dimensional “wicking” test where the absorption of fluid takes place against the gravity load caused by the weight of the fluid. In such a test fluid is made available to the material at the bottom of a column, and the material absorbs as much fluid as the weight of the rising fluid permits. In this example we consider a column of material, kinematically constrained in the horizontal direction so that all deformation will be in the vertical direction; in this sense the problem is one-dimensional. We investigate two cases: one in which the column is not allowed to deform (uncoupled flow problem), and the other in which we consider the deformation of the material (coupled problem).

Problem description

The column of material is 1.0 m high and 0.1 m wide. We model the problem with 10 CPE8RP plane strain elements. In addition, input files containing different element types are included for verification purposes. The mesh is shown in *Figure 1*. We constrain all horizontal displacements. In the deforming column problem we constrain the vertical displacements at the bottom of the column, while in the rigid column problem we constrain all the vertical displacements.

Material

The permeability of the fully saturated material is 3.7×10^{-4} m/sec. The default model is used for the partially saturated permeability. This assumes that the permeability varies as a cubic function of saturation. The specific weight of the fluid is 10^4 N/m^3. The capillary action in the porous medium is defined by the absorption/exsorption curves shown in *Figure 2*. These curves give the (negative) pore pressure versus saturation relationship for absorption and exsorption behavior. The transition between absorption and exsorption and vice-versa takes place along a scanning slope which in this example is set by default to 1.05 times the largest slope of any branch in the absorption/exsorption curves. The fluid is assumed to be water, with a bulk modulus of 2 GPa. For the mechanical properties we assume the material is elastic with Young's modulus 50000 Pa and Poisson's ratio 0.0. The dry mass density of the material is 100 kg/m^3.

The initial condition for saturation is 5% throughout the column. The initial conditions for pore pressure must have a gradient that is equal to the specific weight of the fluid so that, according to Darcy's law, there is no initial flow. For this purpose we assume the initial pore pressures vary linearly from −12000 Pa at the bottom of the column to −22000 Pa at the top of the column. These initial conditions satisfy the pore pressure/saturation relationship in that they are between the absorption and exsorption curves, as shown in *Figure 2*. The initial void ratio is 5.0 throughout the column. In the deforming column case the initial conditions for effective stress are calculated from the density of the dry material and fluid, the initial saturation and void ratio, and the initial pore pressures using equilibrium considerations and the effective stress principle. The procedure used is detailed in *Geostatic Stress State*. It is important to specify the correct initial conditions for this type of problem; otherwise, the system may be so far out of equilibrium initially that it may fail to start because converged solutions cannot be found.

Loading and controls

The weight is applied by gravity loading. In the case of the deforming column, an initial geostatic step is performed to establish the initial equilibrium state. The initial conditions in the column exactly balance the weight of the fluid and dry material so that no deformation or fluid flow takes place. Then the bottom of the column is exposed to fluid by prescribing zero pore pressure (corresponding to full saturation) at those nodes during a transient soils consolidation step. The fluid will seep up the column until the pore pressure gradient is equal to the weight of the fluid, at which time equilibrium is established.

The transient analysis is performed using automatic time incrementation. The pore pressure tolerance that controls the automatic incrementation is set to a large value since we expect the nonlinearity of the material to restrict the size of the time increments during the transient stages of the analysis and we do not wish to impose any further control on the accuracy of the time integration. The check on displacement and pore pressure changes is relaxed using solution controls. The analysis can also be done with the default tolerances, but Abaqus iterates a lot more without any gain in solution accuracy.

The choice of initial time increment in these transient partially saturated flow problems is important to avoid spurious solution oscillations for some element types (see*Partially saturated flow in a porous medium*). As discussed in *Coupled Pore Fluid Diffusion and Stress Analysis*, the criterion for a minimum usable time increment in partial-saturation conditions is

$$\Delta t > \frac{\gamma n^0}{6 k_s k} \frac{ds}{du} (\Delta \ell)^2,$$

where γ is the specific weight of the wetting liquid, n^0 is the initial porosity of the material, k is the fully saturated permeability of the material, k_s is the permeability-saturation relationship, ds/du is the rate of change of saturation with respect to pore pressure as defined in the absorption/exsorption material behavior (*Sorption*), and $\Delta \ell$ is a typical element dimension. For our model we have $\Delta \ell =$ 0.1 m (the size of an element side), $\gamma =$ 1.0×10^4 N/m^3, $k =$ 3.7×10^{-4} m/sec, $k_s = s^3$, and $n^0 =$ 5/6. Adjacent to where we apply the fully saturated boundary condition, elements will span a region from initial to full saturation early in the transient. A conservative estimate of the minimum time increment is found by choosing the initial saturation of 0.05. From this, we compute k_s, ds/du, and a value of Δt of about 2700 sec. We find, in practice, that an initial increment of 1000 sec is adequate to avoid oscillations in this problem. For the remaining input files the initial time increment is chosen as 1 second.

In this analysis the prevailing pore pressure in the medium approaches the magnitude of the stiffness of the material skeleton elastic modulus. When reduced-integration elements are used in such cases, the default choice for the hourglass stiffness control, which is based on a scaling of skeleton-material constitutive parameters, may not be adequate to control hourglassing in the presence of the relatively large pore pressure fields. An appropriate hourglass control setting in these cases should scale with the expected magnitude of pore pressure changes over an element and must be defined explicitly by the user.

Results and discussion

The results for the rigid (uncoupled problem) and deforming (coupled problem) column cases are similar since the deformation of the column is small. The time history of the volume of fluid absorbed by the column is shown in *Figure 3*. The pore pressures at the two bottom nodes are tied with an equation constraint so that the total volume of fluid absorbed by the column is given directly in the output by the reaction (RVT) to the pore pressure at node 1. *Figure 4* shows the time history of the pore pressures at six nodes along the height of the column of material. In *Figure 5* we show time histories of fluid saturation at the six integration points closest to the six nodes for which the pore pressure histories are plotted. At steady state the pore pressure gradient must equal the weight of the fluid so that pore pressure varies linearly with height and saturation, therefore, varies in the same way (according to the absorption behavior) with respect to pressure or height. Thus, points close to the bottom

of the column are fully saturated, while those at the top are still at 5% saturation. This is illustrated in *Figure 6*, which is exactly the absorption curve of *Figure 2*.

Input files

wicking_cpe8rp_deform.inp	Case of the deforming column (element type CPE8RP).
wicking_cpe8rp_rigid.inp	Case of the rigid column (element type CPE8RP). Initial pore pressure specified with user subroutine *UPOREP*.
wicking_cpe8rp_rigid.f	User subroutine *UPOREP* used in wicking_cpe8rp_rigid.inp.
wicking_cpe4p_deform.inp	Element type CPE4P (deforming column).
wicking_cax4p_deform.inp	Element type CAX4P (deforming column).
wicking_c3d4p_deform.inp	Element type C3D4P (deforming column).
wicking_c3d4ph_deform.inp	Element type C3D4PH (deforming column).
wicking_c3d4pt_deform.inp	Element type C3D4PT (deforming column).
wicking_c3d4pht_deform.inp	Element type C3D4PHT (deforming column).
wicking_c3d6p_deform.inp	Element type C3D6P (deforming column).
wicking_c3d6ph_deform.inp	Element type C3D6PH (deforming column).
wicking_c3d6pt_deform.inp	Element type C3D6PT (deforming column).
wicking_c3d6pht_deform.inp	Element type C3D6PHT (deforming column).
wicking_c3d8ph_deform.inp	Element type C3D8PH (deforming column).
wicking_c3d8rp_rigid.inp	Element type C3D8RP (rigid column).
wicking_c3d8rp_rigid.f	User subroutine *UPOREP* used in wicking_c3d8rp_rigid.inp.

Figures

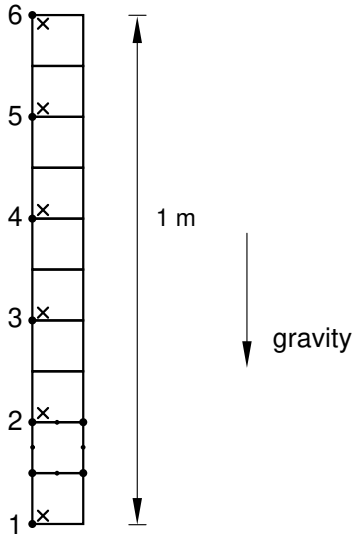


Figure 1: Finite element model for wicking example.

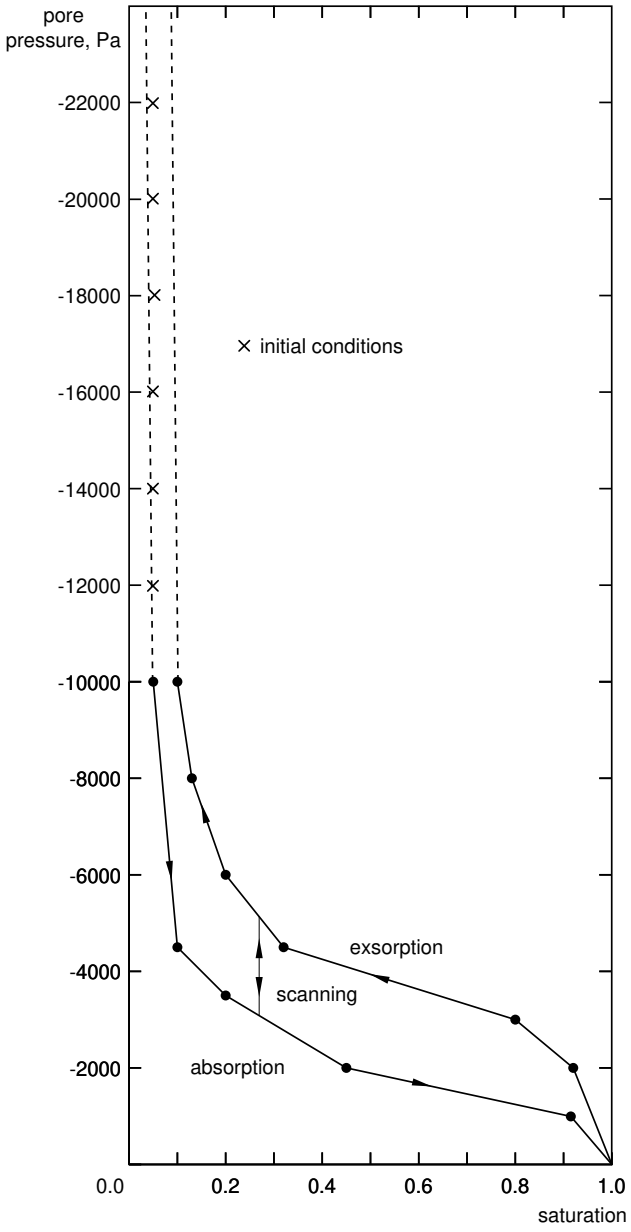


Figure 2: Absorption/exsorption curves and initial conditions.

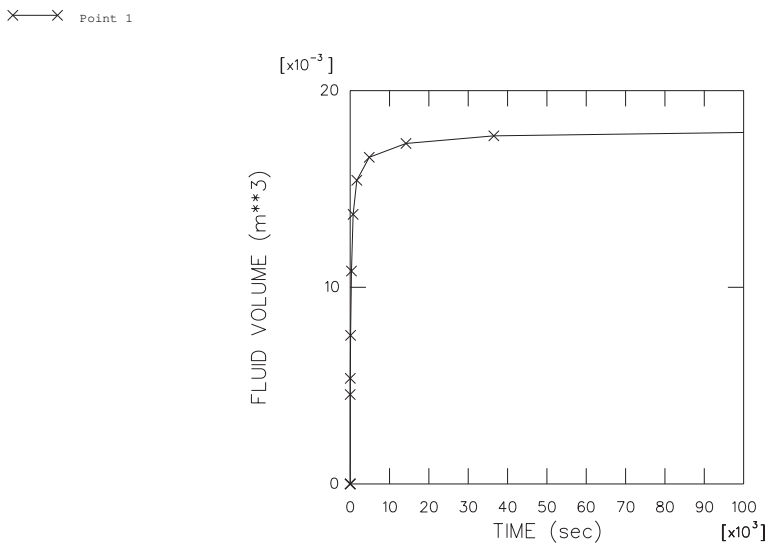


Figure 3: History of fluid volume absorbed at bottom of column.

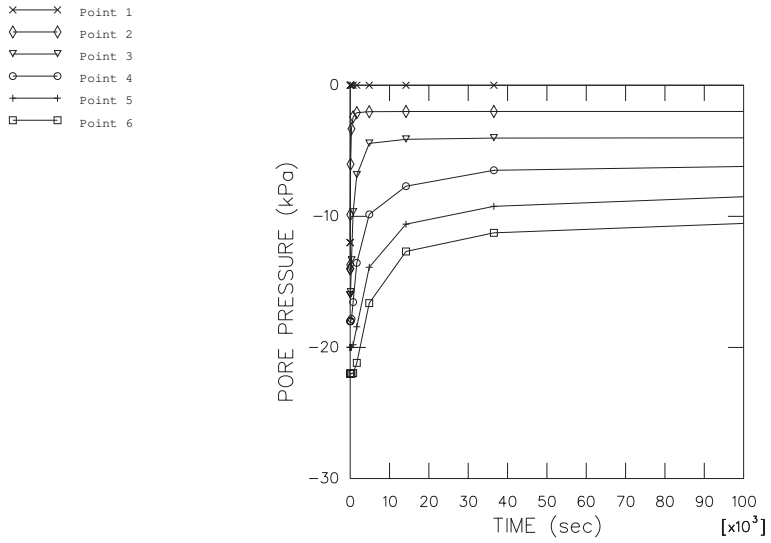


Figure 4: Pore pressure histories.

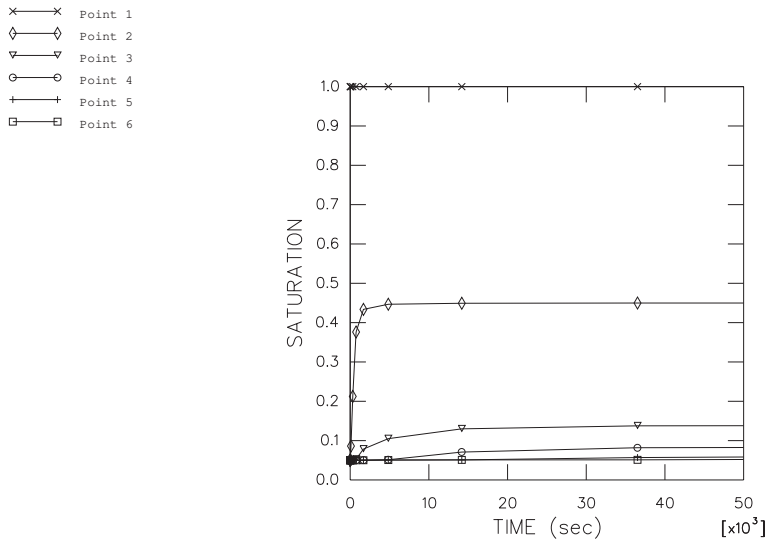


Figure 5: Saturation histories.

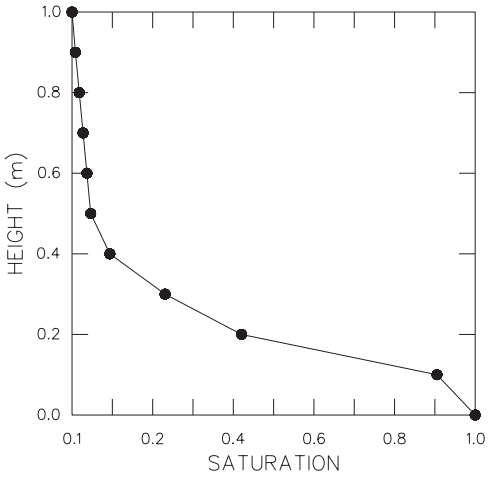


Figure 6: Saturation profile at steady-state conditions.

Desaturation in a column of porous material

Products: Abaqus/Standard

Overview

The purpose of this example is to validate the Abaqus capability to solve coupled fluid flow problems in partially saturated porous media where the effects of gravity are important.

For this example we compare Abaqus results with the experimental work of Liakopoulos (1965). Most experiments of this type are performed to check the hydraulic parameters; and, therefore, some assumptions about the mechanical behavior have to be made in the numerical model. Schrefler and Simoni (1988) have provided a numerical solution for the Liakopoulos experiment, and this example follows their assumptions about the mechanical behavior.

The Liakopoulos experiment consists of the drainage of water from a vertical column of sand. A column of perspex, 1 m high, is filled with Del Monte sand and instrumented to measure the moisture pressure at various points along the height of the column. Prior to the start of the experiment water is added continually at the top of the column and allowed to drain freely at the bottom of the column. The flow is regulated until zero pore pressure readings are obtained throughout the column. At this point flow is stopped and the experiment starts: the top of the column is made impermeable and the water is allowed to drain out of the column, under gravity. Pore pressure profiles in the column are measured during the drainage transient.

We investigate two cases: one in which the column is not allowed to deform (uncoupled flow problem), and the other in which we consider the deformation of the sand (coupled problem). The latter is expected to be a closer representation of the physical experiment.

Problem description

The column of material is 1 m high and 0.1 m wide. We model the problem with 10 CPE8RP plane strain elements. In addition, input files containing different element types are included for verification purposes. The mesh is shown in *Figure 1*. We constrain all horizontal displacements (the flow problem is one-dimensional). In the deforming column problem we constrain the vertical displacements at the bottom of the column, while in the rigid column problem we constrain all the vertical displacements.

Material

The properties used in this example pertaining to the partially saturated flow behavior of the material are taken from Liakopoulos (1965) and are as used by Schrefler and Simoni (1988): the pore pressure/saturation relationship is shown in *Figure 2*, and the permeability of the fully saturated material is 4.5×10^{-6} m/sec. The partially saturated permeability decreases linearly from this value to a value of 3.0×10^{-6} m/sec at a saturation of 0.85 and remains constant below that. A bulk modulus of 2 GPa is used for the water. The mechanical properties for the sand are not given by Liakopoulos. Following Schrefler and Simoni (1988), we assume the material is elastic with Young's modulus 1.3 MPa and Poisson's ratio 0. We also assume that the mass density of the dry material is 1500 kg/m^3, which is typical of sand.

The initial void ratio of the material is 0.4235. The initial conditions for pore pressure and saturation correspond to the fully saturated state of the sand at the beginning of the experiment: the initial saturation is 1.0, and the initial pore pressure is 0.0. There is some steady-state flow under these initial conditions because the zero gradient in pore pressure does not equilibrate the specific weight of the fluid.

In the deforming column case the initial conditions for effective stress are calculated from the density of the dry material and fluid, the initial saturation and void ratio, and the initial pore pressures using equilibrium considerations and the effective stress principle. The procedure used is detailed in *Geostatic Stress State*. It is

important to specify the correct initial conditions for this type of problem; otherwise, the system may be so far out of equilibrium initially that it may fail to start because converged solutions cannot be found.

Loading and controls

The weight is applied by gravity loading. In the case of the deforming column an initial step of a geostatic analysis is performed to establish the initial equilibrium state; the initial conditions in the column exactly balance the weight of the fluid and dry material so that no deformation takes place, while the zero pore pressure boundary conditions enforce the initial steady-state of fluid flow. Then the fluid is allowed to drain through the bottom of the column by prescribing zero pore pressures at these nodes during a transient soils consolidation step. The fluid will drain until the pressure gradient is equal to the weight of the fluid, at which time equilibrium is established.

The transient analysis is performed using automatic time incrementation. The pore pressure tolerance that controls the automatic incrementation is set to a large value since we expect the nonlinearity of the material to restrict the size of the time increments during the transient stages of the analysis and we do not wish to impose any further control on the accuracy of the time integration.

The choice of initial time step in these transient partially saturated flow problems is important. This is discussed in *Partially saturated flow in a porous medium*. For the parameters of this problem the initial time increment is chosen as 20 seconds.

Results and discussion

The profiles of pore pressure obtained in the coupled analysis (deformable column case) at different times during the drainage process are compared to the experimental results in *Figure 3*. *Figure 4* shows the corresponding comparison for the uncoupled analysis (rigid column). The results of the coupled analysis are closer to the experiment than those of the uncoupled analysis; in particular, the uncoupled analysis tends to overestimate the pore pressures in the early stages of the transient. As the transient continues, the material deformation slows (see the displacement histories of six points along the height of the column in *Figure 5*) and, therefore, the rigid column assumption becomes closer to reality; as steady-state is approached, both numerical solutions are in good agreement with the experiment. At steady state, the pore pressure gradient is equal to the weight density of the fluid, as required by Darcy's law. The time histories of the volume of fluid lost through the bottom of the column are shown in *Figure 6* for both the deformable and rigid columns: as expected, more fluid is lost in the deforming column case. *Figure 7* and *Figure 8* show the time history of fluid saturation and pore pressure at six points along the height of the column.

Input files

desaturation_c3d4p_deform.inp	Element type C3D4P (deforming column).
desaturation_c3d4ph_deform.inp	Element type C3D4PH (deforming column).
desaturation_c3d4pt_deform.inp	Element type C3D4PT (deforming column).
desaturation_c3d4pht_deform.inp	Element type C3D4PHT (deforming column).
desaturation_c3d6p_deform.inp	Element type C3D6P (deforming column).
desaturation_c3d6ph_deform.inp	Element type C3D6PH (deforming column).
desaturation_c3d6pt_deform.inp	Element type C3D6PT (deforming column).
desaturation_c3d6pht_deform.inp	Element type C3D6PHT (deforming column).
desaturation_c3d8p_deform.inp	Element type C3D8P (deforming column).
desaturation_c3d8p_deform_po.inp	**POST OUTPUT* analysis.
desaturation_c3d10mp_deform.inp	Element type C3D10MP (deforming column).
desaturation_cax4ph_deform.inp	Element type CAX4PH (deforming column).

desaturation_cax6mp_deform.inp	Element type CAX6MP (deforming column).
desaturation_cpe4p_deform.inp	Element type CPE4P (deforming column).
desaturation_cpe4rp_rigid.inp	Rigid column (element type CPE4RP).
desaturation_cpe6mph_rigid.inp	Rigid column (element type CPE6MPH).
desaturation_cpe8rp_deform.inp	Deforming column (element type CPE8RP).
desaturation_cpe8rp_rigid.inp	Rigid column (element type CPE8RP).
desaturation_cpe8rp_deftorigid.inp	Rigid column simulated by declaring CPE8RP elements as rigid.

References

- Liakopoulos, A. C., Transient Flow Through Unsaturated Porous Media, D. Eng. dissertation, University of California, Berkeley, 1965.
- Schrefler, B. A., and L. Simoni, "A Unified Approach to the Analysis of Saturated-Unsaturated Elastoplastic Porous Media," *Numerical Methods in Geomechanics*, vol. 1, pp. 205–212, 1988.

Figures

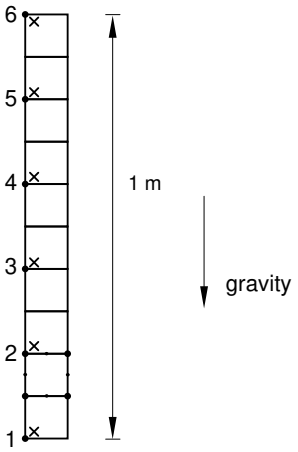


Figure 1: Finite element model for desaturation example.

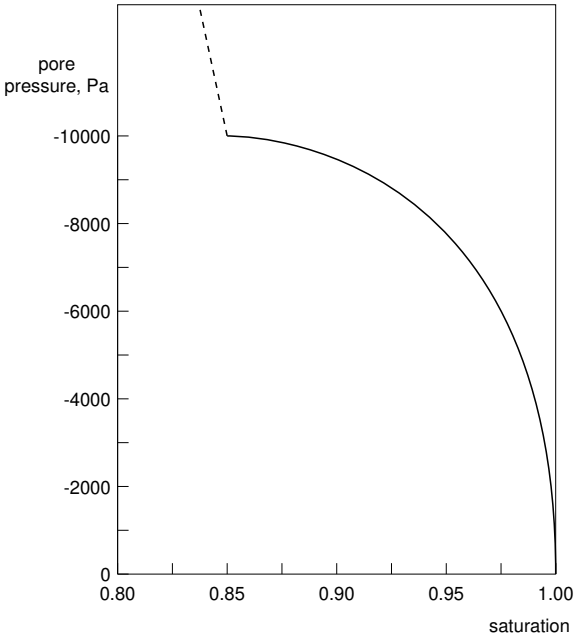


Figure 2: Absorption/exsorption curve for the porous material.

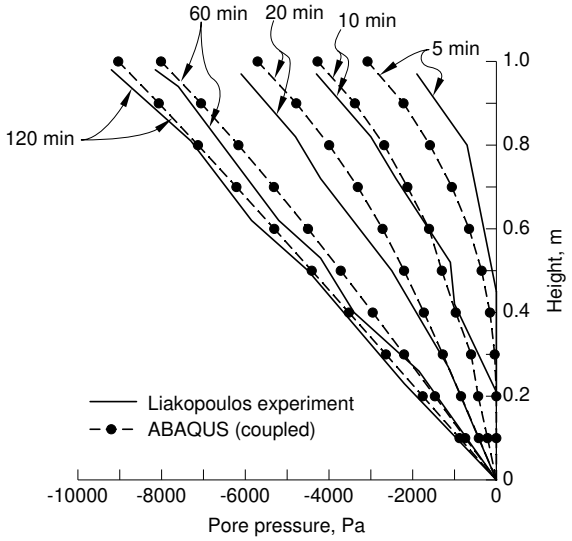


Figure 3: Pore pressure transient profiles (deformable column).

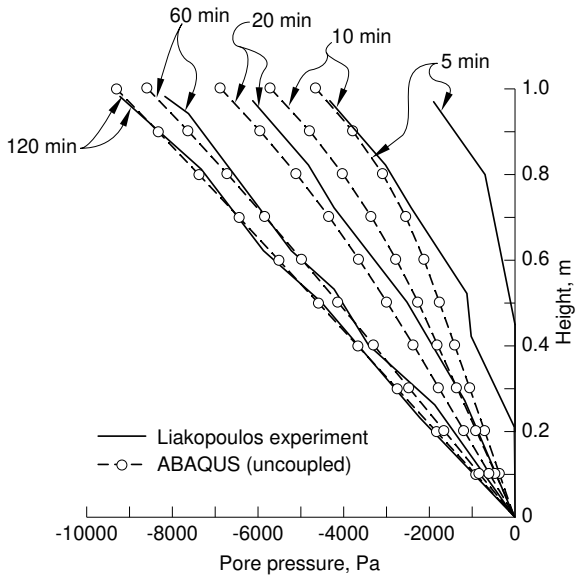


Figure 4: Pore pressure transient profiles (rigid column).

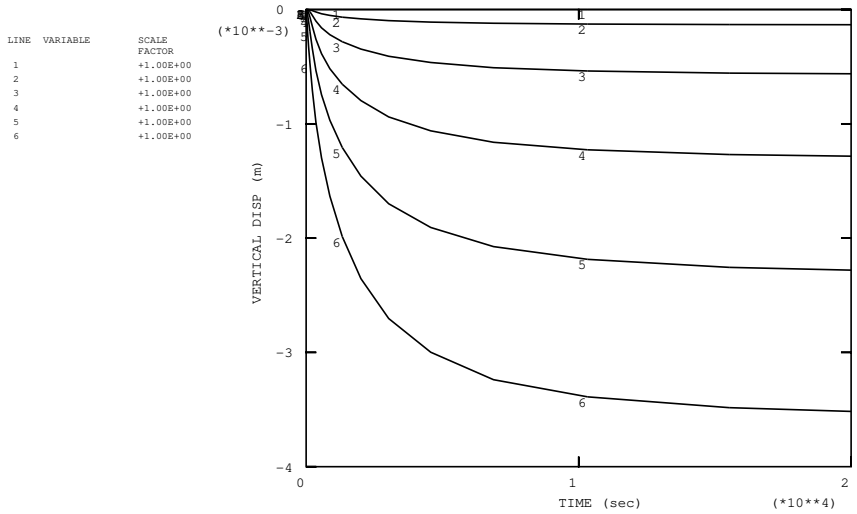


Figure 5: Vertical displacement histories.

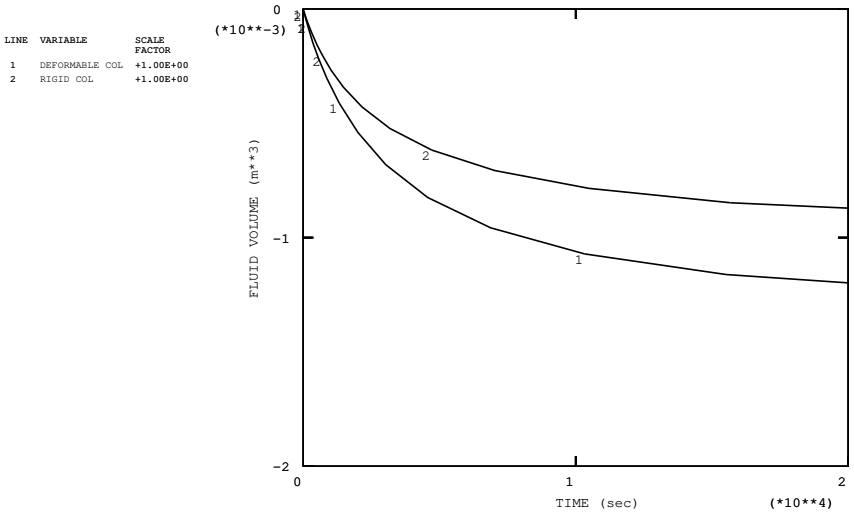


Figure 6: History of fluid volume lost at bottom of column (deformable and rigid columns).

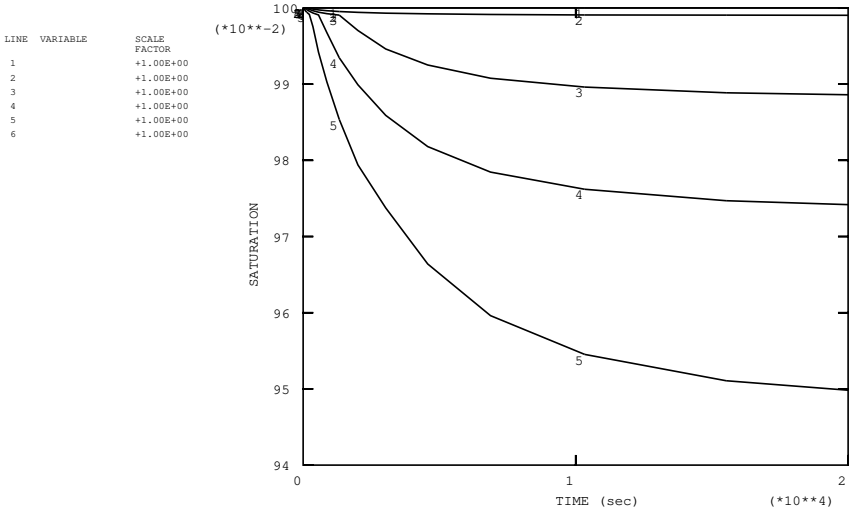


Figure 7: Saturation histories.

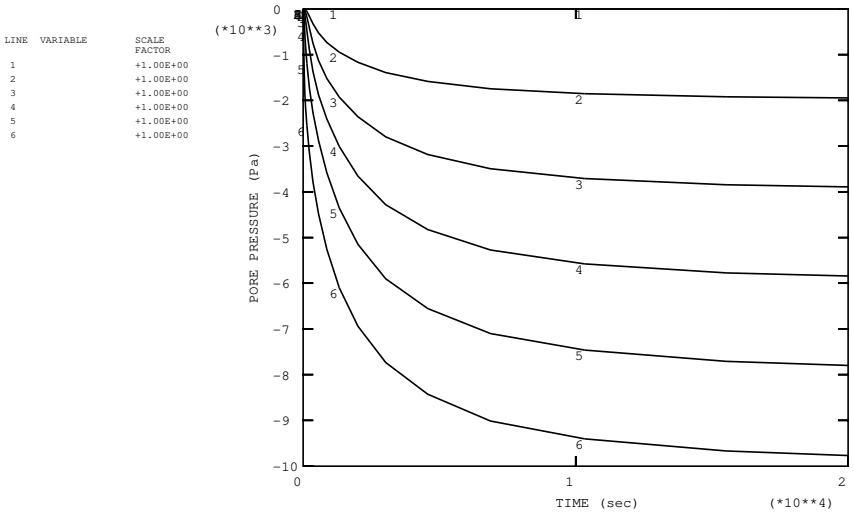


Figure 8: Pore pressure histories.

Substructures and mass diffusion analysis

In this section:

- *Analysis of a frame using substructures*
- *Thermomechanical diffusion of hydrogen in a bending beam*

Analysis of a frame using substructures

Products: Abaqus/Standard

Overview

This example demonstrates and verifies the substructuring capabilities in Abaqus.

Substructures provide significant savings in computer time in certain cases. They can be used in a linear analysis to divide a problem into several smaller analyses or to save parts of the structure that are not modified every time the analysis is done. Substructures can also be useful in a nonlinear analysis. As demonstrated in *The Hertz contact problem*, if any part of the structure remains linear during the analysis, the stiffness of that part can be calculated once and saved. Substructures can also be used in the analysis of linear perturbations around a state reached as a result of nonlinear response, as shown in *Vibration of a rotating cantilever plate*, where the vibration frequencies are calculated for a prestressed structure.

By default, Abaqus writes the substructures to the files whose name is defined by the job name and the step number. Alternatively, you can define the substructure name explicitly in the substructure generation analysis and when defining elements. The input files for this example show the use of these options.

Problem description

The example is a frame consisting of two columns connected by a beam (*Figure 1*). The columns and the beam are each modeled with 10 elements of type B21. The frame is loaded by a distributed load applied on the beam.

One-level substructure analysis

Once a substructure has been generated, it can be used just as any other element in an analysis. The substructure is connected to the rest of the model through specified degrees of freedom (the retained degrees of freedom). During substructure generation it is also possible to define load cases for the substructure. These load cases can then be scaled and applied during the analysis. In this example the loading on the beam is defined as a substructure load case and then applied at the global level.

Many structures include parts with the same geometry. In such cases the part can be generated as a single substructure that is then used several times. In this case the two columns are identical, so a substructure can be formed once for the column geometry and then used twice.

The analysis includes two steps. The first is a static load case in which the distributed load is applied to the structure. For this step the substructure analysis gives the same solution that is obtained by using the model without substructures, since the static reduction to a substructure stiffness matrix does not change that matrix.

In the second step a frequency analysis is performed to extract the first few vibrational modes of the structure. The technique of generating a substructure so that its dynamic response is modeled with reasonable accuracy is called “Component Mode Synthesis.” In this approach the dynamic response of the substructure is made up from a combination of its static modes of deformation and some of its natural modes of vibration obtained with its retained degrees of freedom fully constrained. In the case of a substructure with many internal degrees of freedom it may be expensive to extract these internal modes of vibration. In that case a sufficiently accurate dynamic representation of the substructure can often be obtained by “Guyan reduction.” This means that additional degrees of freedom that are not needed for the physical connection of the substructure to the rest of the mesh are retained simply to improve the dynamic modeling. The drawback to this method is that a judicious choice of these degrees of freedom is necessary to get the most benefit from the larger size of the substructure matrix. Component mode synthesis is, thus, a more reliable alternative but requires a frequency analysis of the substructure, which can be expensive for a large model. The modes are obtained by extracting some of the eigenfrequencies of the substructure

in a frequency step, with all of the retained degrees of freedom restrained with a boundary condition. The *n* modes to be used in the substructure generation are specified with *n* not greater than the number of modes that have been extracted. These extra modes are then added when the substructure is used in the global analysis. In general, this technique improves the representation of the substructure's dynamic behavior significantly, even with just a few additional modes. In this example we have added two extra modes in each of the substructures. *substrframe_1level_2mode_gen1.inp* and *substrframe_1level_2mode_gen2.inp* show the input data when extra dynamic modes are added.

Multilevel substructure analysis

Substructures can be used within substructures. We illustrate this capability by dividing each column into two identical parts. A substructure is then generated for the half-column, and a higher-level substructure is then generated for the full column by combining two substructures on the lowest level. The same process is followed for the beam. In this way a total of nine substructures are used to model the structure, although the repetitive structure requires very little calculation. Abaqus places no limit on the number of levels of substructures that can be used. *substrframe_2level_gen1.inp*, *substrframe_2level_gen2.inp*, *substrframe_2level_gen3.inp*, and *substrframe_2level_gen4.inp* show the input data for this example with two levels of substructures.

Substructure rotation and mirroring

Substructure stiffness and mass matrices are created based on the location and orientation of the elements when the substructure is generated. These matrices can be used anywhere in space as long as they are only translated. If the substructure is needed in a rotated configuration, the matrices must be transformed. Rotation and mirroring of a substructure is defined using a substructure property. The example illustrates this option with the three-dimensional frame structure shown in *Figure 2*. Element 82 on the usage level has been created by rotating substructure Z201 by 90°. Element 81 on the usage level has been created by mirroring substructure Z201 through the plane $\boldsymbol{x} = -\boldsymbol{y}$. Element 72 on the global level could have been created by a simple translation of substructure Z201 but has instead been created by both a rotation and a mirroring of the substructure. *substrframe_b31_gen1.inp* and *substrframe_b31_gen2.inp* show the input data for this analysis.

Results and discussion

The analysis without substructures provides “correct” results for this example (that is, the results are correct for the finite element discretization used). For the static loading case, substructure analysis does not imply any approximation; and the results with substructures are, therefore, identical. For the frequency analysis the substructure mass matrix is an approximate representation, so some accuracy is lost. *Table 1* and *Table 2* show comparisons of the lowest frequencies estimated by the planar and three-dimensional models, respectively. They show the significant improvement in results that is obtained by including extra modes.

Input files

substrframe_1level_2modes.inp	One level of substructures and two dynamic modes added for each substructure.
substrframe_1level_2mode_gen1.inp	Substructure generation referenced by the analysis substrframe_1level_2modes.inp.
substrframe_1level_2mode_gen2.inp	Substructure generation referenced by the analysis substrframe_1level_2modes.inp.
substrframe_2level.inp	Two levels of substructures.
substrframe_2level_gen1.inp	Substructure generation referenced by the analysis substrframe_2level.inp.

substrframe_2level_gen2.inp	Substructure generation referenced by the analysis substrframe_2level.inp.
substrframe_2level_gen3.inp	Substructure generation referenced by the analysis substrframe_2level.inp.
substrframe_2level_gen4.inp	Substructure generation referenced by the analysis substrframe_2level.inp.
substrframe_b31.inp	Analysis of the three-dimensional frame structure using substructures. In this case element 82 is created by a rotation, element 81 by mirroring in the plane $\boldsymbol{x} = -\boldsymbol{y}$, and element 72 by rotation followed by mirroring.
substrframe_b31_gen1.inp	Substructure generation referenced by the analysis substrframe_b31.inp.
substrframe_b31_gen2.inp	Substructure generation referenced by the analysis substrframe_b31.inp.
substrframe_nosubstr.inp	Analysis without substructures.
substrframe_1level.inp	Analysis with one level of substructures.
substrframe_1level_gen1.inp	Substructure generation referenced by the analysis substrframe_1level.inp.
substrframe_substrpath.inp	Shows the use of the two substructure files generated earlier in redoing the global analysis of the frame.
substrframe_3d_nosubstr.inp	Analysis of the three-dimensional frame structure without substructures.
substrframe_rotz201.inp	Analysis of the three-dimensional frame structure using substructures. The two structures in the *y*-direction have been created by rotating substructure Z201 by 90° about the *z*-axis.
substrframe_rotz201_gen1.inp	Substructure generation referenced by the analysis substrframe_rotz201.inp.
substrframe_rotz201_gen2.inp	Substructure generation referenced by the analysis substrframe_rotz201.inp.
substrframe_rotz201_2modes.inp	Same as substrframe_rotz201.inp with two extra dynamic modes added for each substructure.
substrframe_rotz201_2mode_gen1.inp	Substructure generation referenced by the analysis substrframe_rotz201_2modes.inp.
substrframe_rotz201_2mode_gen2.inp	Substructure generation referenced by the analysis substrframe_rotz201_2modes.inp.

Tables

Table 1: Frequencies of the first three modes, planar model (all frequencies in cycles/time).

	Mode 1	Mode 2	Mode 3
Without substructures	7.0176	7.2431	20.309
Three substructures	7.2609	11.640	33.291
Error	3.5%	61%	64%
Three substructures, with two dynamic modes	7.0173	7.2444	20.314
Error	0.004%	0.004%	0.004%

	Mode 1	Mode 2	Mode 3
Two levels of	7.2609	11.640	33.291
substructures			
Error	3.5%	61%	64%

Table 2: Frequencies of the first three modes, three-dimensional model (all frequencies in cycles/time).

	Mode 1	Mode 2	Mode 3
Without substructures	4.6277	4.6277	4.6320
Eight substructures	5.0416	5.0416	5.4384
Error	8.9%	8.9%	17.4%
Eight substructures, with two dynamic modes	4.6301	4.6301	4.6321
Error	0.05%	0.05%	0.05%

Figures

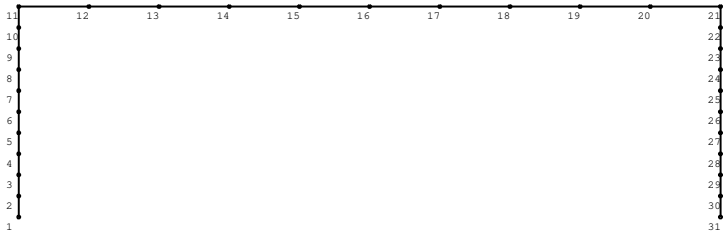


Figure 1: Frame model.

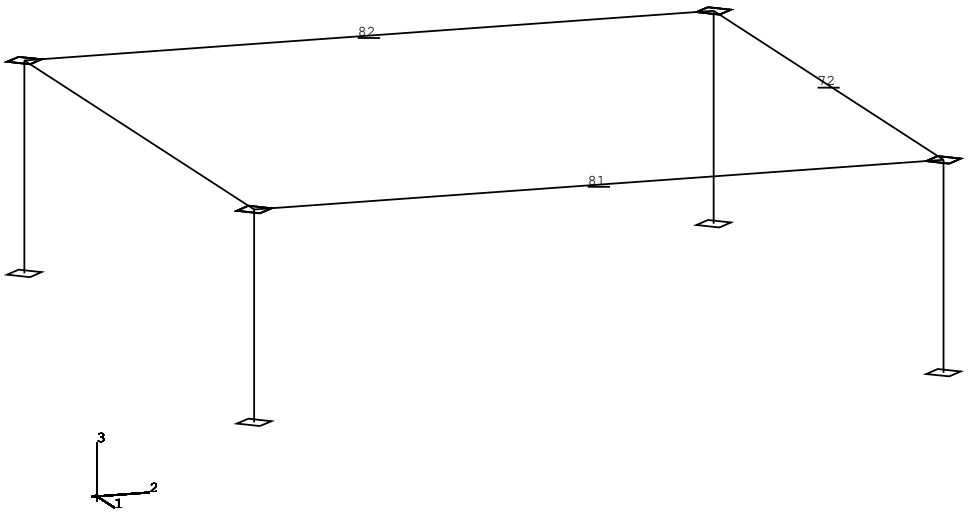


Figure 2: Three-dimensional model.

Thermomechanical diffusion of hydrogen in a bending beam

Products: Abaqus/Standard

References:

- *Mass Diffusion Analysis*
- *Mass diffusion analysis*

Overview

This example uses a two-dimensional problem to provide a simple demonstration and verification of the sequentially coupled, thermomechanical mass diffusion capability in Abaqus.

Problem description

The physical problem considered here is that of a cantilever beam subjected to thermal and mechanical loading with simple hydrogen concentration boundary conditions. Diffusion is driven by the gradients of temperature and equivalent pressure stress. In this example we are concerned with the hydrogen diffusion aspect of the problem; fictitious diffusion properties are chosen.

The problem geometry and boundary conditions are shown in *Figure 1*, and the finite element mesh is shown in *Figure 2*. The specimen is 1-mm thick, 10-mm high, and 100-mm long. The hydrogen concentration is specified at both ends of the beam at the neutral axis, and flux through all other surfaces is assumed to be zero.

The sequentially coupled mass diffusion analysis consists of a coupled temperature-displacement analysis followed by a mass diffusion analysis. Equivalent pressure stresses from the temperature-displacement analysis are written to the results file as nodal averaged values. Temperatures from the temperature-displacement analysis are stored on the results file as nodal values. Subsequently, these pressure stress and temperature fields are read in during the course of the mass diffusion analysis to provide driving mechanisms for mass diffusion.

The material properties for mass diffusion were selected to verify the sequentially coupled mass diffusion procedure and are not intended to model true properties of the material. Solubility, s, is defined as unity so that concentration, c, and normalized concentration, ϕ, are equivalent. This assumption is acceptable since no material interfaces are present. Diffusivity, D, is specified as $3.6 \times 10^{-6}\text{m}^2/\text{h}$; temperature dependence of the diffusivity is not accounted for in this example. Stress-assisted diffusion is specified by defining the pressure stress factor, κ_p, as

$$\kappa_p = 6.67 \times 10^{-9}\ \phi,$$

where ϕ is the normalized concentration. The Soret effect factor, κ_s, which allows temperature-driven mass diffusion, is defined as

$$\kappa_s = -\kappa_p \left(\theta - \theta^Z\right) \left(\frac{\partial \theta}{\partial x}\right)^{-1} \left(\frac{\partial p}{\partial x}\right)_{max},$$

where $(\theta - \theta^Z)$ is the absolute temperature, $(\partial p/\partial x)_{max}$ is the maximum gradient of equivalent pressure stress applied to the beam, and $\partial\theta/\partial x$ is the maximum temperature gradient applied to the beam. This definition of κ_s, although not realistic, allows the mass diffusion behavior to be verified easily when both equivalent pressure stress and temperature are read from the results file: when the gradients of both temperature and equivalent pressure stress are applied simultaneously, these properties indicate that mass diffusion should be driven by

concentration gradients alone. The concentration dependence of κ_p and κ_s is entered in Abaqus in tabulated form, as shown in the input listings.

The following properties are also used in the coupled temperature-displacement analysis: elastic modulus, $E = 2.0 \times 10^{11}\text{Nm}^{-2}$; Poisson's ratio, $\nu = 0.3$; and conductivity, $k = 1.0 \times 10^{3}\text{W m}^{-1}\text{K}^{-1}$.

The specimen is initially at a constant temperature of $\theta - \theta^z = 273$ K, and an initial concentration of 50 ppm is applied over the entire beam. In the first step the concentration at the free end of the beam is ramped to 100 ppm over the step and the steady-state distribution is determined. During the second step a bending moment is applied at the end of the beam to achieve a maximum tensile stress of 7.5 MPa, corresponding to a maximum equivalent pressure stress gradient of 650 MPa/m. The bending moment is applied as a ramp over 10% of the mass diffusion step and maintained until steady-state mass diffusion conditions are reached. In the third step the temperature field is applied to the mass diffusion analysis. A temperature gradient of 3.0×10^4 K/m is applied as a ramp over 10% of the mass diffusion step and maintained until steady-state mass diffusion conditions are reached.

Results and discussion

During the first step the steady-state analytical distribution of normalized concentration along the length of the beam can be obtained by direct integration of the governing equations:

$$\phi = \phi_l + (\phi_r - \phi_l)\,\frac{x}{l},$$

where $\phi_l = 50$ ppm is the concentration at the left end of the beam ($x = 0$) and $\phi_r = 100$ ppm is the concentration at the right end of the beam ($x = l$).

Based on the definition of κ_p used in this problem, the analytical solution for stress-assisted diffusion takes a form similar to that given by Liu (1970):

$$\phi = \phi_o \exp\left(-6.67 \times 10^{-9} p\right),$$

where ϕ_o is the normalized concentration obtained in the unstressed state and p is the equivalent pressure stress in the cantilever beam. For a beam in bending $\phi_o = \phi_l + (\phi_r - \phi_l)\,x/l$ represents the concentration on the neutral axis. *Figure 3* and *Figure 4* show the final distribution of equivalent pressure stress and concentration predicted by the Abaqus analysis at x=7.5 cm (ϕ_o=87.5 ppm). The finite element results show good agreement with the analytical solutions.

The Soret effect factor, κ_s, was selected to allow the applied temperature gradient to counteract the driving force imposed by the applied pressure gradient. When the concentrations reach steady state in Step 3, the solution has returned to the linear distribution obtained in Step 1. This confirms that the temperature and equivalent pressure stress fields are applied properly when read from the results file.

Input files

thermomechdiffusion_tempdisp.inp	Coupled temperature-displacement analysis that generates the temperature and equivalent pressure stress fields for the mass diffusion analysis shown in thermomechdiffusion_massdiff.inp.
thermomechdiffusion_massdiff.inp	Mass diffusion analysis using temperature and equivalent pressure stress fields read from the results file of thermomechdiffusion_tempdisp.inp.

References

- 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

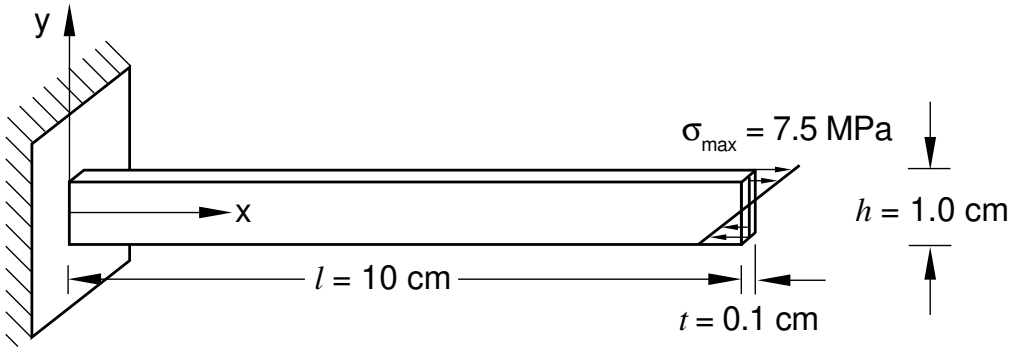


Figure 1: Cantilever beam geometry and boundary conditions.

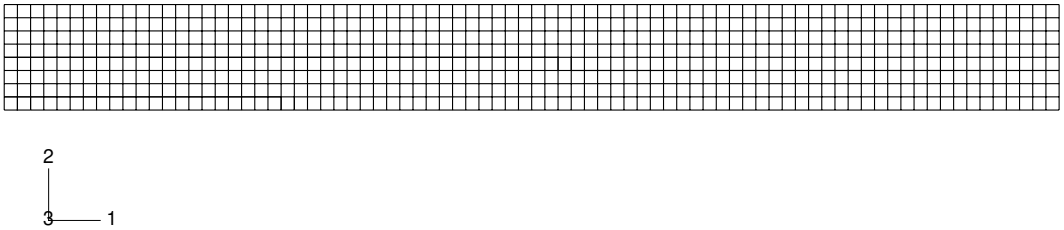


Figure 2: Finite element model of cantilever beam.

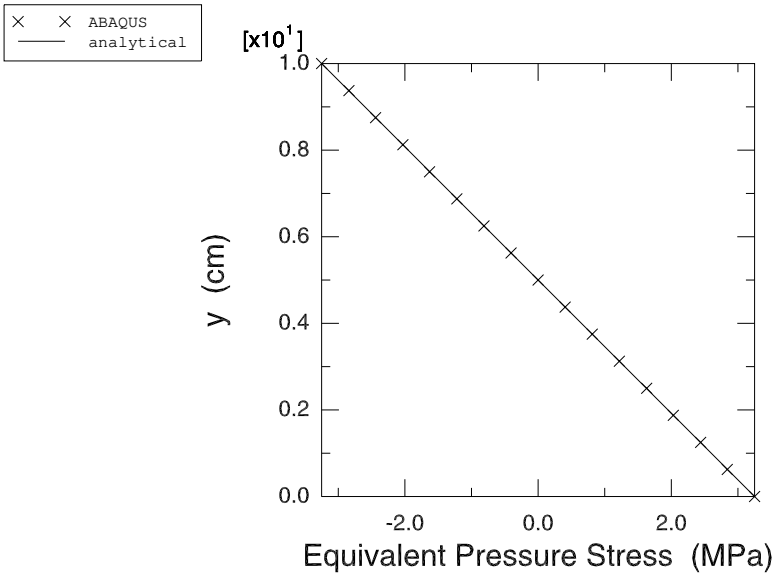


Figure 3: Equivalent pressure stress at the end of Step 2 (x=7.5 cm).

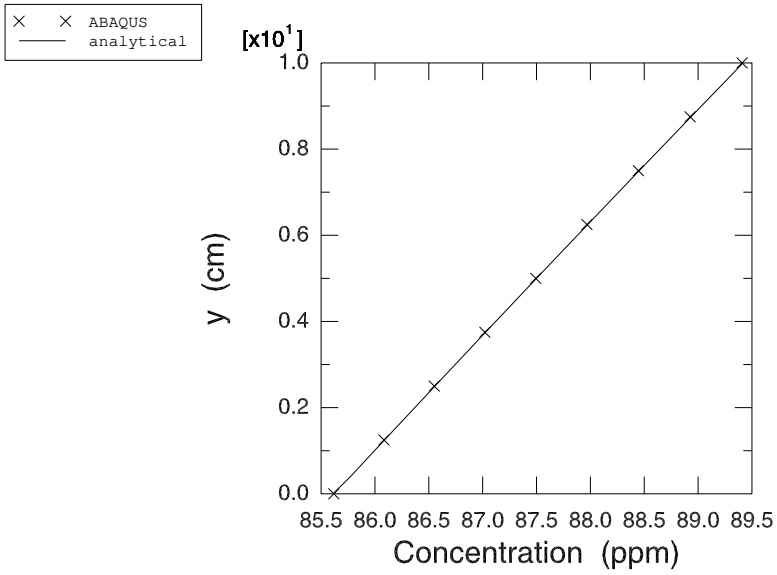


Figure 4: Steady-state concentration at the end of Step 2 (x=7.5 cm).

Acoustic analysis

In this section:

- *A simple coupled acoustic-structural analysis*
- *Analysis of a point-loaded, fluid-filled, spherical shell*
- *Acoustic radiation impedance of a sphere in breathing mode*
- *Acoustic-structural interaction in an infinite acoustic medium*
- *Acoustic-acoustic tie constraint in two dimensions*
- *Acoustic-acoustic tie constraint in three dimensions*
- *A simple steady-state dynamic acoustic analysis*
- *Acoustic analysis of a duct with mean flow*
- *Real exterior acoustic eigenanalysis*
- *Coupled exterior acoustic eigenanalysis*
- *Acoustic scattering from a rigid sphere*
- *Acoustic scattering from an elastic spherical shell*

A simple coupled acoustic-structural analysis

Products: Abaqus/Standard

References:

- *Coupled acoustic-structural medium analysis*

Overview

This example shows an elementary case of coupled acoustic-structural vibration.
This example is included because a closed form solution is easily calculated, thus providing verification of the various options for this type of analysis.

Problem description

The model is shown in *Figure 1*. No particular set of units is used in this case: all units used are assumed to be consistent. A point mass, m, of magnitude 4 is attached to a linear spring whose stiffness, k, is 1 and a dashpot that has a damping coefficient, c, of 0.08. The other ends of the spring and dashpot are fixed. The point mass is exposed to a one-dimensional acoustic medium of unit cross-sectional area and of length $l = 5$, in which the acoustic pressure is assumed to vary linearly with respect to position. We model the acoustic medium with one element of type AC1D2, which has a "lumped mass." The analytical solution is obtained on this basis. The far end of the acoustic medium is constrained to have zero acoustic pressure. The acoustic fluid has a density, ρ_f, of 0.4008 and a bulk modulus, K_f, of 10. It flows in a medium that offers volumetric drag, γ, of 0.04. The end of the acoustic medium adjacent to the structure also has an impedance boundary condition, for which $1/c_1 = 0.01$ and $1/k_1 = 0.25$. These values are chosen so that the natural frequency of the undamped mass-spring system vibrating alone is

$$\omega_s = \sqrt{\frac{k}{m}} = \frac{1}{2} \text{ radians/time (0.08 cycles/time)},$$

and the natural frequency of the acoustic medium with the impedance boundary condition, modeled with a single one-dimensional acoustic element (AC1D2), is

$$\omega_f = \sqrt{\left(\frac{1}{l} + \frac{\gamma}{c_1}\right) \Big/ \rho_f \left(\frac{l}{2\mathrm{K}_f} + \frac{1}{k_1}\right)} = 1.0 \text{ radians/time (0.16 cycles/time)}.$$

The natural frequency of the mass-spring system is confirmed by using the eigenfrequency extraction procedure in Abaqus. Since impedance boundary conditions and volumetric drag are not considered in a frequency analysis, the natural frequency of the acoustic medium cannot be confirmed. The natural frequency calculated by Abaqus for the acoustic medium alone is 0.22 cycles/time.

The damping coefficient of the dashpot is chosen to be 2% of critical damping of the mass-spring-dashpot system vibrating alone. The volumetric drag coefficient in the acoustic medium, together with the coefficient $1/c_1$ of the impedance boundary condition, provides just under 6% of critical damping for the one degree of freedom "lumped mass" model of the acoustic fluid, vibrating alone.

The equilibrium equations of this coupled system, excited by a force P applied to the point mass, can be written in terms of the displacement, u, of the point mass and the acoustic pressure, p, acting on the mass as

$$m_f\ddot{p} + c_f\dot{p} + k_f p + \ddot{u} = 0$$

and

$$m\ddot{u} + c\dot{u} + ku - p = P,$$

where for the steady-state solution we have defined

$$m_f = \frac{1}{\mathrm{K}_f}\frac{l}{2} + \frac{1}{k_1}\,, \quad c_f = \frac{1}{l}\left(\frac{\frac{\gamma}{\Omega^2}}{\rho_f^2+\frac{\gamma^2}{\Omega^2}}\right) + \frac{1}{c_1}, \text{ and } k_f = \frac{1}{l}\left(\frac{\rho_f}{\rho_f^2+\frac{\gamma^2}{\Omega^2}}\right),$$

and for the frequency analysis,

$$m_f = \frac{1}{\mathrm{K}_f}\frac{l}{2} + \frac{1}{k_1}\,, \quad c_f = \frac{\gamma}{\rho_f}\left(\frac{l}{2}\frac{1}{\mathrm{K}_f} + \frac{1}{k_1}\right) + \frac{1}{c_1}, \text{ and } k_f = \frac{1}{\rho_f}\left(\frac{1}{l} + \frac{\gamma}{c_1}\right).$$

These parameters—m_f, c_f, and k_f—are the (2, 2) entries in the AC1D2 mass, damping, and stiffness matrices, respectively. These are the only significant entries in these matrices, since the pressure at the first node is constrained to be zero.

Steady-state vibration

Steady-state vibration is caused by a harmonic loading:

$$P = \overline{P}\,\exp(i\Omega t),$$

where Ω (radians/second) is 2π times the frequency (cycles/second) of excitation. The response is also harmonic and is defined from the equilibrium equations above as

$$\begin{bmatrix} -m_f + \frac{1}{\Omega^2}k_f & -\frac{1}{\Omega}c_f & 1 & 0 \\ -\frac{1}{\Omega}c_f & m_f - \frac{1}{\Omega^2}k_f & 0 & -1 \\ 1 & 0 & k-\Omega^2 m & -\Omega c \\ 0 & -1 & -\Omega c & -\left(k-\Omega^2 m\right) \end{bmatrix} \begin{Bmatrix} \Re(p) \\ \Im(p) \\ \Re(u) \\ \Im(u) \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \\ \overline{P} \\ 0 \end{Bmatrix},$$

where $\Re(p)$ is the real part of the pressure amplitude, $\Im(p)$ is the imaginary part of the pressure amplitude, and $\Re(u)$ and $\Im(u)$ are the real and imaginary components of the displacement amplitude.

We first study the system by uncoupling the fluid from the structural elements and excite the fluid harmonically by specifying an inward volume acceleration with an amplitude, $\overline{P}_p$, of 1:

$$\begin{bmatrix} -m_f + \frac{1}{\Omega^2}k_f & -\frac{1}{\Omega}c_f & 0 & 0 \\ -\frac{1}{\Omega}c_f & m_f - \frac{1}{\Omega^2}k_f & 0 & 0 \\ 0 & 0 & k-\Omega^2 m & -\Omega c \\ 0 & 0 & -\Omega c & -\left(k-\Omega^2 m\right) \end{bmatrix} \begin{Bmatrix} \Re(p) \\ \Im(p) \\ \Re(u) \\ \Im(u) \end{Bmatrix} = \begin{Bmatrix} \frac{1}{\Omega^2}\overline{P}_p \\ 0 \\ \overline{P} \\ 0 \end{Bmatrix}.$$

Results and discussion

The uncoupled system response is obtained by conducting a frequency sweep using the direct-solution and subspace-based steady-state dynamic procedures. Since the undamped individual systems resonate at about 0.08 and 0.16 cycles/time, we request a frequency sweep over the range 0.01 to 0.3 cycles/time, using a linear frequency scale with solution points at intervals of 0.01 cycles/time throughout this range. *Figure 2* shows the amplification

of the pressure p and the displacement u throughout this frequency range. As would be expected, the displacement shows significantly higher amplification around resonance because that system has less damping.

The fully coupled system is investigated with the same frequency sweep. Three steady-state dynamic analyses are performed: one with the subspace-based method based on the coupled acoustic-structural modes, one with the direct-solution method, and a final step with the subspace-based method but based on the uncoupled modes. The system is excited by the mechanical loading, P, only. The response of the fully coupled system is shown in *Figure 3*. The resonances are now separated compared to the uncoupled systems: the lower resonance occurs at about 0.06 cycles/time (compared to the natural frequency of 0.08 cycles/time for the uncoupled structural system), while the higher resonance is at 0.2 cycles/time (compared to the natural frequency of 0.16 cycles/time for the uncoupled acoustic element). The results of all three steady-state dynamic procedures are coincident.

The system is also studied without volumetric drag in the fluid (by choosing $\gamma = 0$). The response is shown in *Figure 4*. The lower resonance has about the same peak amplitude as the system with drag, but the higher resonance now has a peak amplitude about three times that of the system with fluid drag. The displacement amplification is almost zero at the frequency corresponding to the resonance of the uncoupled fluid system (0.16 cycles/time). From the equations given above it is clear that, at this frequency, with $\gamma = 0$ the displacement amplitude is $1/(c_1\Omega)$ times the pressure amplitude. Since $1/c_1$ is fairly low ($1/c_1 = 0.01$), the displacement has a very small value at this frequency.

Input files

acousticstructvibration.inp	Fully coupled case.
acousticstructvibration_uncoup.inp	Uncoupled case.
acousticstructvibration_nodrag.inp	Fully coupled case with no volumetric drag ($\gamma = 0$).
acousticstructvibration_depend.inp	Uncoupled case with temperature and field variable dependence of the density and the acoustic medium properties, as well as frequency dependence of the spring and dashpot properties.

Figures

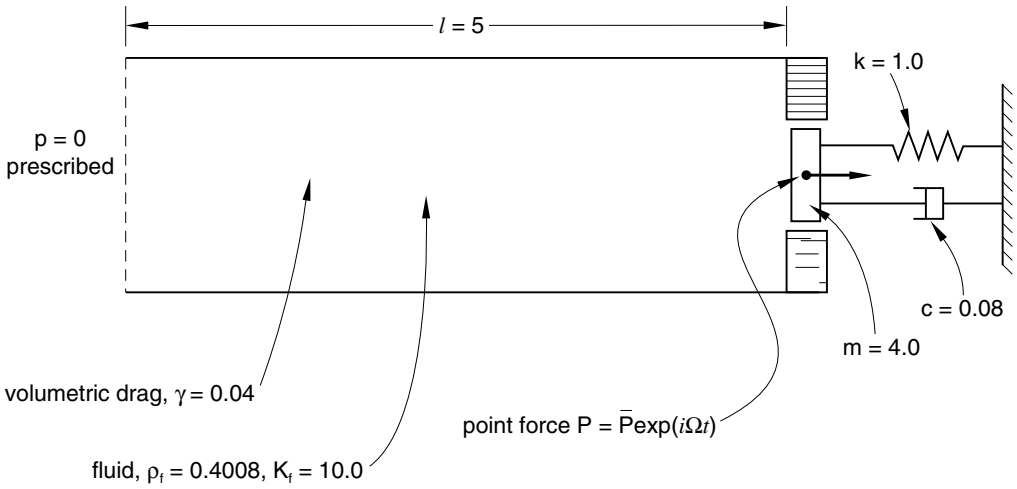


Figure 1: Coupled acoustic-structural vibration model.

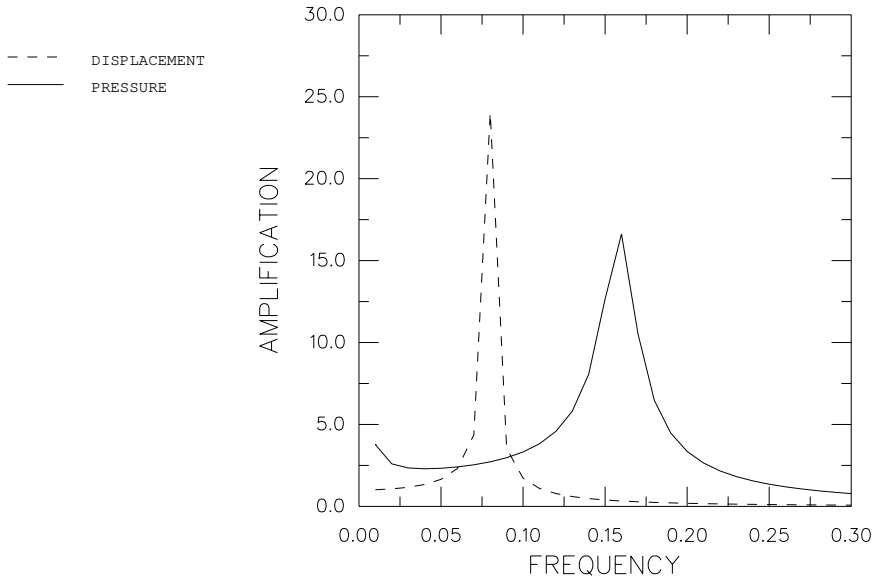


Figure 2: Steady-state response of the uncoupled acoustic-structural system.

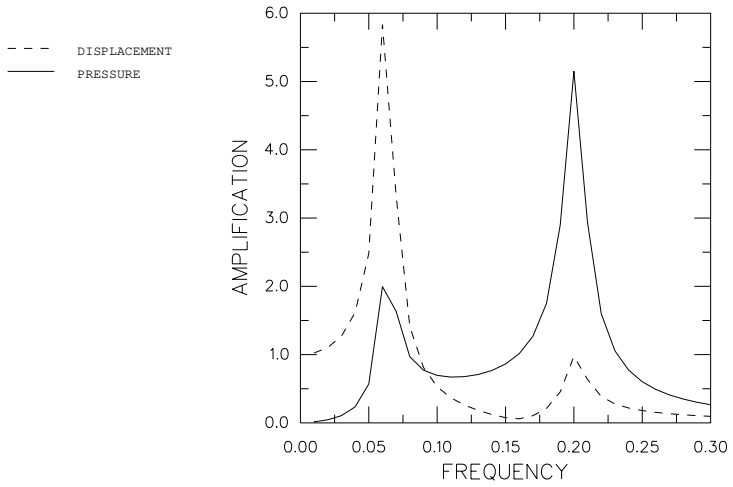


Figure 3: Steady-state response of the coupled acoustic-structural system.

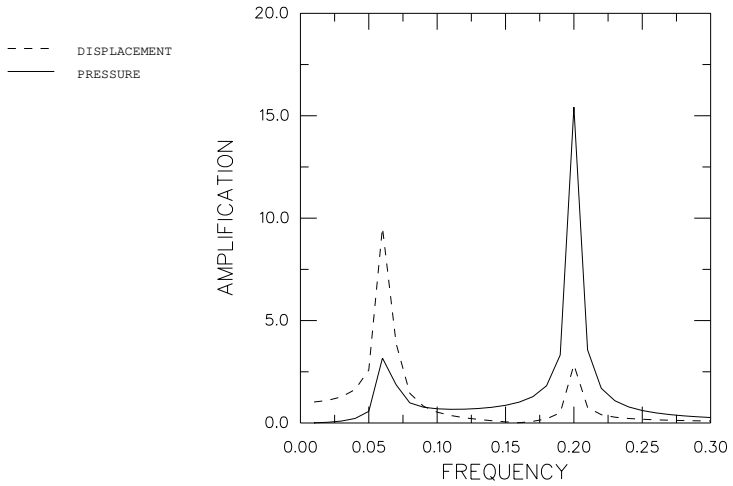


Figure 4: Steady-state response of the coupled acoustic-structural system without volumetric drag.

Analysis of a point-loaded, fluid-filled, spherical shell

Products: Abaqus/Standard

References:

- *Coupled acoustic-structural medium analysis*
- *Coupled Acoustic-Structural Analysis*

Overview

This example shows the steady-state vibration of a point-loaded spherical shell coupled to an acoustic fluid that fills its interior.
The problem in this example is modeled using axisymmetric shell and acoustic elements. The closed form solution of Stepanishen and Cox (2000) is used for validation of the analysis.

Problem description

The model is a semicircular shell and fluid mesh of radius 2.286 m. A point load on the symmetry axis of magnitude 1.0 N is applied to the shell. The shells are 0.0254 m in thickness and have a Young's modulus of 206.8 GPa, a Poisson's ratio of 0.3, and a mass density of 7800.0 kg/m^3. The acoustic fluid has a density, $\boldsymbol{\rho_f}$, of 1000 kg/m^3 and a bulk modulus, $\mathbf{K_f}$, of 2.25 GPa. The response of the coupled system is calculated for frequencies ranging from 100 to 1000 Hz in 5 Hz increments. There are two different finite element meshes used: one with explicitly defined acoustic-structural interaction elements and one that uses a tie constraint. The former model consists of 220 SAX1 elements surrounding a mesh of 15848 ACAX4 elements. Coupling is effected using 220 ASI2A elements. The latter model uses 80 SAX2 elements surrounding a mesh of 965 ACAX8 elements. For this mesh, coupling is effected using a tie constraint to generate the acoustic-structural interaction elements internally.

A dummy part is included in the models to ensure that the analytical solution appears in the output database. This part consists of a single point mass, uncoupled from the model described above, with a displacement boundary condition on degree of freedom 1. This imposed displacement uses an amplitude table consisting of the Stepanishen/Cox analytical solution for the drive point admittance.

Results and discussion

The response is obtained by conducting a frequency sweep, using the direct-solution steady-state and the mode-based steady-state dynamic procedures. A frequency sweep over the range 100.0 to 1000.0 cycles per second, using a linear frequency scale with solution points at intervals of 5 cycles per second throughout this range, is requested.

The finite element solutions for drive point admittance, the ratio of the velocity to the applied load, are shown in *Figure 1*. This plot was created in the Visualization module of Abaqus/CAE by multiplying the steady-state displacement by the frequency and taking the logarithm (base 10) of the data. Comparison to the analytical results of Stepanishen and Cox, as shown, is quite good throughout the frequency range. The analytical data appear as the displacement of degree of freedom 1 for the dummy mass element in the output database. The results obtained in the direct-solution steady-state dynamic analysis are nearly identical to the results obtained in the mode-based steady-state dynamic analysis although only 50 eigenmodes have been retained.

The natural frequencies of the coupled fluid-filled sphere are extracted in a frequency step. These frequencies correspond well with frequencies for the peak amplitudes for acoustic pressure and drive point admittance, as

seen in *Figure 1*. The frequency results in the figure correspond to the case using the tie constraint; the results for the other case are very similar.

Figure 2 shows a sample contour plot of acoustic pressure at 980 Hz.

Input files

acousticfilledsphere.inp	Coupled analysis using acoustic structural interaction elements.

References

- Stepanishen, P., and D. L. Cox, "Structural-Acoustic Analysis of an Internally Fluid-Loaded Spherical Shell: Comparison of Analytical and Finite Element Modeling Results," *NUWC Technical Memorandum 00-118, Newport, Rhode Island*, 2000.

Figures

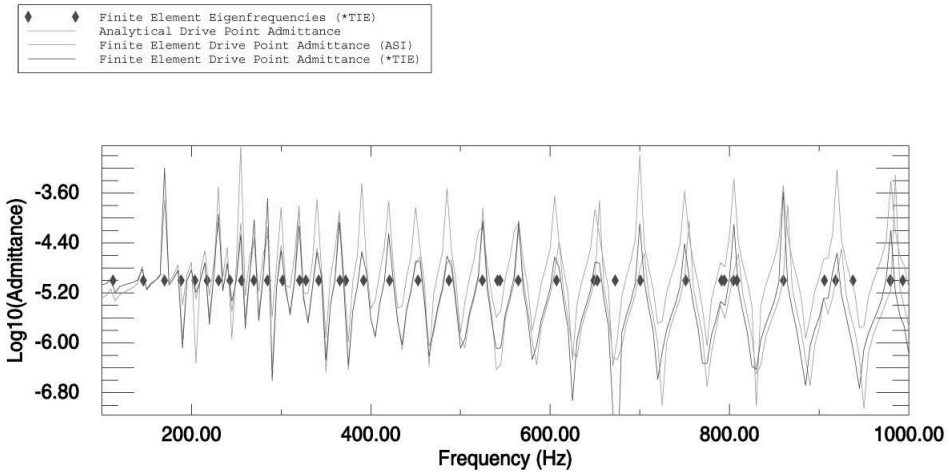


Figure 1: Coupled acoustic-structural vibration model.

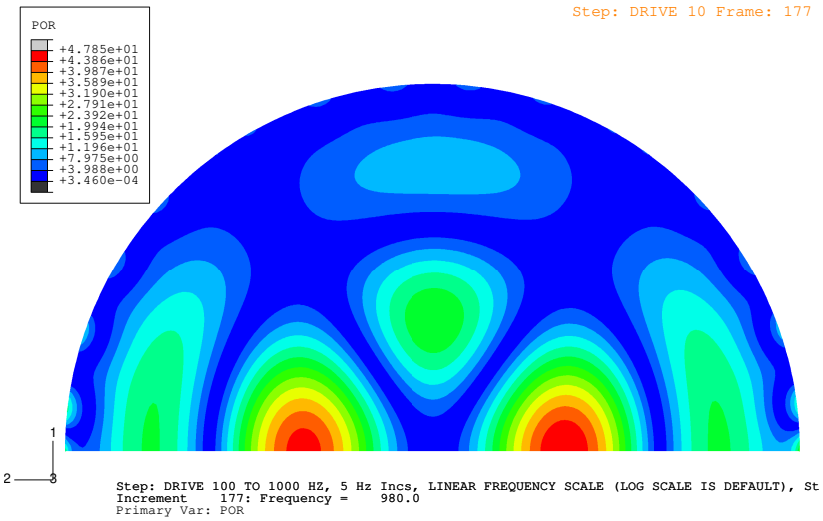


Figure 2: Steady-state pressure magnitudes for the acoustic-structural system at 980 Hz.

Acoustic radiation impedance of a sphere in breathing mode

Products: Abaqus/Standard Abaqus/Explicit

Overview

The example illustrates the use of a simple absorbing boundary condition in conjunction with acoustic continuum and interface elements and acoustic infinite elements.
In this example we calculate the acoustic radiation from a sphere in the "breathing" mode; that is, when the motion consists of uniform radial velocity. The results are compared with classical results.

Problem description

A spherical cavity of unit radius in an unbounded acoustic medium is subjected to a uniform radial velocity on its inner surface. The analytical solution for the acoustic pressure is of the form

$$p(r) = -\rho_f c_f v_n \frac{e^{-ik(r-1)}}{r},$$

where p is the acoustic pressure, ρ_f is the fluid density, $c_f = \sqrt{K_f/\rho_f}$ is the speed of sound, K_f is the bulk modulus, $k = 2\pi f/c_f$ is the acoustic wave number, f is the frequency, r is the radial coordinate, $\mathbf{n}$ is the normal unit vector pointing into the fluid, and $v_n = -\dot{\mathbf{u}}^f \cdot \mathbf{n}$ is the prescribed inward particle velocity on the spherical boundary. The ratio of the pressure to the velocity on the boundary is called the acoustic impedance (see Junger and Feit, 1972); for the zeroth (breathing) mode the impedance is given by

$$Z_0 = \rho_f c_f v_n \frac{ik}{ik+1}.$$

The imaginary part of the impedance is given by

$$\mathfrak{I}(Z_0) = \frac{|Z_0|}{\sqrt{1+k^2}},$$

where

$$|Z_0| = \frac{\rho_f c_f v_n k}{\sqrt{1+k^2}}$$

is the magnitude of the impedance.

The absorbing boundary condition used here is the first-order condition of Bayliss, Gunzberger, and Turkel (1982):

$$\nabla p \cdot \mathbf{n} = \left(\frac{1}{r_1} + ik\right) p,$$

where r_1 is the radius of the spherical truncation boundary of the finite element mesh. This boundary condition is theoretically exact for all vibration frequencies in the breathing mode. Since conservation of linear momentum requires

$$\nabla p = -\rho_f \ddot{\mathbf{u}}^f$$

and the impedance enforces the condition

$$-\dot{\mathbf{u}}^f \cdot \mathbf{n} = \frac{1}{c_1} p + \frac{1}{k_1} \dot{p},$$

the first-order Bayliss et al. boundary condition can be enforced in the steady-state dynamic procedure by specifying impedance parameters given by

$$\frac{1}{k_1} = \frac{-1}{r_1 \rho_f (2\pi f)^2},$$

and

$$\frac{1}{c_1} = \frac{1}{\rho_f c_f}.$$

In Abaqus this spherical radiation impedance can also be defined by specifying the radius of the terminating spherical mesh boundary.

Figure 1 shows the finite element mesh using a single layer of 15 ACAX8 elements, with ASI3A coupling elements on the inner surface and the impedance applied to the outer faces of the ACAX8 elements. Uniform radial velocities are applied to the inner surface transforming the spherical coordinates to a local coordinate system before specifying the magnitudes. The dimensions and acoustic properties of this problem are chosen to facilitate comparison with the analytical value of the impedance, Z_0 With $\rho_f = 1$ and $K_f = 4\pi^2$, $c_f = 2\pi$, and the acoustic wave number $k = 2\pi f / c_f$ is equal to the analysis frequency in Hertz, f. Moreover, if the imposed normal velocity $v_n = 1/2\pi$, the value of the pressure variable on the inner boundary will equal the value for the impedance, Z_0. This normal velocity is imposed using a boundary condition in conjunction with a transformed coordinate system, as shown in *acousticimpsphere_acax8_bayliss.inp*. Since the degrees of freedom representing the tangential velocity components have no stiffness associated with them, they must be constrained to prevent solver problems.

The outer boundary is placed at $r_1 = 1 + 1/2\pi$ to provide a good aspect ratio for the elements; its value has no other significance for this problem. For comparison, the same mesh was reanalyzed using the default plane wave absorbing condition. This plane wave absorber is equal to the limiting value of the Bayliss et al. first-order condition as $r_1 \to \infty$ and is theoretically exact for absorption of planar wavefronts normally incident on a planar truncation boundary.

A steady-state dynamic analysis is performed in Abaqus/Standard over a range of frequencies from 0 to 20 Hz. The transient simulations are also performed in Abaqus/Explicit with an excitation frequency of 1 Hz. Different excitation frequencies can be tested by changing the parameters defined in the input files.

The steady-state analysis was also performed in Abaqus/Standard using three-dimensional and axisymmetric acoustic infinite elements. These acoustic infinite elements use infinite-direction basis functions that are based on radiating modes of a sphere (see *Acoustic infinite elements*). Consequently, the impedance for the spherical breathing mode is modeled accurately by these elements, and the results obtained using acoustic infinite elements should closely match the results obtained using spherical radiation impedance conditions. However, for more complex wave shapes the acoustic infinite elements are expected to give more accurate results, owing to the richness of the basis functions used to model the variation of the acoustic field in the infinite direction. The infinite element mesh required to model this problem consists only of acoustic infinite elements on the spherical surface, with acoustic-structural interface elements added (with the same mesh topology) to facilitate imposition of the velocity boundary condition at the surface of the sphere. The reference point for the infinite elements is located at the center of the sphere. For this problem a single input file, *acousticimpsphere_acin.inp*, includes sections of the sphere modeled with each of the three-dimensional acoustic infinite elements: ACIN3D3, ACIN3D4, ACIN3D6, and ACIN3D8. Two additional input files, *acousticimpsphere_acinax2.inp* and *acousticimpsphere_acinax3.inp*, model the problem using semicircular meshes of ACINAX2 and ACINAX3 elements, respectively.

For the transient analyses in Abaqus/Explicit the Bayliss et al. boundary condition is enforced using spherical radiation impedance conditions. On the inner radius of the mesh the shell elements in the three-dimensional case

and beam elements in the axisymmetric case are connected to the acoustic domain with a tie constraint. The transient analysis is also performed in Abaqus/Explicit using three-dimensional and axisymmetric acoustic infinite elements instead of defining the impedance. Two additional input files, *acousticimpsphere_acin3d4_xpl.inp* and *acousticimpsphere_acinax2_xpl.inp*, model the three-dimensional and axisymmetric problems, respectively.

Results and discussion

For the steady-state dynamic analysis the finite element results for the Bayliss et al. and plane wave absorbing boundary conditions are shown in *Table 1*, where they are compared with the analytical values. The ratios of the finite element results to the analytical solutions for the pressure magnitude (output variable POR) and the imaginary part of pressure are presented. In the table the definition of nodes per wavelength, N, is

$$\mathbf{N} = \frac{8\pi^2}{f},$$

which is appropriate for quadratic elements of length $\mathbf{1/2\pi}$. The Bayliss et al. results agree with the analytical solution, except for the highest frequencies, where errors are attributable to the small number of nodes per wavelength in the radial direction. In contrast, the numerical results for the model with the plane wave absorbing boundary condition are not as accurate. Because the absorbing boundary has been placed extremely close to the radiating surface, the assumption that wavefronts incident on the truncation boundary are planar is invalid.

A graph of the pressure magnitude (output variable POR) as a function of frequency is shown in *Figure 2* for the Bayliss et al. boundary condition.

The results for the acoustic infinite elements also agree with the analytical solution.

The results from the Abaqus/Explicit transient tests with the impedance defined agree with those obtained by Abaqus/Standard. For the axisymmetric case, the pressure variation in time at a sample location on the outer boundary is shown in *Figure 3* (for a clear comparison the Abaqus/Standard analysis is also performed as a transient simulation). The results obtained with acoustic infinite elements agree closely with the results obtained using impedance, as shown in the figure. The figure shows that the peak pressure magnitude for the transient analyses is about .60. This value is quite close to the pressure magnitude predicted by the steady-state analysis for an excitation frequency of 1 Hz, as can be seen from *Figure 2*.

Input files

Abaqus/Standard input files

acousticimpsphere_acax8_bayliss.inp	Model that uses ACAX8 and ASI3A elements with the Bayliss et al. boundary condition.
acousticimpsphere_acax8_auto.inp	Model that uses ACAX8 and ASI3A elements with the Bayliss et al. boundary condition, defined using spherical radiation conditions.
acousticimpsphere_acax8_planew.inp	ACAX8 elements with the plane wave boundary condition.
acousticimpsphere_acax6_bayliss.inp	Same problem as acousticimpsphere_acax8_bayliss.inp but with ACAX6 elements.
acousticimpsphere_ac3d15.inp	Version of the spherical model in three dimensions using AC3D15 elements.
acousticimpsphere_ac3d20.inp	Same problem as acousticimpsphere_ac3d15.inp but with AC3D20 elements.
acousticimpsphere_ac3d10.inp	Same problem as acousticimpsphere_ac3d20.inp but with AC3D10 elements.

acousticimpsphere_acin.inp	Same problem as acousticimpsphere_ac3d20.inp but with spherical segments modeled with ACIN3D3, ACIN3D4, ACIN3D6, and ACIN3D8 elements.
acousticimpsphere_acinax2.inp	Same problem as acousticimpsphere_acax.inp, but with spherical segments modeled with ACINAX2 elements.
acousticimpsphere_acinax3.inp	Same problem as acousticimpsphere_acax.inp, but with spherical segments modeled with ACINAX3 elements.

Abaqus/Explicit input files

acousticimpsphere_acax4r_xpl.inp	Model that uses ACAX4R elements with the Bayliss et al. boundary condition.
acousticimpsphere_ac3d8r_xpl.inp	Version of the spherical model in three dimensions using AC3D8R elements.
acousticimpsphere_acin3d4_xpl.inp	Model that uses ACAX4R and ACINAX2 elements.
acousticimpsphere_acinax2_xpl.inp	Version of the spherical model in three dimensions using AC3D8R and ACIN3D4 elements.

References

- Bayliss, A., M. Gunzberger, and E. Turkel, "Boundary Conditions for the Numerical Solution of Elliptic Equations in Exterior Regions," *SIAM Journal of Applied Mathematics*, vol. 42, no. 2, pp. 430–451, 1982.
- Junger, M., and D. Feit, Sound, Structures, and Their Interaction, The MIT Press, 1972.

Tables

Table 1: Finite element results.

f	N	BGT POR Ratio	PWA POR Ratio	BGT Im Ratio	PWA Im Ratio
0.05000	1579.137	0.99999	14.90379	0.99999	0.03438
0.10000	789.568	0.99999	7.48000	0.99999	0.03477
0.20000	394.784	0.99999	3.79570	0.99999	0.03574
0.30000	263.189	0.99999	2.59122	0.99999	0.03744
0.40000	197.392	0.99999	2.00555	0.99999	0.03985
0.50000	157.914	0.99999	1.66626	0.99999	0.04292
0.60000	131.595	0.99999	1.44914	0.99999	0.04666
0.70000	112.795	0.99999	1.30096	0.99999	0.05110
0.80000	98.696	0.99999	1.19515	0.99999	0.05619
0.90000	87.730	0.99999	1.11699	0.99998	0.06199
1.00000	78.957	0.99999	1.05772	0.99998	0.06843
1.30000	60.736	0.99999	0.94678	0.99998	0.09174
1.60000	49.348	0.99999	0.88868	0.99998	0.12096
1.90000	41.556	0.99999	0.85587	0.99998	0.15591
2.00000	39.478	0.99999	0.84834	0.99999	0.16879
3.00000	26.319	0.99997	0.81832	0.99998	0.32989

f	N	BGT POR Ratio	PWA POR Ratio	BGT Im Ratio	PWA Im Ratio
5.00000	15.791	0.99963	0.84443	0.99989	0.79138
7.00000	11.280	0.99799	0.90195	1.00078	1.32394
10.00000	7.896	0.98983	0.98764	1.04189	1.87622
15.00000	5.264	0.96791	1.02461	1.86250	2.01852
20.00000	3.948	1.03001	1.03899	5.28096	4.49785

Figures

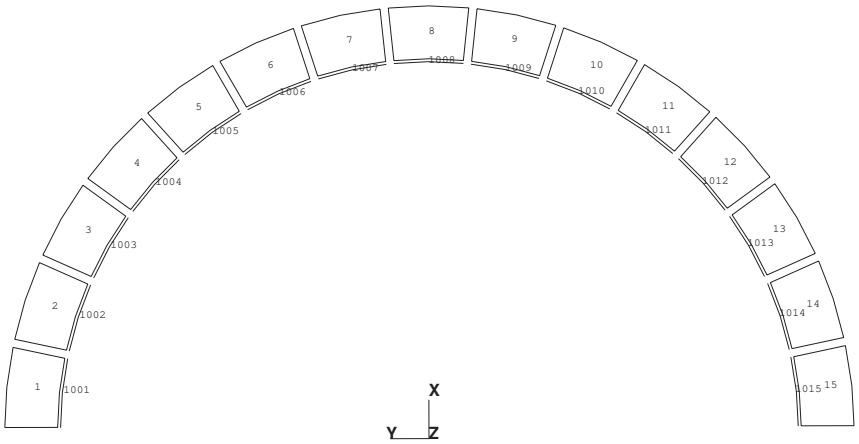


Figure 1: Acoustic model finite element mesh used in Abaqus/Standard.

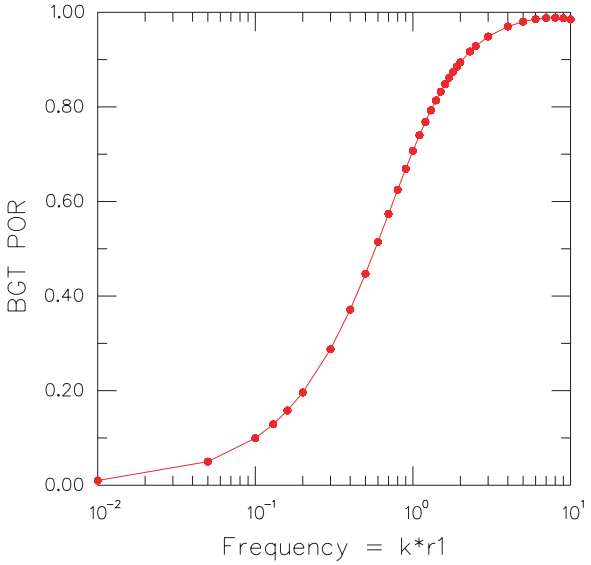


Figure 2: Pressure magnitude (POR) versus frequency.

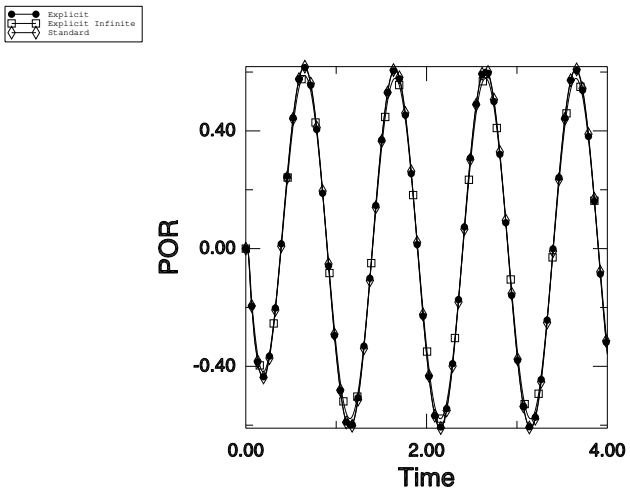


Figure 3: Pressure variation on the outer boundary for the transient analysis.

Acoustic-structural interaction in an infinite acoustic medium

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example is intended to illustrate and verify the use of simple absorbing boundaries and acoustic infinite elements in a coupled exterior acoustic-structural system.
Problems involving infinite regions of acoustic media, solved using absorbing boundary conditions and infinite elements, are compared to converged solutions.

Problem description

This benchmark problem is a simplified version of a characteristic problem in exterior acoustics: a thin shell structure immersed in a fluid. Since the intention here is only to test the effectiveness of the boundary conditions, the structure's purpose in this model is to introduce nontrivial dynamics in the acoustic mesh. In particular, the field incident on the radiating boundary should include significant spatial variation for the test to be meaningful.

The absorbing boundary conditions intended for three-dimensional applications are tested here in axisymmetric meshes. The two-dimensional elliptical and circular type radiation conditions are also applicable in three-dimensional analyses with radiating boundaries in the form of right-elliptic or right-circular cylinders.

Figure 1 shows the test mesh used for the circular and spherical boundary condition tests. On the inner radius of the mesh there is a circular structure of 4 units in radius consisting of shell elements in the axisymmetric case and beam elements in the two-dimensional case. The shells are connected to the acoustic domain with a tie constraint. The structure has stiffeners to disrupt the symmetry of the solution and to produce a nontrivial spatial variation in the pressure field incident on the absorbing boundary. Surrounding the structure is a fluid modeled with acoustic elements and terminated at the outer edge by the absorbing boundary conditions. The mesh shown uses an outer radius of eight units; it is stretched along the Y-axis by a factor of 1.2 for the elliptical and prolate spheroidal boundary condition tests.

Much larger meshes, 28 units in radius, are also run in both the axisymmetric and two-dimensional cases. The results from these meshes provide reference solutions against which to compare the results from the test meshes. Both steady-state and transient dynamic conditions are considered. The transient analyses are also performed using Abaqus/Explicit.

For tests of the acoustic infinite elements, the acoustic infinite elements are coupled directly to the structural model using a tie constraint. The center of the shell is used as the reference point for the acoustic infinite elements (see *Coupled Acoustic-Structural Analysis*). While in this case the acoustic infinite elements are coupled directly to the structural model, an alternative modeling strategy would involve defining an intermediate region comprising acoustic elements between the structure and the acoustic infinite elements. In certain situations the second approach may lead to improved accuracy.

The acoustic medium has a bulk modulus of 2.25E9 Pa and a density of 1000 (the properties of water). For the steady-state analyses the acoustic material has no volumetric drag. For the transient analyses a volumetric drag parameter is introduced with a numerical value equal to approximately 10% of the speed of sound. This value is not negligible in the sense of the approximation made in the derivation of this boundary condition (see *Coupled acoustic-structural medium analysis*); therefore, this case is a good test of the formulation.

Loading

The structure is excited in all cases with a concentrated load applied to degree of freedom 2 on the node at the free (innermost) end of the shell stiffener.

The absorbing boundary tests use nonreflecting boundary conditions to generate the necessary data automatically at the absorbing boundary. For the two-dimensional circular case and reference solutions and the spherical (axisymmetric) case and reference solutions, only the radius of the terminating circle or sphere need be specified to impose corresponding nonreflecting boundary conditions. In the two-dimensional elliptic case and the three-dimensional prolate spheroidal case, additional data need to be included to specify the size and orientation of the ellipse or spheroid.

For the Abaqus/Standard steady-state analyses using infinite elements, the infinite elements model radiation damping as well as the added mass effect, when coupled to the structural elements.

Results and discussion

The steady-state analyses are performed using the direct-solution steady-state dynamic procedure. The analyses are performed for two frequencies: the frequency corresponding to $ka = 1$ to test the effectiveness of the radiation boundary conditions, and the frequency corresponding to $ka = 7$ (in two dimensions) or $ka = 40$ (in three spatial dimensions) to test the limits of the acoustic mesh discretization (less challenging frequencies for the boundary conditions). Two steps are used. The impedance boundary condition is applied in the first step (for both frequencies). In the second step these impedances are removed and surfaces impedances of the same value are applied. The results with both options are identical.

Each analysis shows good agreement between the reference solution and the solutions using the absorbing boundary conditions or the acoustic infinite elements on the smaller test meshes. The lower frequency is a more challenging test for the boundary conditions, particularly in two dimensions, where the boundary conditions are only asymptotic. *Figure 2* through *Figure 5* show the pressure amplitudes on the surface of the shell as a function of angular position. The angle is measured from the *Y*-axis, as shown in *Figure 1*. Each figure shows three curves; where differences between them are visible, the circular and elliptical condition solutions overlay the reference solution, and the spherical and prolate spheroidal conditions deviate by small amounts. The results from the acoustic infinite elements are similar.

The transient analyses are performed with an excitation frequency of 100 Hz using two steps and a fixed time increment of 2.5×10^{-5} units, approximately one two-hundredth of the wavespeed. The analyses are run for 0.003 time units in the first step, where the impedance boundary condition is applied. This time period is long enough for the wave to reach the boundary of the test meshes. In the second step the simulation runs for another 0.003 units, this time with the boundary condition applied using a surface impedance. The total simulation time, 0.006 units, is not long enough for the wavefronts emanating from the structure to reach the boundary of the reference meshes, so the impedance conditions there do not play a role in the simulation. *Figure 6* through *Figure 9* show the pressure amplitudes in the acoustic domain at selected times. In each case the reference solution is on the left. Very good agreement is achieved in all cases; the three-dimensional boundary conditions perform slightly better, for reasons discussed in *Coupled acoustic-structural medium analysis*.

The Abaqus/Explicit transient analyses performed using acoustic infinite elements give very similar results to the Abaqus/Explicit transient analyses performed using nonreflecting boundary conditions. The test using two-dimensional acoustic infinite elements gives very similar results to the test using the circular radiation condition, while the test using axisymmetric acoustic infinite elements gives very similar results to the test using the spherical radiation condition. *Figure 10* shows the pressure amplitudes on the surface of the shell at an angle of 90° to the vertical for the two-dimensional analysis (for a clear comparison the Abaqus/Standard transient analysis results are also included). The pressures are seen to match closely. The tests with acoustic infinite elements do not include any acoustic elements to model the fluid surrounding the structure. However, the additional computational cost due to the acoustic infinite elements offsets any savings obtained from not including the acoustic elements. For this example it was found that the acoustic infinite element analyses were about 12% more expensive than the corresponding analyses without acoustic infinite elements; i.e., with acoustic elements and nonreflecting boundary conditions.

Input files

Abaqus/Standard input files

acoustic_bc_2dref.inp	Circular boundary condition, two-dimensional reference solution, steady state.
acoustic_bc_acin2d2.inp	Linear, two-dimensional acoustic infinite elements, steady state.
acoustic_bc_acin2d3.inp	Quadratic, two-dimensional acoustic infinite elements, steady state.
acoustic_bc_acinax2.inp	Linear, axisymmetric acoustic infinite elements, steady state.
acoustic_bc_acinax3.inp	Quadratic, axisymmetric acoustic infinite elements, steady state.
acoustic_bc_circular.inp	Circular boundary condition, test mesh, steady state.
acoustic_bc_circular_ams.inp	Circular boundary condition, test mesh, steady-state dynamic analysis following uncoupled AMS eigensolution.
acoustic_bc_circular_ams_restart.inp	Steady-state dynamic analysis restarting from uncoupled AMS eigensolution.
acoustic_bc_ellipse.inp	Elliptical boundary condition, test mesh, steady state.
acoustic_bc_3dref.inp	Spherical boundary condition, three-dimensional reference solution, steady state.
acoustic_bc_spherical.inp	Spherical boundary condition, test mesh, steady state.
acoustic_bc_prolate.inp	Prolate spheroidal boundary condition, test mesh, steady state.
acoustic_bc_2dref_trans.inp	Circular boundary condition, two-dimensional reference solution, transient.
acoustic_bc_circular_trans.inp	Circular boundary condition, test mesh, transient.
acoustic_bc_ellipse_trans.inp	Elliptical boundary condition, test mesh, transient.
acoustic_bc_3dref_trans.inp	Spherical boundary condition, three-dimensional reference solution, transient.
acoustic_bc_spherical_trans.inp	Spherical boundary condition, test mesh, transient.
acoustic_bc_prolate_trans.inp	Prolate spheroidal boundary condition, test mesh, transient.

Abaqus/Explicit input files

acoustic_bc_circular_xpl.inp	Circular boundary condition, test mesh.
acoustic_bc_circular_xpl_subcyc.inp	Circular boundary condition, test mesh for the sole purpose of testing the performance of subcycling.
acoustic_bc_ellipse_xpl.inp	Elliptical boundary condition, test mesh.
acoustic_bc_spherical_xpl.inp	Spherical boundary condition, test mesh.
acoustic_bc_prolate_xpl.inp	Prolate spheroidal boundary condition, test mesh.
acoustic_bc_acin2d2_xpl.inp	Two-dimensional acoustic infinite elements, test mesh.
acoustic_bc_acinax2_xpl.inp	Axisymmetric acoustic infinite elements, test mesh.

Figures

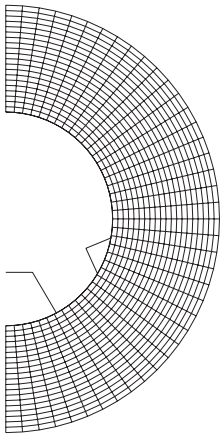


Figure 1: Mesh configuration.

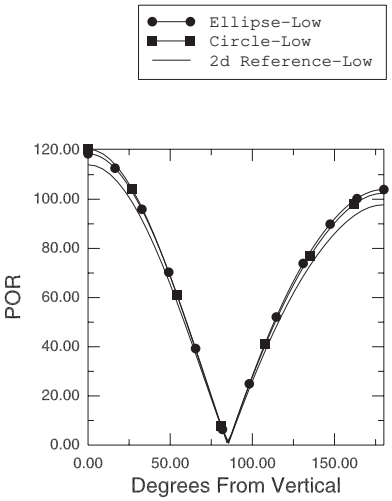


Figure 2: Pressure amplitude comparisons at $ka = 1$, two dimensions.

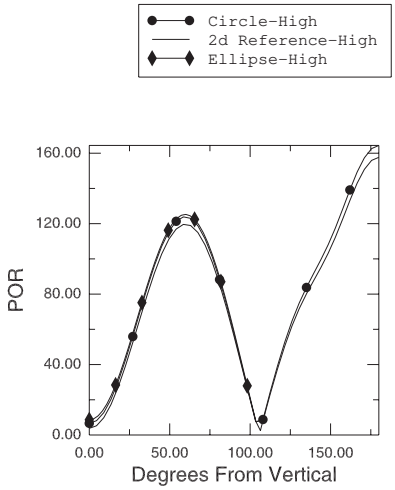


Figure 3: Pressure amplitude comparisons at $ka = 7$, two dimensions.

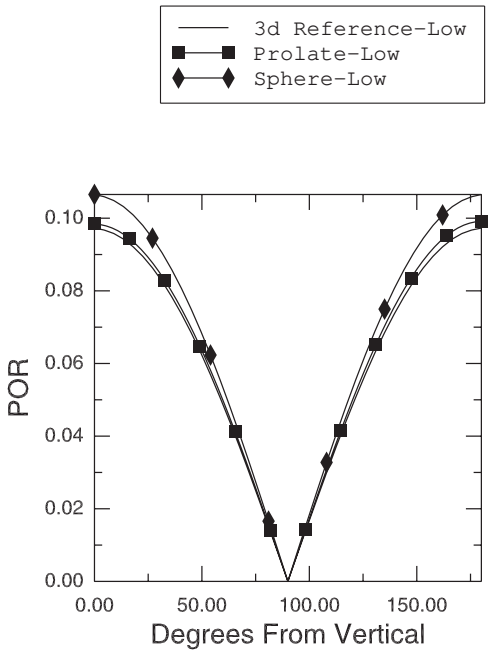


Figure 4: Pressure amplitude comparisons at $ka = 1$, three dimensions.

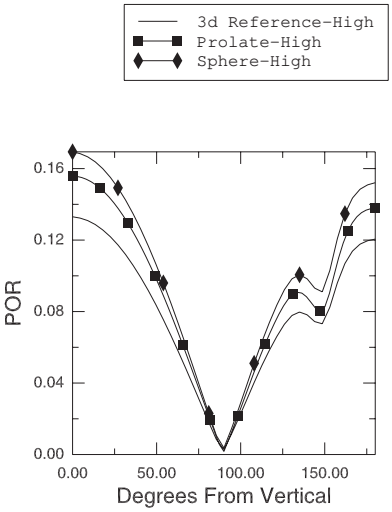


Figure 5: Pressure amplitude comparisons at $ka = 40$, three dimensions.

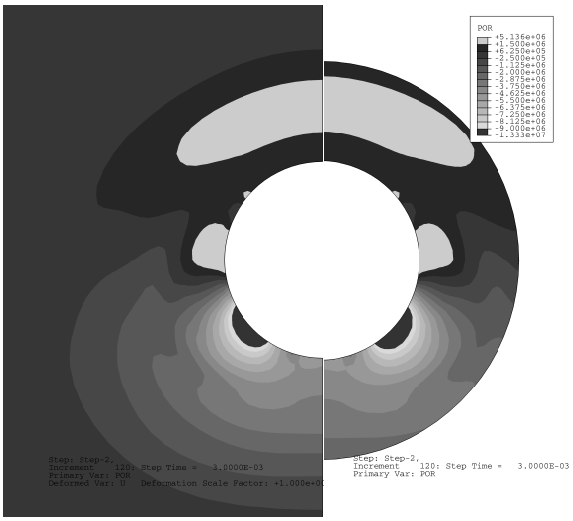


Figure 6: Pressure amplitude comparisons at $t = 0.003$, two dimensions, circular boundary compared to reference.

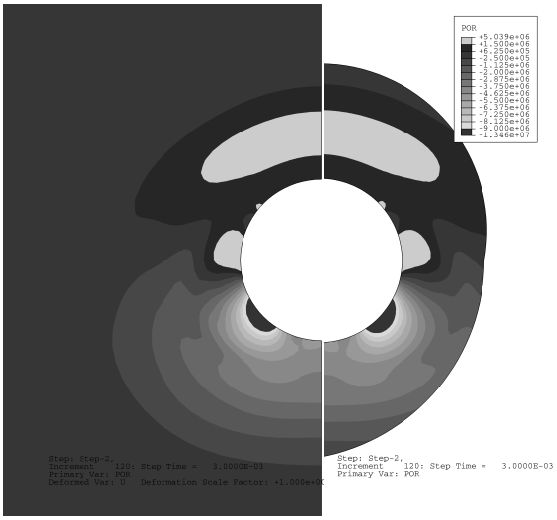


Figure 7: Pressure amplitude comparisons at $t = 0.003$, two dimensions, elliptical boundary compared to reference.

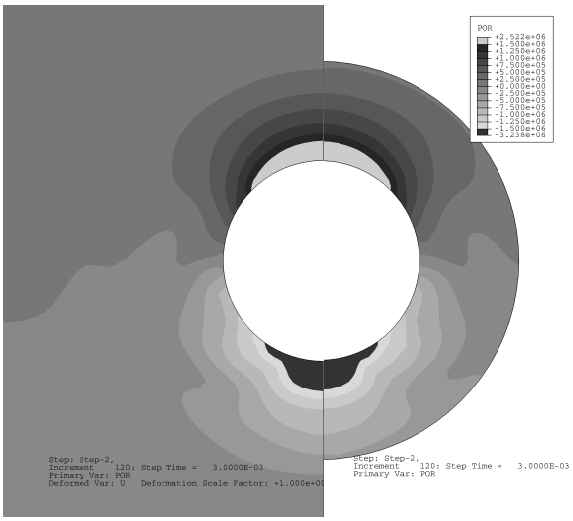


Figure 8: Pressure amplitude comparisons at $t = 0.003$, three dimensions, spherical boundary compared to reference.

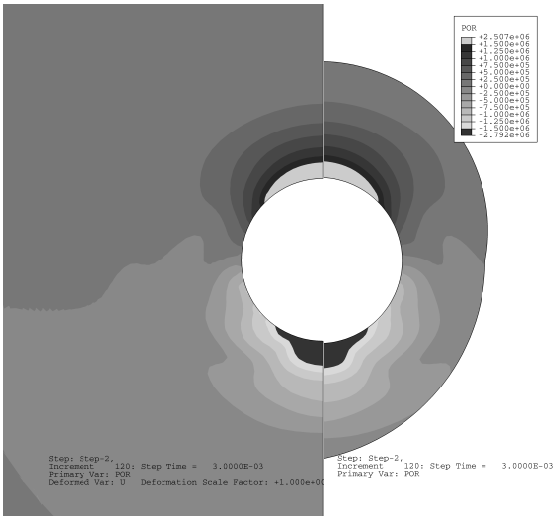


Figure 9: Pressure amplitude comparisons at $t = 0.003$, three dimensions, prolate spheroidal boundary compared to reference.

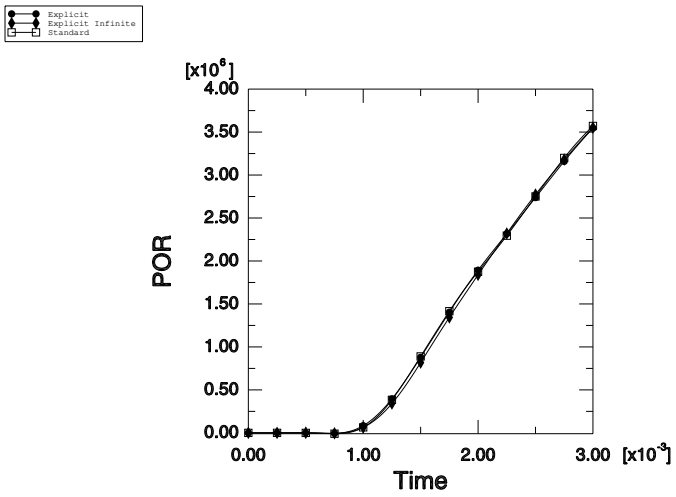


Figure 10: Pressure amplitude comparisons for the transient analysis, two dimensions.

Acoustic-acoustic tie constraint in two dimensions

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example illustrates and verifies the use of the tie constraint in a simple acoustic system, using several procedures.

Problem description

The problem considers steady-state and transient wave propagation in a rectangular duct. *Figure 1* shows the two-dimensional test mesh. There are two unconnected regions: the lower region has the tie constraints, and the simpler upper region is used for comparison.

The lower region is made up of five subregions, totaling 5000 mm long by 1000 mm wide. In Abaqus/Standard the first (leftmost) subregion consists of AC2D6 elements, the second of AC2D3 elements, the third of AC2D4 elements, the fourth (far right, upper) of AC2D4 elements, and the last of AC2D8 elements. The nodes on the intersecting surfaces do not coincide. The upper reference domain is 5000 mm long by 80 mm wide and consists of AC2D4 elements. In Abaqus/Explicit the AC2D4 and AC2D8 elements defined above are replaced by AC2D4R elements, and the AC2D6 elements are replaced with AC2D3 elements. Both domains are made of an acoustic material with a bulk modulus of 0.142 MPa and a density of 1.21 kg per cubic meter. The subregions of the lower domain are connected with tie constraints. In each tie constraint pair, the more refined surface is defined as the secondary surface.

Loading

For the steady-state dynamic simulations both meshes are excited by applying a uniform volumetric acceleration on degree of freedom 8 to the nodes on the left edges of the models. Consistent nodal loads are applied.

For the transient dynamic problems the same concentrated load distribution is applied to the same edges but with a sinusoidal amplitude curve.

In all instances a plane wave absorbing impedance condition is imposed on the rightmost edges of the two domains.

Results and discussion

The steady-state analyses are performed using the direct-solution and subspace-based steady-state dynamic procedures using modes extracted in a frequency step. The analyses are performed at selected frequencies from 30 to 343 Hz.

In the dynamic problem the Abaqus/Standard analysis uses a fixed time step of 0.0004 seconds and is run for a total simulation time of 0.012 seconds.

Each analysis shows good agreement between the reference domain and the domain using tie constraints. The resonant frequencies and mode shapes for the frequency analysis also agree very well, although more modes exist in the frequency band of interest for the wider, lower domain. *Figure 2* shows the phase angle of the acoustic pressure (variable PPOR) at 180.6 Hz in the steady-state response. Since PPOR is the phase angle of the complex-valued solution, defined from −180° to 180°, sharp differences in contour plot color appear at the negative solution values of greatest magnitude. Although the field is continuous, these sharp color lines can be thought of as defining wavefronts in the solution. The wavefronts in the contour plot appear at the same locations for the reference and tied meshes, and the tie constraints do not distort the waves. *Figure 3* and *Figure 4* show the acoustic pressure magnitude (variable POR) in the transient case at 0.012 seconds, before the wave has

reached the end of the meshes. Again, the wavefronts coincide for the tied and reference meshes, and the constraints do not distort the solution.

Input files

Abaqus/Standard input files

acoustic_tie_2d_ssdyn.inp	Steady-state problems.
acoustic_tie_2d_trans.inp	Transient problem.

Abaqus/Explicit input file

acoustic_tie_2d_trans_xpl.inp	Transient analysis.

Figures

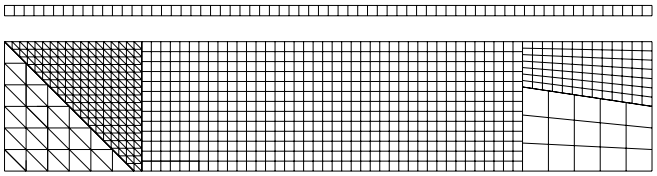


Figure 1: Mesh configuration.

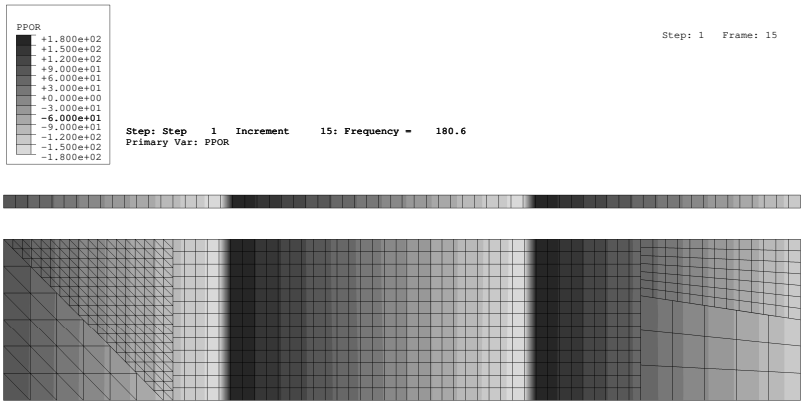


Figure 2: Pressure phase at 180.6 Hz, using a direct-solution steady-state dynamic procedure.

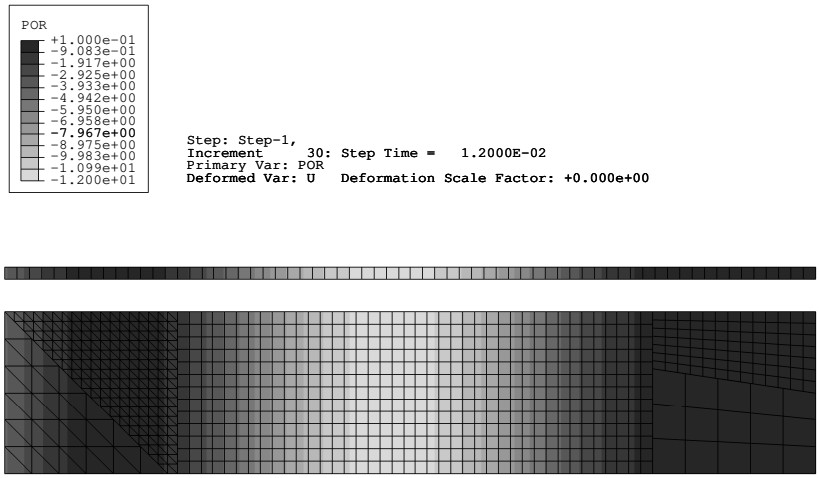


Figure 3: Pressure magnitude at 0.012 seconds, using a dynamic procedure.

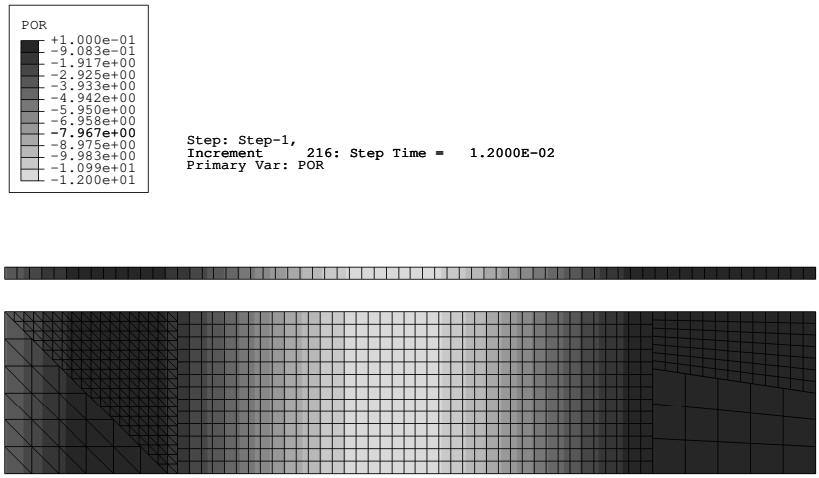


Figure 4: Pressure magnitude at 0.012 seconds, using an explicit dynamic procedure.

Acoustic-acoustic tie constraint in three dimensions

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example illustrates and verifies the use of the tie constraint in a simple three-dimensional acoustic system, using several procedures.

Problem description

This problem examines the natural frequencies of and the steady-state and transient wave propagation in a rectangular duct 20 meters in length and 1 meter square in cross-section. *Figure 1* shows the three-dimensional test mesh. The model is split into two regions: one region has AC3D15 triangular prism elements (AC3D6 triangular prism elements in Abaqus/Explicit), while the other has AC3D4 tetrahedral elements. Both regions are 10 meters long and are connected through tie constraints. Both regions are made of an acoustic material with a bulk modulus of 0.142 MPa and a density of 1.21 kg per cubic meter. The surface on the AC3D15 (AC3D6 in Abaqus/Explicit) side is defined as the secondary surface in the constraint pair.

Loading

The frequency analysis uses no imposed boundary conditions or loads; in acoustic analysis this corresponds to rigid-wall (Neumann) boundary conditions on all exterior surfaces.

In the steady-state dynamic analyses the nodes on the right (unconstrained) end of the AC3D4 mesh are excited using a boundary condition on degree of freedom 8. A plane wave absorbing impedance condition is imposed on the left end of the AC3D15 domain.

In the transient dynamic problems the same boundary condition is applied but with a sinusoidal amplitude. The plane wave condition is imposed here, in the same manner as for the steady-state dynamic analyses.

Results and discussion

The calculated frequencies obtained from the frequency analysis correspond to analytic values, indicating that the constraint transmits the pressure between the mesh regions correctly.

Direct-solution steady-state dynamic analyses are performed at selected frequencies from 5 to 100 Hz. *Figure 2* shows the percentage error in the variable POR (pressure field magnitude) at 20 Hz. The mesh is viewed from the point of view opposite to that of *Figure 1* to show the area of maximum error. The errors in the vicinity of the constraint are on the order of hundredths of a percent; the response in other regions of the mesh is more accurate.

In the dynamic problem the Abaqus/Standard analysis uses a fixed time increment of 0.0005 seconds. *Figure 3* and *Figure 4* show the variable POR (pressure magnitude) at a time of 0.04 seconds, shortly after the wavefront has crossed the tie boundary between the tetrahedra and the wedges. The tie constraints introduce minimal distortion and error in the solution.

Input files

Abaqus/Standard input files

acoustic_tie_3d_ssdyn.inp	Steady-state and natural frequency analysis.

acoustic_tie_3d_trans.inp	Transient analysis.

Abaqus/Explicit input file

acoustic_tie_3d_trans_xpl.inp	Transient analysis.

Figures

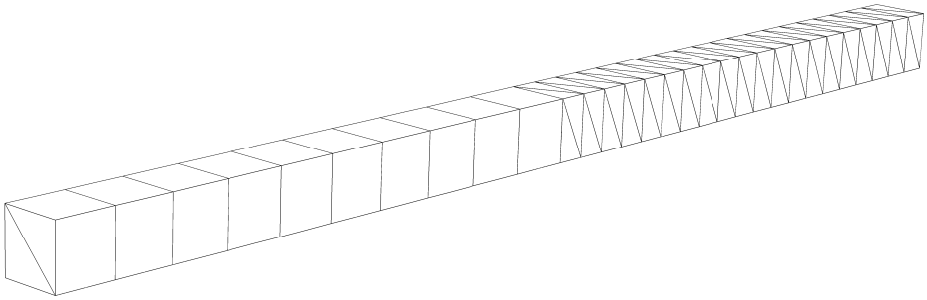


Figure 1: Mesh configuration.

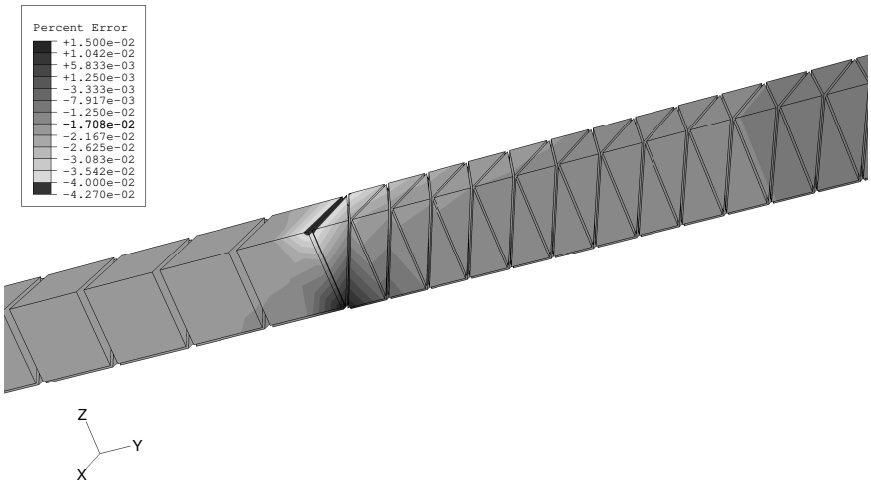


Figure 2: Pressure magnitude error at 20 Hz, using a direct-solution steady-state dynamic procedure.

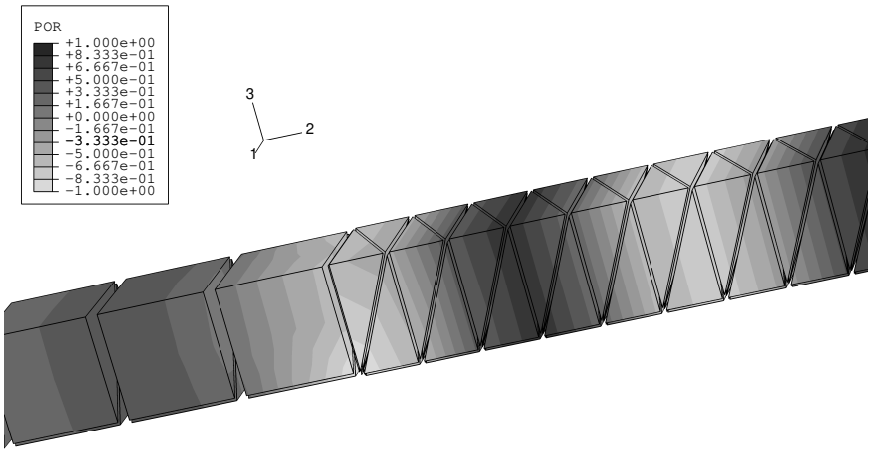


Figure 3: Pressure magnitude at 0.04 seconds, using a dynamic procedure.

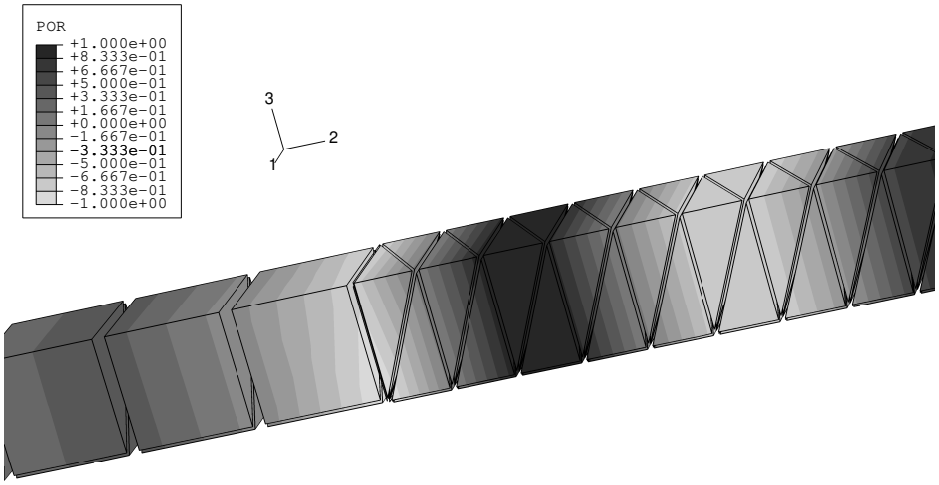


Figure 4: Pressure magnitude at 0.04 seconds, using an explicit dynamic procedure.

A simple steady-state dynamic acoustic analysis

Products: Abaqus/Standard

Overview

This example analyzes the acoustic behavior of a baffled planar piston using the direct-solution steady-state dynamic procedure.
An analytical solution for this problem also exists and is provided for comparison with the numerical result obtained.

Problem description

The model is shown in *Figure 1*. No particular set of units is used in this case: all units used are assumed to be consistent. The acoustic model is a cylinder with radius r = 10 and height h = 15. The density and the bulk modulus are assumed to be 1.0 and 39.4784, respectively, which corresponds to a sound speed of 2π. Default nonreflecting boundary conditions are applied on the top and the lateral surfaces of the cylinder. The baffled condition is imposed by specifying no load or boundary condition at all on the planar baffle. The interaction between the piston and the acoustic medium is simulated by applying a uniform volumetric acceleration on the interface. Since the response of the model is axisymmetric, ACAX8 elements are adopted to model the acoustic medium. The analytical solution of the sound pressure along the axis of axisymmetry is given by the following equation:

$$p = i2p_0 \sin\left(k\frac{\sqrt{z^2+a^2}-z}{2}\right)e^{-ik\left(\sqrt{z^2+a^2}/2\right)},$$

with p_0 defined by

$$p_0 = \rho_0 c_0 v_0,$$

where ρ_0 is the mass density of the material, c_0 is the sound speed of the material, $k = \omega/c_0$ is the wave number, v_0 is the particle velocity, a is the radius of the baffled planar piston, and z is the z-coordinate along the axis of symmetry.

Results and discussion

The response is obtained by conducting a direct-solution steady-state dynamic analysis step. The analysis frequency is chosen as $\omega =$ 20 Hertz, and the amplitude of the volumetric acceleration is determined by the following equation:

$$a_0 = \omega v_0,$$

where the particle velocity is calculated from $p_0 = \rho_0 c_0 v_0$ by setting $p_0 = 1.0$. The variation of sound pressure along the axis of symmetry is shown in *Figure 2*. Both analytical and numerical data are plotted. As can be seen, the numerical result agrees well with the analytical solution. The numerical data in *Figure 2* are obtained in the Visualization module of Abaqus/CAEby creating the *X*–*Y* data along a predefined path, while the analytical data are read into Abaqus/CAE from an ASCII file. The analytical data file is provided along with the input file for this model.

Input files

acousticssdd.inp	Direct-solution steady-state dynamic analysis.

acousticssdd_theory.inp Analytical result.

References

- Kamakura, T., “Fundamentals of Nonlinear Acoustics,” *in Japanese*, 1996.

Figures

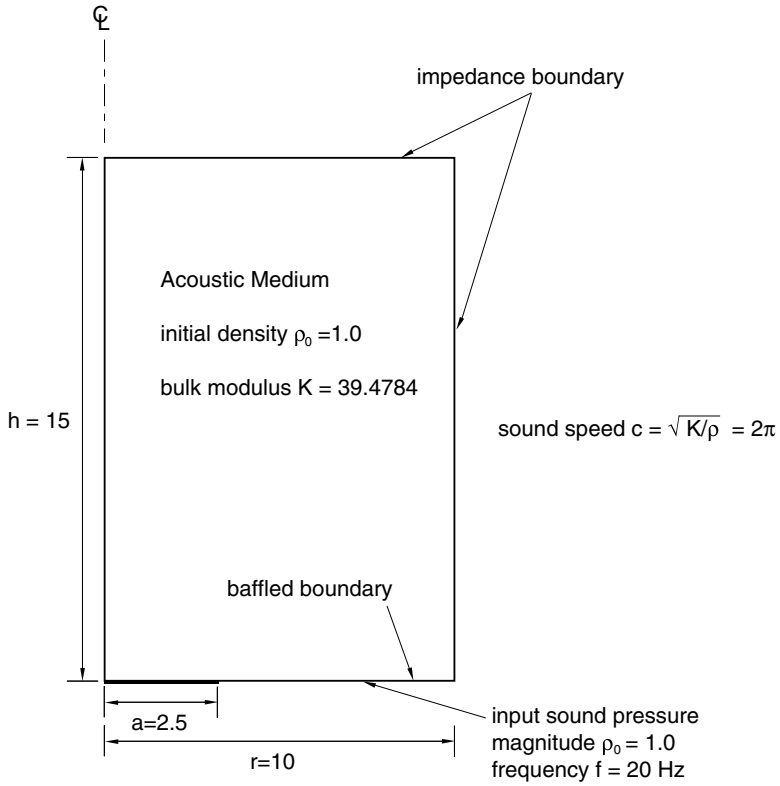


Figure 1: Baffled rigid piston model.

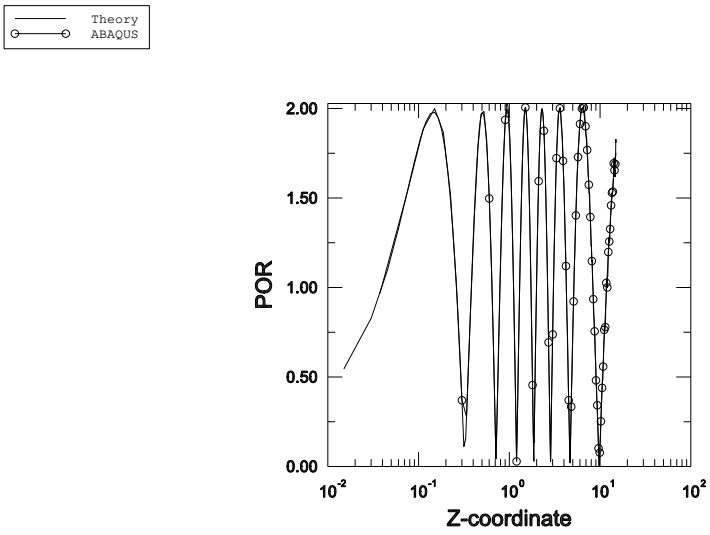


Figure 2: Steady-state response of the system.

Acoustic analysis of a duct with mean flow

Products: Abaqus/Standard

Overview

This example analyzes the acoustic field in a duct with mean flow at high subsonic speed using the direct-solution steady-state dynamic procedure.
Analytical solutions for this problem also exist and are provided for comparison with the numerical results obtained. Real and complex frequency analysis results for the reverberant case are also examined.

Problem description

The model is a simple column of elements oriented along the *x*-axis. The units used in this case are consistent with air: $\rho_f = 1.21$, $K_f = 0.1424 \times 10^6$, the column length is 4, and the Mach number is 0.5. The frequency range of interest is 50 to 300 cycles per second.

Two physical cases are examined: a reverberant end condition and an open condition. In both cases the real and imaginary parts of the acoustic pressure are prescribed at one end of the duct. Default nonreflective impedance conditions are applied on the opposite end of the duct to simulate the open case; no loads, boundary conditions, or impedance conditions are required for the reverberant case.

The general analytical solution of the steady-state sound pressure along the length of the duct with uniform flow at (subsonic) Mach number M is given by

$$\tilde{p} = C_1 e^{-\frac{ikx}{1+M}} + C_2 e^{\frac{ikx}{1-M}},$$

where $k = \Omega/c_0$, $c_0 = \sqrt{K_f/\rho_f}$, and the constants C_1 and C_2 are defined by the prescribed load and end conditions. At the $x = 0$ end, we set $\tilde{p} \equiv 1 + i$ for both the reverberant and open cases. At $x = 4$, the reverberant and open conditions depend on the Mach number. To see this, recall the variational form of the acoustic equation with flow, as used in the derivation of the finite elements for these problems in Abaqus:

$$\int_{V_f} \left[\frac{1}{\Omega^2 \rho_f} \nabla \delta\tilde{p} \cdot (I - \tilde{\mathbf{v}}\tilde{\mathbf{v}}) \cdot \nabla \tilde{p} - \frac{i}{\Omega \rho_f c_0} \left(\nabla \delta\tilde{p} \cdot \tilde{\mathbf{v}} \tilde{p} - \delta\tilde{p}\, \tilde{\mathbf{v}} \cdot \nabla \tilde{p} \right) - \frac{1}{K_f} \delta\tilde{p}\, \tilde{p} \right] dV_f =$$

$$- \int_{S_f} \frac{1}{\Omega^2 \rho_f} \delta\tilde{p} \left[\mathbf{n}^- \cdot (I - \tilde{\mathbf{v}}\tilde{\mathbf{v}}) \cdot \nabla \tilde{p} - \frac{i\Omega}{c_0} \mathbf{n}^- \cdot \tilde{\mathbf{v}}\, \tilde{p} \right] dS_f.$$

In this one-dimensional problem the right hand side for the boundary traction at $x = 4$ reduces to

$$- \int_{S_f} \frac{1}{\Omega^2 \rho_f} \delta\tilde{p} \left[- \left(1 - M^2\right) \frac{\partial \tilde{p}}{\partial x} + \frac{i\Omega}{c_0} M \tilde{p} \right] dS_f.$$

The default reverberant condition in Abaqus sets this boundary traction to zero. This can be satisfied in an analytical solution by enforcing the strong condition

$$\left(1 - M^2\right) \frac{\partial \tilde{p}}{\partial x} = \frac{i\Omega}{c_0} M \tilde{p},$$

and using this, in conjunction with the boundary condition at $x = 0$, to establish the following constants:

$$C_1 = (1+i) \Big/ \left(1 + e^{-4ik\left(\frac{2}{1-M^2}\right)}\right),$$

$$C_2 = C_1 \left(e^{-4ik\left(\frac{2}{1-M^2}\right)}\right).$$

For the open-end case with the same boundary condition at $x = 0$, only right-traveling waves exist; that is,

$$\tilde{p} = (1+i)\, e^{-\frac{ikx}{1+M}}.$$

In Abaqus the open-end condition has to be enforced using a radiation impedance. Applying the right-traveling plane wave $\tilde{p}_{right} = e^{-\frac{ikx}{1+M}}$ to the boundary traction integral and simplifying, we obtain the boundary term as

$$-\int_{S_f} \frac{i\Omega}{\Omega^2 \rho_f} \delta\tilde{p}\ \tilde{p}\ dS_f,$$

which is the same as the Abaqus default radiation condition for quiescent acoustic fluids. Consequently, the open-end analysis in Abaqus for this problem, with Mach number 0.5, is performed the same as for a stationary fluid.

The real and complex frequency analyses are performed separately for the reverberant physical case.

Results and discussion

The responses for the reverberant and open-ended cases are obtained by conducting direct-solution steady-state dynamic analysis steps. The analysis frequencies are chosen between 50 and 300 cycles per second. Analytic and computed results agree as expected.

The results for the real and complex frequency analysis also agree with the expected results.

Input files

acousticflowduct2d.inp	Direct-solution steady-state dynamic analysis, all two-dimensional elements tested.
acousticflowduct3d.inp	Direct-solution steady-state dynamic analysis, all three-dimensional elements tested.
acousticflowduct2d_eig.inp	Real and complex eigenanalysis, all two-dimensional elements tested.
acousticflowduct3d_eig.inp	Real and complex eigenanalysis, all three-dimensional elements tested.

Real exterior acoustic eigenanalysis

Products: Abaqus/Standard

Overview

This example analyzes the symmetric acoustic resonances of a rigid unit sphere immersed in an acoustic fluid of infinite extent using nonreflecting impedance conditions and acoustic infinite elements.

In the case studied a set of real-valued solutions are sought. Analytical solutions for this problem are provided for comparison with the numerical results obtained. Real frequency analysis results for the reverberant case are also examined for comparison.

Problem description

The first model is a small sector of axisymmetric acoustic elements with an inner radius of 1 and an outer radius of 3. The units used in this case are consistent with water: $\rho_f = 997$ and $K_f = 2.12812 \times 10^9$. The frequency range of interest is 21 to 3730 cycles per second. The second model is identical to the first except that it is terminated with an acoustic infinite element.

Two physical cases are examined: a reverberant end condition and an open condition. In the former case no infinite element or impedance condition is used. In the latter case a spherical nonreflective impedance condition or an acoustic infinite element is applied on the opposite end of the duct.

The general analytical solution of the steady-state sound pressure along the radius of the sphere is given by the following:

$$\tilde{p} = \left(\frac{1}{r}\right)\left(C_1 e^{-ikr} + C_2 e^{ikr}\right),$$

with boundary conditions

$$\frac{\partial \tilde{p}}{\partial r} = 0, r = r_0,$$

and

$$\frac{\partial \tilde{p}}{\partial r} = \frac{-1}{r}\tilde{p}\ , \quad r = r_1.$$

The latter equation is the real part of the spherical nonreflecting impedance condition used in Abaqus. Seeking characteristic solutions, the constants C_1 and C_2 are eliminated and the resonant wave numbers k are defined implicitly by the characteristic equation

$$\frac{ikr_0 - 1}{ikr_0 + 1} = e^{2ik(r_1 - r_0)}.$$

For the reverberant case the boundary condition at $r = r_1$ becomes

$$\frac{\partial \tilde{p}}{\partial r} = 0\ , \quad r = r_1,$$

and the resonant wave numbers are defined by

$$\left[\frac{ikr_1 + 1}{ikr_1 - 1}\right]\left[\frac{ikr_0 - 1}{ikr_0 + 1}\right] = e^{2ik(r_1 - r_0)}.$$

The results for the eigenfrequencies are calculated approximately for both analytical formulae using a Newton method.

Results and discussion

The results for the reverberant and open-ended cases are obtained by conducting real-valued frequency extraction steps. The analysis frequencies are chosen between 120 and 3800 cycles per second. Guidelines for acoustic analysis recommend using at least 10 elements per wavelength for accurate solutions. For the mesh used in this problem this corresponds roughly to a limit of 1800 Hz. However, in this problem all of the analytic and computed results agree well, despite the fact that the analysis proceeds well beyond the 10 elements per wavelength frequency of the mesh. Results using the nonreflecting impedance condition are shown in *Table 1* below; results from the infinite element analysis are similar.

Table 1: Analytic and computed results for the nonreflecting impedance condition.

Analytic exterior	Abaqus exterior	Analytic reverberant	Abaqus reverberant
125.2	125.21	—	—
423.6	423.33	394.5	403.63
764.8	762.78	749.7	751.32
119.0	1112.9	1110.0	1105.0
1479.0	1463.7	1472.0	1457.6
1841.0	1810.9	1835.0	1806.0
2204.0	2152.3	2199.0	2148.1
2567.0	2485.9	2563.0	2482.2
2931.0	2810.1	2929.0	2806.9
3295.0	3123.5	3292.0	3120.5
3660.0	3424.6	3657.0	3421.9
4024.0	3712.2	4022.0	3709.6

Input files

exteig_real_ax4_inf.inp	Solution using acoustic infinite element.
exteig_real_ax4_imp.inp	Solution using nonreflecting impedance.

Coupled exterior acoustic eigenanalysis

Products: Abaqus/Standard

Overview

This example analyzes the symmetric acoustic resonances of an elastic spherical shell immersed in an acoustic fluid of infinite extent using acoustic infinite elements.

A set of real-valued and complex-valued solutions are sought. An analytical solution for this problem is provided for comparison with the numerical results obtained.

Problem description

The model consists of a layer of linear axisymmetric acoustic elements with an inner radius of 1 and an outer radius of 1.3. On the inner surface of the layer, linear axisymmetric shell elements of thickness 0.01 are coupled to the water layer. On the outer surface of the water layer, linear axisymmetric acoustic infinite elements are used. The units used are consistent with water ($\rho_f = 997$ and $K_f = 2.12812 \times 10^9$) and with steel ($\rho_f = 7647$, $E = 1.8509 \times 10^{11}$, and $\nu = 0.3$). The frequency range of interest is 36 to 3000 cycles per second, which contains resonances in both the upper and lower branch of the system (see Chapter 10.3 of Junger and Feit, 1972).

Two Abaqus eigenanalysis procedures are used: regular eigenvalue extraction and complex eigenvalue extraction. In the former case the analysis considers the acoustic contributions due to the acoustic finite and infinite element mass and stiffness matrices but not the radiation damping. In addition, the infinite element stiffness matrices are rendered symmetric for compatibility with the real-valued eigensolver. Therefore, these modes are real valued and correspond to the analytical solutions computed using the acoustic accession to inertia only. In the complex eigenvalue extraction procedure the real-valued modes are used as a basis, and the entire finite and infinite element matrix contribution is projected onto this basis. Thus, the radiation damping term is included in the analysis. The shell resonant mode shapes are identical whether or not the shell is coupled to the fluid, which allows the numerically computed modes to be identified with the analytical solution by inspection.

The analytical results for the eigenfrequencies are calculated using the material properties described above.

Results and discussion

Analytic and computed results agree well for both branches, as shown in *Table 1*. Mode shapes also correspond to the analytical solutions. The complex eigenvalue extraction results are more accurate, due to the inclusion of the nonsymmetric acoustic infinite element stiffness matrix and the important radiation damping term.

Table 1: Analytic and computed results for coupled exterior acoustic eigenanalysis.

Branch	Mode	Analytic exterior	Abaqus exterior, real	Abaqus exterior, complex
1	2	261	274	275
1	3	322	339	339
1	4	378	382	382
1	5	428	416	416
1	6	460	444	444
1	7	480	468	468

Branch	Mode	Analytic exterior	Abaqus exterior, real	Abaqus exterior, complex
1	8	496	490	491
1	9	510	512	512
2	1	1001	1228	1225
2	2	1494	1429	1427
2	3	2093	1593	1591

Input files

exteig_cpld_ax4_inf.inp Solution using acoustic infinite element.

References

- Junger, M., and D. Feit, Sound, Structures, and Their Interaction, Massachusetts Institute of Technology, 1972.

Acoustic scattering from a rigid sphere

Products: Abaqus/Standard

Overview

This example calculates the acoustic near field scattered from a sphere when impinged by a plane wave. The example illustrates the use of a simple absorbing boundary condition in conjunction with acoustic continuum elements. The results are compared with a classical solution.

Problem description

A rigid spherical obstacle of radius r_0 = 0.1 m in an unbounded acoustic medium is subjected to an incident plane wave. The analytical solution for the acoustic scattered pressure is of the form

$$p_{scat}(r,\theta) = -P_{incident}\sum_{n=0}^{\infty}(2n+1)\,i^n\,P_n(\cos\theta)\,\frac{j_n'(kr_0)}{h_n'(kr_0)}h_n(kr),$$

where p_{scat} is the scattered acoustic pressure, $P_{incident}$ is the coefficient of the incident plane wave, $P_n(\cos\theta)$ are Legendre polynomials, $j_n(kr_0)$ are spherical Bessel functions of the first kind, $h_n(kr_0)$ are spherical Hankel functions of the first kind, $k = 2\pi f/c_f$ is the acoustic wave number, $c_f = \sqrt{K_f/\rho_f}$ is the speed of sound, and f is the frequency. The orientation is shown in *Figure 1*; the incident field is defined as having zero phase at the origin, which lies at the center of the sphere. The analytical solution is derived in Junger and Feit, but its complex conjugate is used for comparison to conform to the Abaqus sign convention for time-harmonic problems.

Figure 2 shows the finite element mesh using seven layers of AC3D15 elements (252 in total), with an outer radius of r_1 = 0.4 m and a circumferential angle of 10°. Since the problem is axisymmetric, this is sufficient to resolve the field. Planar incident wave loads of unit reference magnitude are applied to the inner surface, with the standoff point defined at the center of the sphere and the source point defined at a point along the positive *x*-axis. Specifying the load in this way means that Abaqus will apply loads on the surface corresponding to an incident pressure field having a value of $1 + 0 \times i$ at the standoff point. A spherical radiation condition is imposed on the outer surface. The acoustic properties of this problem are chosen as follows: K_f = 2.0736 GPa, ρ_f = 1000 kg/m^3, so that the acoustic wave speed is c_f = 1440 m/s. The analysis is run using the direct-solution steady-state dynamic procedure in the range from 30 to 9000 Hertz.

Results and discussion

The finite element results for the scattered pressure in the near field, at $\theta = \pi$, are shown in *Figure 3*, where they are compared with the analytical values. The real and imaginary parts of the solutions show excellent agreement.

Input files

acoustic_scat_sph.inp	Model that uses AC3D15 elements with the Bayliss et al. boundary condition.

References

- Bayliss, A., M. Gunzberger, and E. Turkel, “Boundary Conditions for the Numerical Solution of Elliptic Equations in Exterior Regions,” *SIAM Journal of Applied Mathematics*, vol. 42, no. 2, pp. 430–451, 1982.
- Junger, M., and D. Feit, Sound, Structures, and Their Interaction, The MIT Press, 1972.

Figures

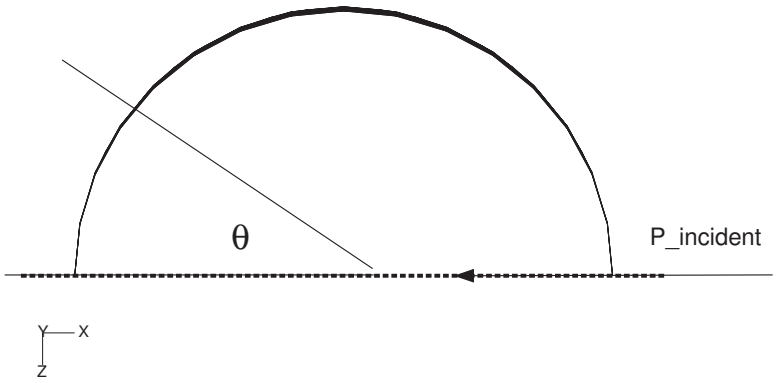


Figure 1: Orientation of the incident wave with respect to the sphere.

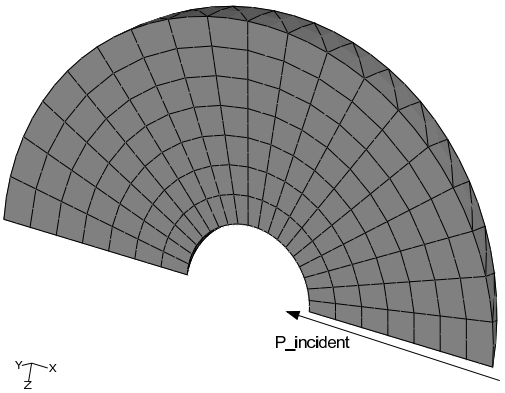


Figure 2: Abaqus mesh using AC3D15 elements.

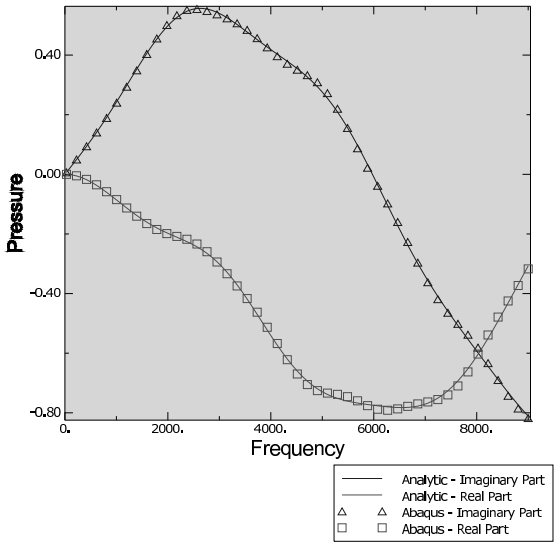


Figure 3: Pressure (POR) versus frequency—real and imaginary parts.

Acoustic scattering from an elastic spherical shell

Products: Abaqus/Standard

Overview

This example calculates the acoustic near field scattered from an elastic spherical shell when impinged by a plane wave.
The example illustrates the use of simple absorbing boundary conditions, acoustic continuum elements, acoustic infinite elements, tie constraints, and incident wave interactions. The results are compared with a classical solution.

Problem description

A thin spherical shell of radius r_0 = 0.1 m and thickness h = 0.001 m in an unbounded acoustic medium is subjected to an incident plane wave. The analytical solution for the acoustic scattered pressure is of the form

$$p_{scat} = p_{elastic} + p_{rigid},$$

where

$$p_{rigid}(r,\theta) = -P_{incident}\sum_{n=0}^{\infty}(2n+1)\,i^n\,P_n(\cos\theta)\,\frac{j_n'(kr_0)}{h_n'(kr_0)}\,h_n(kr),$$

$$p_{elastic}(r,\theta) = P_{incident}\,\frac{\rho_f c_0}{k^2 r_0^2}\sum_{n=0}^{\infty}(2n+1)\,i^n\,P_n(\cos\theta)\,\frac{h_n(kr)}{h_n'(kr_0)}\,\frac{1}{Z_n+z_n}.$$

The elastic pressure term uses the in-vacuo modal impedance of the shell,

$$Z_n = \frac{ih\rho_f c_0}{r_0\Omega}\,\frac{\left[\Omega^2-\Omega_{n,1}^2\right]\left[\Omega^2-\Omega_{n,2}^2\right]}{\Omega^2-\left(1+\beta^2\right)(\nu+n(n+1)-1)},$$

and the specific acoustic modal impedance,

$$z_n = i\rho_f c_0\,\frac{h_n(kr_0)}{h_n'(kr_0)}.$$

Definitions of the terms in the expressions above are found in *Table 1*. The orientation of the incident wave with respect to the sphere is shown in *Figure 1*; the incident field is defined as having zero phase at the origin, which lies at the center of the sphere. The analytical solution is derived in Junger and Feit, but its complex conjugate is used for comparison to conform to the Abaqus sign convention for time-harmonic problems.

The finite element mesh uses AC3D20 elements to model the fluid, with an outer radius of r_1 = 0.25 m and a circumferential angle of 10°. Since the problem is axisymmetric, this is sufficient to resolve the field. The shell is meshed with S8R elements, and this mesh is coupled to the acoustic mesh using a tie constraint. Planar incident wave loads of unit reference magnitude are applied to the inner acoustic and outer shell surfaces, with the standoff point defined at the center of the sphere and the source point defined at a point along the positive x-axis. Specifying the load in this way means that Abaqus will apply loads on the surface corresponding to an incident pressure field having a value of $1 + 0 \times i$ at the standoff point. Two Abaqus models are created: in one, a spherical nonreflecting condition is imposed on the outer surface; in the other, acoustic infinite elements are created and coupled to the acoustic finite elements using a tie constraint. The material properties used in this problem are shown in *Table 2*. The analysis is run using the direct-solution steady-state dynamic procedure in the range from 1500 to 5000 Hertz.

Results and discussion

The finite element results for the scattered pressure in the near field, at a frequency of 1500 Hz, are shown in *Figure 2*, where they are compared with the analytical values. The figure depicts the analytic near field on the upper annulus and the finite element solution on the lower one. The real parts of the solutions show very good agreement. The analytic solution was not plotted using Abaqus/CAE and has a slightly different color scale.

Input files

aco_elas_scat_inf.inp	Model that uses AC3D20 elements and acoustic infinite elements.
aco_elas_scat_nri.inp	Model that uses AC3D20 elements with the Bayliss et al. boundary condition.

References

- Bayliss, A., M. Gunzberger, and E. Turkel, "Boundary Conditions for the Numerical Solution of Elliptic Equations in Exterior Regions," *SIAM Journal of Applied Mathematics*, vol. 42, no. 2, pp. 430–451, 1982.
- Junger, M., and D. Feit, Sound, Structures, and Their Interaction, The MIT Press, 1972.

Tables

Table 1: Variable definitions.

Variable	Definition
p_{scat}	Scattered acoustic pressure
$p_{elastic}$	Elastic contribution to scattered pressure
p_{rigid}	Rigid contribution to scattered pressure
$P_{incident}$	Incident plane wave coefficient
P_n	Legendre polynomial
$j_n\left(kr_0\right)$	Spherical Bessel functions of the first kind
$h_n\left(kr_0\right)$	Spherical Hankel functions of the first kind
$k = 2\pi f/c_f$	Acoustic wave number
$c_f = \sqrt{K_f/\rho_f}$	Speed of sound
f	Frequency
Ω	$\frac{2\pi f r_0}{c_p}$
$\Omega_{n,1}$	n^{th} resonant frequency of shell *in-vacuo*, first branch
$\Omega_{n,2}$	n^{th} resonant frequency of shell *in-vacuo*, second branch

Variable	Definition
β^2	Thin-shell section parameter, $\frac{h^2}{12r_0^2}$
c_p	Plate wave speed, $\sqrt{\frac{E}{(1-\nu^2)\rho_s}}$

Table 2: Material properties.

Parameter	Value
K_f	2.0736 GPa
ρ_f	1000 kg/m^3
c_f	1440 m/s
E	180.3 GPa
ν	0.3
ρ_s	7670 kg/m^3

Figures

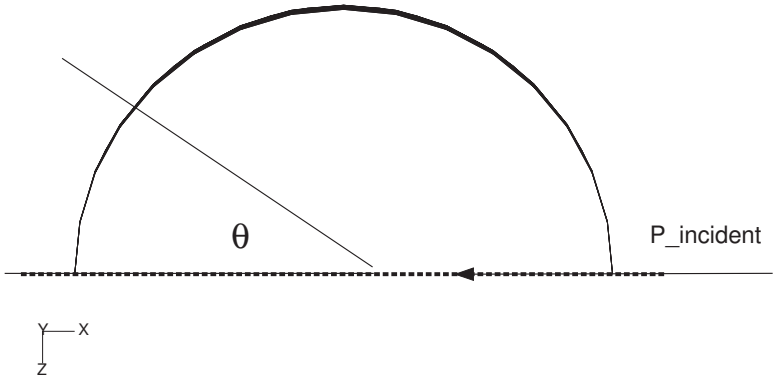


Figure 1: Orientation of the incident wave with respect to the sphere.

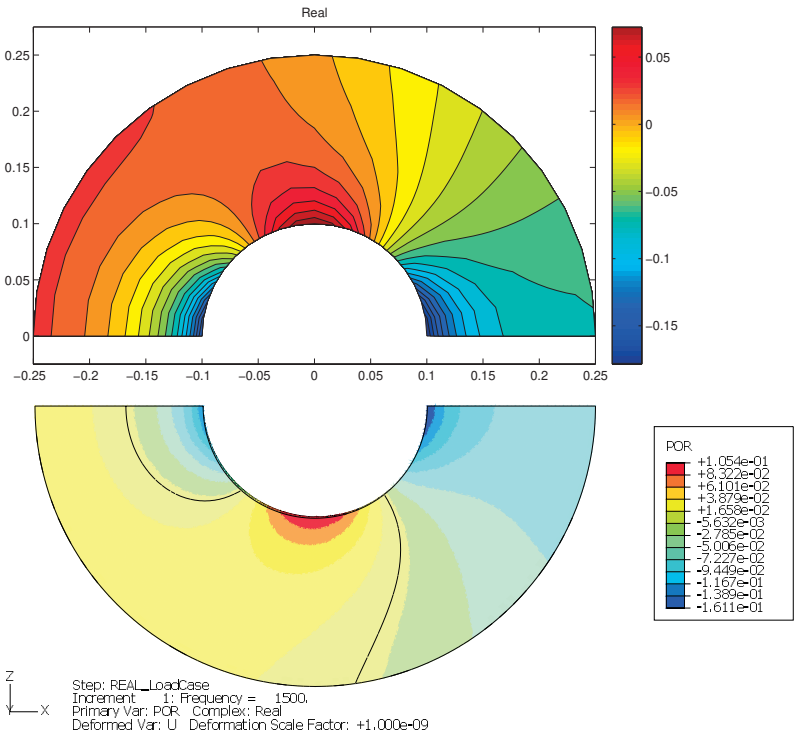


Figure 2: Pressure (POR) at 1500 Hz—real part.

Adaptivity analysis

In this section:

- *Indentation with different materials*
- *Wave propagation with different materials*
- *Adaptivity patch test with different materials*
- *Wave propagation in a shock tube*
- *Propagation of a compaction wave in a shock tube*
- *Advection in a rotating frame*
- *Water sloshing in a pitching tank*

Indentation with different materials

Products: Abaqus/Explicit

Overview

This example illustrates the use of adaptive meshing in indentation problems with different materials.

Problem description

The finite element model for the problem is axisymmetric, as shown in *Figure 1*. The model consists of a rigid punch and a deformable blank. The blank, which is meshed with CAX4R elements, has a radius of 600 mm and a height of 300 mm. The punch is modeled as an analytical rigid surface. Coulomb friction is modeled between the punch and the blank with a friction coefficient of 0.1. Symmetry boundary conditions are defined at r=0 for the blank. The bottom of the blank is also fully constrained.

Several analyses are performed using the following material models for the blank: hyperelasticity, hyperelasticity with viscoelasticity, hyperfoam, Hill plasticity, Mises plasticity, rate-dependent Mises plasticity, Drucker-Prager plasticity, Drucker-Prager cap plasticity, crushable foam plasticity, low-density foam, and porous metal plasticity. The parameters and constants used for each material model can be found in the input files that are included with the Abaqus release. The punch is fully constrained except in the vertical direction, in which motion is prescribed such that the maximum indentation depth is 250 mm. The blank is indented dynamically when it is modeled with a rate-dependent material (hyperelasticity with viscoelasticity or rate-dependent Mises plasticity). A ramped velocity profile is prescribed such that the maximum velocity is 2000 mm/sec. The blank is indented quasi-statically when it is modeled with the remaining (rate-independent) material models. A smooth-step amplitude curve is used to specify the displacement of the punch and promote a quasi-static solution.

Adaptive meshing

A single adaptive mesh domain that incorporates the entire blank is used. The symmetry boundary conditions are defined as Lagrangian boundary regions (the default), and contact surfaces are defined as sliding boundary regions (the default). For some materials a large amount of localized deformation occurs immediately under the punch; therefore, nondefault values are used to specify the frequency of adaptive meshing and the number of mesh sweeps to be performed in each adaptive mesh increment. These values are chosen to solve the problem economically yet retain a good mesh throughout the simulation.

Results and discussion

With the exception of the hyperfoam material, an indentation of this depth cannot be simulated using a pure Lagrangian simulation. The crushable foam material can be indented a majority of this depth using a pure Lagrangian approach. For each of the materials continuous adaptive meshing maintains the quality of the mesh throughout the analysis, and contours of key variables appear to be reasonable. For the hyperfoam material comparisons of the results from the adaptive mesh simulation and a pure Lagrangian simulation can be made. *Figure 2* and *Figure 3* show contours of the y-component of nominal strain in the blank after punching using the adaptive mesh and pure Lagrangian approaches, respectively. The results are in good agreement.

Input files

ale_matverif_indentmises.inp	Mises plasticity.
ale_matverif_indenthyper.inp	Hyperelasticity.

ale_matverif_indentvishyper.inp	Hyperelasticity with viscoelasticity.
ale_matverif_indenthyperfoam.inp	Hyperfoam.
ale_matverif_indentplfoam.inp	Crushable foam plasticity with volumetric hardening.
ale_matverif_indentcrushfoam.inp	Crushable foam plasticity with isotropic hardening.
ale_matverif_indentporous.inp	Porous plasticity.
ale_matverif_indenthill.inp	Hill plasticity.
ale_matverif_indentdrucker.inp	Drucker-Prager plasticity.
ale_matverif_indentcap.inp	Drucker-Prager cap plasticity.
ale_matverif_indentratedep.inp	Mises plasticity with rate dependence.
ale_matverif_indentlowdenfoam.inp	Low-density foam.
lag_matverif_indenthyperfoam.inp	Pure Lagrangian analysis of the hyperfoam model.

Figures

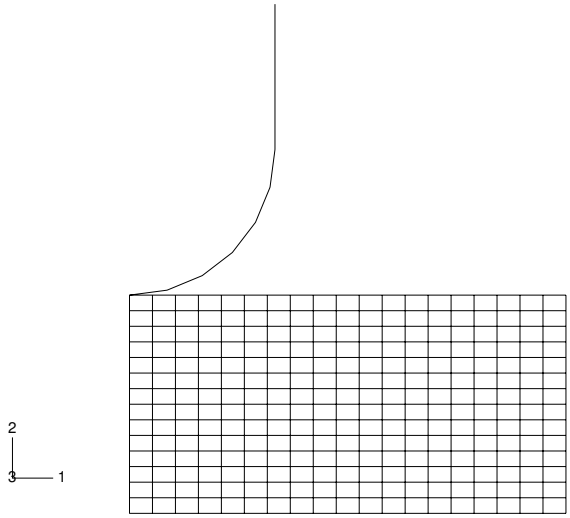


Figure 1: Initial configuration.

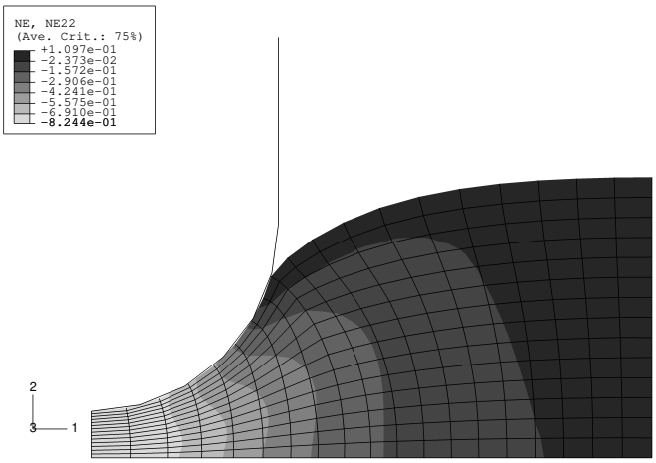


Figure 2: Nominal strain in the y-direction for the hyperfoam model using adaptive meshing.

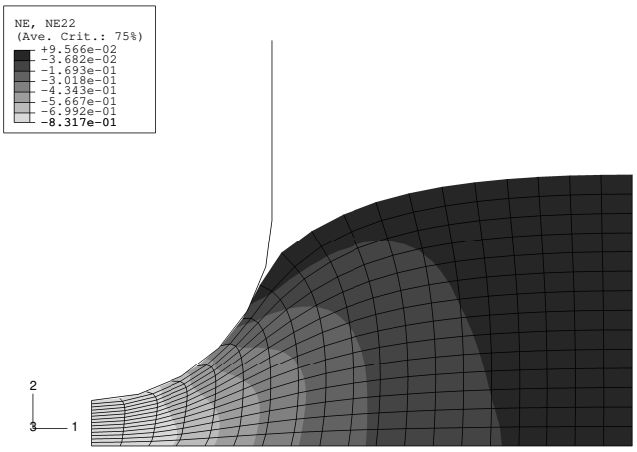


Figure 3: Nominal strain in the y-direction for the hyperfoam model using a pure Lagrangian approach.

Wave propagation with different materials

Products: Abaqus/Explicit

Overview

This example verifies the use of adaptive meshing to analyze wave propagation through a bar.

Problem description

The model consists of a bar through which a one-dimensional wave is propagated. The bar is given an initial rigid body velocity of 57.14 m/sec in the positive *x*-direction. The analysis is run in two steps. In the first step a hat-shaped velocity pulse is defined at the left end of the bar, and a wave form is generated. In the second step the velocity at the left end of the bar is held constant at 57.14 m/sec, and the wave propagates through the bar. The wavelength of the pulse is chosen to be relatively short (over about 10 elements) for a more severe test of the advection algorithms applied to wave propagation. Problems with wavelengths that span large numbers of elements are not as difficult because overall diffusion and dispersion effects are less pronounced.

Two different techniques are used to solve the problem and are shown schematically in *Figure 1*.

1. Both steps are run as a pure Lagrangian analysis.
2. The first step is run as a pure Lagrangian analysis to generate the wave form. The second step is run in an Eulerian fashion by defining an adaptive mesh domain that incorporates the bar. Eulerian boundaries are defined at either end of the bar. Adaptive mesh constraints are used to hold the mesh in place at both the inflow and outflow boundaries. The mesh is held stationary on the interior of the domain by positioning the nodes at the end of the previous adaptive mesh increment using adaptive mesh controls (the default behavior for Eulerian adaptive mesh domains). The mesh is pulled back to its position after the last adaptive mesh increment, which has the effect of holding the mesh stationary for a uniform mesh with no boundary deformation. The default momentum advection method (element center projection) is changed to the half-index shift method because a more accurate momentum advection technique is desirable for wave propagation problems such as these. Although the differences in the two methods are very slight for this problem, the half-index shift is expected to lead to marginally better dispersion characteristics than the element center projection.

Three different geometric models with four different material behaviors for each model are analyzed for both cases. In the two-dimensional, plane strain model the bar measures 27.6 m in length by 0.575 m in width. In the axisymmetric model the bar measures 27.6 m axially (length direction) by 0.575 m radially. In the three-dimensional model the bar measures 27.6 m in length, .575 m in width, and 5 m in depth. For all models the wave is propagated through the bar in the length direction.

The material models used in each analysis include an equation of state having the properties of water, a Mooney-Rivlin hyperelastic material having the properties of rubber, a low-density material having the properties of foam, and a von Mises elastic-plastic material having the properties of steel. The parameters and constants used for each material model can be found in the input files that are included with the Abaqus release. The maximum velocities of the wave pulse in the water, the rubber, and the steel are 492, 28, and 250 m/sec (Mach 0.3, 0.12, and 0.035), respectively. These velocities are typical for shock waves traveling through each type of material.

Results and discussion

Path plots of the pressure along the length of the bar at three equivalent points in time ($t_1 < t_2 < t_3$) are compared for both cases using the equation of state model in *Figure 2*, *Figure 3*, and *Figure 4*. The results are in good agreement, considering the fact that the wave is a strong shock (Mach = 0.3). The adaptive meshing solution predicts slightly smaller peak values; however, the width of the pulse exactly matches the Lagrangian solution and no erroneous phase shifts are introduced. *Figure 5*, *Figure 6*, and *Figure 7* show path plots at three different times for the Mooney-Rivlin hyperelastic material. In this case the wave has a Mach number of 0.12. Again, the pure Lagrangian and adaptive meshing results are in good agreement. The peak values predicted by the adaptive meshing technique for this material are in closer agreement with the pure Lagrangian solution than those predicted for the equation of state material because of the weaker shock wave. Strong shocks are extremely sensitive to any flux limiting introduced during material transport; while the algorithms used are designed to prevent significant diffusion, some inaccuracies are to be expected.

Input files

ale_wave_eos3d.inp	Three-dimensional model with the EOS material.
ale_wave_mises3d.inp	Three-dimensional model with Mises plasticity.
ale_wave_hyper3d.inp	Three-dimensional model with hyperelasticity.
ale_wave_lowden3d.inp	Three-dimensional model with low-density foam.
ale_wave_eosgpe.inp	Plane strain model with the EOS material.
ale_wave_misesgpe.inp	Plane strain model with Mises plasticity.
ale_wave_hypergpe.inp	Plane strain model with hyperelasticity.
ale_wave_lowdengpe.inp	Plane strain model with low-density foam.
ale_wave_eosaxi.inp	Axisymmetric model with the EOS material.
ale_wave_misesaxi.inp	Axisymmetric model with Mises plasticity.
ale_wave_hyperaxi.inp	Axisymmetric model with hyperelasticity.
ale_wave_lowdenaxi.inp	Axisymmetric model with low-density foam.

Figures

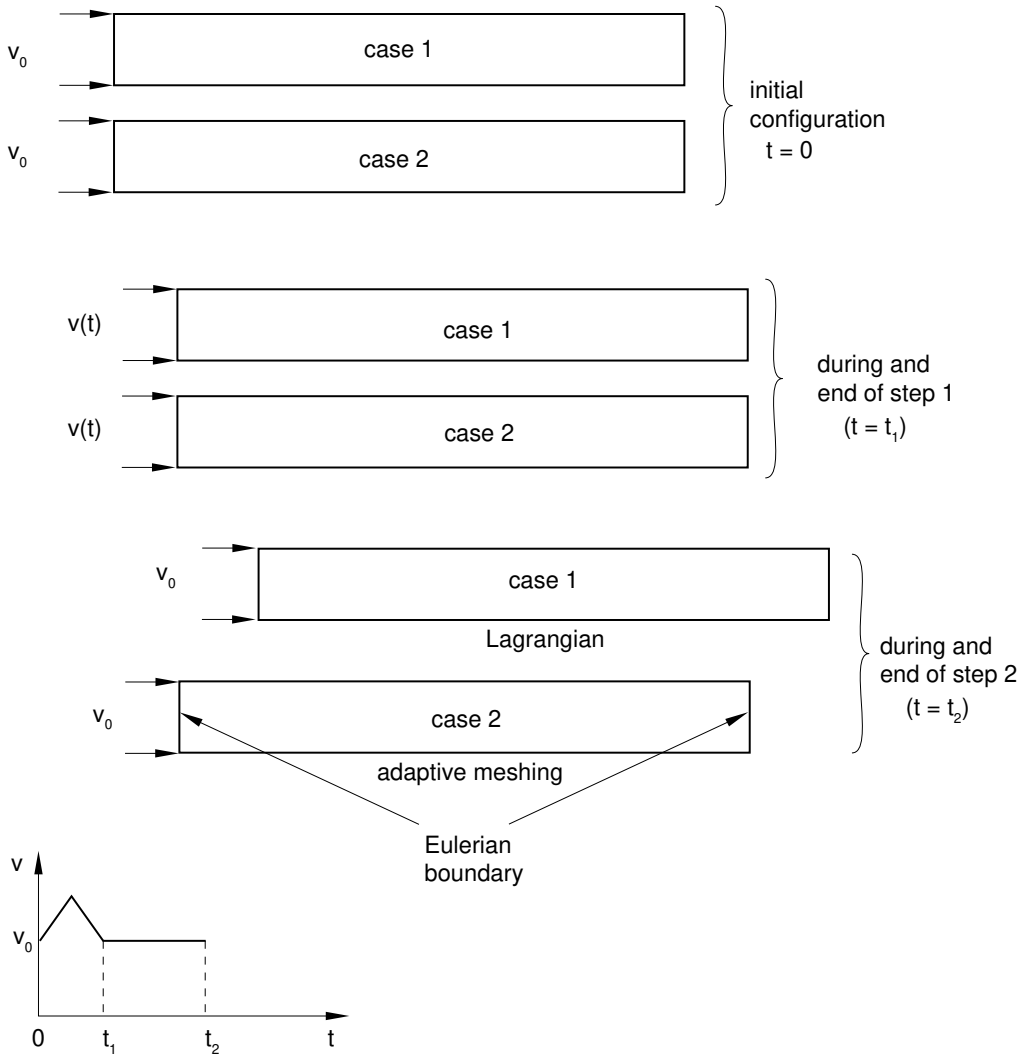


Figure 1: Problem description for both analysis techniques.

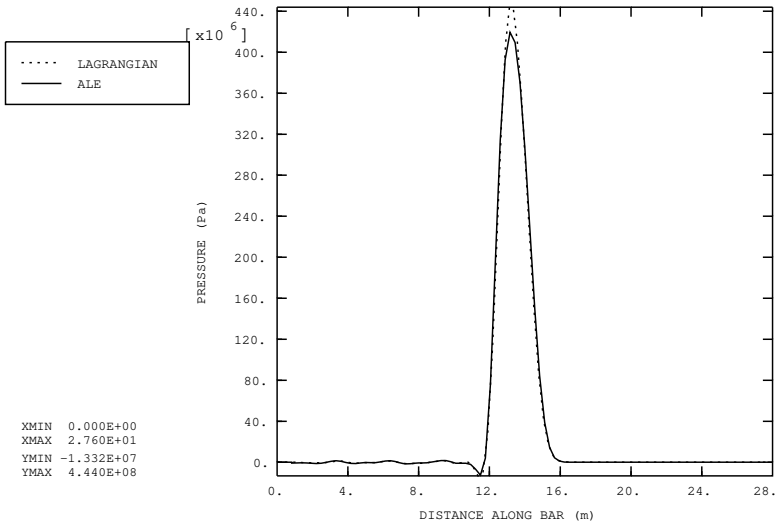


Figure 2: Path plot of pressure at 45% of the total travel time for Cases 1 and 2 (EOS material).

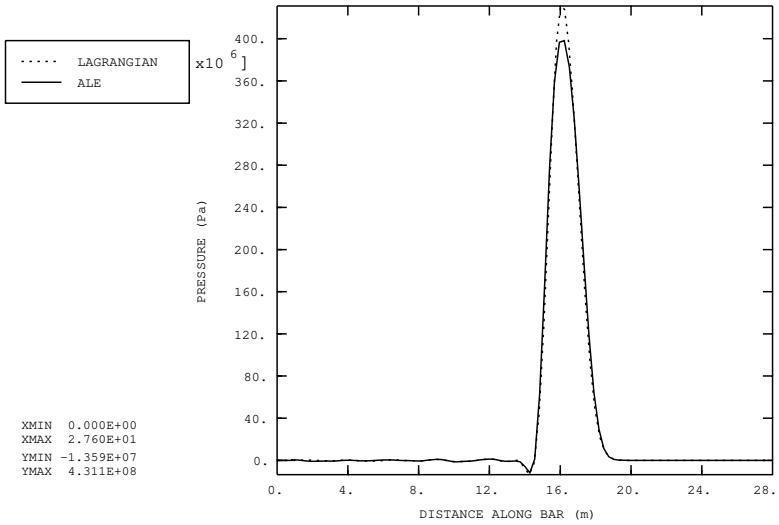


Figure 3: Path plot of pressure at 67% of the total travel time for Cases 1 and 2 (EOS material).

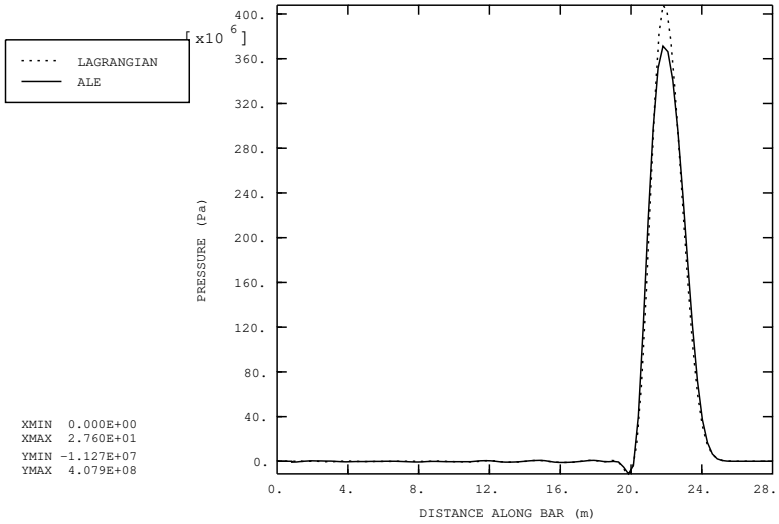


Figure 4: Path plot of pressure at 90% of the total travel time for Cases 1 and 2 (EOS material).

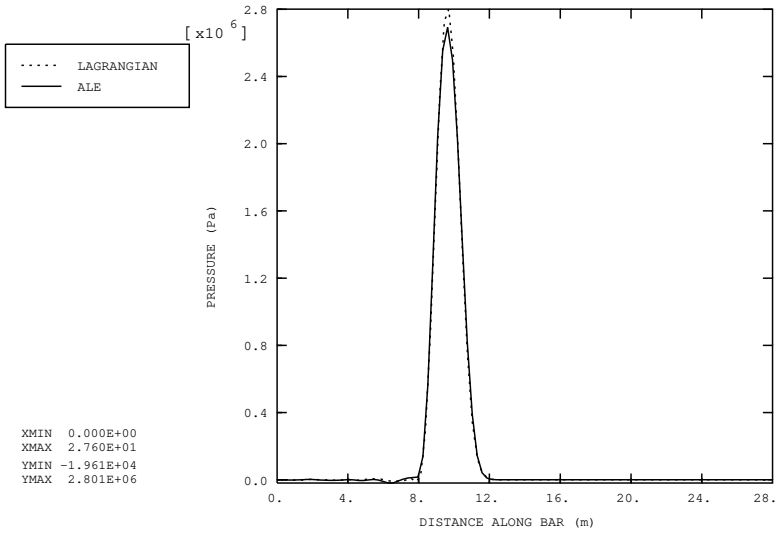


Figure 5: Path plot of pressure at 44% of the total travel time for Cases 1 and 2 (Mooney-Rivlin).

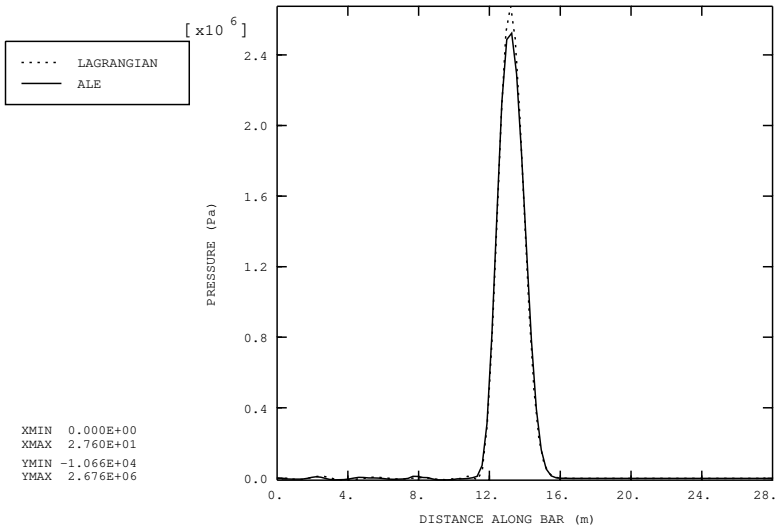


Figure 6: Path plot of pressure at 58% of the total travel time for Cases 1 and 2 (Mooney-Rivlin).

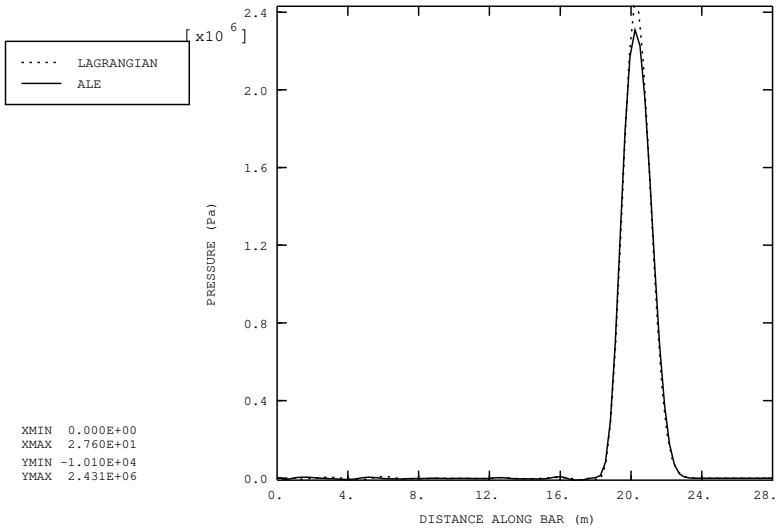


Figure 7: Path plot of pressure at 86% of the total travel time for Cases 1 and 2 (Mooney-Rivlin).

Adaptivity patch test with different materials

Products: Abaqus/Explicit

Overview

This example tests the advection algorithms by loading a small patch of elements.

Problem description

Two-dimensional (plane strain and plane stress) and three-dimensional geometries are considered. For the two-dimensional geometries a square block with an edge length of 4 m is meshed with CPE4R (plane strain) or CPS4R (plane stress) elements, as shown in *Figure 1*. For the three-dimensional geometry a cube with an edge length of 4 m is meshed with C3D8R elements, as shown in *Figure 2*. Both geometries are meshed with initially distorted elements.

All three meshes are tested for the following material models: hyperelasticity, hyperelasticity with viscoelasticity, hyperfoam, Hill plasticity, Mises plasticity, Drucker-Prager plasticity, Drucker-Prager cap plasticity, crushable foam plasticity, low-density foam, and porous metal plasticity. The parameters and constants used for each material model can be found in the input files that are included with the Abaqus release.

The loading is similar for all geometries. The first step is run in a pure Lagrangian fashion. A linearly varying displacement field, acting in the negative y-direction, is prescribed on the bottom edge or face of the mesh, as shown in *Figure 3*. The displacements normal to all the remaining edges/faces are fully constrained. The prescribed displacement is ramped on using a smooth-step amplitude curve to promote a quasi-static response to the loading.

In the second step an adaptive mesh domain is defined for each patch to allow adaptive meshing to occur. The loading and boundary conditions remain unchanged. Adaptive meshing is performed at every increment. At this frequency the mesh distortion is eliminated in a few increments. An accurate advection algorithm ensures that the distribution of solution variables in the patch remains approximately the same before and after adaptive meshing occurs.

Results and discussion

The deformed configurations at the completion of the first and second steps are shown in *Figure 4* and *Figure 5*, respectively, for the plane strain model with Mises plasticity. Corresponding contour plots of equivalent plastic strain are shown in *Figure 6* and *Figure 7*. It is apparent from these plots that adaptive meshing creates a much more uniform mesh without affecting the solution. Similar results (not presented here) are observed for the plane stress and three-dimensional geometries using each material model. For the hyperelasticity with viscoelasticity model the solution does not reach a steady value at the end of the first step because of ongoing stress relaxation. The solution is continuous upon adaptive meshing and continues to relax for the duration of the second step.

Input files

ale_patch_mises2d.inp	Two-dimensional geometry and Mises plasticity.
ale_patch_hyper2d.inp	Two-dimensional geometry and hyperelasticity.
ale_patch_vishyper2d.inp	Two-dimensional geometry and hyperelasticity with viscoelasticity.
ale_patch_hyperfoam2d.inp	Two-dimensional geometry and hyperfoam.
ale_patch_plfoam2d.inp	Two-dimensional geometry and crushable foam plasticity with volumetric hardening.

ale_patch_crushfoam2d.inp	Two-dimensional geometry and crushable foam plasticity with isotropic hardening.
ale_patch_porous2d.inp	Two-dimensional geometry and porous plasticity.
ale_patch_hill2d.inp	Two-dimensional geometry and Hill plasticity.
ale_patch_drucker2d.inp	Two-dimensional geometry and Drucker-Prager plasticity.
ale_patch_cap2d.inp	Two-dimensional geometry and Drucker-Prager cap plasticity.
ale_patch_lowdenfoam2d.inp	Two-dimensional geometry and low-density foam.
ale_patch_mises3d.inp	Three-dimensional geometry and Mises plasticity.
ale_patch_hyper3d.inp	Three-dimensional geometry and hyperelasticity.
ale_patch_vishyper3d.inp	Three-dimensional geometry and hyperelasticity with viscoelasticity.
ale_patch_hyperfoam3d.inp	Three-dimensional geometry and hyperfoam.
ale_patch_plfoam3d.inp	Three-dimensional geometry and crushable foam plasticity with volumetric hardening.
ale_patch_crushfoam3d.inp	Three-dimensional geometry and crushable foam plasticity with isotropic hardening.
ale_patch_porous3d.inp	Three-dimensional geometry and porous plasticity.
ale_patch_hill3d.inp	Three-dimensional geometry and Hill plasticity.
ale_patch_drucker3d.inp	Three-dimensional geometry and Drucker-Prager plasticity.
ale_patch_cap3d.inp	Three-dimensional geometry and Drucker-Prager cap plasticity.
ale_patch_lowdenfoam3d.inp	Three-dimensional geometry and low-density foam.

Figures

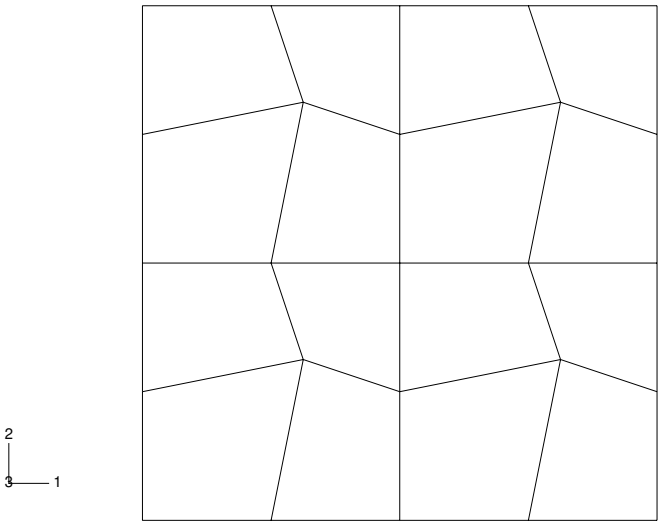


Figure 1: Initial finite element mesh for the two-dimensional geometries.

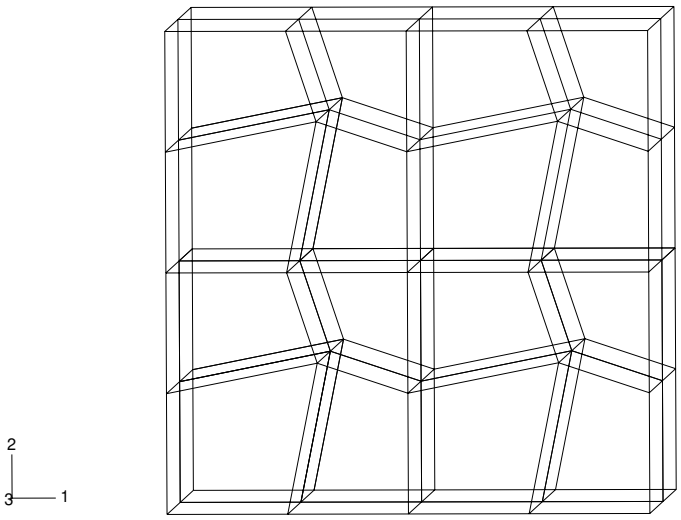


Figure 2: Initial finite element mesh for the three-dimensional geometry.

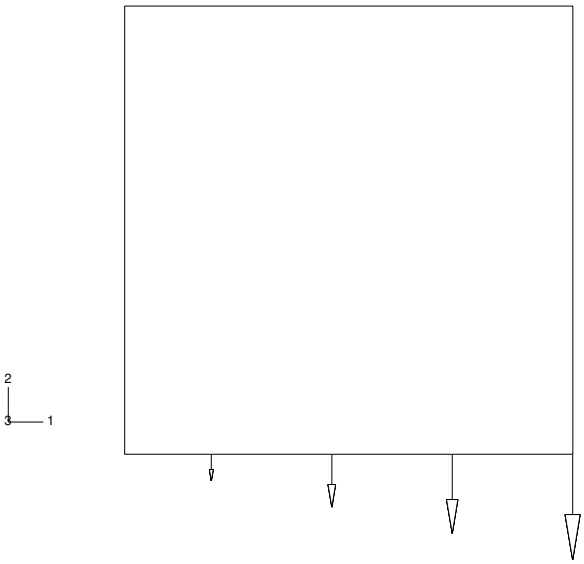


Figure 3: Schematic diagram showing the applied displacement field.

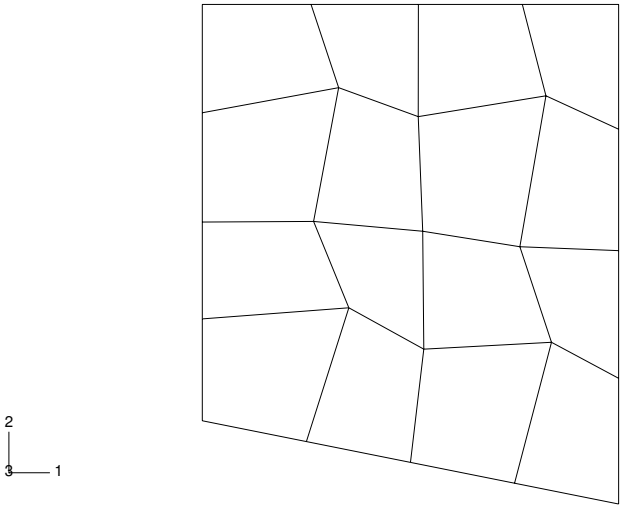


Figure 4: Deformed mesh at the end of the first step for the plane strain geometry with Mises plasticity.

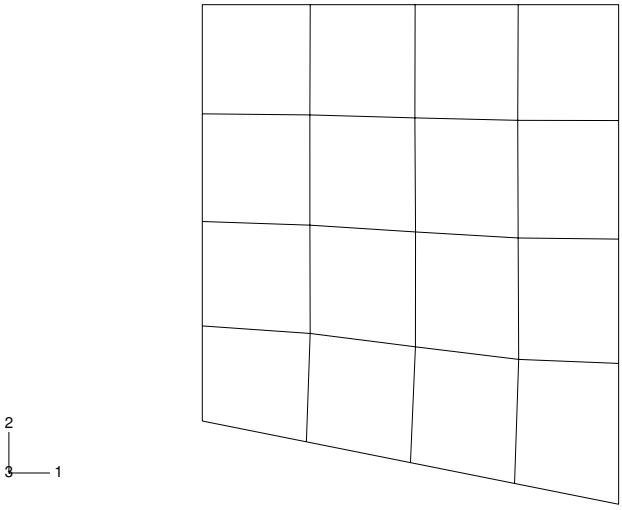


Figure 5: Deformed mesh at the end of the second step for the plane strain geometry with Mises plasticity.

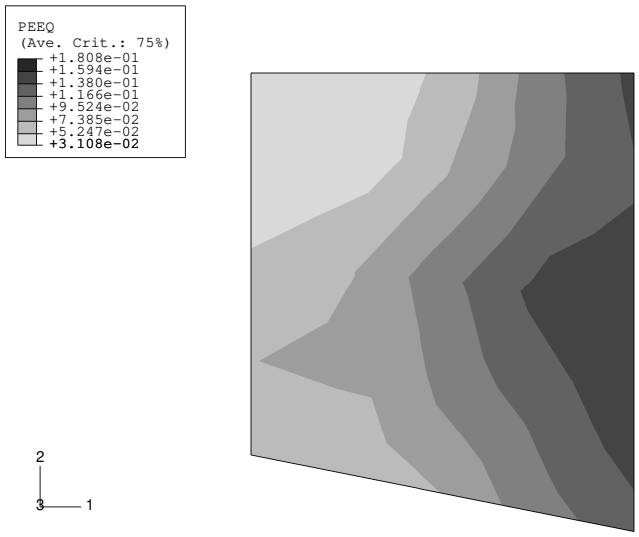


Figure 6: Contours of equivalent plastic strain at the end of the first step for the plane strain geometry with Mises plasticity.

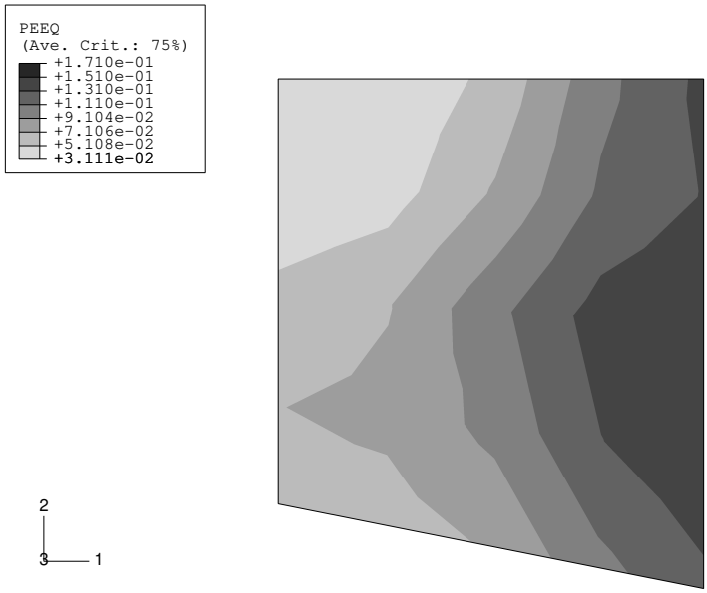


Figure 7: Contours of equivalent plastic strain at the end of the second step for the plane strain geometry with Mises plasticity.

Wave propagation in a shock tube

Products: Abaqus/Explicit

Overview

This example models the one-dimensional propagation of a shock wave in a polytropic gas.
An analytical solution is given in Amsden, Ruppel, and Hirt (1980).

Problem description

A schematic of the model is shown in *Figure 1*; the model consists of a compartmentalized shock tube filled with a polytropic gas. A diaphragm separates the tube into two equal compartments. Both compartments are filled with the same gas, but the ratio of the density of the gas in compartment A to the density of the gas in compartment B is 2:1. The diaphragm separating the compartments is instantaneously removed, causing a shock wave to advance into compartment B and a rarefaction wave to propagate back into compartment A.

The plane strain finite element model used for the analysis is shown in *Figure 2*. While the shock tube is not meshed, the gas in the tube is meshed with CPE4R elements and fills a volume of dimensions $20 \times 0.333 \times 1$. The Abaqus/Explicit ideal gas equation of state model is used with a gas constant of 0.2 and a constant specific heat at constant volume of 0.3. The initial state of the gas is determined from an initial uniform temperature of 0.6, an initial density of 0.2 for compartment A, and an initial density of 0.1 for compartment B. The initial specific energy of 0.18 is also given only for the purpose of total specific energy output.

Symmetry boundary conditions are prescribed on all four outer boundary walls of the tube throughout the analysis. The analysis is continued until $t = 10$ sec, at which time the shock wave has propagated halfway through the original compartment B.

The following three different techniques are used to solve the problem:

1. Pure Lagrangian: The problem is solved with a pure Lagrangian analysis; no adaptive meshing is performed.
2. Adaptive meshing with two domains: An adaptive mesh domain is defined for each compartment, and continuous adaptive meshing is performed within each domain. The interface between the two compartments remains Lagrangian because of the boundary region between the two domains. The net effect of this constraint is that there is no mixing of the gas contained in each compartment. The frequency of adaptive meshing is changed to 1 from a default value of 10 because of the substantial material flow through the mesh that occurs when the shock wave propagates.
3. Adaptive meshing with one domain: The analysis is performed using a traditional Eulerian approach. A single adaptive mesh domain is defined that encompasses both compartments. This allows the gas from the two compartments to mix freely when the diaphragm is removed. As the shock wave moves through the tube, the mesh can be held stationary using one of two techniques: (1) applying spatial adaptive mesh constraints on every node or (2) performing adaptive meshing based on the positions of nodes at the end of the previous adaptive mesh increment, which has the effect of holding the mesh stationary for a uniform mesh with no boundary deformation. The latter technique is adopted here. The frequency of adaptive meshing is changed to 1 from a default value of 10 because of the substantial material flow through the mesh that occurs when the shock wave propagates.

Although this simple one-dimensional problem can be solved satisfactorily with all three approaches, two- and three-dimensional shock wave problems involving compressible gases almost always require the third approach that uses one adaptive mesh domain to include both the high energy (density) and low energy gases. This approach is required because of the substantial amount of material flow resulting from the expansion of the high energy (density) gas into the area occupied by the low energy gas.

Results and discussion

Figure 3 shows the deformed configuration for each of the three cases at the end of the analysis. The magnitude of the shock wave can be realized by studying the size of the elements along the length of the tube for Case 1 (the pure Lagrangian analysis). Path plots of the density along the length of the tube at the end of the analysis are shown in *Figure 4* for each case. The results are identical. In *Figure 5* the path plot of density for Case 3 (the analysis using one adaptive mesh domain) is compared to the solution given by Amsden, et al. The results are in close agreement.

Input files

lag_shocktube.inp	Case 1.
ale_shocktube.inp	Case 2.
ueul_shocktube.inp	Case 3.

References

- Amsden, A. A., H. M. Ruppel, and C. W. Hirt, "SALE: A Simplified ALE Computer Program for Fluid Flow at All Speeds," *Los Alamos Scientific Laboratory*, 1980.

Figures

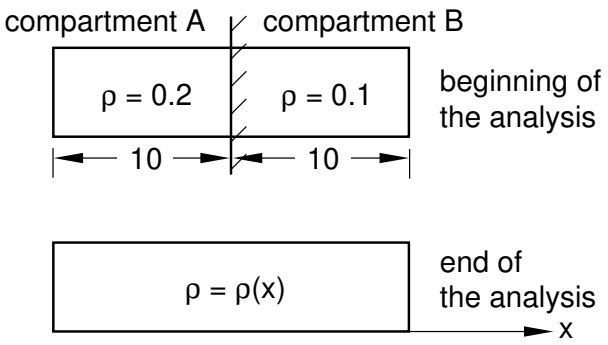


Figure 1: Schematic drawing of the shock tube.



Figure 2: Initial configuration.



Figure 3: Deformed configuration at the end of the second step for Cases 1–3.

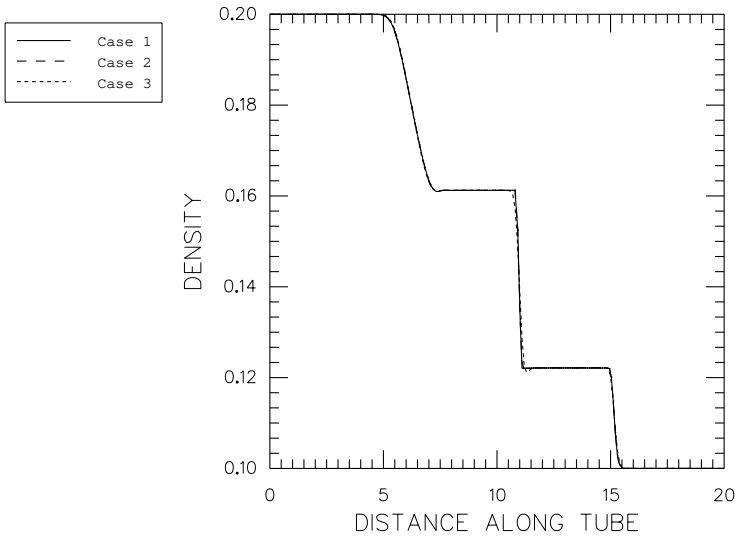


Figure 4: Comparison of the density along the length of the shock tube at the end of the second step for Cases 1–3.

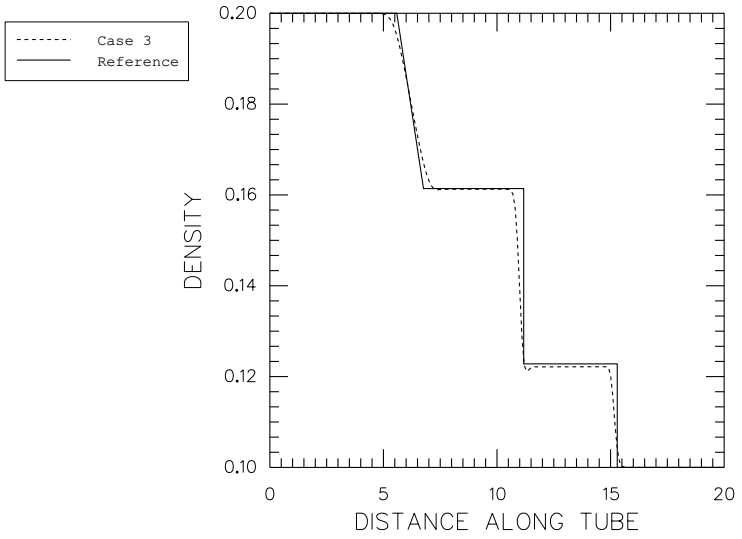


Figure 5: Comparison of the density along the length of the shock tube at the end of the second step for Case 3 and the analytical solution.

Propagation of a compaction wave in a shock tube

Products: Abaqus/Explicit

Overview

This example shows the one-dimensional propagation of a compaction wave in partially saturated sand. The results are compared with the solution given in Wardlaw, McKeown, and Chen (1996).

Problem description

The problem under consideration is similar to that discussed in *Wave propagation in a shock tube*. Here we consider the case of a compartmentalized shock tube filled with sand, as shown in *Figure 1*. A diaphragm separates the tube into two equal compartments. The left compartment is filled with compacted (fully saturated) sand with initial pressure $p =$ 423 MPa. The right compartment is filled with partially saturated sand with initial pressure $p =$ 0. The diaphragm separating the compartments is instantaneously removed, causing a compaction wave to advance into the right compartment and a rarefaction wave to propagate back into the left compartment. The compaction wave causes permanent compaction of the sand in the right compartment.

The mechanical response of the sand is modeled using the $P - \alpha$ equation of state material model (*Equation of State*). Sand consists of sand grains, water, and air or voids. It is assumed that the air does not carry any pressure. The solid part of the $P - \alpha$ model represents the sand grain-water mixture, while the porous part accounts for the air-void content. In the absence of the air-void content, sand is said to be fully saturated, or fully compacted, which in the framework of the $P - \alpha$ model corresponds to $\alpha = 1$. The constitutive behavior of fully saturated sand is described by a Mie-Grüneisen equation of state. At its virgin state, under zero pressure, sand usually contains an initial void volume fraction and is said to be partially saturated. As the pressure increases, sand undergoes irreversible compaction and permanent (plastic) volume change. Sand becomes fully compacted when the pressure reaches the compaction pressure p_s.

The following material properties are used:

Solid phase (Mie-Grüneisen)		Compaction properties	
ρ_{s0}	2070 kg/m^3	C_e	600 m/sec
C_s	1480 m/sec	n_0	0.049758
s	1.93	p_e	0.0 MPa
Γ_0	0.880	p_s	6.5 MPa

The sand in the left compartment is assumed to be fully compacted ($\alpha =$ 1), with initial pressure $p =$ 423 MPa ($p > p_s$) and initial specific energy $E_m =$ 5000 j oule/Kg. On the other hand, the sand in the right compartment is initially at the virgin state: $p = 0$, $E_m =$ 0, and porosity $n = n_0 =$ 0.049758 (or $\alpha = \alpha_0 =$ 1.052364).

Plane strain CPE4R elements are used to mesh the sand in the tube, which fills a volume of 5 × 0.1 × 1 m^3. Symmetry boundary conditions are prescribed on all four outer boundary walls of the tube throughout the analysis. The analysis is continued until $t =$ 0.001 sec.

The following three techniques are used to solve the problem:

1. Pure Lagrangian: The problem is solved with a pure Lagrangian analysis; no adaptive meshing is performed.

2. Adaptive meshing with two domains: An adaptive mesh domain is defined for each compartment, and continuous adaptive meshing is performed within each domain. The interface between the two compartments remains Lagrangian because of the boundary region between the two domains. The net effect of this constraint is that there is no mixing of the sand contained in each compartment. The frequency of adaptive meshing is changed to 1 from a default value of 10 because of the substantial material flow through the mesh that occurs when the shock wave propagates.
3. Adaptive meshing with one domain: The analysis is performed using a traditional Eulerian approach. A single adaptive mesh domain is defined that encompasses both compartments. This allows the sand from the two compartments to mix freely when the diaphragm is removed. As the shock wave moves through the tube, the mesh can be held stationary using one of two techniques: (1) applying spatial adaptive mesh constraints on every node or (2) performing adaptive meshing based on the positions of nodes at the end of the previous adaptive mesh increment, which has the effect of holding the mesh stationary for a uniform mesh with no boundary deformation. The latter technique is adopted here. The frequency of adaptive meshing is changed to 1 from a default value of 10 because of the substantial material flow through the mesh that occurs when the shock wave propagates.

Results and discussion

Path plots along the length of the tube at the end of the analysis ($t = 0.001$ sec) are shown in *Figure 2* and *Figure 3* for the density, ρ, and the distension, α, respectively. The results from all analyses (Cases 1, 2, and 3) are nearly identical, which indicates that the one-dimensional problem can be solved satisfactorily with all three approaches. However, two- and three-dimensional shock wave problems usually require the third approach. *Figure 3* shows that as the compaction wave advances toward the right compartment, it leaves behind a trail of fully saturated sand ($\alpha = 1$). In *Figure 4* the path plot of density for Case 3 (the analysis using one adaptive mesh domain) is compared to the solution given by Wardlaw, et al. The results are in close agreement.

Input files

lag_compaction.inp	Lagrangian analysis (Case 1).
ale_compaction.inp	Adaptive meshing analysis with two domains (Case 2).
ueul_compaction.inp	Adaptive meshing analysis with one domain (Case 3).

References

- Wardlaw, A. B., R. McKeown, and H. Chen, "Implementation and application of the $P - \alpha$ equation of state in the DYSMAS code," *Naval Surface Warfare Center, Dahlgren Division, Report Number: NSWCDD/TR-95/107*, May 1996.

Figures

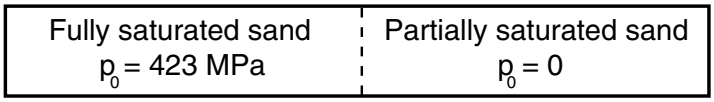


Figure 1: Schematic drawing of the shock tube.

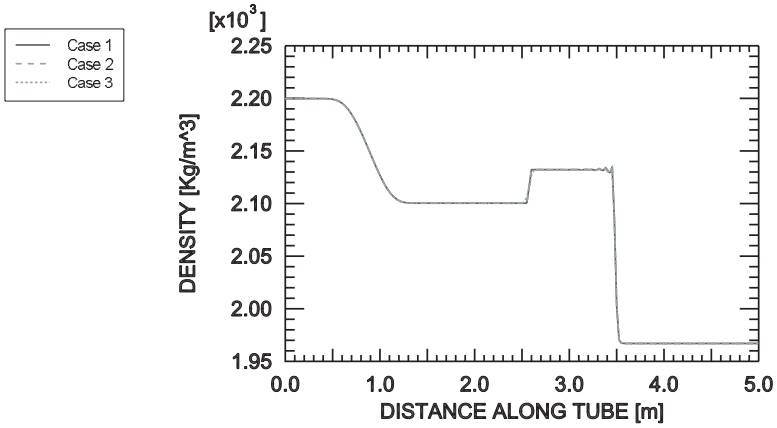


Figure 2: Comparison of the density along the length of the shock tube at the end of the analysis for Cases 1–3.

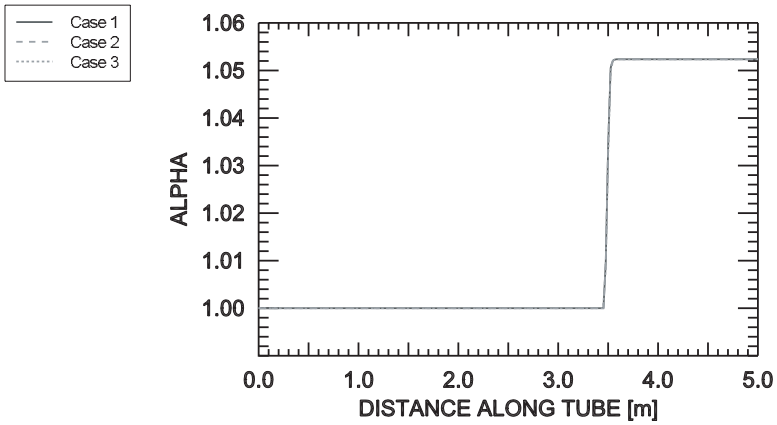


Figure 3: Comparison of the distension α along the length of the shock tube at the end of the analysis for Cases 1–3.

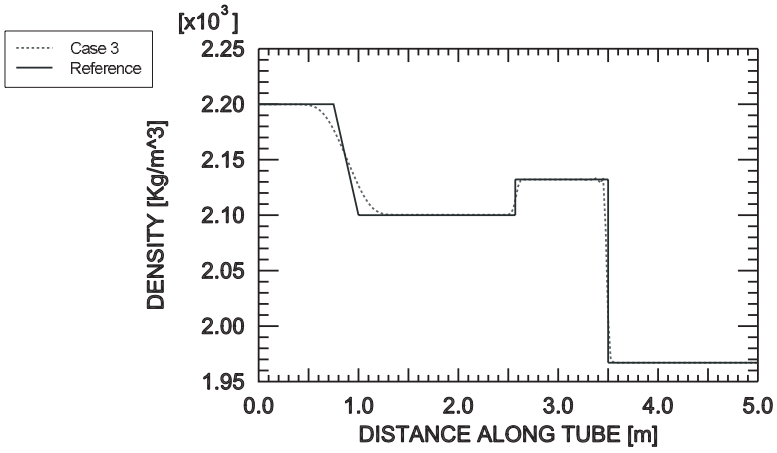


Figure 4: Comparison of the density along the length of the shock tube at the end of the analysis for Case 3 and the reference solution.

Advection in a rotating frame

Products: Abaqus/Explicit

Overview

This example tests the accuracy of the advection algorithms used in adaptive meshing by studying the advection of a single scalar variable, adiabatic temperature, in a rotating flow field.

Problem description

Adiabatic temperature is a convenient scalar variable for this type of test because its spatial distribution can be held constant over a step and it is remapped when adaptive meshing is used. The rotating flow field is generated by either holding the mesh fixed while rotating the material or holding the material fixed while rotating the mesh.

The finite element model consists of a two-dimensional domain with dimensions 2.0 × 2.0 meshed with CPS4R elements. The mesh density is 80 × 80, and the origin is located at the center of the square domain. The initial configuration is shown in *Figure 1* with contours of initial temperature on the mesh. The initial temperature distribution, θ_0, is a function of the coordinates and is given as

$$\theta_0 (X, Y) = .01^{4r^2},$$

where $r = \sqrt{(X + 1/2)^2 + Y^2}$.

As shown in the figure, the temperature distribution has a peak value of 1.0 that occurs along the x-axis at $x = -0.5$. The temperature tends to zero as the distance from the peak increases. The temperature in the first and fourth quadrants of the model and along all the edges is less than 0.01. Tracer particles are defined to monitor the material motion and temperature throughout the analysis. As shown in *Figure 1*, the tracer particles are located initially along the negative x-axis.

An adiabatic procedure is used, and the material is modeled as von Mises elastic-plastic. The Young's modulus and yield strength are chosen so that the material undergoes very little deformation and stays in the elastic regime; therefore, the temperature field remains unchanged from its initial condition.

A rotating flow field is generated using one of the following two techniques:

1. The mesh is held fixed, and the material is given a rotation about the origin. The material is assumed to extend beyond the boundaries of the finite element mesh. All the elements are included in a single adaptive mesh domain, and an Eulerian boundary is defined along the entire perimeter of the model. The mesh is held fixed spatially by applying adaptive mesh constraints at the nodes along the Eulerian boundary and also by performing adaptive meshing based on the position of nodes at the end of the previous adaptive mesh increment, which has the effect of holding the mesh stationary for a uniform mesh with no overall deformation.

   An initial rotational velocity of 1256.64 rad/sec about the origin is specified for all nodes, so that the material rotates a full 360° in 0.005 s. The geometry and problem definition should result in conservation of angular momentum, even though material flows into and out of the domain through the Eulerian boundary. The rotary inertia of the system is based only on the mass distribution within the model boundaries (not on material outside the Eulerian boundaries), which remains constant throughout the analysis. Thus, with proper advection, the angular velocity should also remain constant throughout the analysis.
2. The material is fixed, and the mesh is given a rotation about the origin. As in Case 1, the material is assumed to extend beyond the boundaries of the mesh. All elements are included in a single adaptive mesh domain, and an Eulerian boundary is defined along the entire perimeter of the model. For this case the mesh domain

is rotated by applying an adaptive mesh constraint at nodes along the Eulerian boundary. The motion of each node along the boundary is prescribed by defining a separate amplitude curve. Adaptive meshing is performed every 10 increments, and the number of mesh sweeps per increment is 5. With these settings nodes on the interior of the mesh follow the rotation only approximately, lagging slightly behind. The lagging of the mesh can be minimized by increasing the number of mesh sweeps or lowering the frequency value. However, lagging is intentionally allowed here as a verification of the advection algorithms for a geometrically complex mesh pattern.

Results and discussion

Figure 2 to *Figure 4* show the mesh configuration and temperature distribution at $t =$ 0.0015 s, $t =$ 0.0035 s, and at the final time of $t =$ 0.005 s for Case 1. Although the mesh does not move, the contours of adiabatic temperature and the tracer particles clearly demonstrate the rotation of the material about the center of the domain. The shape and levels of the contours show that the adiabatic temperature distribution is advected throughout the rotational motion with minimal error. The initially straight line array of tracer particles, which are fixed to material points, remains straight throughout the rotation. These results also verify the momentum advection algorithm since angular momentum must be conserved for the material to come full circle in 0.005 s. *Figure 5* shows the time histories of adiabatic temperature at four selected tracer particles representing a range of temperatures. With perfect advection the temperatures at these particles should remain constant. For the 360° rotation the peak temperature value is reduced by 7%. The temperatures at locations moving away from the peak remain nearly constant.

Figure 6 to *Figure 8* show the mesh configuration and temperature distribution at $t =$ 0.0015 s, $t =$ 0.0035 s, and at the final time of $t =$ 0.005 s for Case 2. Although the material does not move, the mesh motion is apparent from the figures. The contour distribution and tracer particles remain stationary because the material does not move. However, this case does verify the accuracy of the tracer particle tracking algorithms and adiabatic temperature advection algorithms (the mesh moves relative to the material). *Figure 9* shows time histories of adiabatic temperature at four selected tracer particles. This case shows a level of accuracy nearly identical to that of Case 1, as expected.

Input files

ale_rotmat_81x81.inp	Case 1.
ale_rotmesh_81x81.inp	Case 2.
ale_rottemp_81x81.inp	Data file containing the nodal temperatures that is read by the two files above.
ale_rotmat_41x41.inp	Case 1 for a smaller mesh.
ale_rotmesh_41x41.inp	Case 2 for a smaller mesh.
ale_rottemp_41x41.inp	Data file containing the nodal temperatures that is read by the two files above.

Figures

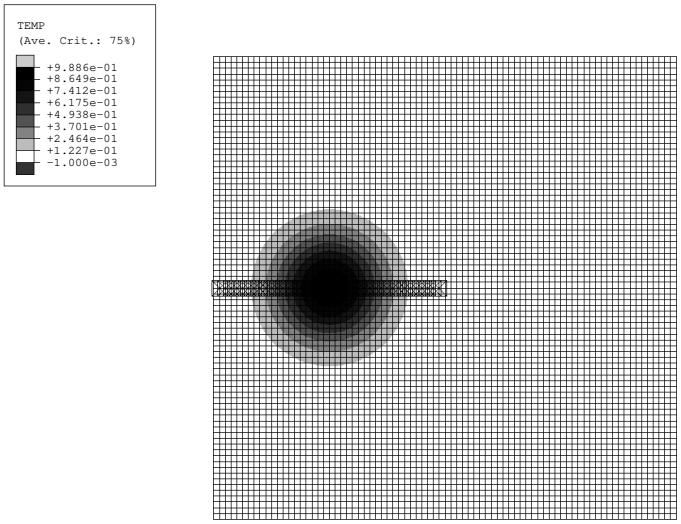


Figure 1: Initial configuration and adiabatic temperature distribution for Cases 1 and 2.

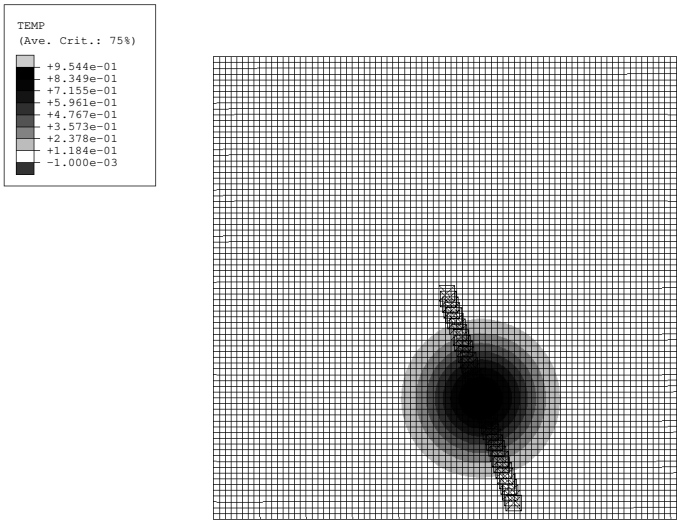


Figure 2: Configuration and adiabatic temperature distribution at $t =$ 0.0015 s for Case 1.

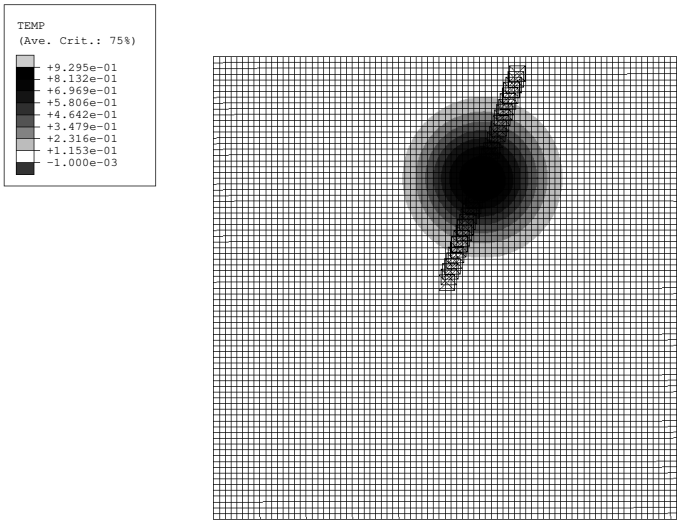


Figure 3: Configuration and adiabatic temperature distribution at $t = 0.0035$ s for Case 1.

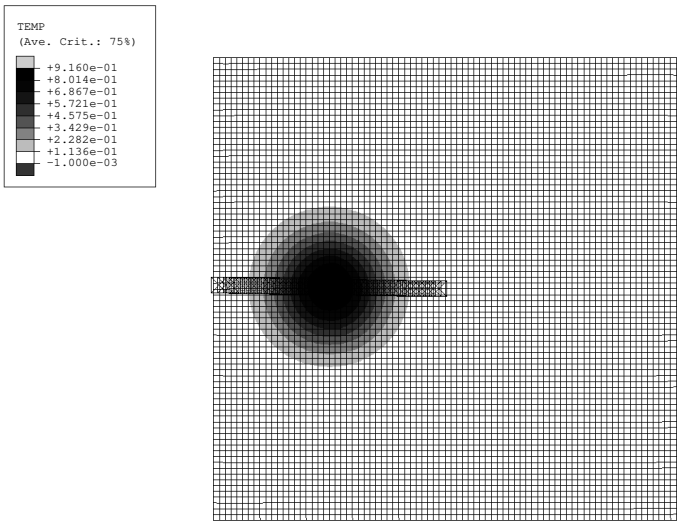


Figure 4: Configuration and adiabatic temperature distribution at the end of the simulation for Case 1.

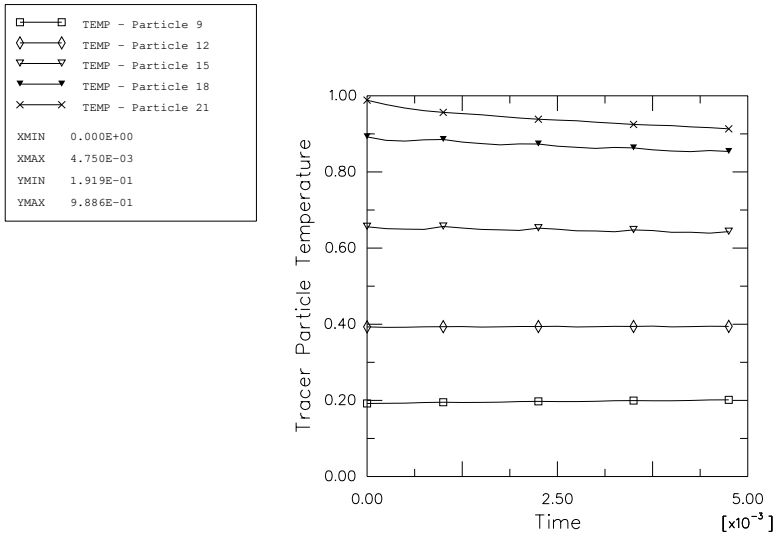


Figure 5: Time histories of adiabatic temperature at selected tracer particles for Case 1.

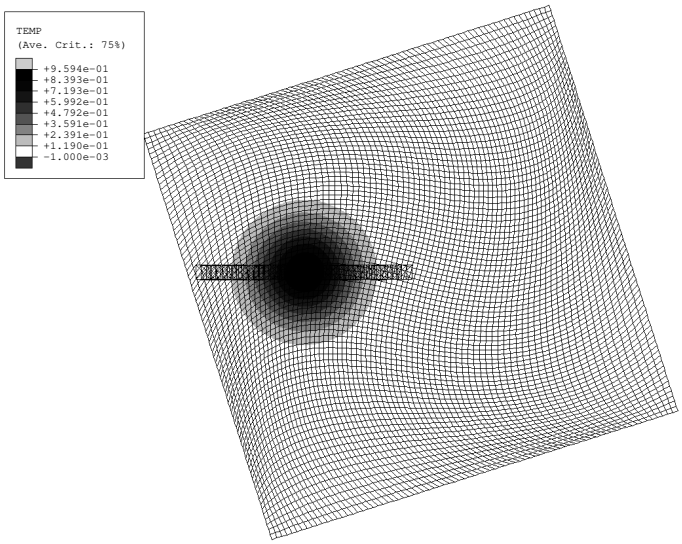


Figure 6: Configuration and adiabatic temperature distribution at $t = 0.0015$ s for Case 2.

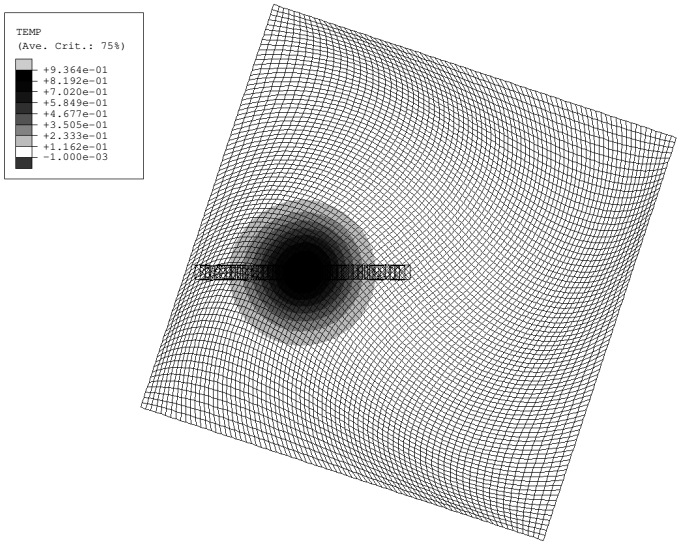


Figure 7: Configuration and adiabatic temperature distribution at $t =$ 0.0035 s for Case 2.

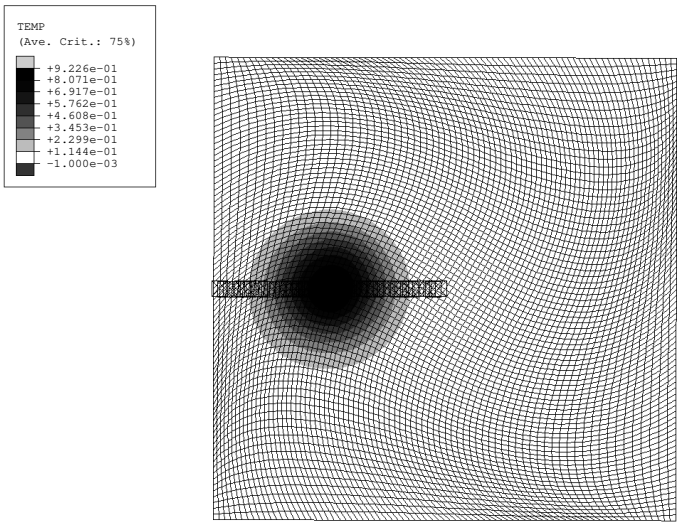


Figure 8: Configuration and adiabatic temperature distribution at the end of the simulation for Case 2.

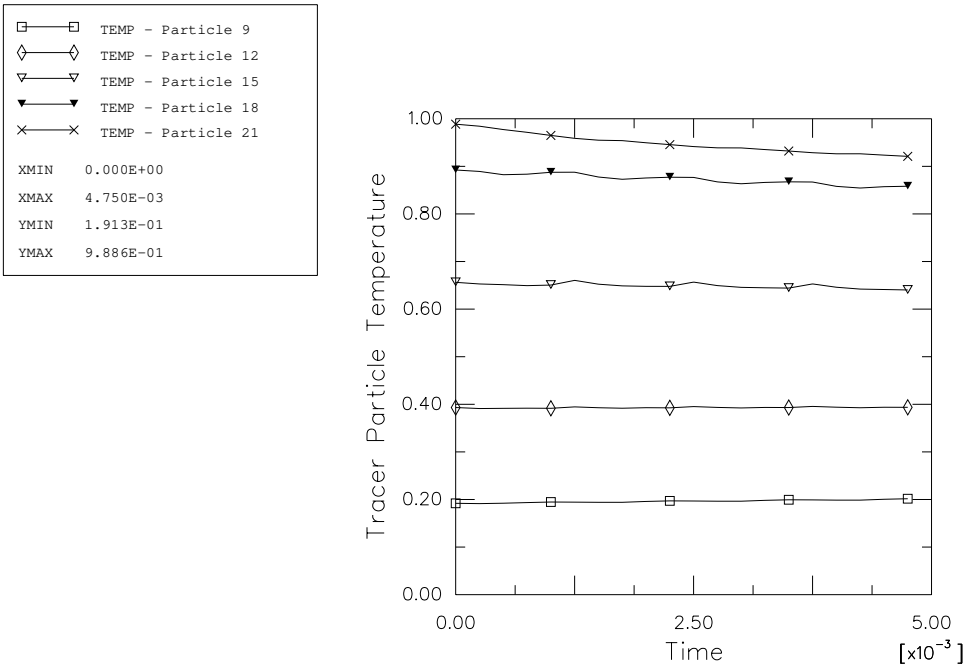


Figure 9: Time histories of adiabatic temperature at selected tracer particles for Case 2.

Water sloshing in a pitching tank

Products: Abaqus/Explicit

Overview

This example verifies the use of adaptive meshing to solve basic fluid-structure interaction problems involving sloshing.

A forced oscillation problem is solved. Results are compared to a numerically obtained reference solution given by Nakayama and Washizu (1980).

Problem description

The model geometry for the problem is shown in *Figure 1*. A rigid tank is partially filled with water to a height of 60 cm. The tank measures $90 \times 80 \times 60$ cm and is initially inclined 0.8 degrees with respect to the y-axis. The tank is rotated about a point at the water surface midway between the two tank walls. The rotation is prescribed as

$$\theta = 0.01396 \cos 5.5t,$$

which gives rise to a sinusoidal pitching oscillation of the water in the tank. Although there are physical stiffness contributions from surface tension forces in the water, these effects are minimal compared to the fluid-structural coupling effects and, consequently, are not modeled.

The finite element model is shown in *Figure 2*. The water is modeled with CPE4R elements, and the rigid tank is modeled with R2D2 elements. Frictionless contact is defined between the water and the tank. The motion of the tank is prescribed at the rigid body reference node, which is located on the axis of rotation. Gravity loading is defined for the water. Consequently, an initial geostatic stress field is defined to equilibrate the stresses caused by the self-weight of the water. The sloshing analysis is performed for 14 seconds, although adaptive meshing allows an analysis such as this to be carried out much further.

For this class of 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. In this problem the Newtonian viscous deviatoric model is used, and the shear viscosity specified is on the order of an equivalent shear modulus, calculated as mentioned earlier, scaled by the expected stable time increment. This keeps the deviatoric shear stresses several orders of magnitude lower than the pressure in the water.

In addition, when 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 strength 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.

For this problem the linear $U_s - U_p$ equation of state is used with a wave speed of 45 m/s and density of 983.2 kg/m^3. This wave speed corresponds to a bulk modulus of 2.07 MPa (three orders of magnitude less than the actual bulk modulus of water, 2.07 GPa). The shear viscosity is chosen as 13E−4 Pa sec.

Adaptive meshing

A single adaptive mesh domain that incorporates the water is used for each model. A sliding boundary region is used for the contact surface definition on the water (the default). Because of the large amounts of shearing induced by the sloshing motion, the frequency and intensity of adaptive meshing must be increased to provide a smooth mesh. The frequency of adaptive meshing is reduced from the default of 10 to 5, and the number of mesh sweeps to be performed in each adaptive mesh increment is increased from the default of 1 to 3.

Results and discussion

Figure 3 shows the deformed mesh configuration at various times. As the figure shows, there is significant sloshing of the water. The use of adaptive meshing acts to prevent excessive element distortion from occurring over time. *Figure 4* shows a time history of the water displacement in the global *y*-direction at the extreme right end of the free surface (node 275 highlighted in *Figure 3*). Results are compared to the reference solution reported in Nakayama and Washizu (1980) and are seen to be similar, even though the approaches are quite different. A well-known, nonlinear characteristic of the fluid motion is observed: the upward wave amplitude becomes greater than the downward amplitude as the wave amplitude becomes larger.

Input files

ale_water_oscillation.inp	Input data for this analysis.

References

- Nakayama, T., and K. Washizu, “Nonlinear Analysis of Liquid Motion in a Container Subjected to Forced Pitching Oscillation,” *International Journal for Numerical Methods in Engineering*, vol. 15, pp. 1207–1220, 1980.

Figures

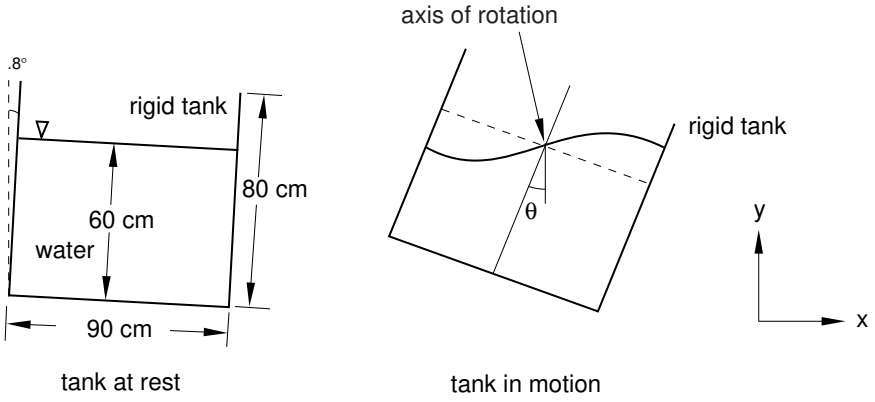


Figure 1: Model geometry.

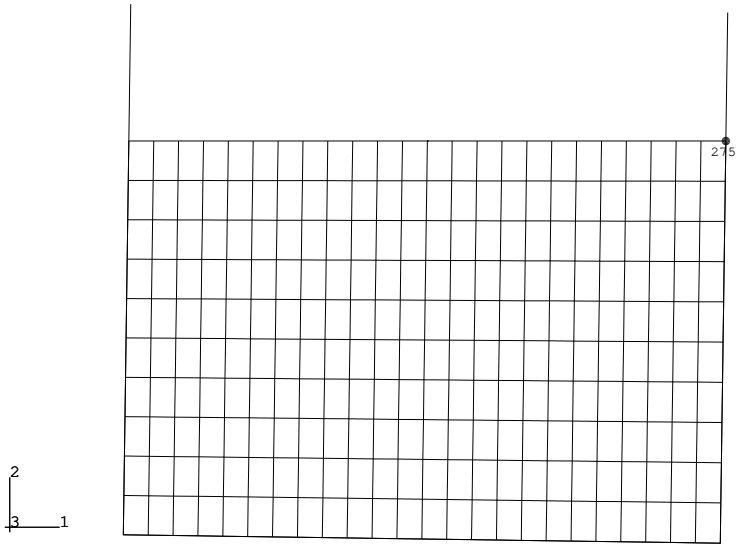


Figure 2: Finite element model.

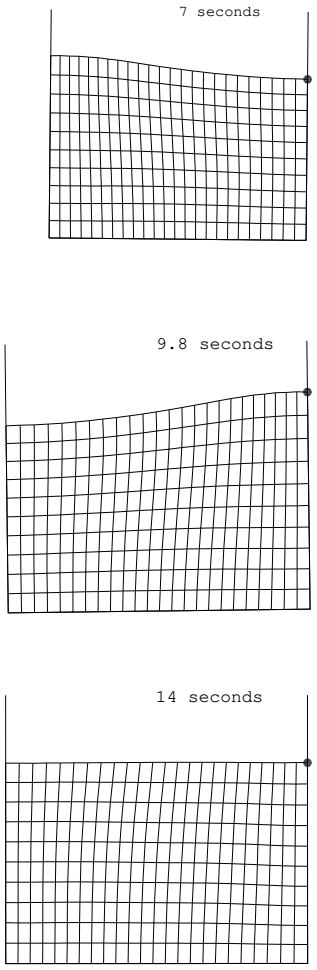


Figure 3: Deformed configurations at various times.

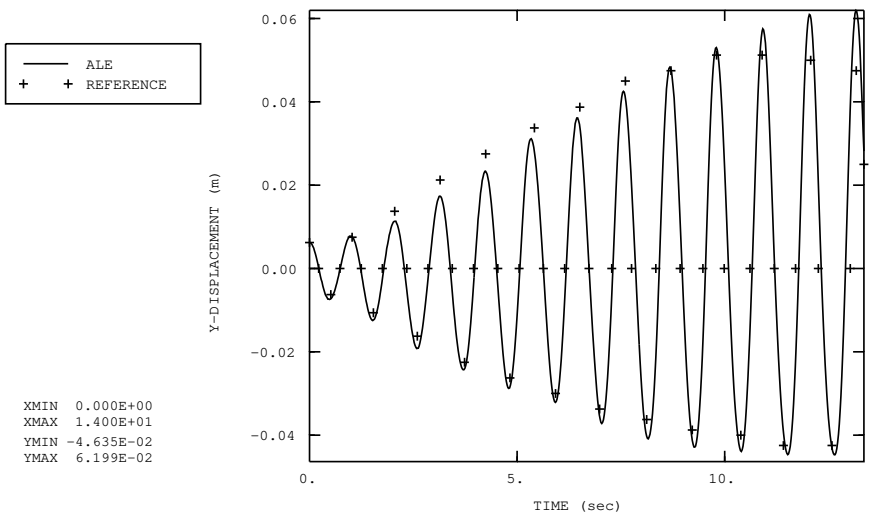


Figure 4: Time history of the y-displacement at the right end of the free surface.

Abaqus/Aqua analysis

In this section:

- *Pull-in of a pipeline lying directly on the seafloor*
- *Near bottom pipeline pull-in and tow*
- *Slender pipe subject to drag: the “reed in the wind”*

Pull-in of a pipeline lying directly on the seafloor

Products: Abaqus/Standard Abaqus/Aqua

Overview

This example verifies the use of the anisotropic friction model in Abaqus to simulate the pull-in of a pipeline.

One problem encountered in offshore pipeline installation is the response of a pipeline lying directly on the seabed and being moved by winching one end or the other toward a point. Since the pipeline lies directly on the ground or seabed, frictional effects are important. Field measurements suggest that, for pipes lying directly on the seabed, the resistance to motion per unit length (the friction coefficient) is higher for motion transverse to the pipe than for motion parallel to the pipe. Abaqus allows the modeling of this effect through its anisotropic friction option. This is a Coulomb friction model with different friction coefficients for motion transverse and parallel to the pipe. A stiffness method is used in this model; and, by default, Abaqus will choose an "elastic slip" to occur during sticking. The value of the elastic slip is chosen as a small fraction of the average interface element length in the model. Alternatively, the user has the option of specifying the magnitude of the elastic slip to occur during sticking friction. A reasonably small elastic slip value should be specified for appropriate frictional interface behavior. Too small a value will result in excessive iteration. A reasonable value to choose for this elastic slip is a typical small model dimension—for example, the diameter of the beam. Friction coefficients are usually taken from field data. We choose the user-specified elastic slip approach in this case since the average interface element length in the model considers only beam element lengths and not their diameters.

A spherical gap element is used to model a weightless cable attached between the end of the pipeline and a fixed point and is used to winch the pipeline into the fixed point. The length of the cable is then specified as a function of time by specifying contact interference. The cable will only carry tension: if the force in the cable becomes compressive, the cable goes slack and remains slack until the relative positions of the two points is such that the cable will again carry tension. This slack cable concept allows any cable to be made inactive at any time by specifying it to have a very large length.

Problem description

A pipeline 228.6 m (750 ft) long, with outer diameter 254 mm (10 in) and wall thickness 25.4 mm (1 in) is initially straight and stress-free and is assumed to lie on a hard seafloor. One end of the pipeline is attached through a weightless cable to a fixed point, *A*, which is offset from the pipeline end as shown in *Figure 1*. The cable is shortened gradually to simulate a winching process during which the end of the pipeline is pulled toward *A*. The anisotropic seabed friction capability discussed above is used to model the pipe-seabed interaction; the transverse friction coefficient is 1.0 and the tangential friction coefficient is 0.6. The motion is assumed to be in the plane of the model only. Fifteen hybrid beam elements (type B31H) are used to model the pipe. The hybrid beam elements are specifically formulated for modeling very slender beams and are generally recommended for this type of pipeline modeling.

Seafloor modeling

The contact between the pipeline and the seafloor is modeled with a contact pair. The seafloor is modeled as an analytical rigid surface. It is an infinite rigid plane perpendicular to the global z-axis, and the pull-in motion occurs on that plane.

The mechanical interaction between the surface of the pipeline and the seabed is assumed to be anisotropic frictional contact, as discussed previously, with "softened" contact (a nonlinear pressure-clearance surface interaction model). For this class of problem, such a mechanical interaction model is often more realistic than the default assumption of perfectly hard surfaces.

Since the pull-in motion is assumed to take place solely in the $z =$ constant plane, the pipeline is defined to lie a distance of 0.2642 m (0.861 ft) above the seabed and is constrained from motion in the vertical direction. This results in a pressure between the pipe and seabed of 875.63 N/m (60.32 lb/ft). This constraint of the pipe in the direction of the normal to the rigid surface is possible only because we use a softened contact. The default hard contact introduces this constraint automatically whenever a secondary node lies on the rigid surface.

Material

The pipeline is constructed of steel, with a Young's modulus of 206.8 GPa (4.32×10^9 lb/ft^2) and a shear modulus of 103.4 GPa (2.16×10^9 lb/ft^2). The material response is assumed to be elastic, so a general beam section is used to specify the pipe section description. This avoids numerical integration of the beam section. A beam section with numerical integration, would be needed if material nonlinearity must be considered.

Boundary conditions

Point *A*, the point toward which the pipeline is winched, is restrained. The vertical displacement of the pipeline is restrained, as well as the rotations about the *x*- and *y*-directions.

Incrementation

The automatic incrementation option is used to obtain the response history. Since the solution involves friction, it is path dependent; hence, a reasonably large number of increments are required to ensure that the solution follows the actual response path closely. For this reason an upper limit on the increment size of 0.1 of the total pull-in is specified.

Results and discussion

The configuration of the pipeline at the completion of pull-in is shown in *Figure 2*. The figure shows that only part of the pipeline is affected by the pull-in. This is a consequence of the values chosen for the coefficients of friction and the flexibility of the pipe. For lower friction coefficients (or a stiffer pipe), more of the pipeline will be moved by the winching process.

Input files

pipelinepullin.inp	Input data for this example.

Figures

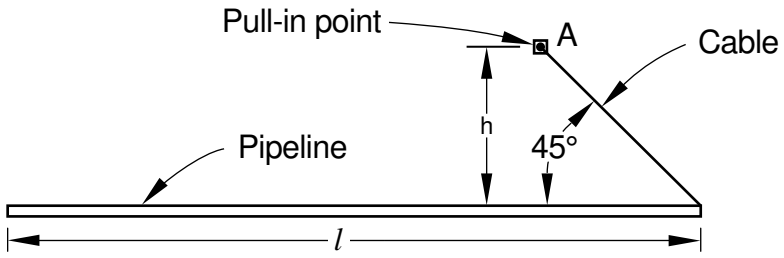


Figure 1: Pipeline pull-in on a frictional seabed.

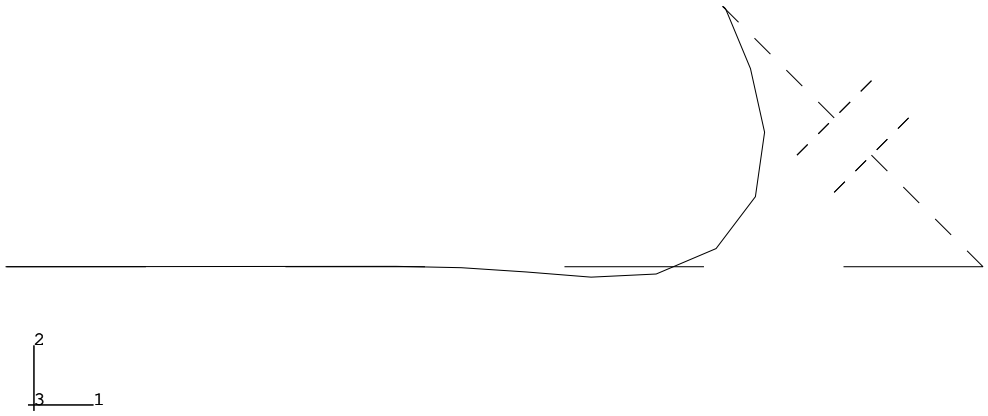


Figure 2: Final pipeline configuration—anisotropic friction.

Near bottom pipeline pull-in and tow

Products: Abaqus/Standard Abaqus/Aqua

Overview

This example verifies that Abaqus correctly simulates the "near-bottom bending" method for installing a pipeline on the seabed floor and the subsequent towing process.

In the "near-bottom bending" approach chains, typically 5–10 m (20–30 ft) long, are attached to the pipeline at intervals along its length. Their weight then balances the buoyancy devices, which are attached to the pipeline, when the pipeline is lowered to a position about 3 m (10 ft) from the seabed. The pipeline is then winched into position at each end, with the lengths of chain lying on the seabed acting as restraints on the motion and, thus, providing some control over the process. One of the analyses in this example is the prediction of configuration and stress in the pipeline throughout such a "pull-in" process, the usual concern being to accomplish a satisfactory final configuration without buckling or overstressing the pipeline at any time during the installation. As a second near-bottom pipeline installation example, the cable is assumed to remain constant in length and motion is prescribed on the unattached end, thereby simulating a towing process.

During the pull-in or towing process the chains typically take the configuration shown in *Figure 1*: a catenary between the attachment point and the seabed, with some length along the seabed (this part of the chain may not lie in a straight line along the seabed: its configuration depends on the previous motion). In Abaqus this is idealized as a single anchor block on the seabed, connected to the attachment point by a catenary (*Figure 2*). When two-dimensional drag chains are used, the model requires the specification of two parameters: the horizontal distance, l_s, between the attachment point and the anchor block when the system is slipping (that is, the maximum possible horizontal distance between these points, since the horizontal force is limited by friction) and this maximum frictional force. Typically, l_s is chosen as the horizontal distance between the attachment point and a point halfway along the horizontal chain lying on the seabed, while the maximum frictional force is $\mu w l_0$, where μ is the friction coefficient, w is the weight of the chain per unit length (in water), and l_0 is the length of chain on the seabed in the actual configuration. The three-dimensional drag chains can also be used. In this case the model requires the specification of three parameters: the total length of the chain, the friction coefficient, and the weight per unit length of chain. The total length of the chain is the sum of the length of the chain on the seabed and the suspended length. In addition, for three-dimensional analyses the seabed must be defined using a rigid surface, which must be flat and parallel to the global X–Y plane.

This idealization of the drag chains is usually satisfactory for motions several times larger than typical lengths associated with these chains (l_0). For small motions (of order l_0) the model is too idealized, and the chain must itself be modeled. Since the majority of installation procedures involve considerable motion, the model is usually adequate.

Problem description

The example used here to illustrate the process consists of a pipe of length 304.8 m (1000 ft), with an outer diameter of 228.6 mm (0.75 ft) and a wall thickness of 7.62 mm (0.025 ft). This is a rather slender beam, and for this reason hybrid beam element type B23H is chosen. (The "hybrid" beams are mixed formulation elements designed for use with very slender or very stiff systems.) One end of the pipeline is winched into an anchor point that is initially offset 121.92 m (400 ft) to one side of the pipeline and set back 91.44 m (300 ft) from the end of the pipeline. The other end of the pipeline is assumed to be built in—that is, already fully attached to some rigid fixture, as indicated in *Figure 3*.

There are six equally spaced drag chains on the pipeline, and so for convenience five elements are used to idealize the pipeline. Drag chains are attached to the nodes. The chain at the end being winched has a mean length at

slip, l_s, of 7.62 m (25 ft) and requires a force of 556 N (125 lb) to slip. The other chains are all of equal size, with a mean length at slip of 1.524 m (5 ft) and a slip force of 111 N (25 lb).

For comparison purposes the analysis is also performed using three-dimensional drag chains. For this case hybrid beam element type B33H is used, and the z-displacements at all of the beam nodes are restrained to reproduce the two-dimensional case. The equivalent three-dimensional parameters are obtained based on the description outlined in *Drag Chains*. The chain at the end of the pipe will have a total length l_c of 39.9 m (131 ft), a friction coefficient μ of 0.3, and a weight per unit length of 58.2 N/m (4.0 lb/ft). The remaining chains have a total length l_c of 7.98 m (26.2 ft), a friction coefficient μ of 0.1, and a weight per unit length of 1455 N/m (100 lb/ft). The height of the beam above the seabed, h, is 3.05 m (10 ft). A cylindrical analytical surface with a fixed reference node is used to simulate the seabed. The reference node is used as the second node of the DRAG3D element to associate the drag chains with the seabed.

The cable is modeled as a spherical gap element, which provides for an inextensible cable supporting tension but no compression. The length of the cable can be changed throughout a step by using a contact interference. This feature is used here to reduce the length to zero over the step and, thus, effect the pull-in.

Material

The pipeline is made of steel, with a Young's modulus of 206.8 GPa (4.32×10^9 lb/ft^2). Since the material response is assumed to remain elastic throughout the process, a general beam section is used: with this section Abaqus integrates the elastic section response exactly. If nonlinear material response is involved, numerical integration of the section is required; hence, a beam section should be used instead.

Boundary conditions

For the pull-in analysis the left-hand end of the pipe is assumed to be held rigidly, including full rotational restraint. For the three-dimensional analysis the beam nodes are also restrained in the z-direction to simulate the two-dimensional case. The rigid surface reference node is fully restrained in all six degrees of freedom. The anchor point node is restrained in all directions in this case, since the pull-in is toward a fixed point.

For the near-bottom tow analysis the pipeline is unrestrained for the analysis using DRAG2D elements; however, when DRAG3D elements are used, the pipeline is restrained in the z-direction as described above. The tow is up the y-axis: the anchor point is fixed in the x-direction, and a motion of 304.8 m (1000 ft) is prescribed in the y-direction. This implies that the pipeline has no restraint (and is, therefore, singular) until the drag chain extends sufficiently to stabilize the pipeline. To overcome numerical difficulties in the early stages of the analysis, soft springs are attached to two pipeline nodes. When the system is no longer singular, the solution proceeds smoothly, with the automatic time incrementation algorithm controlling the increment size.

Results and discussion

Both the two-dimensional and three-dimensional drag chain elements produce the same response. The results for the two examples studied are discussed below.

Pull-in analysis

The configuration of the pipeline at the end of the pull-in analysis is shown in *Figure 4*. It is interesting to notice some swiveling of the pipeline: part of the pipeline moves in the negative y-direction, by as much as 12.2 m (40 ft). Presumably this occurs because of the direction of pull-in and the pointwise resistance to motion provided by the drag chains. It is instructive to contrast this with the results of *Pull-in of a pipeline lying directly on the seafloor*, where the pull-in of a pipeline lying directly on the seabed is simulated.

Towing analysis

The simulation is complicated in this case by the lack of restraint on the pipeline in its initial configuration. At the start of the analysis, as point *A* (see *Figure 3*) moves in the positive *y*-direction, the pipeline moves slightly in the negative *x*-direction, and the parts of the pipeline farthest from the cable also move in the negative *y*-direction. Then, as the analysis proceeds, the pipeline straightens out, taking on the configuration shown in *Figure 5* at the end of the prescribed towing motion.

Actual installation processes usually involve considerably more complex winching and alignment histories than those shown here. Such complex histories can be simulated in a series of steps, each specifying a phase of the installation.

Input files

nearbottompipeline_pullin.inp	Pull-in simulation using DRAG2D elements.
nearbottompipeline_tow.inp	Towing analysis using DRAG2D elements.
nearbottompipeline_pullin_3d.inp	Pull-in simulation using DRAG3D elements.
nearbottompipeline_tow_3d.inp	Towing analysis using DRAG3D elements.

Figures

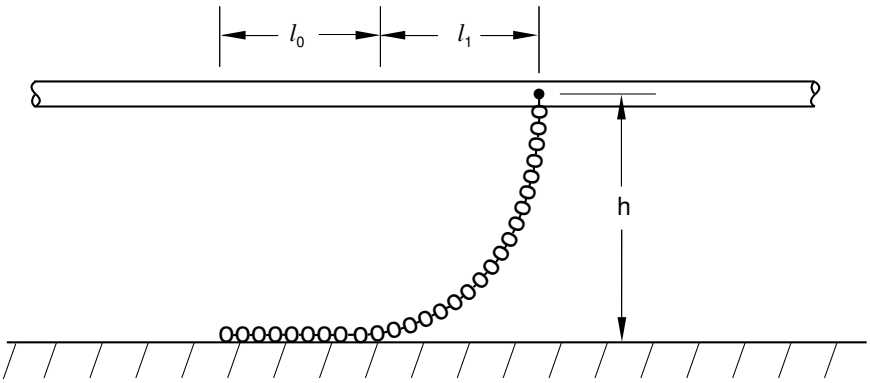


Figure 1: Actual drag chain.

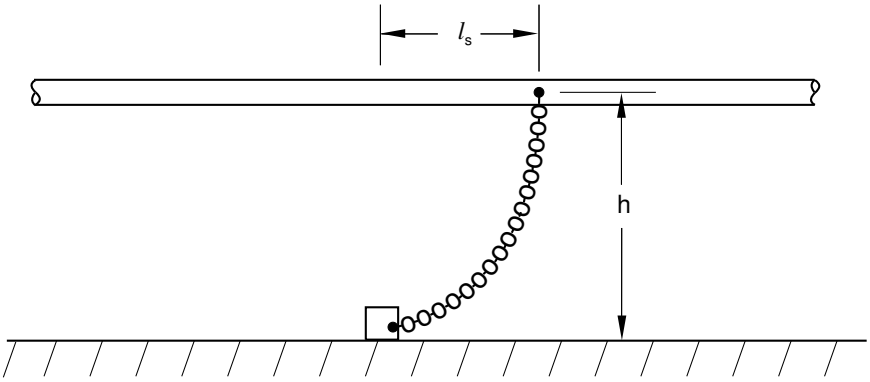


Figure 2: Drag chain model.

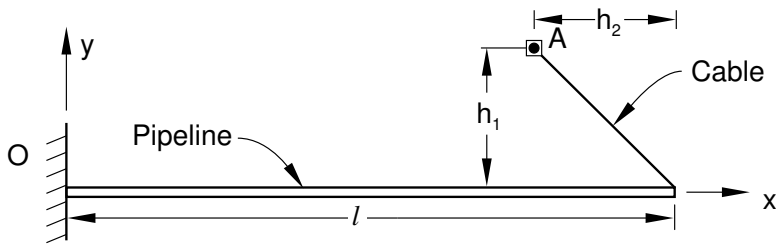


Figure 3: Pipeline pull-in and towing problems (boundary condition only for the pull-in analysis).

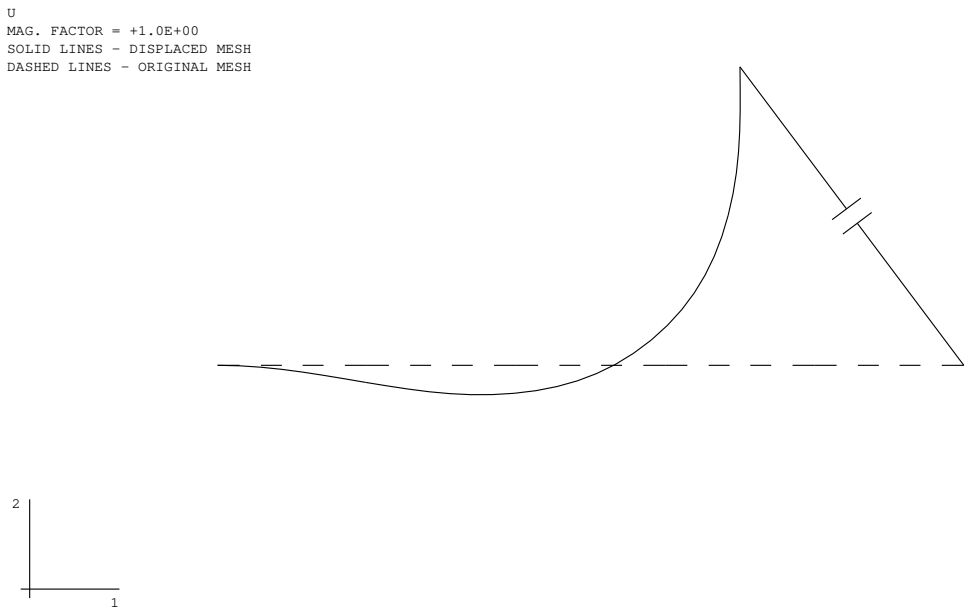


Figure 4: Final configuration—pipeline pull-in, drag chains.

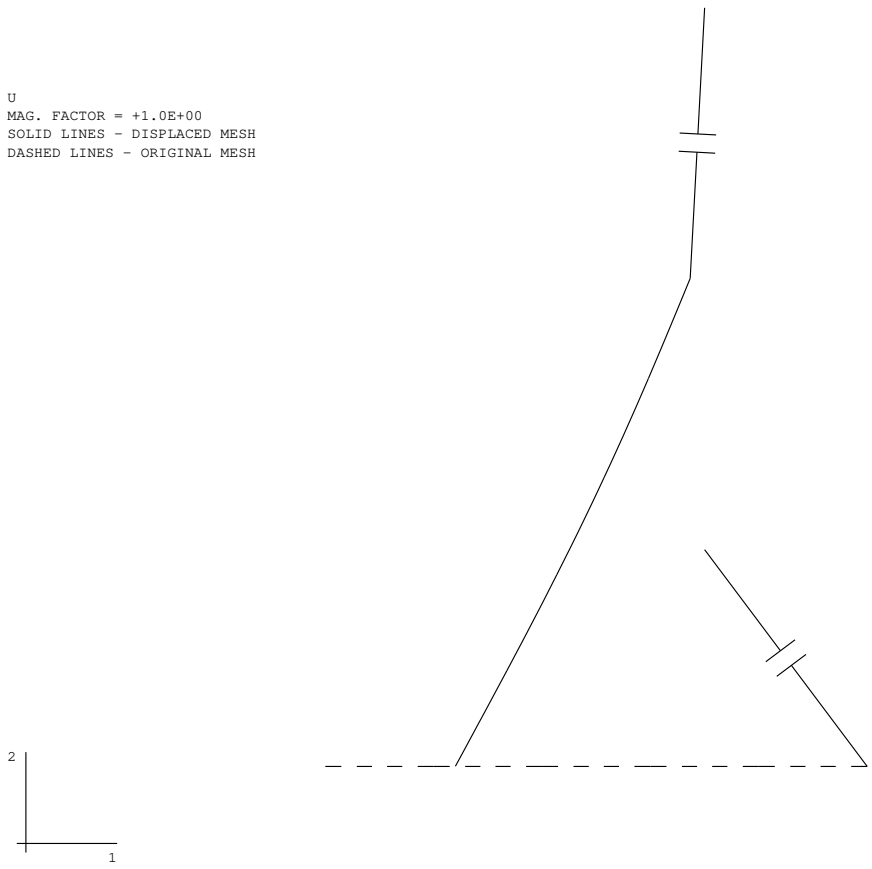


Figure 5: Final configuration—pipeline tow with drag chains.

Slender pipe subject to drag: the “reed in the wind”

Products: Abaqus/Standard Abaqus/Aqua

Overview

This example verifies that Abaqus correctly models the effect of drag loading on a slender pipeline on the seabed floor.

Currents flowing past a pipeline result in drag loading, which must be accounted for in designing restraints and moorings for the pipeline. Drag loadings vary approximately as the square of the relative normal velocity (difference between pipe and fluid velocities), and effects can be dramatic, as illustrated in the present example. Numerically drag loading results in an unsymmetric load stiffness contribution, so the resulting finite element equations are also unsymmetric.

The equation of Morison et al. (1950) is used in Abaqus to account for drag loading. The total drag force is divided into tangential, transverse, inertia, and lift contributions. The first two of these relate the drag force to the square of relative velocity, through the experimentally determined tangential and transverse drag coefficients, respectively. The inertia drag force is an “added mass” contribution based on the relative acceleration. The lift term is relevant when the pipeline lies on the seafloor or near a platform, so that the current velocity varies across the pipe. For details of Morison's drag formulation, see *Drag, inertia, and buoyancy loading*.

Problem description

A pipe 304.8 m (1000 ft) long, initially straight and stress-free, is loaded by a uniform cross-current of 0.762 m/s (2.5 ft/sec) as shown in *Figure 1*. The outside radius of the pipe is 76.2 mm (0.25 ft), and its wall thickness is 15.24 mm (0.05 ft), so the pipe is “slender” and virtually inextensible (its extensional stiffness is much greater than its flexural stiffness). Abaqus provides a family of two-dimensional and three-dimensional hybrid beam elements designed specifically for use in such cases. In this case five B23H elements are used to model the pipeline.

The pipe is made of a linear elastic material, with a Young's modulus of 206.8 GPa (4.32×10^9 lb/ft^2). A general beam section is used for the pipe section specification. This choice avoids numerical integration of the pipe section during the analysis and, hence, reduces computer costs. Numerical integration over the beam section would be necessary if plasticity or other material nonlinearities were to be considered.

The loading on the pipe results from a uniform current flowing in the global y-direction, as shown in *Figure 1*. The current velocity is 0.762 m/s (2.5 ft/s), and the density of the water is 1000 kg/m^3 (1.94 slug/ft^3). The transverse drag coefficient is 1.2, and the tangential drag coefficient is 0.002. The large difference between these coefficients reflects the different effects transverse and tangential flow have on the pipeline, the tangential drag being primarily a skin friction effect only. The pipeline is assumed to be built-in at one end, as shown in *Figure 1*.

Results and discussion

The final pipeline configuration resulting from the current induced drag loads is shown in *Figure 2*. The dramatic effect of the drag loads is evident. It is typical of such monotonically loaded, geometrically nonlinear problems to require smaller load increments for convergence at early stages of the analysis, and the load increments can increase as the analysis progresses. This is because the system is most flexible in its original configuration. By reducing load steps when convergence is difficult, the automatic loading option in Abaqus avoids user experimentation to obtain convergent solutions. This greatly eases the task of obtaining solutions and generally reduces the computer costs of generating such solutions. For this reason automatic incrementation is recommended for problems of this type.

Input files

slenderpipedrag.inp	Input data used in this analysis.

References

- Morison, J. R., L. W. Johnson, M. P. O'Brien, and S. A. Schaaf, “The Force Exerted by Surface Waves on Piles,” *Transactions, American Institute of Mining, Metallurgical and Petroleum Engineers, Inc.*, vol. 189, pp. 149–154, 1950.

Figures

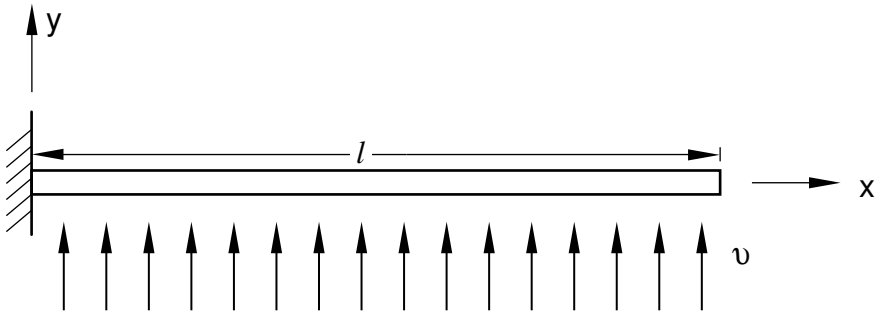


Figure 1: Slender pipe subject to drag.

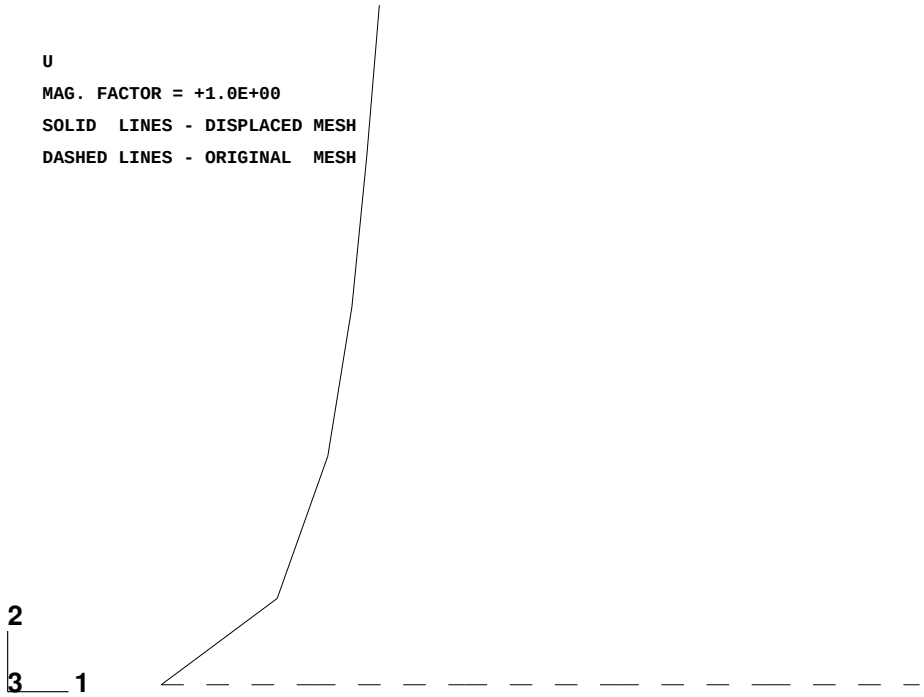


Figure 2: Displaced configuration of pipe subject to drag.

Underwater shock analysis

These benchmark problems validate that Abaqus can accurately simulate the response of submerged structures of simple geometric shapes to various underwater explosions.

In this section:

- *One-dimensional underwater shock analysis*
- *The submerged sphere problem*
- *The submerged infinite cylinder problem*
- *The one-dimensional cavitation problem*
- *Plate response to a planar exponentially decaying shock wave*
- *Cylindrical shell response to a planar step shock wave*
- *Cylindrical shell response to a planar exponentially decaying shock wave*
- *Spherical shell response to a planar step wave*
- *Spherical shell response to a planar exponentially decaying wave*
- *Spherical shell response to a spherical exponentially decaying wave*
- *Air-backed coupled plate response to a planar exponentially decaying wave*
- *Water-backed coupled plate response to a planar exponentially decaying wave*
- *Coupled cylindrical shell response to a planar step wave*
- *Coupled spherical shell response to a planar step wave*
- *Fluid-filled spherical shell response to a planar step wave*
- *Response of beam elements to a planar wave*

One-dimensional underwater shock analysis

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example verifies the underwater shock analysis capability in Abaqus, which uses incident wave loading. The coupling is accomplished by using a tie constraint in which Abaqus calculates both the structural response and the fluid pressure response at the interaction surface.

Underwater shock impinging on a one-dimensional continuum

Problem description

A one–dimensional continuum model is analyzed as described below.

Model description

The Abaqus model (*Figure 1*) consists of a single C3D8R continuum element constrained to deform in one dimension. The stiffness and density have been chosen so that the structure alone has a natural frequency of 1 Hz. A single acoustic element is coupled to the structural element. The continuum element has unit cross sectional area and a thickness of 10 units. Two cases are analyzed here. In the first case the acoustic element thickness is set to 0.01 units. In the second case the thickness is increased to 0.1 units. The surface of the acoustic element and the structural element are tied together. The fluid density and speed of sound are both set to 1.0. A plane wave pressure pulse is applied to the front face of the continuum element and the back face of the acoustic element using incident wave loading. The back face of the continuum element is held fixed and a plane wave absorbing boundary condition is specified on the front face of the acoustic element via surface impedance. The pressure pulse travels along the x-axis and is a step function in time. The sound source is located at (100, 0, 0) for the plane waves, and the stand-off point is located at (10, 0, 0). The analysis is run for 10 seconds. The response of the front face of the continuum element is one-dimensional and oscillatory. The analysis is performed in both Abaqus/Standard (u1d_std_c3d8r_ac3d8.inp) and Abaqus/Explicit (u1d_xpl_c3d8r_ac3d8r.inp). There is no damping applied in either the structure or the fluid model except from the radiation boundary condition. However, in Abaqus/Explicit bulk viscosity is used to introduce damping. The value of linear bulk viscosity is chosen as 0.02, and the value of quadratic bulk viscosity is chosen as 0.5.

The model data are kept the same for the restart analysis. In the initial run the loading is applied for 2 seconds. During the restart run, the initial conditions for the new dynamic step are read from the restart files. The loading is applied for 2 seconds. A restart analysis is performed for both Abaqus/Standard and Abaqus/Explicit.

Results and discussion

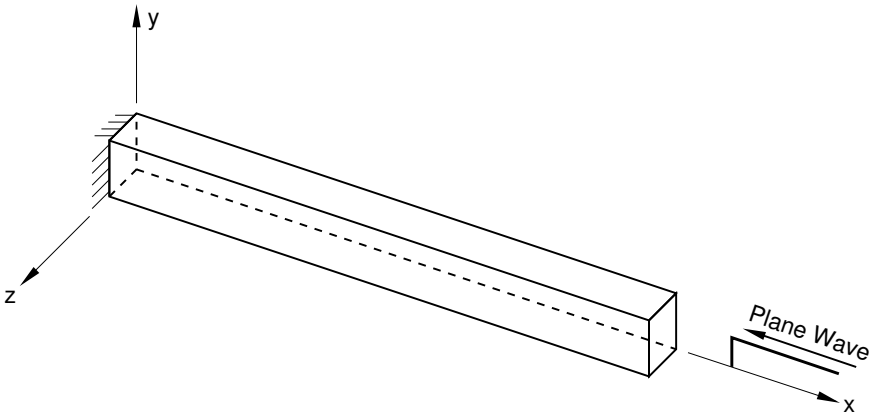
The response comparison is based on the velocity of the front face of the structure. History plots of velocity versus time for Abaqus/Standard and Abaqus/Explicit are shown in *Figure 2* and *Figure 3*. The plots are compared individually to the reference solution. Both Abaqus/Standard and Abaqus/Explicit exactly match the direct integration reference solution.

Input files

u1d_shock_ref.f	Fortran program used to generate the direct integration numerical solution for the one-dimensional continuum problem. The program uses trapezoidal integration.

u1d_std_c3d8r_ac3d8.inp	Abaqus/Standard analysis: C3D8R/AC3D8 model with acoustic element of thickness 0.01 units.
u1d_xpl_c3d8r_ac3d8r.inp	Abaqus/Explicit analysis: C3D8R/AC3D8R model with acoustic element of thickness 0.01 units.
u1d_std_c3d8r_ac3d8_init.inp	Initial Abaqus/Standard run used for restart analysis.
u1d_std_c3d8r_ac3d8_restart.inp	Restart analysis.
u1d_xpl_c3d8r_ac3d8r_init.inp	Initial Abaqus/Explicit run used for restart analysis.
u1d_xpl_c3d8r_ac3d8r_restart.inp	Restart analysis.
u1d_std_c3d8r_ac3d8_diffthick.inp	Abaqus/Standard analysis: C3D8R/AC3D8 model with acoustic element of thickness 0.1 units.
u1d_xpl_c3d8r_ac3d8r_diffthick.inp	Abaqus/Explicit analysis: C3D8R/AC3D8R model with acoustic element of thickness 0.1 units.

Figures



Solid properties		Fluid properties	
area	1.0	incompressible	
thickness	10.0	inviscid	
density	1.0	density	1.0
modulus	1973.92	sound speed	1.0
Nat. frequency	1.0		

Figure 1: One-dimensional continuum model.

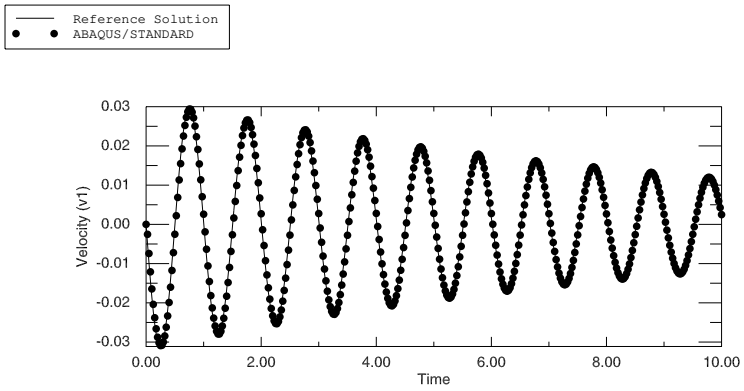


Figure 2: Comparison of Abaqus/Standard model velocity-time history with the reference solution.

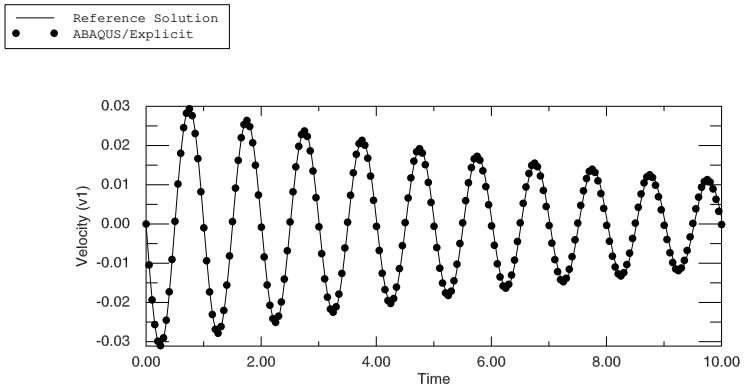


Figure 3: Comparison of Abaqus/Explicit model velocity-time history with the reference solution.

Underwater shock impinging on a one-dimensional plate

Problem description

A one-dimensional plate model is analyzed as described below.

Model description

The Abaqus model (*Figure 4*) consists of a single S4R shell element constrained to translate as a rigid body in one dimension. The planar shell is modeled in the *X*–*Y* plane with a length of 1.5 units on all sides. The shell properties are those of steel. A tie constraint is used to couple the structure to the acoustic fluid. The fluid modeled is water. A plane wave underwater pressure pulse is applied to the front face of the structural element and to the back face of the acoustic element. The source is at (0.75, 0.75, 100), and the stand-off point is at (0.75, 0.75, 0). The pulse travels along the z-axis and is a step function in time. The reaction of the plate is rigid body translation

in one dimension with constant acceleration. The response comparison is based on the velocity time history of the plate. The analysis is performed in both Abaqus/Standard (u1d_std_s4r_ac3d8.inp) and Abaqus/Explicit (u1d_xpl_s4r_ac3d8r.inp).

Results and discussion

The AbaqusS4R/AC3D8 model results are identical to the reference solution. History plots of velocity versus time for Abaqus/Standard and Abaqus/Explicit are shown in *Figure 5* and *Figure 6*.

Input files

u1d_std_s4r_ac3d8.inp	Abaqus/Standard analysis: S4R/AC3D8 model.
u1d_xpl_s4r_ac3d8r.inp	Abaqus/Explicit analysis: S4R/AC3D8R model.

Figures

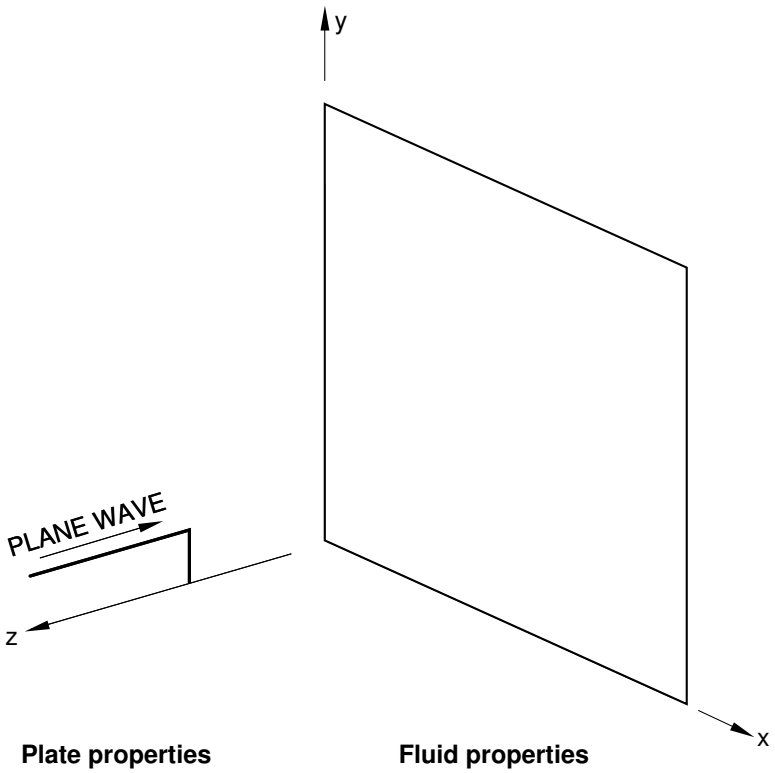


Plate properties

area	2.25
thickness	1.0
density	5.3×10^{-4}
modulus	30.0×10^{6}

Fluid properties

incompressible	
inviscid	
density	9.3×10^{-5}
sound speed	5.7×10^{4}

Boundary conditions

$u_x = 0$ $u_y = 0$ all rotations fixed

Figure 4: One-dimensional plate model.

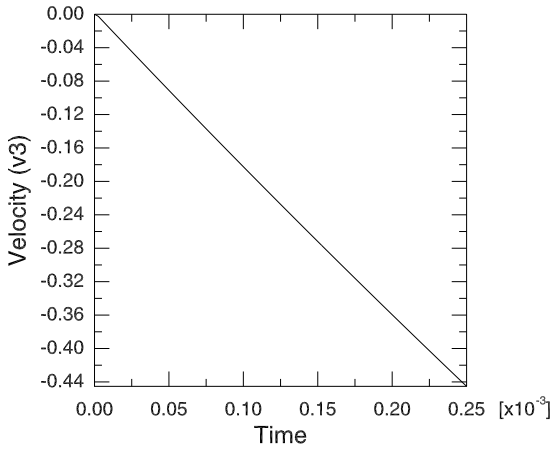


Figure 5: Abaqus/Standard model velocity-time history.

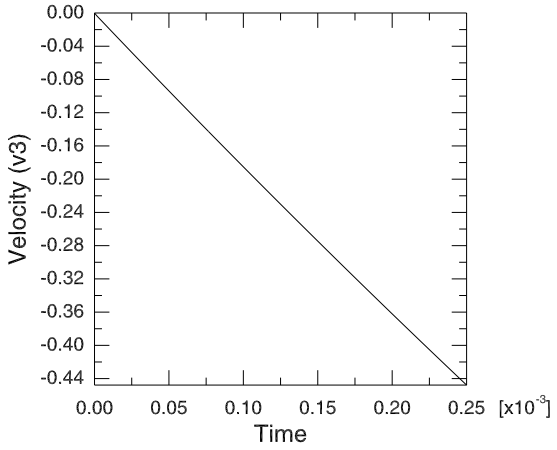


Figure 6: Abaqus/Explicit model velocity-time history.

The submerged sphere problem

Products: Abaqus/Explicit

Overview

This example verifies the use of the underwater shock simulation capabilities of Abaqus to model a submerged spherical shell structure excited by a planar sphere shock wave with a step function profile (plane wave).
The response of the submerged spherical shell has been analytically calculated by Huang (Huang and Mair, 1996). In this example the test problems are analyzed using both the total and scattered wave formulations, and the results calculated by Abaqus/Explicit are compared to analytical results.

Problem description

The quarter-sphere model is shown in *Figure 1*. The sphere is entirely submerged in a fluid, deep enough so that free surface effects are unimportant. The figure shows the sphere modeled with 4-node shell elements, enclosed in a fluid mesh modeled with acoustic fluid elements. The material properties used are based on a steel spherical shell and water as the fluid, but the properties have been nondimensionalized. Symmetry boundary conditions are imposed on the structural model. A consistent set of boundary conditions need not be given for the fluid since they are applied by default. The coupling between the structure and the fluid is enforced using a tie constraint, which does not require compatible meshes. The mesh density for the fluid is chosen such that the shock is captured accurately. The condition for non-reflection is applied on the outer surface of the fluid mesh via the predefined spherical radiation boundary condition.

An explosion occurs on the y-axis far from the structure. The incident wave loading is applied to the structure and the fluid at their interface as part of the history data, specifying the location of the charge and the standoff point. Since the front is planar, the charge location is used only to compute the direction of the incoming shock. The standoff point is chosen as the point on the structure closest to the charge, so that the simulation begins when the front is just about to impinge on the structure. The pressure profile of the shock wave measured at the charge standoff point is given by an amplitude curve. For this problem the pressure profile is a step function with a magnitude of 1.0×10^{-4}.

The sphere has been modeled with 273 S4R shell elements. The fluid has been meshed with 3250 AC3D8R acoustic fluid elements. To capture the shock front accurately, each fluid element measures 0.03 units radially, while it spans 10° in both the polar and azimuthal directions. The elements must be small enough that significant solution features, such as peaks in POR, span at least several elements. If the computed results show POR (or corresponding structural degrees of freedom) varying greatly over less than several element lengths, mesh refinement is probably required.

Results and discussion

The Abaqus models use time increments of 0.01 units and a total step time of 10 units. The results are presented as the value of the radial velocity of the leading node (nearest to the charge) and the trailing node (farthest from the charge). The Abaqus/Explicit results are shown along with the reference results in *Figure 2* and *Figure 3*. The results compare well in both cases.

Input files

us_xpl.inp	Spherical shell model, scattered wave formulation.
us_tot_xpl.inp	Spherical shell model, total wave formulation.

References

- Huang, H., and H. U. Mair, "The ROSEHIPS Program (Response Of a Spherical Elastic Shell to an Incident Plane Step pressure wave) for UNDEX Simulation Validation," *67th Shock and Vibration Symposium Proceedings*, 1996.

Figures

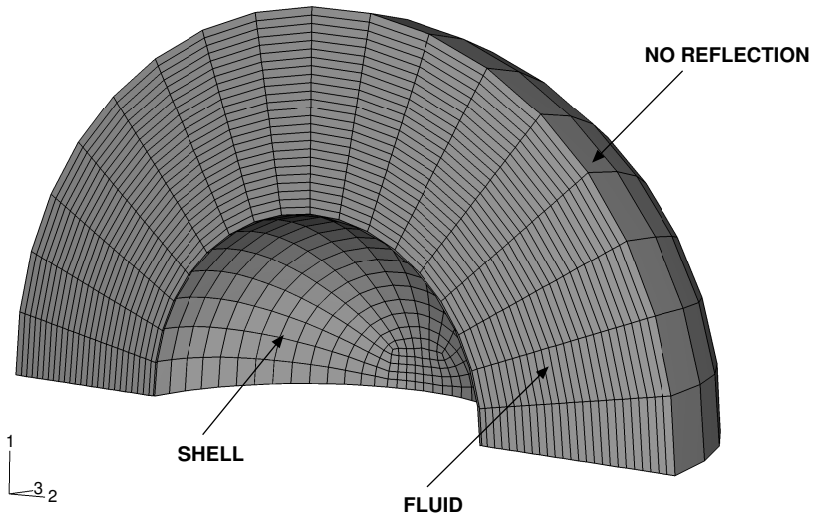


Figure 1: Quarter-sphere shell model.

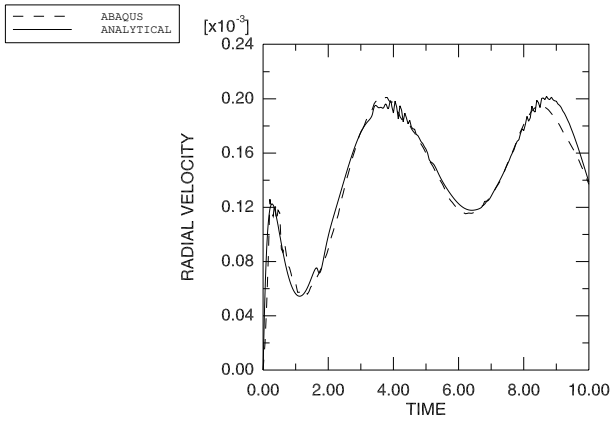


Figure 2: Leading node velocity time histories.

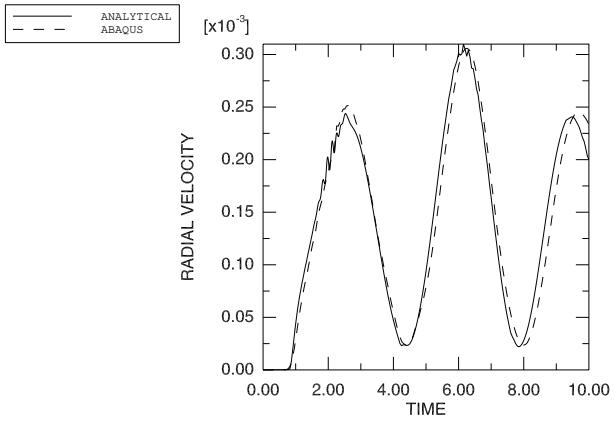


Figure 3: Trailing node velocity time histories.

The submerged infinite cylinder problem

Products: Abaqus/Explicit

Overview

This example analyzes the elastic response of an infinite cylinder excited by a planar shock wave with a step function profile (plane wave).
The response calculated by the USA-STAGS program (DeRuntz and Brogan, 1980) can be found in the paper by DeRuntz (1989). Geers (1972) has presented an analytical solution for the infinite cylinder. The problem is analyzed using both the scattered and total wave formulations, and the response calculated by Abaqus/Explicit is compared to the results given by DeRuntz (1989).

In the second part of this problem we analyze the elastic response of an infinite cylinder in a fluid with a bottom surface. The cylinder is excited by a single spherical shock wave and pressure wave reflections off the bottom surface. The response calculated by Abaqus/Explicit for this analysis is compared to that of a multiple shock wave analysis.

Infinite cylindrical shell in an infinite fluid

Problem description

The infinite cylinder excited by a plane wave is a two-dimensional problem with a single plane of symmetry. The half-model for this problem is shown in *Figure 1*. Plane strain boundary conditions are imposed along the axis of the cylinder, and symmetry boundary conditions are imposed on the x–z plane. Boundary conditions are imposed on the structural model. A consistent set of boundary conditions need not be given for the fluid since they are applied by default. The structure is enclosed in a fluid mesh that models the surrounding fluid. Two different meshes are chosen, and the results are compared. The infinite cylinder is assumed to be fully submerged in the fluid so that free surface effects are negligible. The cylindrical shell is assumed to be made of steel, and the fluid is assumed to be water; however, the material properties have been nondimensionalized. The coupling between the structure and the fluid is enforced using a tie constraint, which does not require compatible meshes. The mesh density for the fluid is chosen such that the shock is captured accurately. The condition for non-reflection is applied on the outer surface of the fluid mesh via the predefined circular radiation boundary condition.

An explosion occurs on the x-axis far from the structure. The incident wave loading is applied at the interface of the structure and the fluid as part of the history data, specifying the location of the charge and the standoff point. Since the front is planar, the charge location is used only to compute the direction of the incoming shock. The standoff point is chosen as the point on the structure closest to the charge, so that the simulation begins when the front is just about to impinge on the structure. The pressure profile of the shock wave measured at the charge standoff point is given by an amplitude curve. For this problem the pressure profile is a step function with a magnitude of 1.0×10^{-4}.

The half-cylinder is modeled with 4-node quadrilateral shell elements. In both models the structure is meshed with 36 S4R elements. The fluid meshes are different in the models displaying variations of mesh density. Fluid meshes are modeled with AC3D8R elements. The analyses are run with double precision in Abaqus/Explicit, using direct user control for the time interval. The bulk viscosities and the interval sizes have been specified to optimize the efficiency of the solution, by reducing the run time and capturing the shock accurately.

Results and discussion

The coarse models use an average mesh size of 0.035 units in the radial direction with each element spanning 10°. In the fine models the average mesh size in the radial direction is 0.02 units, and each element spans 5°. The radiating surface is 2 units away from the center of the shell structure. The results are presented as the value

of the radial velocity of the leading node (nearest to the charge, see *Figure 2*) and the trailing node (farthest from the charge, see *Figure 3*). The Abaqus results are compared with those obtained using the USA-STAGS program. There is good comparison between the codes.

Input files

uc_xpl_coarse.inp	Coarse mesh model, scattered wave formulation.
uc_xpl_fine.inp	Fine mesh model, scattered wave formulation.
uc_xpl_tot_coarse.inp	Coarse mesh model, total wave formulation.
uc_xpl_tot_fine.inp	Fine mesh model, total wave formulation.

References

- DeRuntz, J. A., Jr., Private Communication, 1990.
- DeRuntz, J. A., Jr., "The Underwater Shock Analysis Code and its Applications," *60th Shock and Vibration Symposium Proceedings*, vol. 1, pp. 89–107, 1989.
- DeRuntz, J. A., Jr., and F. A. Brogan, "Underwater Shock Analysis of Nonlinear Structures, A Reference Manual for the USA-STAGS Code (Version 3)," *DNA 5545F, Defense Nuclear Agency, Washington D.C.*, 1980.
- Geers, T. L., "Scattering of a Transient Acoustic Wave by an Elastic Cylindrical Shell," *Journal of the Acoustical Society of America*, vol. 51, no. 5 (part 2), pp. 1640–1651, 1972.

Figures

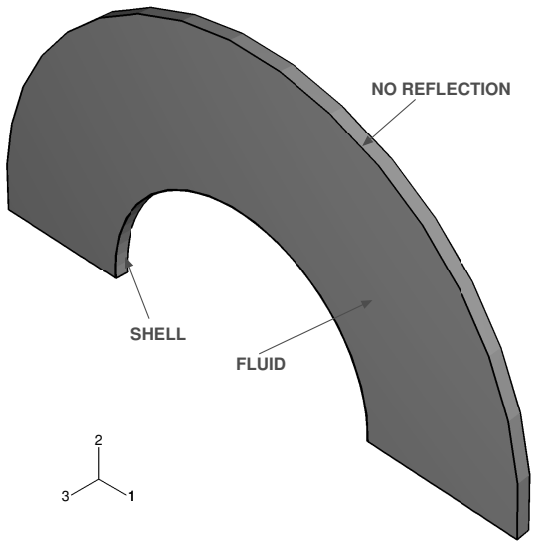


Figure 1: Cylindrical shell model.

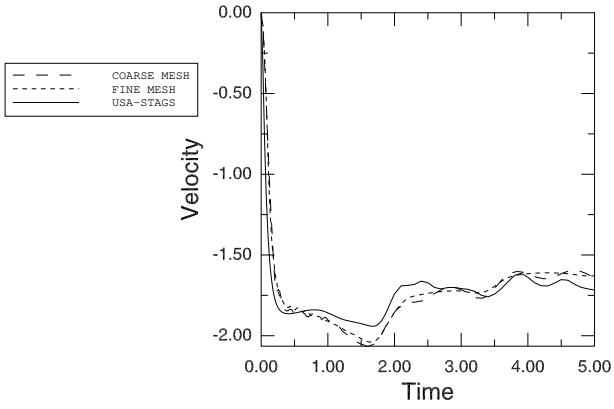


Figure 2: Leading edge velocity time histories.

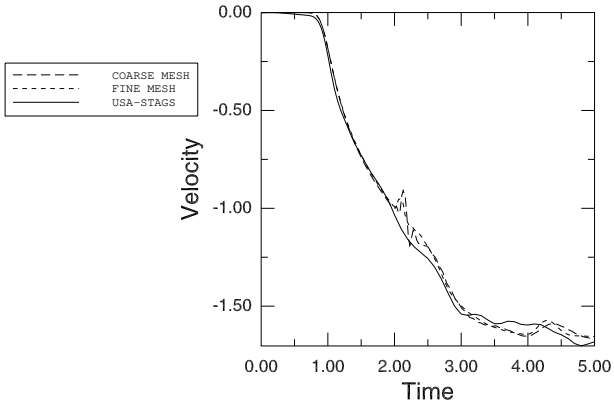


Figure 3: Trailing edge velocity time histories.

Infinite cylindrical shell with bottom reflection

Problem description

The infinite cylinder is excited by a shock wave from a charge that is relatively close to the structure. The effect of the pressure wave reflecting off the bottom surface is included. Bottom surface effects are treated in Abaqus using imaging techniques where the incident pressure wave is made up of both primary and image contributions with the appropriate time delays automatically calculated. The analysis is performed in Abaqus/Explicit using both the total and scattered wave formulations.

The cylinder geometry and fluid properties are the same as those defined in the first part of this example. While the structure is symmetric, the loading is not. Therefore, symmetry is not used in the analysis. A cross-section of the full cylinder model is shown in *Figure 4*. A single charge is located on the x-axis 10 length units away from the cylinder (z) axis. The bottom surface is located 4.6904 length units below the cylinder axis. It is assumed that 80% of the incoming wave is reflected off the “soft” bottom surface. The charge location is included as data with the incident wave property. The bottom surface location is specified using an incident wave reflection. The reflective properties of the surface are converted into equivalent impedance properties and specified using an impedance property. The model consists of 72 4-node quadrilateral shell elements and 2880 AC3D8R fluid

elements. The standoff point is placed along the *x*-axis, at the point where the structure and fluid come into contact.

A two-charge model, wherein the bottom surface is represented by a second “image” charge in addition to the original primary charge, is also considered. The image charge is located at the same *x*-direction coordinate used for the primary charge but below it by twice the distance to the bottom surface. The bottom surface definition is intentionally excluded from this model. Multiple charges are represented by multiple incident wave loading definitions within an analysis step. To specify the time delay due to reflection correctly, the distance between the image charge and its standoff point is set equal to the distance between the primary charge and its standoff point. In addition, the standoff point is located on the line joining the image charge to the center of the circle. To simulate the partial reflection requirement, the image charge magnitude is scaled by the reflection coefficient of 0.8.

Results and discussion

The bottom surface analyses, *uc_xpl_1ch.inp* and *uc_xpl_tot_1ch.inp*, use constant time increments and are conducted in a single step. The results are presented as the values of the radial and tangential velocities of the leading node (nearest to the primary charge) in *Figure 5*. The effects of the pressure wave reflecting off the bottom surface are delayed due to the longer effective standoff distance. The results of the multiple charge analyses, *uc_xpl_2ch.inp* and *uc_xpl_tot_2ch.inp*, are shown to agree exactly with the bottom surface analysis.

Input files

uc_xpl_1ch.inp	Full-cylinder model with a single charge and a bottom surface.
uc_xpl_2ch.inp	Full-cylinder model with two charges and no bottom surface.
uc_xpl_tot_1ch.inp	Full-cylinder model with a single charge, total wave formulation.
uc_xpl_tot_2ch.inp	Full-cylinder model with two charges, total wave formulation.

Figures

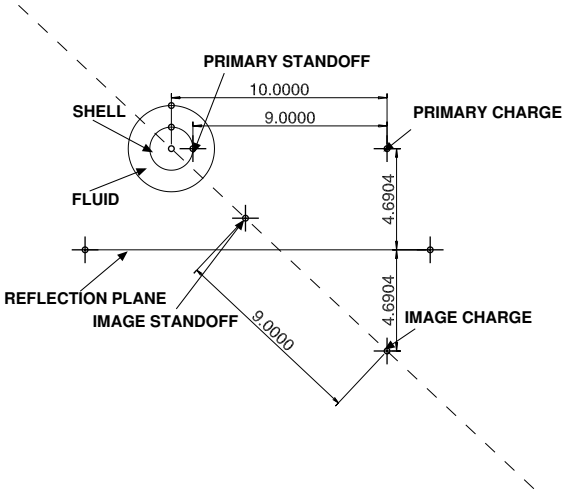


Figure 4: Full-cylinder bottom surface model.

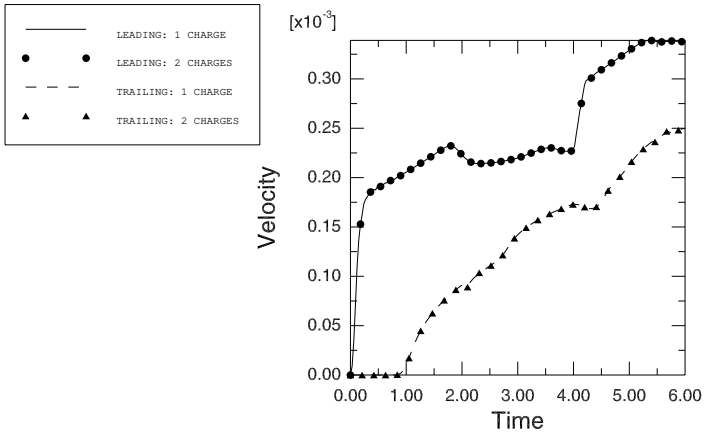


Figure 5: Velocity time history comparison.

The one-dimensional cavitation problem

Products: Abaqus/Explicit

Overview

This example illustrates the use of Abaqus/Explicit to model a one-dimensional cavitation problem. A fluid column supporting a floating mass-spring system is studied, and the results obtained using Abaqus/Explicit are compared with those obtained by Bleich and Sandler (1970) and Sprague and Geers (2001).

When an underwater explosion occurs, a compressive wave is generated. If the wave reaches the free surface of the water, the reflected wave is dilational, causing tensile stress in the water. Water cannot sustain a high value of tension and can disassociate, creating a region of cavitation that has a substantial influence on the response of marine structures. In this example the ability of Abaqus/Explicit to model this situation accurately is illustrated using a one-dimensional problem.

Problem description

A one-dimensional fluid column in a rigid pipe with a constant cross-sectional area is modeled with AC2D4R elements. At the top of the column the fluid is coupled to an idealized floating structure, represented by two vertically oriented masses (M_1 and M_2) connected by a spring with stiffness K. *Figure 1* shows a schematic of the model. The fluid-solid system is excited by a plane, upward-propagating, step-exponential wave, applied at the bottom of the fluid column. A plane wave absorbing boundary is also applied at the bottom of the column, which is exact for one-dimensional acoustic waves.

To simulate the mass M_1, one point mass element is attached to a node vertically aligned with each of the uppermost two nodes of the fluid column. In the mass-spring models two other point masses are attached to nodes 5.08 m above the uppermost fluid nodes to simulate the mass M_2, and the corresponding point mass nodes are linked with spring elements. At the top of the fluid column, the fluid response is coupled to that of the structure using a tie constraint. At the bottom of the fluid column, a plane wave, nonreflecting boundary condition is applied. The step-exponential wave is applied on the bottom surface of the fluid column using incident wave loading, which refers to an amplitude curve that contains the discretized pressure-time history of the wave at the standoff point (the bottom of the fluid column). The point masses are constrained in all directions except the vertical (degree of freedom 2). The nondefault total wave formulation is used to capture the effects of cavitation. The cavitation pressure limit for the acoustic medium is set to zero, thus initiating cavitation whenever the absolute pressure (sum of the incident, scattered, and static pressure) becomes negative. The initial acoustic static pressure in the fluid is specified.

Single-mass case

In the first part of this example M_2 is set to zero, duplicating the model problem published in Bleich and Sandler (1970).

The fluid column depth is 3.81 m. A single stack of AC2D4R elements is used to mesh this column, with all elements 38.1 mm in width and 1.0 m in out-of-plane thickness. The draft of the floating mass is 0.145 m. Atmospheric pressure is 0.101 MPa. The fluid density is 989.0 kg/m^3, the acoustic velocity is 1451.0 m/s, and the acceleration of gravity is 9.81 m/s^2. The fluid properties yield a bulk modulus of 2.082242 GPa, which is the value that is specified along with the fluid density. The initial conditions are specified in such a manner that the pressure at the free surface is the sum of the atmospheric pressure and the pressure caused by the floating mass. Hence, the initial pressure is applied as a linear variation from $p = P_{atm}$ = 0.101 MPa at a height of 0.145

m (to include the effect of the floating mass) to $p = P_{atm} + \rho_f (g) (depth)$ = 0.13937177 MPa at the bottom of the fluid column. The maximum pressure in the step-exponential wave is 0.7106 MPa, and the decay time is 0.9958 ms. This model is studied using a coarse mesh of 100 elements, each 38.1 mm in height, and a fine mesh of 381 elements, each 10 mm in height.

In addition, two time increment sizing methods are compared: that used by Sprague and Geers, and the time increment size automatically computed by Abaqus/Explicit. Sprague and Geers have used a fixed time increment $\Delta t = \Delta t_{cfl}/2$, where Δt_{cfl} is the Courant-Friedrichs-Lewy time increment limit, computed as $\Delta t_{cfl} = \Delta x/c$, where Δx is the element height. Thus, for the 38.1 mm case this requirement yields a time increment of 13.12887 μs, while for the 10 mm case it yields a time increment of 3.445889 μs.

Multiple-mass case

In the second part of this example the ratio of M_2/M_1 is varied to study the cases of Sprague and Geers (2001). Four cases are examined: $M_2/M_1 = 0$, $M_2/M_1 = 1$, $M_2/M_1 = 5$, and $M_2/M_1 = 25$. The results obtained are compared with those obtained by Sprague and Geers (2001) with Cavitating Acoustic Finite Element (CAFE) models.

The fluid column depth is 3.0 m. A single stack of AC2D4R elements is used to mesh this column, with all elements 2.5 mm in height, 10 mm in width, and 1.0 m in out-of-plane thickness. Atmospheric pressure is 0.101 MPa. The fluid density is 1025.0 kg/m^3, the speed of sound is 1500.0 m/s, and the acceleration of gravity is 9.81 m/s^2. The draft is 5.08 m for all the models, so that the mass of displaced fluid is equal to the displaced volume times the fluid mass density, or 52.07 kg. In the case where $M_2/M_1 = 0$, this mass is entirely assigned to M_1. In the second case $M_2/M_1 = 1$. To keep the draft constant at 5.08 m, the total mass of M_1 and M_2 must equal 52.07 kg. Dividing it equally between M_1 and M_2 yields $M_1 = M_2 = 26.035$ kg. For the case $M_2/M_1 = 5$, M_1 is assigned a mass of 8.678333 kg, while M_2 is assigned a mass of 43.391667 kg. For the fourth case $M_2/M_1 = 25$; hence, M_1 is 2.0026923 kg, while M_2 is 50.067308 kg.

The spring constants defined in the reference paper are such that the fixed-base natural frequency of mass M_2 is 5 Hz in all the cases: $K = (5 \times 2\pi)^2 M_2$. Thus, for the case $M_2/M_1 = 1$, K = 12847.758 kg/s^2; for $M_2/M_1 = 5$, K = 21412.929 kg/s^2; and for $M_2/M_1 = 25$, K = 24707.226 kg/s^2.

The initial conditions are specified as in the single-mass case, with atmospheric pressure of 0.101 MPa applied 5.08 m above the free surface and 0.212412 MPa applied at a depth of 6 m. The maximum pressure in the step-exponential wave is 16.15 MPa, and the decay time is 0.423 ms.

For all the mass ratio cases, analyses are performed using a fixed time increment size $\Delta t = \Delta t_{cfl}/2$, which is 0.8333 μs. For the case where $M_2/M_1 = 5$, the effect of using a smaller time increment size ($\Delta t = \Delta t_{cfl}/20$ = 0.08333 μs) while holding all other parameters constant is also analyzed.

Submodeling

These types of analyses can also be performed using the acoustic-to-structure submodeling technique; this study includes a case where the results obtained by performing an analysis with the submodeling technique are compared with those obtained using the default global analysis technique. The submodeling technique illustrated here is useful in situations where the structural response is of primary interest and the presence of the fluid is required mainly for the application of the underwater explosive load onto the structure. In such situations it is possible to perform a single global analysis with a fluid mesh, followed by multiple submodel analyses without the fluid mesh wherein the structural parameters are varied and the effects analyzed. Due to the absence of the fluid mesh in the submodel analyses, computational effort may be significantly reduced. In this study the submodeling

technique is illustrated for the case $M_2/M_1 = 5$ and is run for a step time of 5 ms. First, a global analysis is performed and the structural displacements (U) and acoustic pressures (POR) at the top of the fluid stack are written to the results file. Following the completion of the global analysis, the submodel analysis is performed, wherein the model consists of the structure only and no fluid mesh is present. This structural submodel is driven by the acoustic pressure results extracted from the global analysis.

Modeling cavitation using displacement-based elements

In certain underwater explosion situations—for example, when an underwater explosion occurs near a submarine—the explosion can cause large structural displacements of the submarine hull. In cases where the structural displacements are extremely large, as occurs during plastic failure of the hull, the fluid migrates to fill the displaced volume. This large motion of the fluid is best modeled with displacement-based continuum elements that can be adaptively meshed to avoid extreme mesh distortions in Abaqus/Explicit. To demonstrate this modeling technique, the cavitation of the one-dimensional fluid column is modeled using a single stack of CPE4R elements instead of AC2D4R elements. The geometry and material properties used are identical to the multiple-mass case. The effect of the hydrostatic pressure is simulated using gravity loading on the fluid column, with an additional distributed pressure load defined on the top of the fluid column to account for the effect of the atmospheric pressure and the draft of the floating masses. To establish an initial equilibrated state, geostatic initial stresses are specified. An equation-of-state material of the linear $U_s - U_p$ form is used to model the fluid, and a tensile failure model is used to simulate cavitation in the fluid medium. The material parameters are chosen to closely match the acoustic medium properties used for the acoustic element simulation. At the bottom of the fluid column, nonreflecting boundary conditions are applied by defining a displacement-based infinite element of type CINPE4. Adaptive meshing is used to adaptively remesh the fluid domain to prevent excessive mesh distortions. The loading is identical to that used for the multiple-mass case.

Results and discussion

The results are analyzed by comparing the predictions made by running double precision Abaqus/Explicit to those in the referenced literature.

Single-mass case

We compare the upward velocity of mass M_1 and the spatiotemporal variation of the cavitated region in the fluid obtained using Abaqus/Explicit to those same quantities obtained by Bleich and Sandler. *Figure 2* shows the results obtained by Abaqus/Explicit plotted alongside those obtained analytically by Bleich and Sandler for the coarse mesh consisting of 38.1 mm elements. *Figure 3* shows the comparison for the finer mesh consisting of 10 mm elements, while *Figure 4* shows the comparisons of the cavitation region. The results obtained by Abaqus/Explicit show good comparison with the theoretical results. It is also found that the difference in results between using a predetermined fixed time increment size and the automatic time incrementation scheme in Abaqus/Explicit is insignificant.

Multiple-mass case

Figure 5 through *Figure 11* show the Abaqus/Explicit results alongside those calculated numerically by Sprague and Geers. We compare velocities v_1 and v_2 and the cavitated region. There is good comparison in all cases. In the $M_2/M_1 = 5$ case the velocity obtained by using a reduced time increment size is also shown. There is no significant effect of reducing the time increment size in Abaqus/Explicit on the velocities v_1 and v_2. *Figure 12* through *Figure 16* show the cavitation region comparisons for the four different mass ratios. Comparing *Figure 14* and *Figure 16*, we see that the cavitation region computed by Abaqus/Explicit shows a dependence on the time increment size, which is in agreement with the findings of Sprague and Geers.

Submodeling

Figure 17 and *Figure 18* show the comparisons between the results obtained from the global analysis and those obtained from the submodel analysis for the $M_2/M_1 = 5$ case. As can be seen, the results are identical.

Modeling cavitation using displacement-based elements

Figure 19 and *Figure 20* show the comparisons between the results obtained from the multiple-mass case analysis using acoustic elements and those obtained using displacement-based elements. The results are shown for a mass ratio $M_2/M_1 = 5$. The results from the two analyses are seen to be in good agreement.

Input files

1_mass_coarse.inp	Bleich and Sandler model, coarse mesh.
1_mass_fine.inp	Bleich and Sandler model, fine mesh.
2_mass_0_1.inp	$M_2/M_1 = 0$.
2_mass_1_1.inp	$M_2/M_1 = 1$.
2_mass_5_1.inp	$M_2/M_1 = 5$.
2_mass_25_1.inp	$M_2/M_1 = 25$.
2_mass_5_1_global.inp	$M_2/M_1 = 5$, global model.
2_mass_5_1_sub.inp	$M_2/M_1 = 5$, submodel.
2_mass_5_1_displbased.inp	$M_2/M_1 = 5$, displacement-based elements.

References

- Bleich, H. H., and I. S. Sandler, “Interaction between Structures and Bilinear Fluids,” *International Journal of Solids and Structures*, vol. 6, pp. 617–639, 1970.
- Sprague, M. A., and T. L. Geers, “Computational Treatments of Cavitation Effects in Near-Free-Surface Underwater Shock Analysis,” *72nd Shock and Vibration Symposium Proceedings*, 2001.

Figures

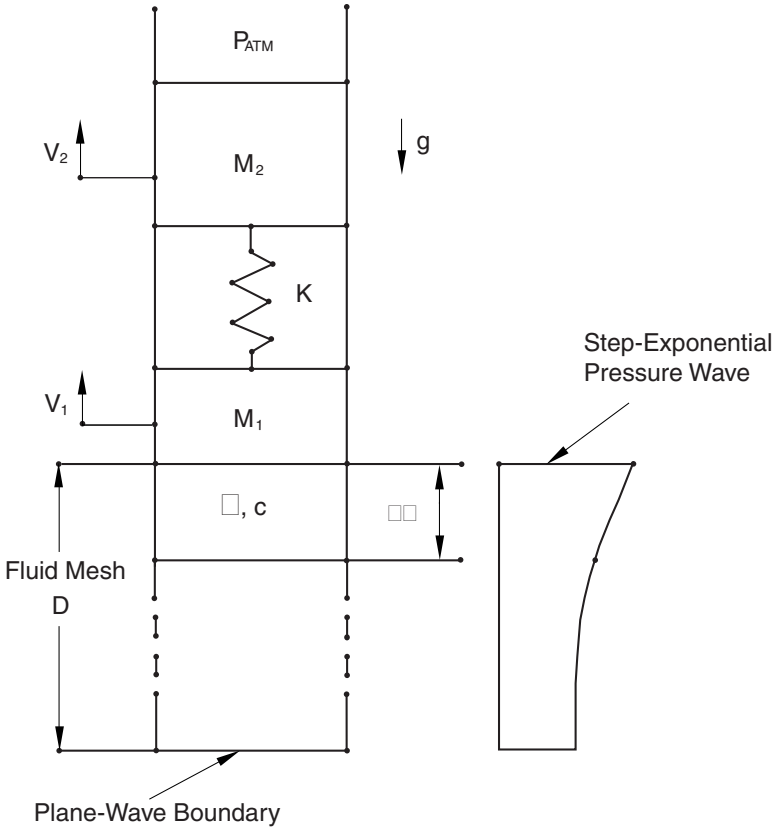


Figure 1: Schematic for a 2-mass oscillator floating on a fluid column.

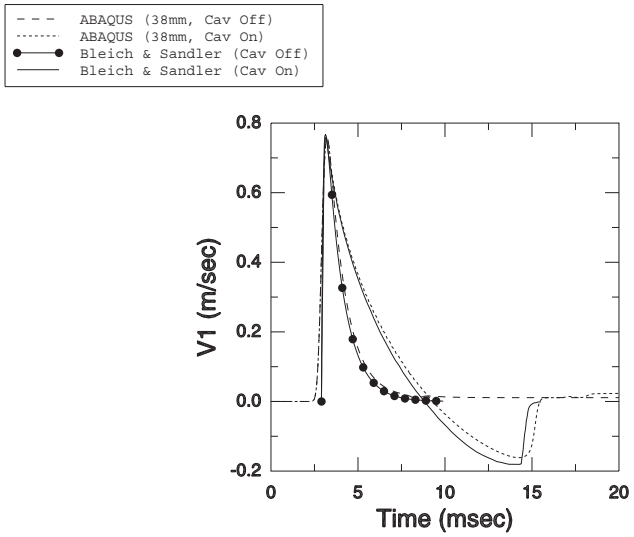


Figure 2: Velocity v_1 comparison for the Bleich and Sandler model with coarse mesh.

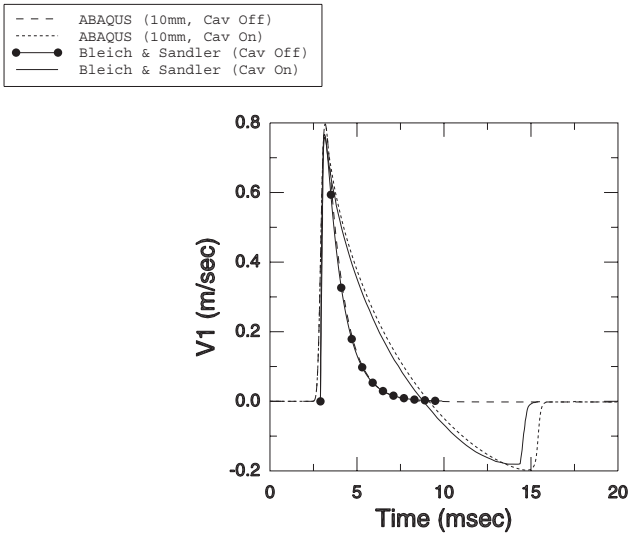


Figure 3: Velocity v_1 comparison for the Bleich and Sandler model with fine mesh.

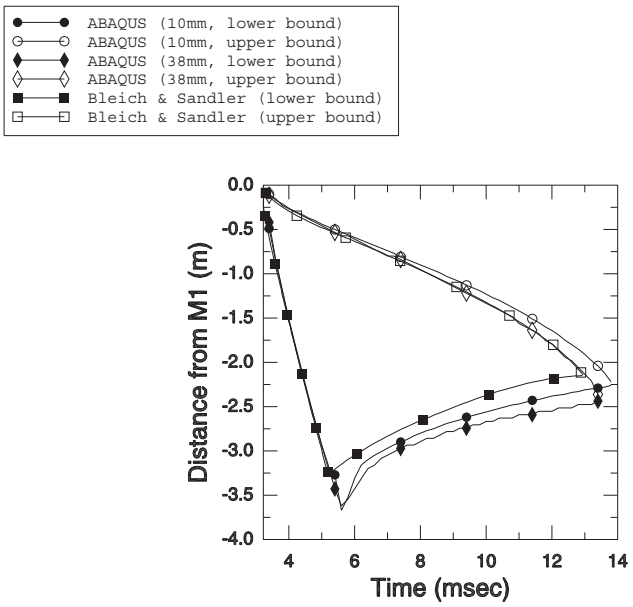


Figure 4: Cavitation region comparison for the Bleich and Sandler model.

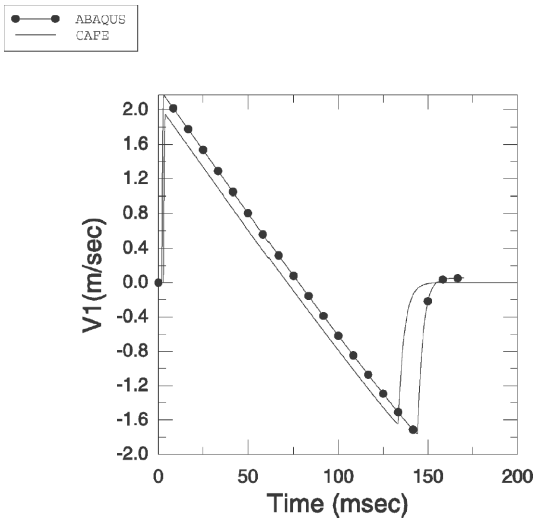


Figure 5: Velocity v_1 comparison for the M_2/M_1 = 0 case.

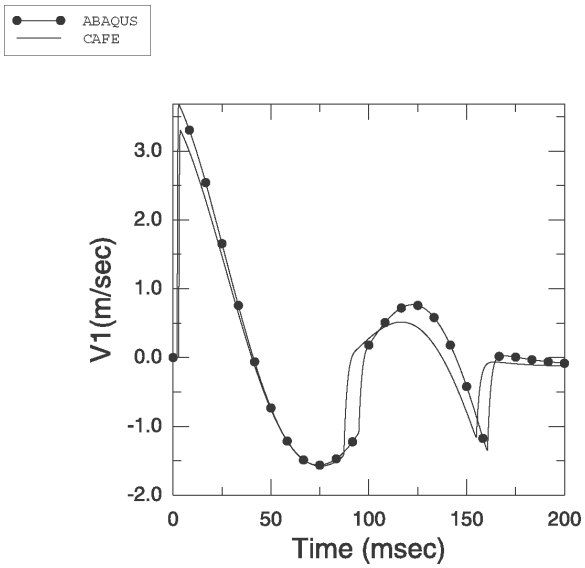


Figure 6: Velocity v_1 comparison for the M_2/M_1 = 1 case.

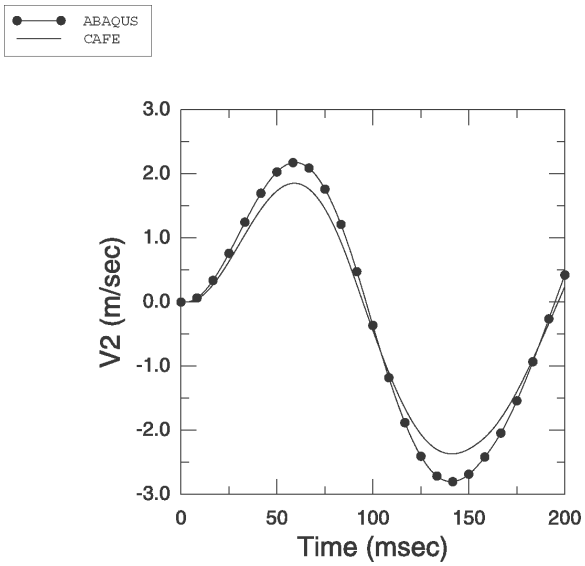


Figure 7: Velocity v_2 comparison for the M_2/M_1 = 1 case.

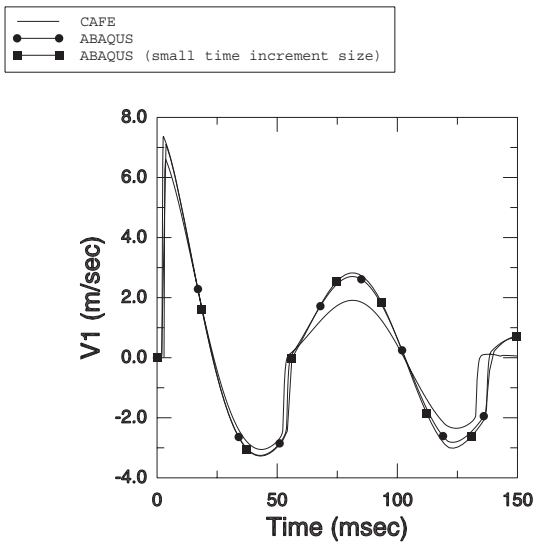


Figure 8: Velocity v_1 comparison for the M_2/M_1 = 5 case.

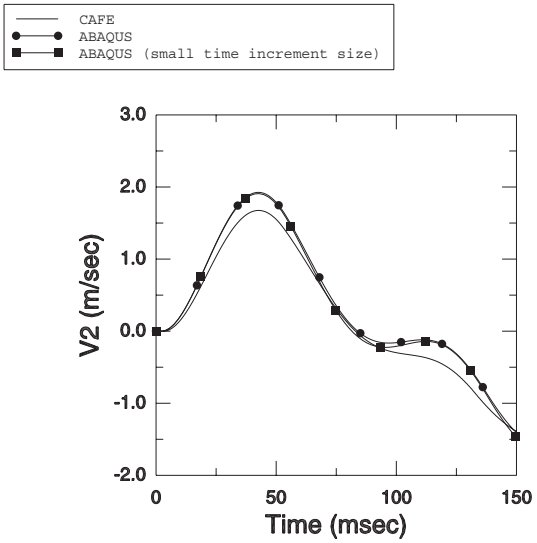


Figure 9: Velocity v_2 comparison for the M_2/M_1 = 5 case.

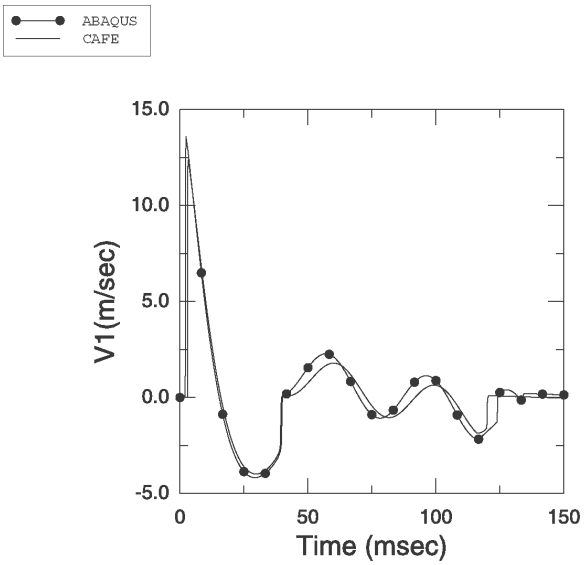


Figure 10: Velocity v_1 comparison for the M_2/M_1 = 25 case.

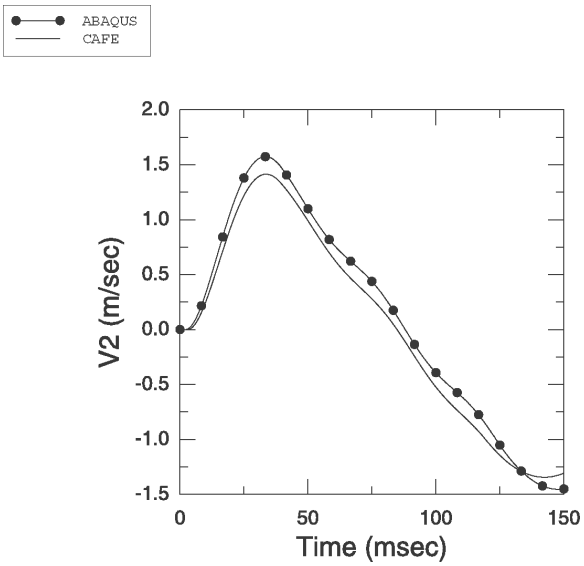


Figure 11: Velocity v_2 comparison for the M_2/M_1 = 25 case.

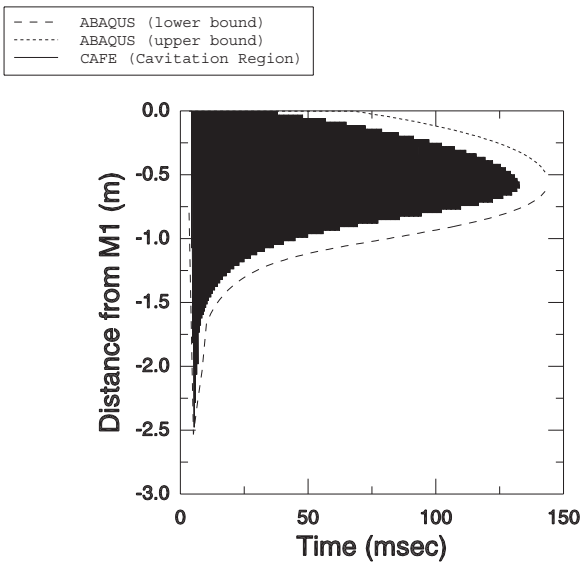


Figure 12: Cavitation region comparison for the M_2/M_1 = 0 case.

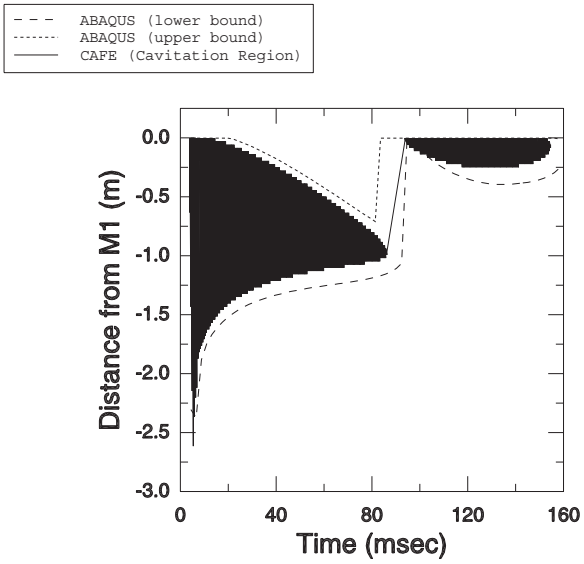


Figure 13: Cavitation region comparison for the M_2/M_1 = 1 case.

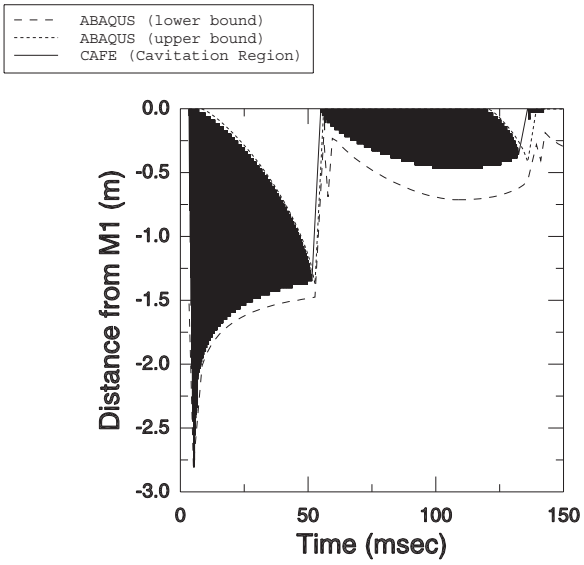


Figure 14: Cavitation region comparison for the M_2/M_1 = 5 case.

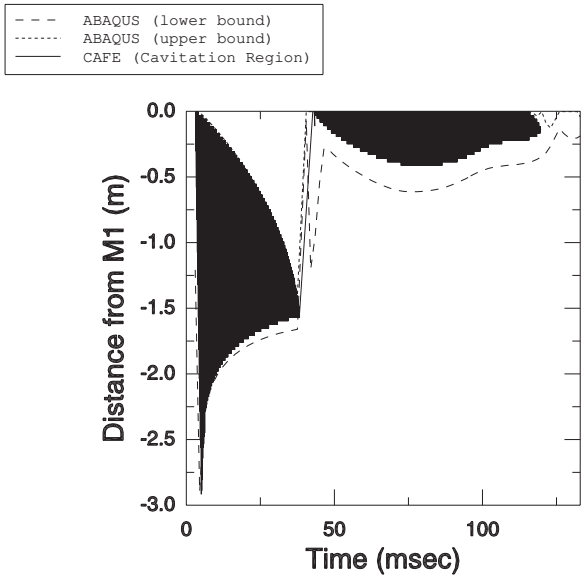


Figure 15: Cavitation region comparison for the M_2/M_1 = 25 case.

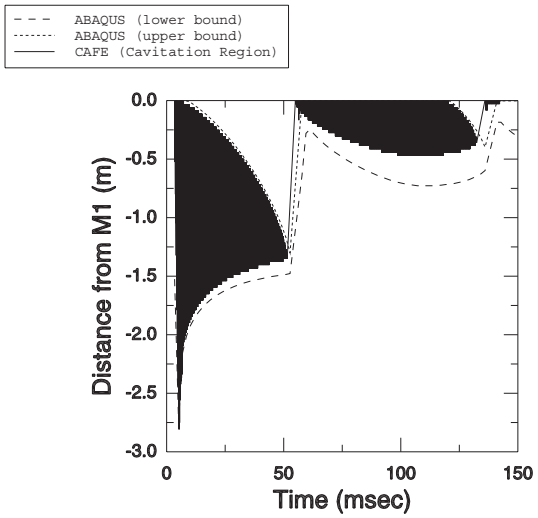


Figure 16: Cavitation region comparison for the M_2/M_1 = 5 case using a smaller time increment size.

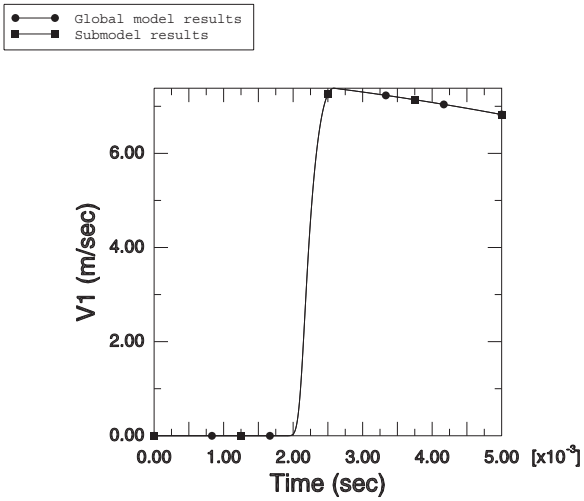


Figure 17: Velocity v_1 comparison between the global and submodel analyses for the case M_2/M_1 = 5.

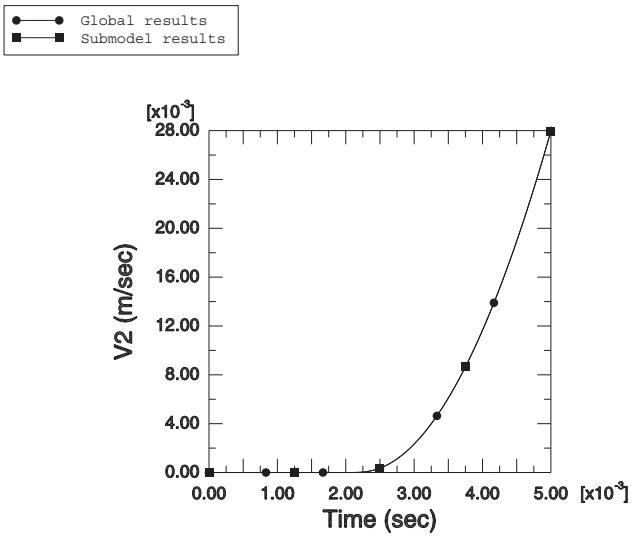


Figure 18: Velocity v_2 comparison between the global and submodel analyses for the case M_2/M_1 = 5.

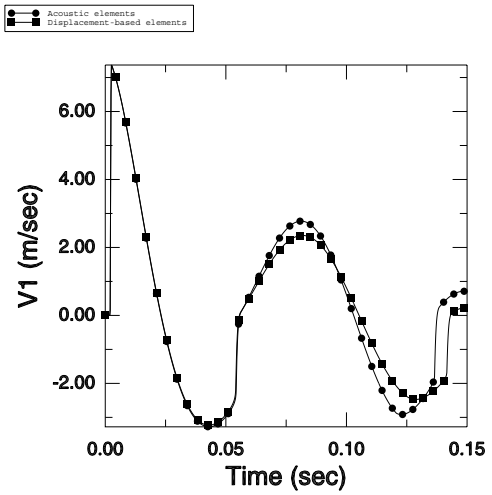


Figure 19: Velocity v_1 comparison between the acoustic element and displacement-based element analyses for the case M_2/M_1 = 5.

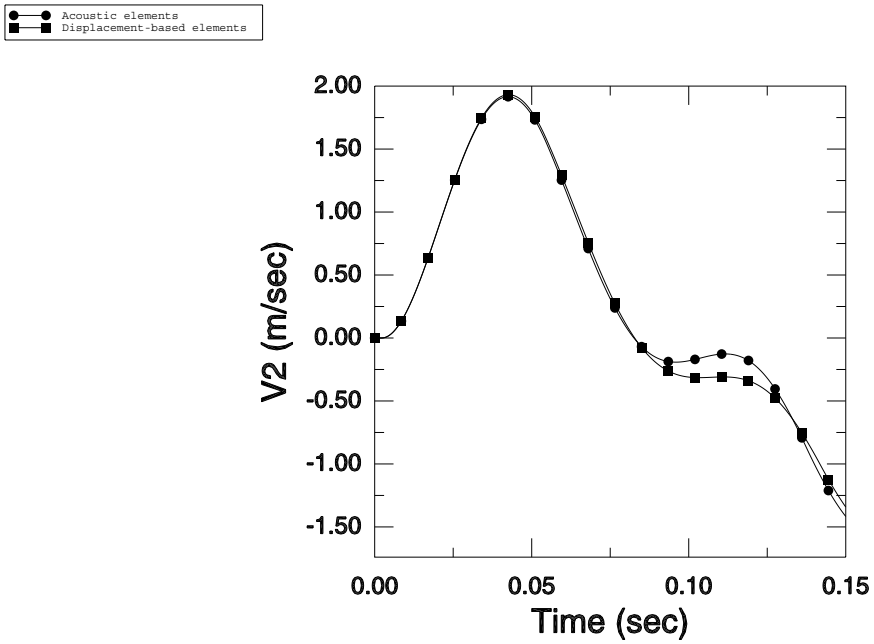


Figure 20: Velocity v_2 comparison between the acoustic element and displacement-based element analyses for the case M_2/M_1 = 5.

Plate response to a planar exponentially decaying shock wave

Products: Abaqus/Explicit

Overview

This example illustrates the use of Abaqus/Explicit to model the interaction between an air-backed elastic plate and a planar exponentially decaying wave.
The results obtained using Abaqus/Explicit are compared with those obtained independently using the Doubly Asymptotic Approximation (Geers (1978), Abaqus/USA 6.1). This problem has been solved analytically by Taylor (1950).

Problem description

When an underwater explosion occurs, a compressive wave is generated. This wave can have a substantial effect on a submerged structure.

This problem models the interaction between an air-backed elastic plate and a weak planar exponentially decaying shock wave with a maximum pressure of 15.4 MPa and a decay time of 0.433 ms. In contrast to Taylor's solution, engineering material parameters for the fluid and solid media are used. The plate is a square of side 1 m and has a thickness of 0.00635 m. The plate is made of steel with a density of 7850 kg/m^3, a Young's modulus of 210.0 GPa, and a Poisson's ratio of 0.3. The fluid is water with a density of 1000 kg/m^3, in which the speed of sound is 1461 m/s. The plate is represented by a single S4R element, and the surrounding fluid is represented by a fluid region that extends from the plate to a distance of 5.5 m away from the plate in the direction of the incoming shock wave. The fluid region is modeled by a single stack of 400 AC3D8R elements. A planar nonreflective impedance boundary condition is imposed on the exterior surface of the fluid region furthest from the plate. The fluid response is coupled to that of the structure using a tie constraint on the fluid surface nearest to the plate and the plate itself. The fluid-solid system is excited by a planar exponentially decaying wave applied at the fluid-solid interface using incident wave loading. In addition, the plate motion is constrained by the use of four springs attached to the nodes of the plate, each possessing a spring constant of 4.919 MN/m. A linear bulk viscosity parameter of 0.25 and a quadratic bulk viscosity parameter of 10.0 are used.

Results and discussion

The results from Abaqus/Explicit show good qualitative comparison with those in the referenced literature. We also compare the translational velocity imparted to the plate obtained using Abaqus/Explicit with that obtained using Abaqus/USA 6.1. As shown in *Figure 1*, the results agree closely.

Input files

undex_plate_ped.inp	Input data for this analysis.

References

- Geers, T., “Doubly Asymptotic Approximations for Transient Motions of Submerged Structures,” *Journal of the Acoustical Society of America*, vol. 64, pp. 1500–1508, 1978.
- Taylor, G. I., “The Pressure and Impulse of Submarine Explosion Waves on Plates,” *Underwater Explosion Research, Office of Naval Research*, vol. 1, pp. 1155–1173, 1950.

Figures

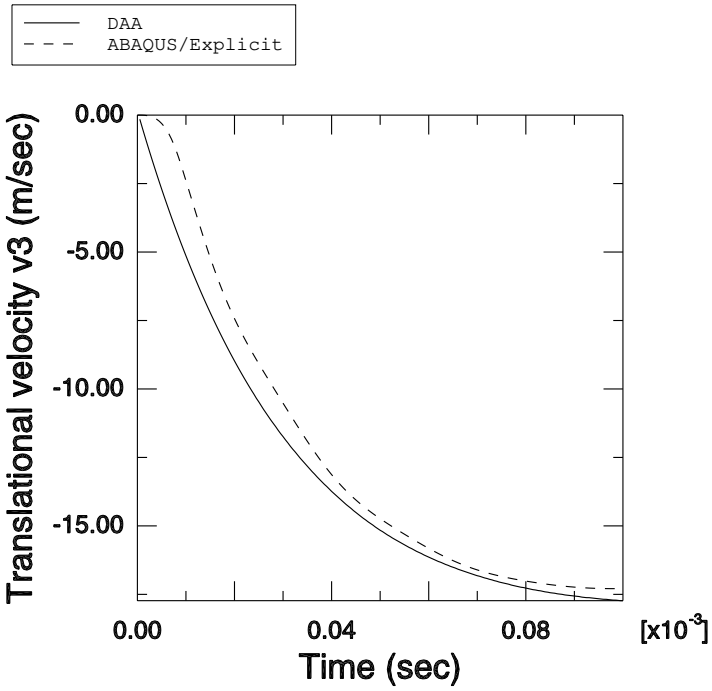


Figure 1: Comparison of the translational velocity of the plate obtained with the Doubly Asymptotic Approximation method and with Abaqus/Explicit.

Cylindrical shell response to a planar step shock wave

Products: Abaqus/Explicit

Overview

This example illustrates the use of Abaqus/Explicit to model the interaction between an air-backed spherical elastic shell and a planar step wave.
The results obtained using Abaqus/Explicit are compared with those obtained independently using the Doubly Asymptotic Approximation (Geers (1978), Abaqus/USA 6.1). This problem has been solved analytically by Huang (1970).

Problem description

This problem models the interaction between an air-backed cylindrical elastic shell and a weak planar step shock wave with a maximum pressure of 1 Pa. The cylindrical shell has a radius of 1 m and a thickness of 0.029 m. The shell is made of steel with a density of 7766 kg/m^3, a Young's modulus of 206.4 GPa, and a Poisson's ratio of 0.3. The fluid is water with a density of 997 kg/m^3, in which the speed of sound is 1524 m/s. A half-symmetry model is used to study this problem. A thin axial section of width 0.0049 m with symmetry boundary conditions is used to represent the infinite length of the actual cylinder. The shell is represented by S4R elements, and the surrounding fluid is represented by a fluid region that extends concentrically from the shell and has a radius of 3 m. The fluid region is modeled with AC3D8R elements. A circular nonreflective boundary condition is imposed on the exterior surface of the fluid using surface impedance. The fluid response is coupled to that of the structure using a tie constraint on the fluid surface nearest to the shell and the shell itself. The fluid-solid system is excited by a planar step wave applied close to the fluid-solid interface using incident wave loading. A linear bulk viscosity parameter of 0.25 and a quadratic bulk viscosity parameter of 10.0 are used.

Results and discussion

The results are analyzed by comparing predictions made by Abaqus/Explicit with those in the referenced literature. We also compare the numerical values for radial velocities at the leading and trailing edges of the shell obtained using Abaqus/Explicit with those obtained using Abaqus/USA 6.1. As shown in *Figure 1* and *Figure 2*, the results agree closely.

Input files

undex_cyl_ps.inp	Input data for this analysis.

References

- Geers, T., “Doubly Asymptotic Approximations for Transient Motions of Submerged Structures,” *Journal of the Acoustical Society of America*, vol. 64, pp. 1500–1508, 1978.
- Huang, H., “An Exact Analysis of the Transient Interaction of Acoustic Plane Waves With a Cylindrical Elastic Shell,” *Journal of Applied Mechanics*, vol. 37, pp. 1091–1099, December 1970.

Figures

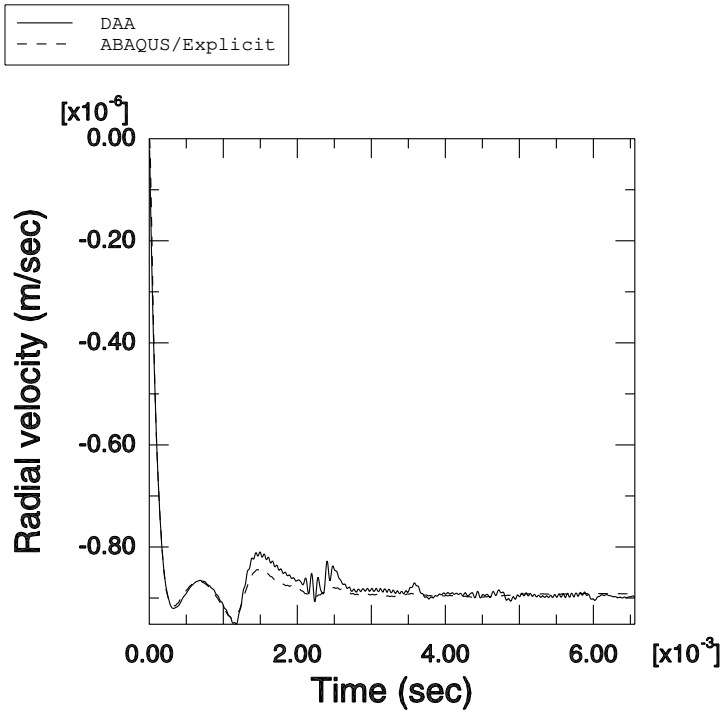


Figure 1: Comparison of radial velocity at the leading edge of the cylindrical shell obtained with the Doubly Asymptotic Approximation method and with Abaqus/Explicit.

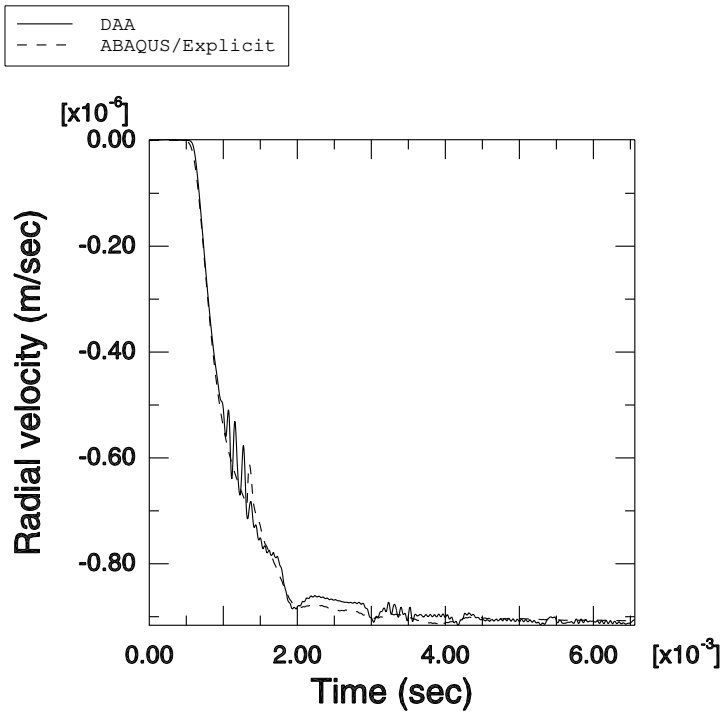


Figure 2: Comparison of radial velocity at the trailing edge of the cylindrical shell obtained with the Doubly Asymptotic Approximation method and with Abaqus/Explicit.

Cylindrical shell response to a planar exponentially decaying shock wave

Products: Abaqus/Explicit

Overview

This example illustrates the use of Abaqus/Explicit to model the interaction between an air-backed cylindrical elastic shell and a planar exponentially decaying wave.
The results obtained using Abaqus/Explicit are compared with those obtained independently using the Doubly Asymptotic Approximation (Geers (1978), Abaqus/USA 6.1). This problem has been solved analytically by Huang (1970).

Problem description

This problem models the interaction between an air-backed cylindrical elastic shell and a weak planar exponentially decaying shock wave with a maximum pressure of 1 MPa and a decay time of 0.0137 ms. The cylindrical shell has a radius of 1 m and a thickness of 0.029 m. The shell is made of steel with a density of 7766 kg/m^3, a Young's modulus of 206.4 GPa, and a Poisson's ratio of 0.3. The fluid is water with a density of 997 kg/m^3, in which the speed of sound is 1524 m/s. A half-symmetry model is used to study this problem. A thin axial section of width 0.049 m with symmetry boundary conditions is used to represent the infinite length of the actual cylinder. The shell is represented by S4R elements, and the surrounding fluid is represented by a fluid region that extends concentrically from the shell and has a radius of 2.029 m. The fluid region is modeled with AC3D8R elements. The fluid mesh density is the same as that of the shell in the circumferential direction; the fluid mesh has 150 elements/m in the radial direction. A circular nonreflective boundary condition is imposed on the exterior surface of the fluid using surface impedance. The fluid response is coupled to that of the structure using a tie constraint on the fluid surface nearest to the shell and the shell itself. The fluid-solid system is excited by a planar exponentially decaying wave applied close to the fluid-solid interface using incident wave loading. A linear bulk viscosity parameter of 0.25 and a quadratic bulk viscosity parameter of 10.0 are used.

Results and discussion

The results are analyzed by comparing predictions made by Abaqus/Explicit with those in the referenced literature. We also compare the numerical values of radial velocities at the leading and trailing edges of the cylindrical shell obtained using Abaqus/Explicit with those obtained using Abaqus/USA 6.1. As shown in *Figure 1* and *Figure 2*, the results agree closely.

Input files

undex_cyl_ped.inp	Input data for this analysis.

References

- Geers, T., “Doubly Asymptotic Approximations for Transient Motions of Submerged Structures,” *Journal of the Acoustical Society of America*, vol. 64, pp. 1500–1508, 1978.
- Huang, H., “An Exact Analysis of the Transient Interaction of Acoustic Plane Waves With a Cylindrical Elastic Shell,” *Journal of Applied Mechanics*, vol. 37, pp. 1091–1099, December 1970.

Figures

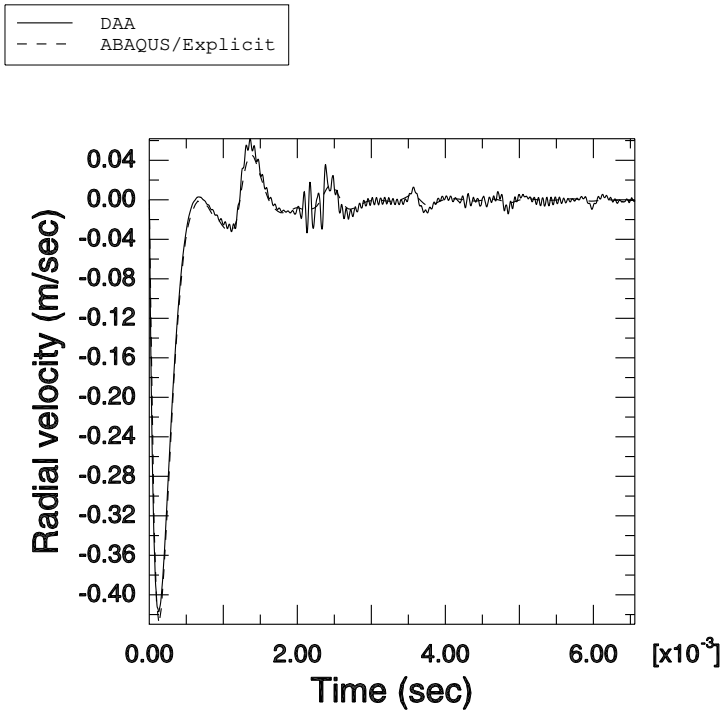


Figure 1: Comparison of radial velocity at the leading edge of the cylindrical shell obtained with the Doubly Asymptotic Approximation method and with Abaqus/Explicit.

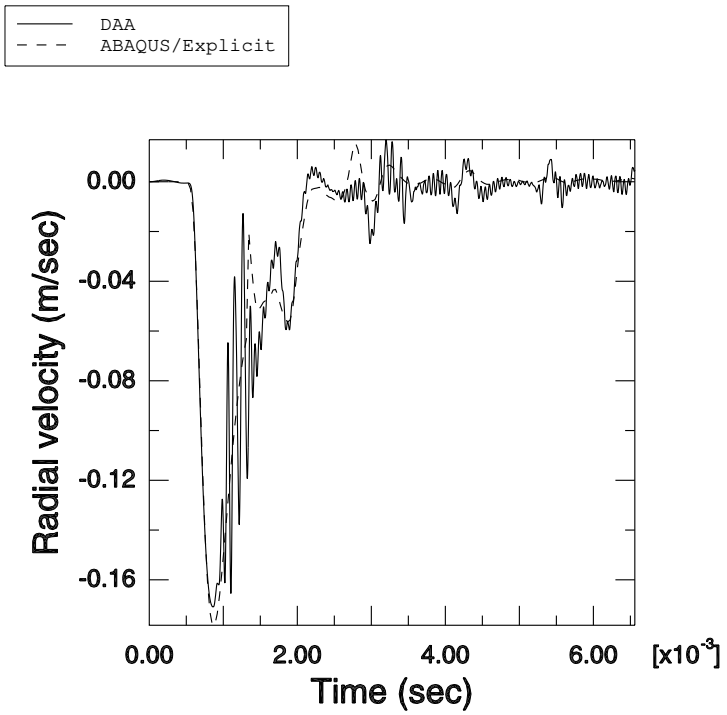


Figure 2: Comparison of radial velocity at the trailing edge of the cylindrical shell obtained with the Doubly Asymptotic Approximation method and with Abaqus/Explicit.

Spherical shell response to a planar step wave

Products: Abaqus/Explicit

Overview

This example illustrates the use of Abaqus/Explicit to model the interaction between an air-backed spherical elastic shell and a planar step wave.
The results obtained using Abaqus/Explicit are compared with those obtained independently using the Doubly Asymptotic Approximation (Geers (1978), Abaqus/USA 6.1). This problem has been solved analytically by Huang (1969).

Problem description

This problem models the interaction between an air-backed spherical elastic shell and a weak planar step shock wave with a maximum pressure of 1 MPa. In contrast to Huang's solution, engineering material parameters for the fluid and solid media are used. The sphere has a radius of 1 m and a thickness of 0.02 m. The sphere is made of steel with a density of 7766 kg/m^3, a Young's modulus of 210.0 GPa, and a Poisson's ratio of 0.3. The fluid is water with a density of 997 kg/m^3, in which the speed of sound is 1462 m/s. An axisymmetric model is used for this analysis. The spherical shell is represented by a semicircular shell, and the surrounding fluid is represented by an acoustic region bounded by two concentric semicircles and the axis of symmetry. The spherical shell is modeled with SAX1 elements, while the surrounding fluid is modeled with ACAX4R elements. The inner semicircle that bounds the fluid region is coincident with the shell, and the outer semicircle has a radius of 3 m. A spherical nonreflective boundary condition is imposed on the outer semicircle using surface impedance. The fluid response is coupled to that of the structure using a tie constraint. The fluid-solid system is excited by a planar step wave applied at the point where the semicircular shell intersects the axis of symmetry using incident wave loading. A linear bulk viscosity parameter of 0.2 and a quadratic bulk viscosity parameter of 1.2 are used.

Results and discussion

The results from Abaqus/Explicit show good qualitative comparison with those in the referenced literature. We also compare the numerical values for radial velocities at the leading and trailing points on the shell obtained using Abaqus/Explicit with those obtained using Abaqus/USA 6.1. As shown in *Figure 1* and *Figure 2*, the results agree closely.

Input files

undex_sph_ps.inp	Input data for this analysis.

References

- Geers, T., “Doubly Asymptotic Approximations for Transient Motions of Submerged Structures,” *Journal of the Acoustical Society of America*, vol. 64, pp. 1500–1508, 1978.
- Huang, H., “Transient Interaction of Plane Acoustic Waves with a Spherical Elastic Shell,” *Journal of the Acoustical Society of America*, vol. 45, pp. 661–670, 1969.

Figures

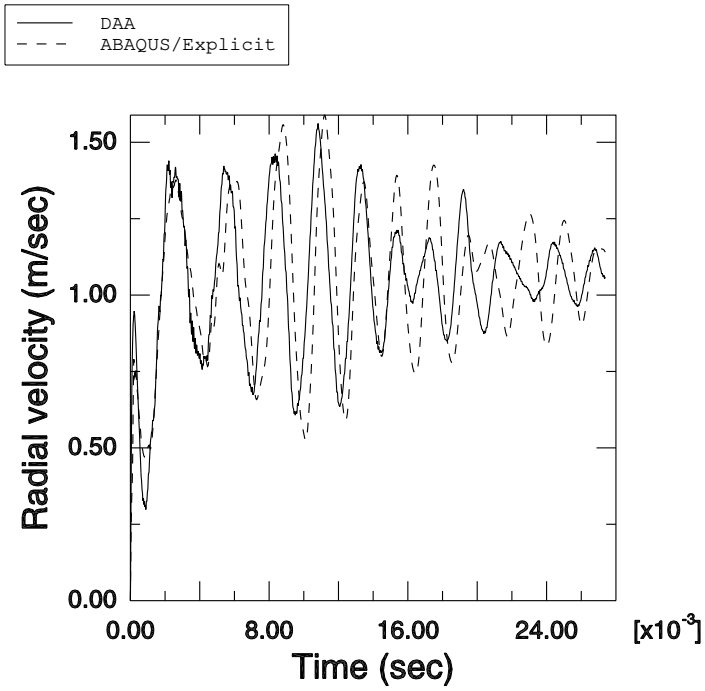


Figure 1: Comparison of the radial velocity at the leading point on the spherical shell obtained with the Doubly Asymptotic Approximation method and with Abaqus/Explicit.

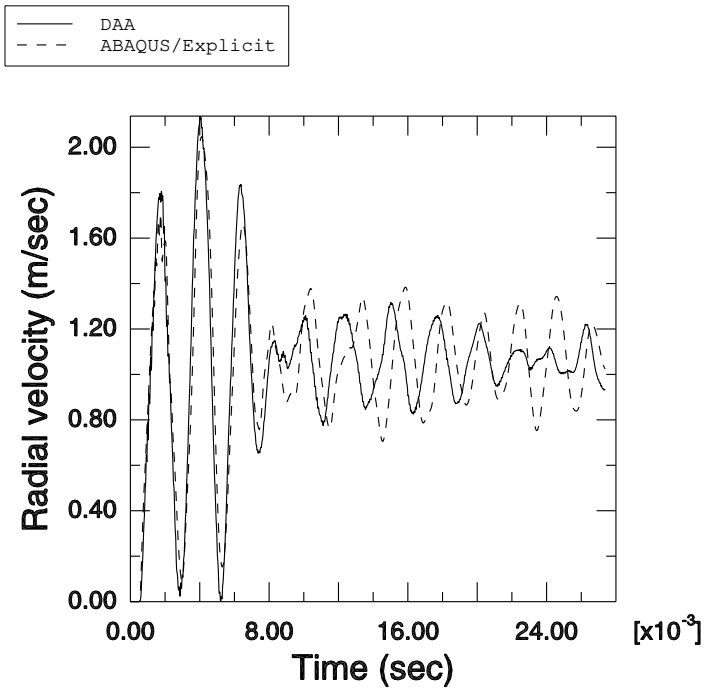


Figure 2: Comparison of the radial velocity at the trailing point on the spherical shell obtained with the Doubly Asymptotic Approximation method and with Abaqus/Explicit.

Spherical shell response to a planar exponentially decaying wave

Products: Abaqus/Explicit

Overview

This example illustrates the use of Abaqus/Explicit to model the interaction between a spherical elastic shell and a planar exponentially decaying wave.
The results obtained using Abaqus/Explicit are compared with those obtained independently using the Doubly Asymptotic Approximation (Geers (1978), Abaqus/USA 6.1). This problem has been solved analytically by Huang (1969).

Simulating the response of submerged structures of simple geometric shapes to various underwater explosions constitutes an important part of the validation of any fluid-structure interaction code.

Problem description

This problem models the interaction between an air-backed spherical elastic shell and a weak planar exponentially decaying shock wave with a maximum pressure of 1 MPa and a decay time of 0.685 ms. In contrast to Huang's solution, engineering material parameters for the fluid and solid media are used. The sphere has a radius of 1 m and a thickness of 0.02 m. The sphere is made of steel with a density of 7766 kg/m^3, a Young's modulus of 210.0 GPa, and a Poisson's ratio of 0.3. The fluid is water with a density of 997 kg/m^3, in which the speed of sound is 1461 m/s. An axisymmetric model is used for this analysis. The spherical shell is represented by a semicircular shell, and the surrounding fluid is represented by an acoustic region bounded by two concentric semicircles and the axis of symmetry. The spherical shell is modeled with SAX1 elements, while the surrounding fluid is modeled with ACAX4R elements. The inner semicircle that bounds the fluid region is coincident with the shell, and the outer semicircle has a radius of 3 m. A spherical nonreflective boundary condition is imposed on the outer semicircle using surface impedance. The fluid response is coupled to that of the structure using a tie constraint. The fluid-solid system is excited by a planar exponentially decaying wave applied at the point where the semicircular shell intersects the axis of symmetry using incident wave loading. A linear bulk viscosity parameter of 0.2 and a quadratic bulk viscosity parameter of 1.2 are used.

Results and discussion

The results from Abaqus/Explicit show good qualitative comparison with those in the referenced literature. We also compare the numerical values for radial velocities at the leading and trailing points on the spherical shell obtained using Abaqus/Explicit with those obtained using Abaqus/USA 6.1. As shown in *Figure 1* and *Figure 2*, the results agree closely.

Input files

undex_sph_ped.inp	Input data for this analysis.

References

- Geers, T., “Doubly Asymptotic Approximations for Transient Motions of Submerged Structures,” *Journal of the Acoustical Society of America*, vol. 64, pp. 1500–1508, 1978.
- Huang, H., “Transient Interaction of Plane Acoustic Waves with a Spherical Elastic Shell,” *Journal of the Acoustical Society of America*, vol. 45, pp. 661–670, 1969.

Figures

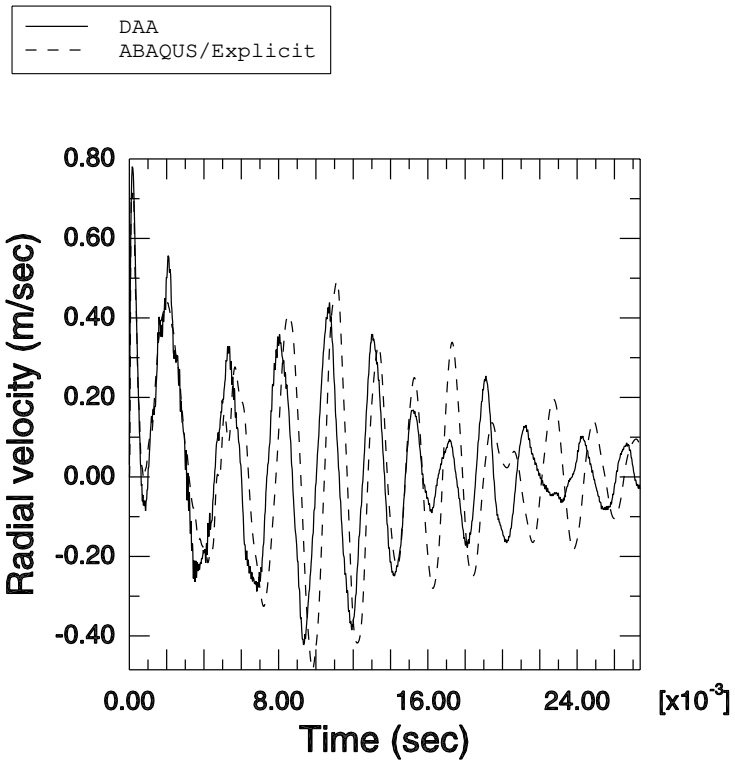


Figure 1: Comparison of the radial velocity at the leading point on the spherical shell obtained with the Doubly Asymptotic Approximation method and with Abaqus/Explicit.

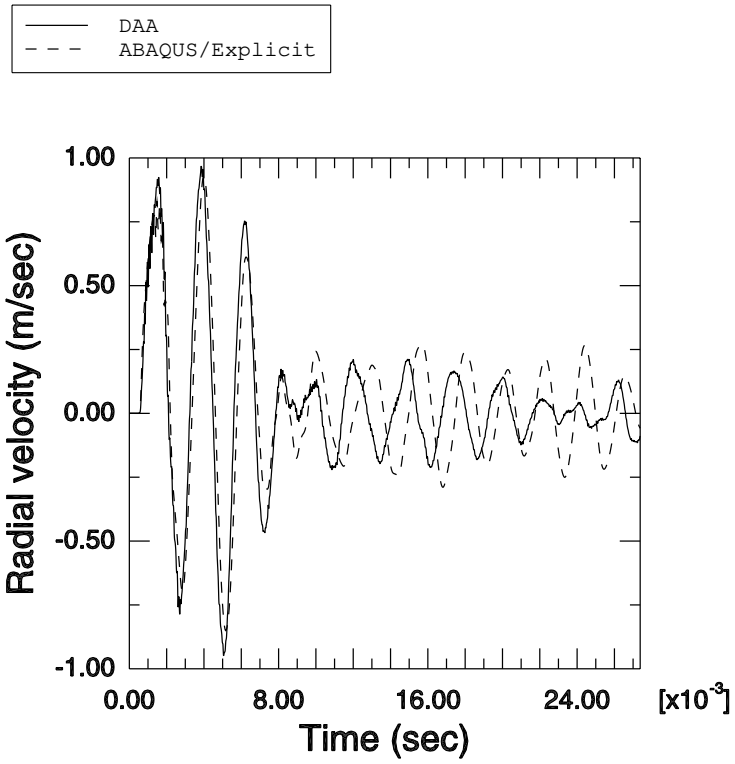


Figure 2: Comparison of the radial velocity at the trailing point on the spherical shell obtained with the Doubly Asymptotic Approximation method and with Abaqus/Explicit.

Spherical shell response to a spherical exponentially decaying wave

Products: Abaqus/Explicit

Overview

This example illustrates the use of Abaqus/Explicit to model the interaction between a spherical elastic shell and a spherical exponentially decaying wave.
The results obtained using Abaqus/Explicit are compared with those obtained independently using the Doubly Asymptotic Approximation (Geers (1978), Abaqus/USA 6.1). This problem has been solved analytically by Huang et al (1971).

Problem description

This problem models the interaction between an air-backed spherical elastic shell and a weak spherical exponentially decaying shock wave with a maximum pressure of 1 MPa and a decay time of 0.685 ms. The source is located 4 m away from the center of the sphere. In contrast to the solution from Huang et al., engineering material parameters for the fluid and solid media are used. The sphere has a radius of 1 m and a thickness of 0.02 m. The sphere is made of steel with a density of 7766 kg/m^3, a Young's modulus of 210.0 GPa, and a Poisson's ratio of 0.3. The fluid is water with a density of 997 kg/m^3, in which the speed of sound is 1461 m/s. An axisymmetric model is used for this analysis. The spherical shell is represented by a semicircular shell, and the surrounding fluid is represented by an acoustic region bounded by two concentric semicircles and the axis of symmetry. The spherical shell is modeled with SAX1 elements, while the surrounding fluid is modeled with ACAX4R elements. The inner semicircle that bounds the fluid region is coincident with the shell, and the outer semicircle has a radius of 3 m. A spherical nonreflective boundary condition is imposed on the outer semicircle using surface impedance. The fluid response is coupled to that of the structure using a tie constraint. The fluid-solid system is excited by a spherical exponentially decaying wave applied at the point where the semicircular shell intersects the axis of symmetry using incident wave loading. A linear bulk viscosity parameter of 0.2 and a quadratic bulk viscosity parameter of 1.2 are used.

Results and discussion

The results from Abaqus/Explicit show good qualitative comparison with those in the referenced literature. We also compare the numerical values for radial velocities at the leading and trailing points on the spherical shell obtained using Abaqus/Explicit with those obtained using Abaqus/USA 6.1. As shown in *Figure 1* and *Figure 2*, the results agree closely.

Input files

undex_sph_sed.inp	Input data for this analysis.

References

- Geers, T., “Doubly Asymptotic Approximations for Transient Motions of Submerged Structures,” *Journal of the Acoustical Society of America*, vol. 64, pp. 1500–1508, 1978.
- Huang, H., Y. P. Lu, and Y. F. Wang, “Transient Interaction of Spherical Acoustic Waves and a Spherical Elastic Shell,” *Journal of Applied Mechanics*, pp. 71–74, March 1971.

Figures

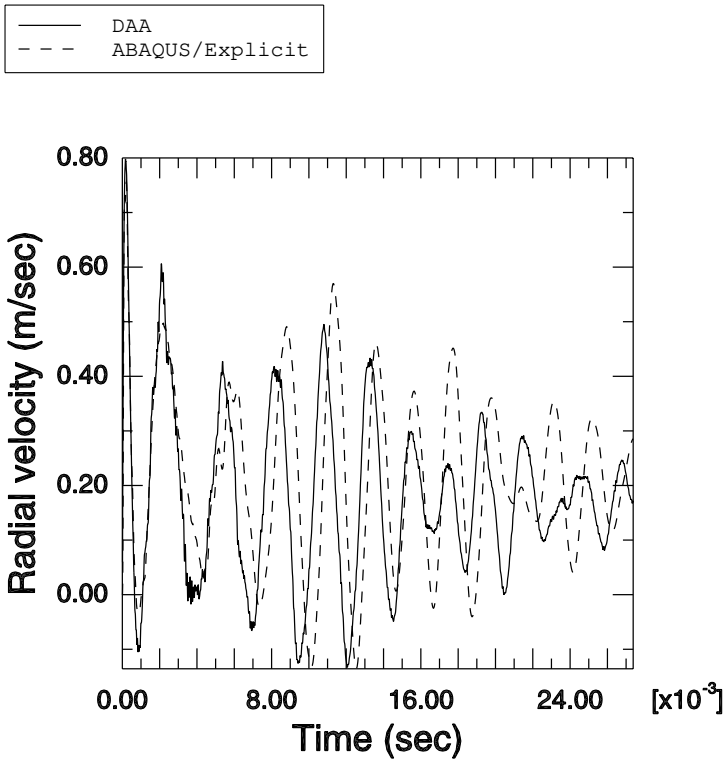


Figure 1: Comparison of the radial velocity at the leading point on the spherical shell obtained with the Doubly Asymptotic Approximation method and with Abaqus/Explicit.

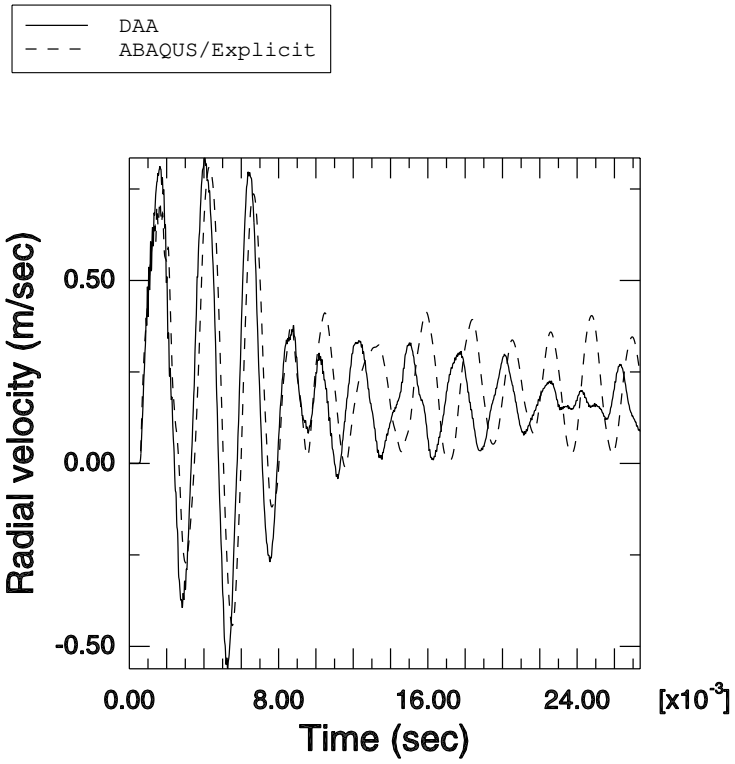


Figure 2: Comparison of the radial velocity at the trailing point on the spherical shell obtained with the Doubly Asymptotic Approximation method and with Abaqus/Explicit.

Air-backed coupled plate response to a planar exponentially decaying wave

Products: Abaqus/Explicit

Overview

This example illustrates the use of Abaqus/Explicit to model the interaction between two fluid-coupled plates and a planar exponentially decaying wave.
The results obtained using Abaqus/Explicit are compared with those obtained independently using the Doubly Asymptotic Approximation (Geers (1978), Abaqus/USA 6.1). This problem has been solved analytically by Schechter and Bort (1981).

Problem description

This problem models the interaction between two fluid-coupled elastic plates and a weak planar exponentially decaying shock wave with a maximum pressure of 1.57 MPa and a decay time of 1.0 ms. The second plate (the plate further from the shock source) is air-backed. In contrast to the solution from Schechter and Bort, engineering material parameters for the fluid and solid media are used. Both plates have a square cross-section of side 1 m and a thickness of 0.016 m. The separation between the plates is 3.2 m. The plates are made of steel with a density of 7850 kg/m^3, a Young's modulus of 210.0 GPa, and a Poisson's ratio of 0.3. The fluid is water with a density of 1026 kg/m^3, in which the speed of sound is 1528 m/s. Each plate is modeled with a single S4R element. A single stack of AC3D8R elements is used to model the fluid in front of the first plate and in between the plates. A planar nonreflective boundary condition is imposed at the end of the outer fluid column using surface impedance. The fluid response is coupled to that of the structure using a tie constraint on the relevant surfaces, with the plate surfaces as main surfaces. The fluid-solid system is excited by a planar exponentially decaying wave applied to the first plate using incident wave loading. A linear bulk viscosity parameter of 0.02 and a quadratic bulk viscosity parameter of 0.5 are used.

Results and discussion

The results from Abaqus/Explicit show good qualitative comparison with those in the referenced literature. We also compare the numerical values for translational velocities of the plates in the direction of the wave obtained using Abaqus/Explicit with those obtained using Abaqus/USA 6.1. As shown in *Figure 1* and *Figure 2*, the results agree closely.

Input files

undex_coupled_plate_air.inp	Input data for this analysis.

References

- Geers, T., “Doubly Asymptotic Approximations for Transient Motions of Submerged Structures,” *Journal of the Acoustical Society of America*, vol. 64, pp. 1500–1508, 1978.
- Schechter, R. S., and R. L. Bort, “The Response of Two Fluid-Coupled Plates to an Incident Pressure Pulse,” *Naval Research Laboratory Memorandum Report*, vol. 4647, October 1981.

Figures

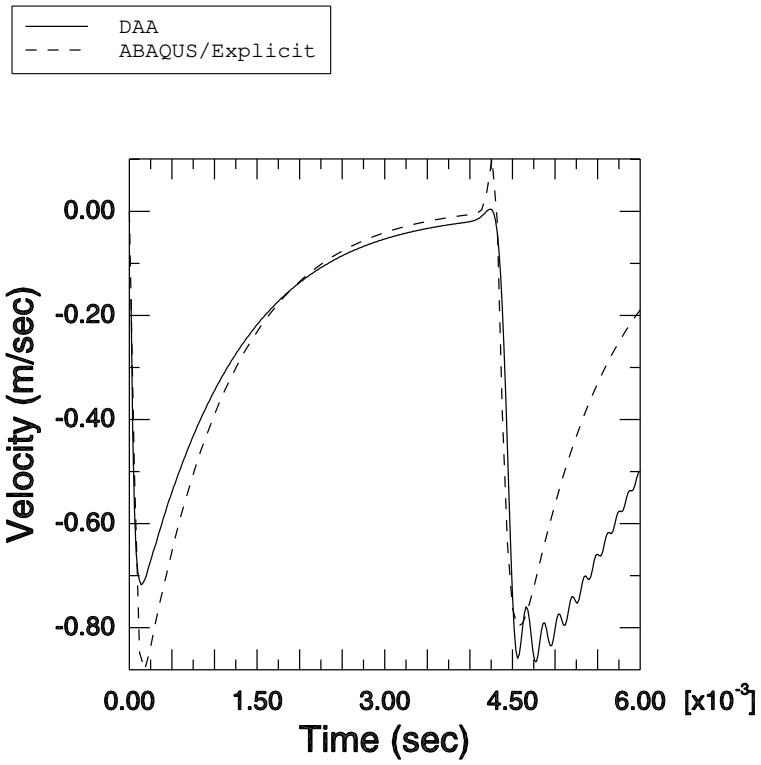


Figure 1: Comparison of the translational velocity of the first plate obtained with the Doubly Asymptotic Approximation method and with Abaqus/Explicit.

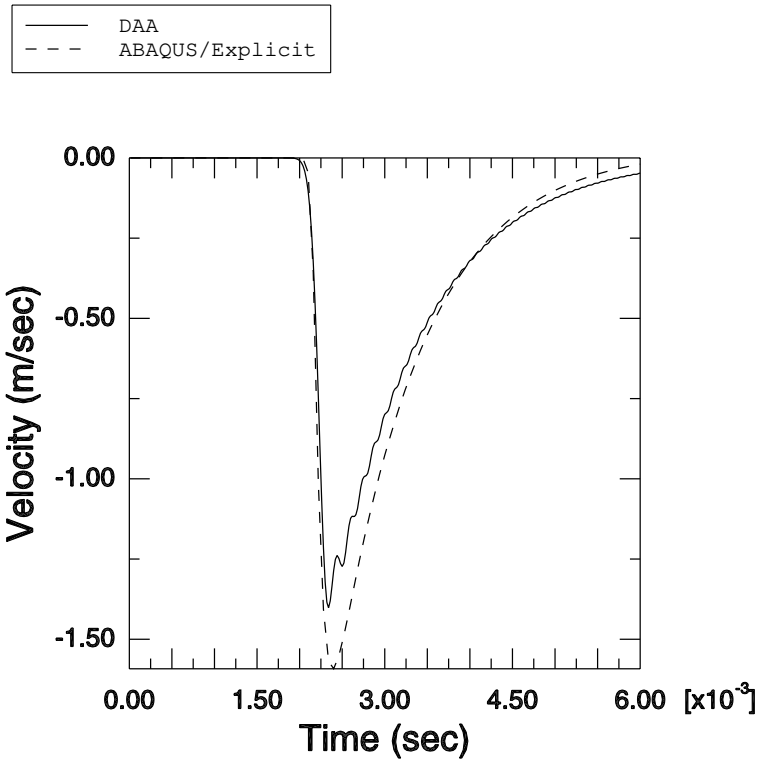


Figure 2: Comparison of the translational velocity of the second plate obtained with the Doubly Asymptotic Approximation method and with Abaqus/Explicit.

Water-backed coupled plate response to a planar exponentially decaying wave

Products: Abaqus/Explicit

Overview

This example illustrates the use of Abaqus/Explicit to model the interaction between two fluid-coupled plates and a planar exponentially decaying wave.
The results obtained using Abaqus/Explicit are compared with those obtained independently using the Doubly Asymptotic Approximation (Geers (1978), Abaqus/USA 6.1). This problem has been solved analytically by Schechter and Bort (1981).

Problem description

This problem models the interaction between two fluid-coupled elastic plates and a weak planar exponentially decaying shock wave with a maximum pressure of 1.57 MPa and a decay time of 1.0 ms. The second plate (the plate further from the shock source) is water-backed. In contrast to the solution from Schechter and Bort, engineering material parameters for the fluid and solid media are used. Both plates have a square cross-section of side 1 m and a thickness of 0.016 m. The separation between the plates is 3.2 m. The plates are made of steel with a density of 7850 kg/m^3, a Young's modulus of 210.0 GPa, and a Poisson's ratio of 0.3. The fluid is water with a density of 1026 kg/m^3, in which the speed of sound is 1528 m/s. Each plate is modeled with a single S4R element. A single stack of AC3D8R elements is used to model the fluid in front of the first plate, behind the second plate, and in between the plates. A planar nonreflective boundary condition is imposed at the ends of the outer fluid columns using surface impedance. The fluid response is coupled to that of the structure using a tie constraint on the relevant surfaces, with the plate surfaces as main surfaces. The fluid-solid system is excited by a planar exponentially decaying wave applied to the first plate using incident wave loading. A linear bulk viscosity parameter of 0.02 and a quadratic bulk viscosity parameter of 0.5 are used.

Results and discussion

The results from Abaqus/Explicit show good qualitative comparison with those in the referenced literature. We also compare the numerical values for translational velocities of the plates in the direction of the wave obtained using Abaqus/Explicit with those obtained using Abaqus/USA 6.1. As shown in *Figure 1* and *Figure 2*, the results agree closely.

Input files

undex_coupled_plate_water.inp	Input data for this analysis.

References

- Geers, T., “Doubly Asymptotic Approximations for Transient Motions of Submerged Structures,” *Journal of the Acoustical Society of America*, vol. 64, pp. 1500–1508, 1978.
- Schechter, R. S., and R. L. Bort, “The Response of Two Fluid-Coupled Plates to an Incident Pressure Pulse,” *Naval Research Laboratory Memorandum Report*, vol. 4647, October 1981.

Figures

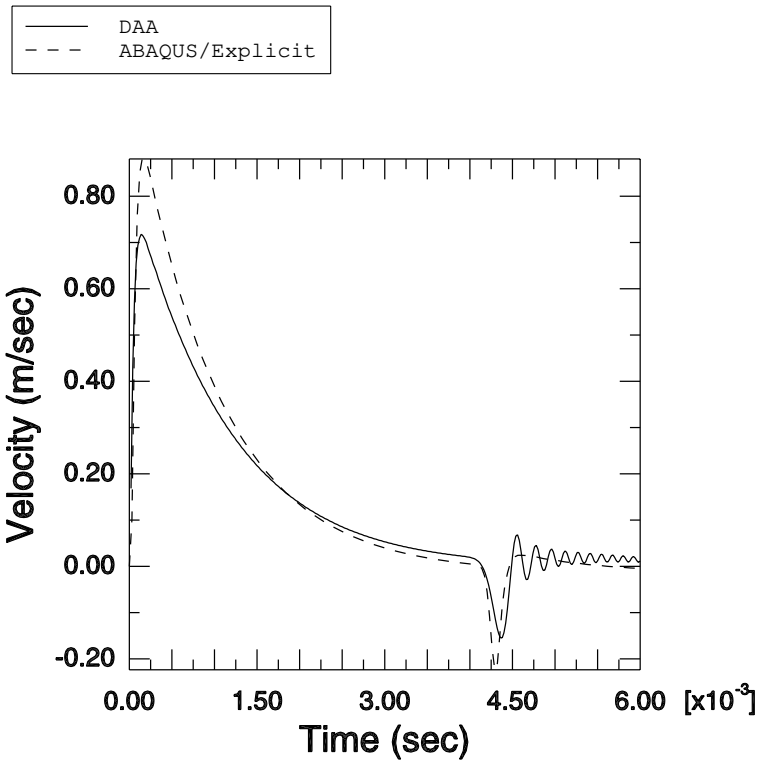


Figure 1: Comparison of the translational velocity of the first plate obtained with the Doubly Asymptotic Approximation method and with Abaqus/Explicit.

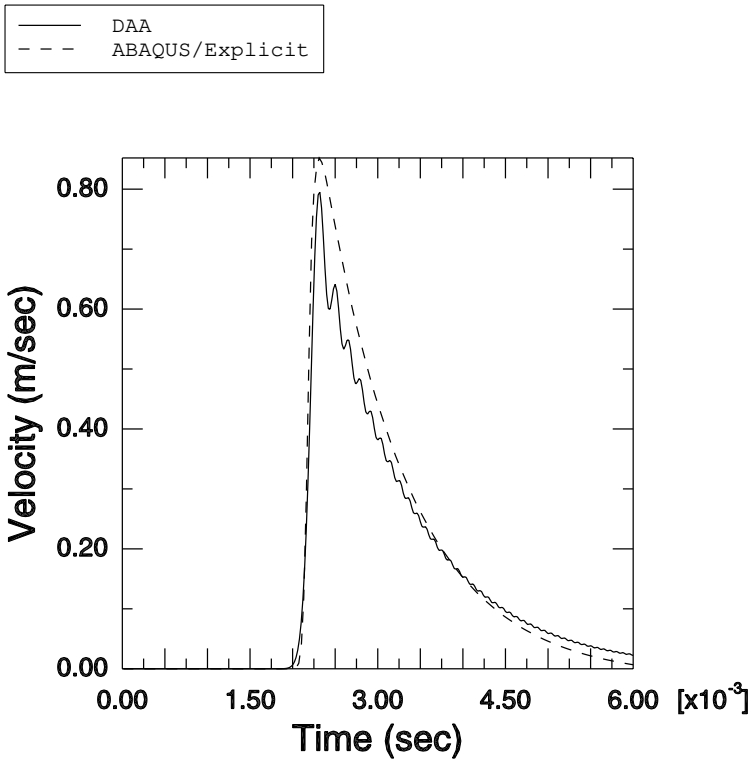


Figure 2: Comparison of the translational velocity of the second plate obtained with the Doubly Asymptotic Approximation method and with Abaqus/Explicit.

Coupled cylindrical shell response to a planar step wave

Products: Abaqus/Explicit

Overview

This example illustrates the use of Abaqus/Explicit to model the interaction between two fluid-coupled concentric cylinders and a planar step wave.
The results obtained using Abaqus/Explicit are compared with those obtained independently using the Doubly Asymptotic Approximation (Geers (1978), Abaqus/USA 6.1). This problem has been solved analytically by Huang (1979).

Problem description

This problem models the interaction between two fluid-coupled concentric elastic cylinders and a weak planar step shock wave with a maximum pressure of 1.0 MPa. The inner cylinder is air-backed. In contrast to Huang's solution, engineering material parameters for the fluid and solid media are used. The inner cylindrical shell has a radius of 0.8 m and a thickness of 23.24 mm, while the outer cylindrical shell has a radius of 1 m and a thickness of 5.81 mm. The shells are made of steel with a density of 7766 kg/m^3, a Young's modulus of 206.4 GPa, and a Poisson's ratio of 0.3. The fluid is water with a density of 997 kg/m^3, in which the speed of sound is 1524 m/s. A half-symmetry model is used. Each cylindrical shell is modeled with 18 S4R elements, with each element spanning 10° in the circumferential direction and 175 mm in the axial direction. Axial symmetry boundary conditions are applied on the edges of the shell elements to represent the infinite axial dimensions of the problem. The fluid in between the cylinders and outside the outer cylinder is meshed with AC3D8R elements, with each acoustic element spanning 10° in the circumferential direction. The exterior fluid region is concentric with the cylinders and has a radius of 2.002 m. A circular nonreflective boundary condition is imposed on the outer surface of the exterior fluid region using surface impedance. The fluid response is coupled to that of the structure using a tie constraint. The fluid-solid system is excited by a planar step wave applied at the outer cylindrical shell using incident wave loading. A linear bulk viscosity parameter of 0.25 and a quadratic bulk viscosity parameter of 10.0 are used.

Results and discussion

The results from Abaqus/Explicit show good qualitative comparison with those in the referenced literature. We also compare the numerical values for radial velocities at the leading edges of the inner and outer cylinders obtained using Abaqus/Explicit with those obtained using Abaqus/USA 6.1. As shown in *Figure 1* and *Figure 2*, the results agree closely.

Input files

undex_coupled_cyl.inp	Input data for this analysis.

References

- Geers, T., “Doubly Asymptotic Approximations for Transient Motions of Submerged Structures,” *Journal of the Acoustical Society of America*, vol. 64, pp. 1500–1508, 1978.
- Huang, H., “Transient Response of Two Fluid-Coupled Cylindrical Elastic Shells to an Incident Pressure Pulse,” *Journal of Applied Mechanics*, vol. 46, pp. 513–518, September 1979.

Figures

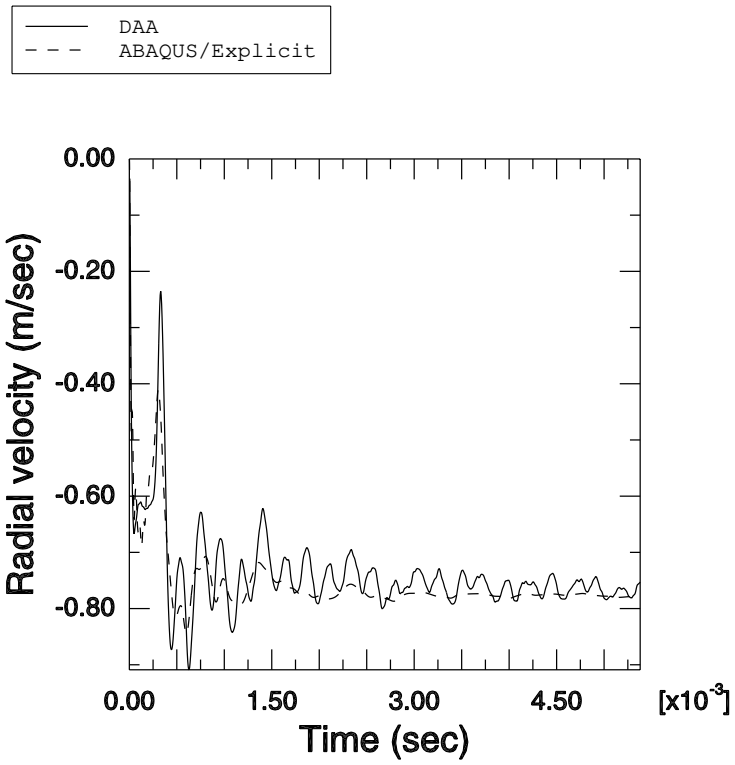


Figure 1: Comparison of the radial velocity at the leading edge of the outer cylindrical shell obtained with the Doubly Asymptotic Approximation method and with Abaqus/Explicit.

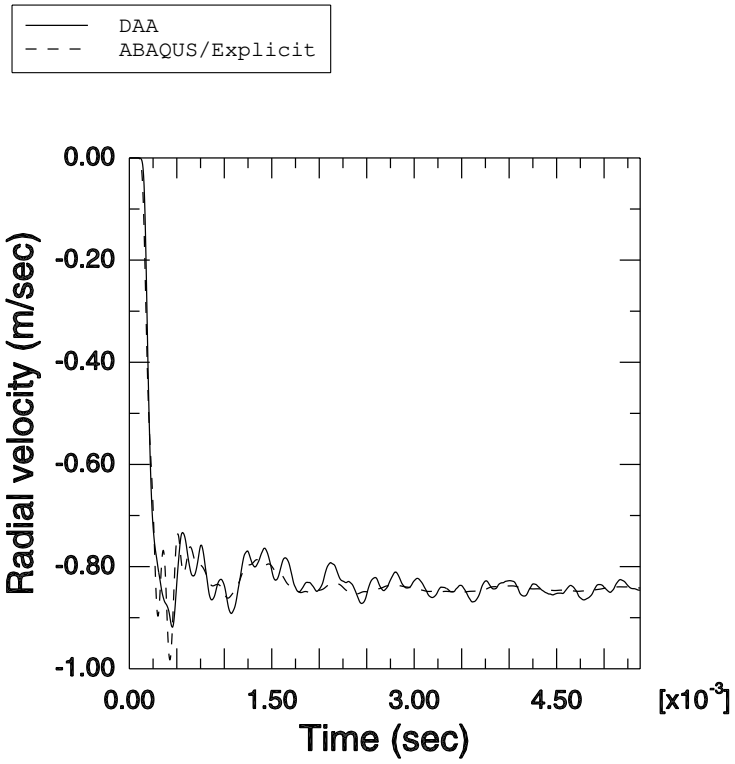


Figure 2: Comparison of the radial velocity at the leading edge of the inner cylindrical shell obtained with the Doubly Asymptotic Approximation method and with Abaqus/Explicit.

Coupled spherical shell response to a planar step wave

Products: Abaqus/Explicit

Overview

This example illustrates the use of Abaqus/Explicit to model the interaction between two fluid-coupled concentric spherical shells and a planar step wave.
The results obtained using Abaqus/Explicit are compared with those obtained independently using the Doubly Asymptotic Approximation (Geers (1978), Abaqus/USA 6.1). This problem has been solved analytically by Huang (1979).

Problem description

This problem models the interaction between two fluid-coupled concentric elastic spherical shells and a weak planar step shock wave with a maximum pressure of 1.0 MPa. The outer cylinder is water-backed. In contrast to Huang's solution, engineering material parameters for the fluid and solid media are used. The inner spherical shell has a radius of 0.8 m and a thickness of 16 mm, while the outer spherical shell has a radius of 1 m and a thickness of 4 mm. The shells are made of steel with a density of 7765 kg/m^3, a Young's modulus of 224.6 GPa, and a Poisson's ratio of 0.3. The fluid is water with a density of 997 kg/m^3, in which the speed of sound is 1524 m/s. An axisymmetric model is used. Each spherical shell is modeled with 64 SAX1 elements. The fluid in between the spherical shells and outside the outer spherical shell is meshed with ACAX4R elements. The exterior fluid region is concentric with the spherical shells and has a radius of 3.002 m. A spherical nonreflective boundary condition is imposed on the outer surface of the exterior fluid region using surface impedance. The fluid response is coupled to that of the structure using a tie constraint on both surfaces of the outer shell and on the outer surface of the inner shell. In both cases the shell surfaces are the main surfaces. The fluid-solid system is excited by a planar step wave applied at the outer spherical shell using incident wave loading. A linear bulk viscosity parameter of 0.2 and a quadratic bulk viscosity parameter of 1.2 are used.

Results and discussion

The results from Abaqus/Explicit show good qualitative comparison with those in the referenced literature. We also compare the numerical values for radial velocities at the leading and trailing edges of the inner spherical shell obtained using Abaqus/Explicit with those obtained using Abaqus/USA 6.1. As shown in *Figure 1* and *Figure 2*, the results agree closely.

Input files

undex_coupled_sph.inp	Input data for this analysis.

References

- Geers, T., “Doubly Asymptotic Approximations for Transient Motions of Submerged Structures,” *Journal of the Acoustical Society of America*, vol. 64, pp. 1500–1508, 1978.
- Huang, H., “Transient Response of Two Fluid-Coupled Spherical Elastic Shells to an Incident Pressure Pulse,” *Journal of the Acoustical Society of America*, vol. 65, pp. 881–887, 1979.

Figures

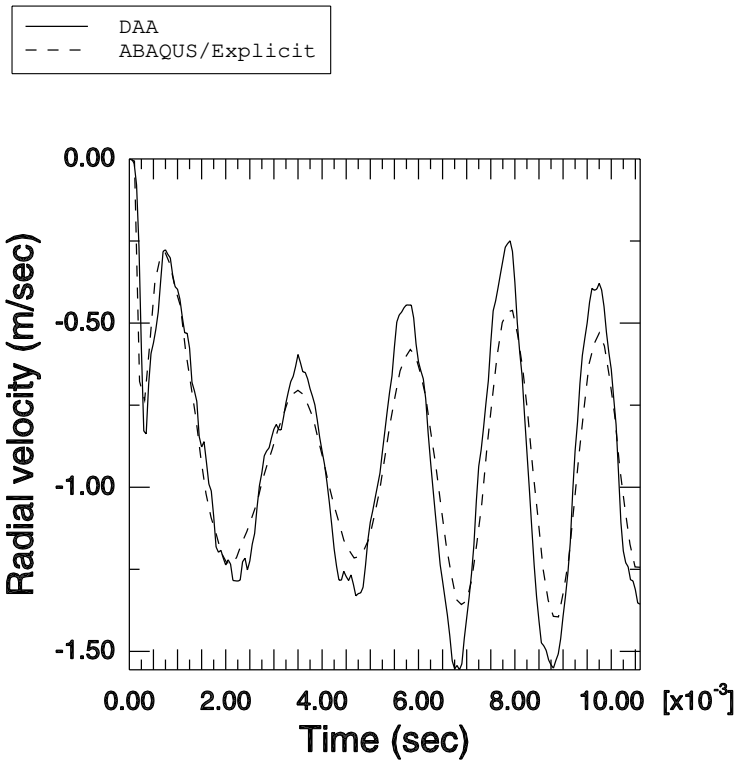


Figure 1: Comparison of the radial velocity at the leading edge of the inner spherical shell obtained with the Doubly Asymptotic Approximation method and with Abaqus/Explicit.

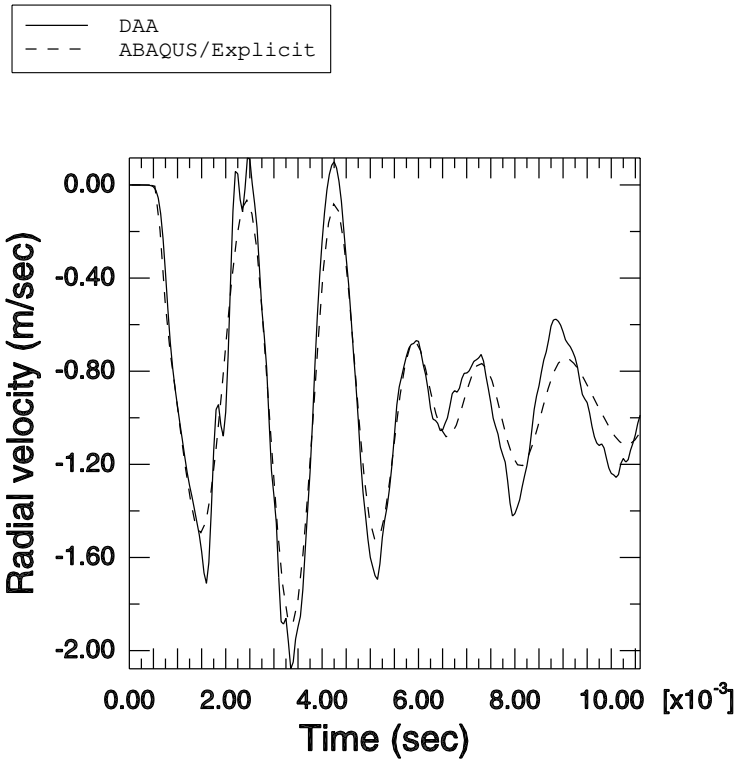


Figure 2: Comparison of the radial velocity at the trailing edge of the inner spherical shell obtained with the Doubly Asymptotic Approximation method and with Abaqus/Explicit.

Fluid-filled spherical shell response to a planar step wave

Products: Abaqus/Explicit

Overview

This example illustrates the use of Abaqus/Explicit to model the interaction between a fluid-filled spherical elastic shell and a planar step wave.
The results obtained using Abaqus/Explicit are compared with those obtained independently using the Doubly Asymptotic Approximation (Geers (1978), Abaqus/USA 6.1). This problem has been solved analytically by Zhang and Geers (1993).

Problem description

This problem models the interaction between a water-backed spherical elastic shell and a weak planar step shock wave with a maximum pressure of 1 MPa. In contrast to the solution from Zhang and Geers, engineering material parameters for the fluid and solid media are used. The sphere has a radius of 1 m and a thickness of 0.01 m. The sphere is made of steel with a density of 7677 kg/m^3, a Young's modulus of 210.0 GPa, and a Poisson's ratio of 0.3. The fluid is water with a density of 997 kg/m^3, in which the speed of sound is 1524 m/s. An axisymmetric model is used for this analysis. The spherical shell is modeled with SAX1 elements, while the enclosed and surrounding fluid is modeled with ACAX4R elements. The inner semicircle that bounds the fluid region is coincident with the shell, and the outer semicircle has a radius of 3 m. A spherical nonreflective boundary condition is imposed on the outer semicircle using surface impedance. The fluid response is coupled to that of the structure using a tie constraint on both sides of the spherical shell, with the shell surfaces as the main surfaces. The fluid-solid system is excited by a planar step wave applied at the point where the semicircular shell intersects the axis of symmetry using incident wave loading. A linear bulk viscosity parameter of 0.2 and a quadratic bulk viscosity parameter of 1.2 are used.

Results and discussion

The results from Abaqus/Explicit show good qualitative comparison with those in the referenced literature. We also compare the numerical values for radial velocities at the leading and trailing points on the shell obtained using Abaqus/Explicit with those obtained using Abaqus/USA 6.1. As shown in *Figure 1* and *Figure 2*, the results agree closely.

Input files

undex_sph_ps_fluid.inp	Input data for this analysis.

References

- Geers, T., “Doubly Asymptotic Approximations for Transient Motions of Submerged Structures,” *Journal of the Acoustical Society of America*, vol. 64, pp. 1500–1508, 1978.
- Zhang, P., and T. L. Geers, “Excitation of a Fluid-Filled, Submerged Elastic Shell by a Transient Acoustic Wave,” *Journal of the Acoustical Society of America*, vol. 93, pp. 696–705, 1993.

Figures

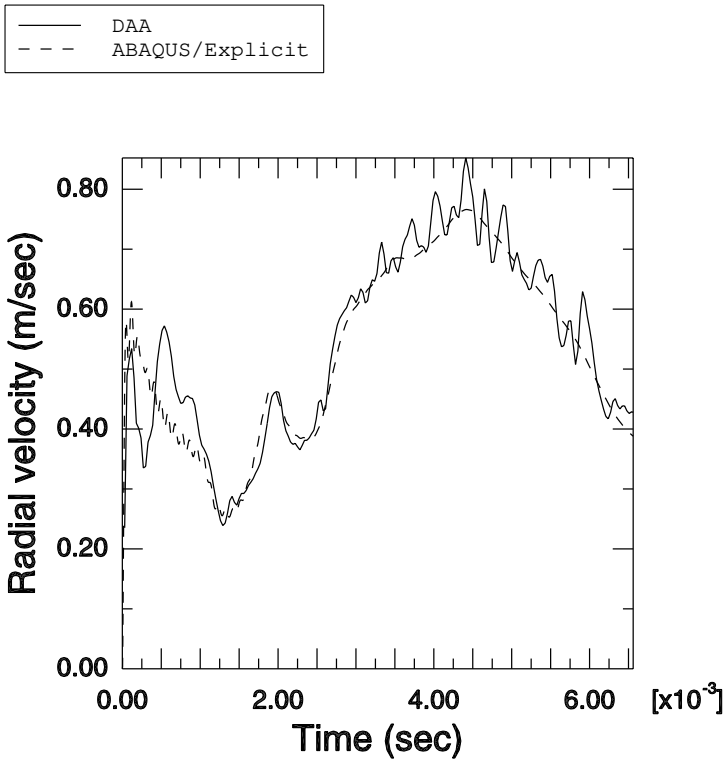


Figure 1: Comparison of the radial velocity at the leading point on the spherical shell obtained with the Doubly Asymptotic Approximation method and with Abaqus/Explicit.

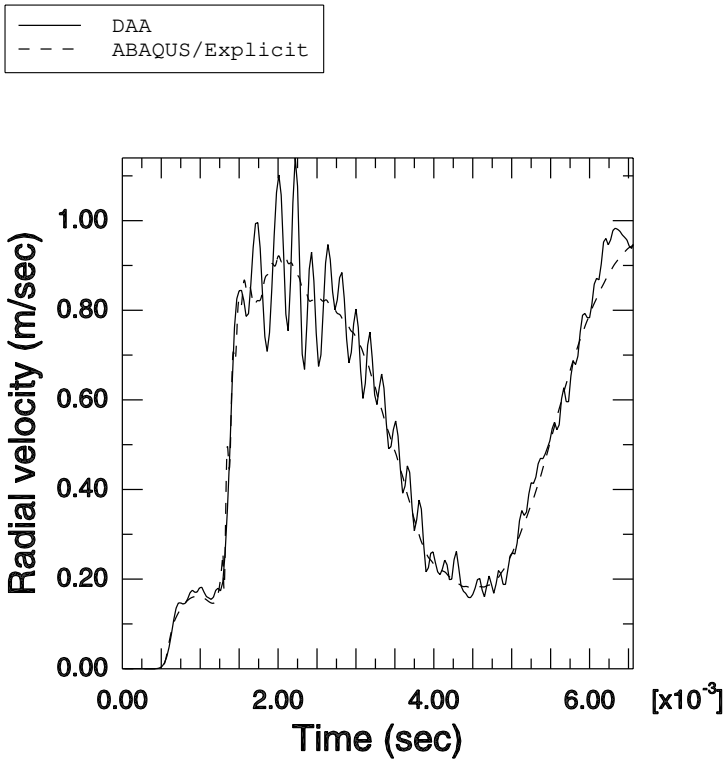


Figure 2: Comparison of the radial velocity at the trailing point on the spherical shell obtained with the Doubly Asymptotic Approximation method and with Abaqus/Explicit.

Response of beam elements to a planar wave

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example illustrates the use of Abaqus/Standard and Abaqus/Explicit to model the response of beam elements with circular cross-sections to a planar linearly increasing wave.
The results obtained using Abaqus are compared with those determined theoretically using the equations of Hicks.

Problem description

To validate the incident wave loading feature, several beams of increasing diameter are subjected to the same incident wave load, athwartships. From Hicks we have expressions describing the actual load on the beam in terms of the cross-sectional area of the beam and the response as a function of the beam's structural mass and the entrained fluid mass. Therefore, the acceleration of a rigid mass is

$$\ddot{\mathbf{u}} = -\mu(r_0)\,\nabla P\frac{1}{\rho_f},$$

where

$$\mu(r_0) \equiv \frac{A}{B} \equiv \frac{(C+\overline{C})\pi r_0^2\rho_f}{m+c_0\pi r_0^2\rho_f}.$$

In these expressions, P is the incident fluid pressure, ρ_f is the fluid mass density, c_0 is the fluid speed of sound, C and $\overline{C}$ are area coefficients for the cross-section, r_0 is the (equivalent) circular radius of the cross-section, and m is the mass of the beam structure. For a beam with a circular cross-section $C \equiv \overline{C} \equiv 0$.

The loading of immersed beams in Abaqus is achieved using incident wave loading. In this test an unconnected array of eight beam elements is subjected to a plane wave incident at 90° from the plane of the array. The amplitude of the wave is linearly increasing, providing a uniform pressure gradient. The beams have identical structural properties, but their wetted cross-sections vary from 0.3 m to 35.0 m. Consequently, the loads generated on the elements due to the incident wave will vary, and the responses will vary due to the different entrained fluid masses. Both Abaqus/Standard and Abaqus/Explicit are tested.

Results and discussion

The acceleration results from Abaqus/Standard and Abaqus/Explicit are summarized in *Table 1*. The results agree closely with the theory.

Input files

iw_bfi_bmk_std.inp	Abaqus/Standard analysis.
iw_bfi_bmk_xpl.inp	Abaqus/Explicit analysis.

References

- Hicks, A. N., "The Theory of Explosion Induced Hull Whipping," *Naval Construction Research Establishment, Dunfermline, Fife, Scotland, Report NCRE/R579, March 1972.*

Tables

Table 1: Finite element results.

r_0	A	B	$\mu(r_0)$	Theoretical Acceleration (Component)	Abaqus/Standard Acceleration (Component)	Abaqus/Explicit Acceleration (Component)
0.3	579.6	3.262e4	1.777e−2	8.17e−9	8.17e−9	8.17e−9
1.0	6440.0	3.555e4	1.811e−1	8.33e−8	8.33e−8	8.33e−8
3.0	5.796e4	6.131e4	9.453e−1	4.35e−8	4.35e−8	4.35e−8
5.0	1.610e5	1.128e5	1.427	6.56e−7	6.56e−7	6.56e−7
7.0	3.155e5	1.901e5	1.660	7.63e−7	7.63e−7	7.63e−7
9.0	5.216e5	2.931e5	1.779	8.18e−7	8.18e−7	8.18e−7
17.0	1.861e6	9.629e5	1.933	8.89e−7	8.89e−7	8.89e−7
35.0	7.889e6	3.977e5	1.9837	9.12e−7	9.12e−7	9.12e−7

Soils analysis

In this section:

- *The Terzaghi consolidation problem*
- *Consolidation of a triaxial test specimen*
- *Finite-strain consolidation of a two-dimensional solid*
- *Limit load calculations with granular materials*
- *Finite deformation of an elastic-plastic granular material*
- *The one-dimensional thermal consolidation problem*
- *Consolidation around a cylindrical heat source*

The Terzaghi consolidation problem

Products: Abaqus/Standard

Overview

This example illustrates the use of the Terzaghi problem to verify the consolidation capability in Abaqus.

This one-dimensional problem has a well-known linear solution (see Terzaghi and Peck, 1948) and, thus, provides a simple verification of the consolidation capability in Abaqus. The analysis of saturated soils requires solution of coupled stress-diffusion equations, and the formulation used in Abaqus is described in detail in *Analysis of porous media* and *Plasticity for nonmetals*. The coupling is approximated by the effective stress principle, which treats the saturated soil as a continuum, assuming that the total stress at each point is the sum of an "effective stress" carried by the soil skeleton and a pore pressure in the fluid permeating the soil. This fluid pore pressure can change with time (if external conditions change, such as the addition of a load to the soil), and the gradient of the pressure through the soil that is not balanced by the weight of fluid between the points in question causes the fluid to flow: the flow velocity is proportional to the pressure gradient in the fluid according to Darcy's law. A typical case is a consolidation problem. Here the addition of a load (usually an overburden) to a body of soil causes pore pressure to rise initially; then, as the soil skeleton takes up the extra stress, the pore pressures decay as the soil consolidates. The Terzaghi problem is the simplest example of such a process. For illustration purposes, the problem is treated with and without finite-strain effects. The small-strain version is the classical case discussed by Terzaghi and Peck (1948), and the finite-strain version has been analyzed numerically by a number of authors, including Carter et al. (1979).

Problem description

The problem is shown in *Figure 1*. A body of soil 2.54 m (100 in) high is confined by impermeable, smooth, rigid walls on all but the top surface. On that surface perfect drainage is possible, and a load is applied suddenly. Gravity is neglected. Because of the boundary conditions, the problem is one-dimensional, the only gradient being in the vertical direction. The purpose of the analysis is to predict the evolution of displacement, effective stress, and pore pressure throughout the soil mass as a function of time following the load application.

Geometry and models



Note: The material properties and loading conditions defined in this problem are only for illustration purposes and are not intended to be representative of any particular geotechnical application.

Abaqus contains no one-dimensional elements for effective stress calculations. Therefore, we use a two-dimensional plane strain mesh, with one element only in the x-direction. Element type CPE4P is chosen to perform the finite-strain analysis, and element type CPE8P is chosen for the small-strain analysis. We recommend the use of linear elements for applications involving finite strain, impact, or complex contact conditions and second-order elements for problems where stress concentrations must be captured accurately or where geometric features such as curved surfaces must be modeled. In this particular example the linear and second-order elements yield almost identical results.

The soil is assumed to be linear elastic, with a Young's modulus of 689.5 GPa (10^8 lb/in^2) and Poisson's ratio of 0.3. The specific weight of the pore fluid is assumed to be 276.8×10^3 N/m^3 (1 lb/in^3). The permeability is assumed to vary linearly with the void ratio, with a value of 8.47×10^{-8} m/sec (2.0×10^{-4} in/min) at a void ratio of 1.5 and a value of 8.47×10^{-9} m/sec (2.0×10^{-5} in/min) at a void ratio of 1.0. The void ratio is assumed to be 1.5 initially throughout the sample. Abaqus uses effective permeability, which is permeability divided by the specific weight of the pore fluid. Therefore, the fluid in this problem is assigned the value 276.8×10^3 N/m^3 (1

lb/in^3) for the specific weight (water, for example, has a specific weight of 9965 N/m^3, 0.036 lb/in^3) and the permeability is scaled accordingly.

The boundary conditions are as follows. On the bottom and two vertical sides, the normal component of displacement is fixed ($u_y = 0$ on the bottom and $u_x = 0$ on the sides), and no flow of pore fluid through the walls is permitted. This latter is the natural boundary condition in the fluid mass conservation equation, so no explicit specifications need to be made (as with zero tractions in the equilibrium equation). On the top surface a uniform downward load (an overburden) is applied suddenly. The magnitude of this load is taken to be 689.5 GPa (10^8 lb/in^2). This large load causes considerable deformation, thus illustrating the difference between the small- and large-strain solutions. This surface allows perfect drainage so that the excess pore pressure is always zero on this surface.

Time stepping

The problem is run in two steps. The first step is a single increment of a transient soils consolidation analysis with an arbitrary time step, with no drainage allowed across the top surface (the natural boundary condition in the mass conservation equation governing the pore fluid flow). This establishes the initial solution: uniform pore pressure equal to the load throughout the body, with no stress carried by the soil skeleton (zero effective stress). The actual consolidation is then done with a second soils consolidation step, using automatic time stepping.

The accuracy of the time integration for the second soils consolidation procedure, during which drainage is occurring, is controlled by specifying the maximum allowable pore pressure change per time step, Δu_w^{max}. Even in a linear problem this value controls the accuracy of the solution, because the time integration operator is not exact (the backward difference rule is used). In this case Δu_w^{max} is chosen as 344.8 GPa (5.0×10^7 lb/in^2), which is a relatively large value and, so, should only give moderate accuracy: this is considered to be adequate for the purposes of the example.

An important issue in such consolidation problems is the choice of initial time step. As the governing equations are parabolic, the initial solution (immediately after the sudden change in load) is a local, "skin effect," solution. In this one-dimensional case the form of the initial solution is sketched in *Figure 2* for illustration purposes. With a finite element mesh of reasonable size for modeling the solution at a later time (when the changes in pore pressure have diffused into the bulk of the body soil), this initial solution will be modeled poorly. With smaller initial time steps the difficulty becomes more pronounced, as sketched in *Figure 2*. 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 give 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 this particular case the issue has been discussed in detail by Vermeer and Verruijt (1981), who suggest the simple criterion

$$\Delta t \geq \frac{\gamma_w}{6Ek}(\Delta h)^2,$$

where Δh is a characteristic element size near the disturbance (that is, near the draining surface in our case), E is the elastic modulus of the soil skeleton, k is the soil permeability, and γ_w is the specific weight of the permeating fluid. For our model we choose $\Delta h =$ 254 mm (10 in); and we have $E =$ 689.5 GPa (10^8 lb/in^2), $k = 8.47 \times 10^{-8}$ m/s (2.0×10^{-4} in/min), $\gamma_w = 2.768 \times 10^5$ N/m^3 (1.0 lb/in^3), which gives $\Delta t \geq$ .05 s (0.833×10^{-3} min). Based on this calculation, an initial time step of .06 sec (0.001 min) is used. This gives an initial solution with no "overshoot" at all, as expected.

In this case we want to continue the analysis to steady-state conditions. This is defined by asking Abaqus to stop when all pore pressure change rates fall below 11.5 KN/m^2/s (100 lb/in^2/min).

Results and discussion

In the small-strain analysis the "steady-state" condition (rate of change of pore pressure with time below the prescribed value) is reached after 20 increments, the last time increment taken being 491 seconds (8.19

min)—about 8000 times the initial time increment. This very large change in time increment size is typical of such diffusion systems and points out the value of using automatic time stepping with an unconditionally stable integration operator for such problems.

The results of the small-strain analysis are summarized in *Figure 3* to *Figure 5*. *Figure 3* shows pore pressure profiles (pore pressure as a function of elevation) at various times in the solution. As we would expect, the solution begins by rapid drainage at the top of the sample and loss of pore pressure in that region. This effect propagates down the sample until the entire sample is steadily losing pore pressure throughout its length. At steady state the solution has zero pore pressure everywhere, with the load being carried as a uniform effective vertical stress. *Figure 4* shows this transfer of load from the fluid to the skeleton at the 1.905 m (75 in) elevation as a function of time. *Figure 5* compares these numerical results with the solution quoted in Terzaghi and Peck (1948). Here the downward displacement of the top surface of the soil, as a fraction of its steady-state value (the "degree of consolidation"), is plotted as a function of normalized time, defined as

$$\text{“time factor”} = \frac{kE}{H^2\gamma_w}t,$$

where k is the permeability of the soil, E is the Young's modulus of the soil, γ_w is the specific weight of the pore fluid, H is the height of the soil sample, and t is time.

Figure 5 shows that the numerical solution agrees reasonably well with the analytical solution, with some loss of accuracy at later times. This latter effect is attributable to the coarse time stepping tolerance chosen. Higher accuracy could be obtained with a tighter tolerance on the allowable pore pressure stress change parameter (Δu_w^{max}). However, the solution is clearly adequate for design use.

In the finite-strain analysis of soils, changes in the void ratio can lead to large changes in permeability, therefore affecting the transient response in a consolidation analysis. Typical soils show a strong dependence of permeability on the void ratio (as soil compacts, it becomes increasingly harder for fluid to pass through it), with the consequence that "plugging" may result. This means that a soil that was relatively permeable in its original state becomes less permeable as it consolidates.

In this example the permeability of the soil is assumed to decrease by an order of magnitude as the void ratio decreases from its initial value of 1.5 to a value of 1.0. Such logarithmic dependence of permeability on the void ratio is common in fully saturated clays. Two finite-strain analyses are run, one with permeability treated as a constant and a second with this variation in permeability. The results are shown in *Figure 6*, together with the results of the small-strain analysis under similar load. The "plugging" effect of void ratio dependence of permeability is clearly seen in this figure. Since the permeability decreases with the consolidation of the soil, the time required for all excess pore pressure to dissipate increases. The final value of displacement under the applied load is not a function of permeability and is correctly predicted by both large-strain analyses. However, this displacement is overpredicted by the small-strain analysis, indicating that nonlinear geometric effects are important and the small-strain assumption is not applicable with such a large load magnitude. (The exact solution for this displacement is very easily calculated.) It is interesting to observe that, if the permeability is not dependent on the void ratio, the finite-strain results show more rapid initial consolidation than the corresponding small-strain analysis.

A separate suite of files (*terzaghi_cpe8p_rigid.inp*, *terzaghi_cpe4p_rigid.inp*, and *terzaghi_cpe8p_ss_rigid.inp*) is provided to illustrate the use of a contact pair in problems involving pore pressure elements. Three rigid surfaces are used to model the three impermeable sides of the specimen shown in *Figure 1*, thus replacing the boundary conditions used in *terzaghi_cpe8p.inp*, *terzaghi_cpe4p.inp*, and *terzaghi_cpe8p_ss.inp*.

Input files

terzaghi_cpe8p.inp	Small-strain analysis (element type CPE8P).
terzaghi_cpe4p.inp	Finite-strain case with permeability depending on the void ratio (element type CPE4P).
terzaghi_cpe8p_ss.inp	Small-strain steady-state solution (element type CPE8P).

terzaghi_cpe8p_perm.inp	Small-strain case with velocity-dependent permeability (Forchheimer flow) and velocity coefficient depending on the void ratio.
terzaghi_postoutput1.inp	**POST OUTPUT* postprocessing of terzaghi_cpe8p.inp.
terzaghi_postoutput2.inp	**POST OUTPUT* postprocessing of terzaghi_cpe8p_perm.inp.
terzaghi_cpe8p_rigid.inp	Identical to terzaghi_cpe8p.inp except that rigid surfaces are used to impose the boundary conditions.
terzaghi_cpe4p_rigid.inp	Identical to terzaghi_cpe4p.inp except that rigid surfaces are used to impose the boundary conditions.
terzaghi_cpe8p_ss_rigid.inp	Identical to terzaghi_cpe8p_ss.inp except that rigid surfaces are used to impose the boundary conditions.

References

- Carter, J. P., J. R. Booker, and J. C. Small, "The Analysis of Finite Elasto-Plastic Consolidation," *International Journal for Numerical and Analytical Methods in Geomechanics*, vol. 3, pp. 107–129, 1979.
- Terzaghi, K., and R. B. Peck, Soil Mechanics in Engineering Practice, John Wiley and Sons, New York, 2nd, 1948.
- 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

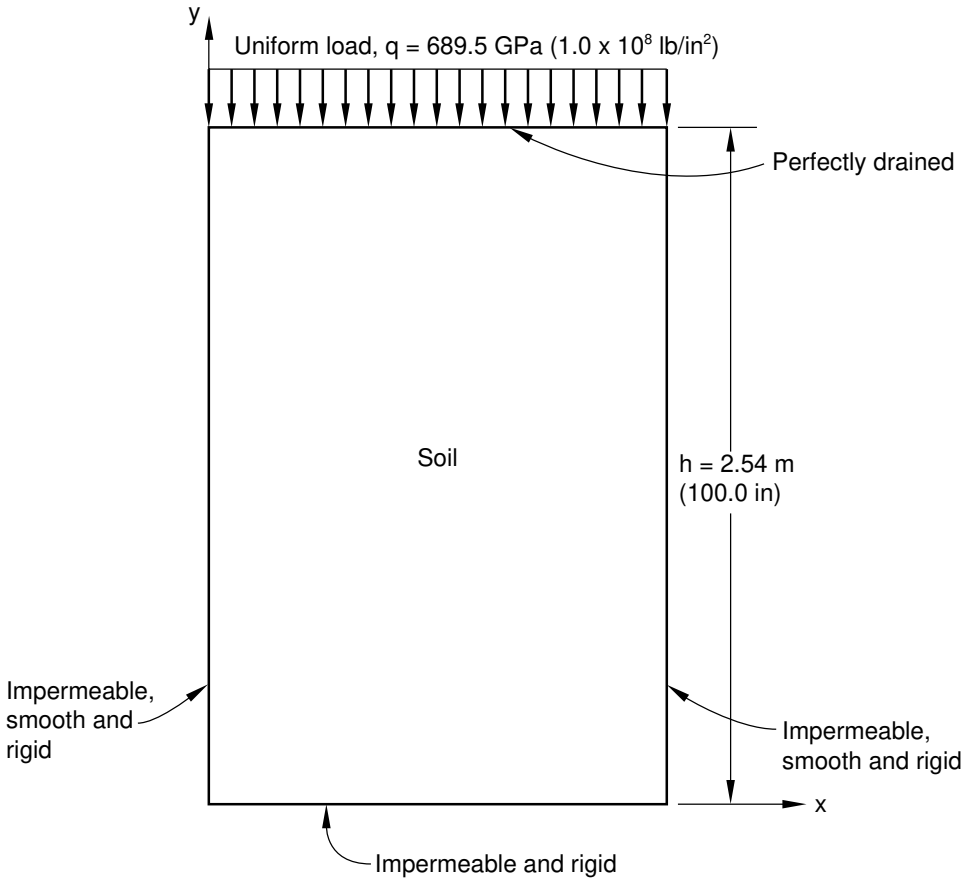


Figure 1: Terzaghi consolidation problem definition.

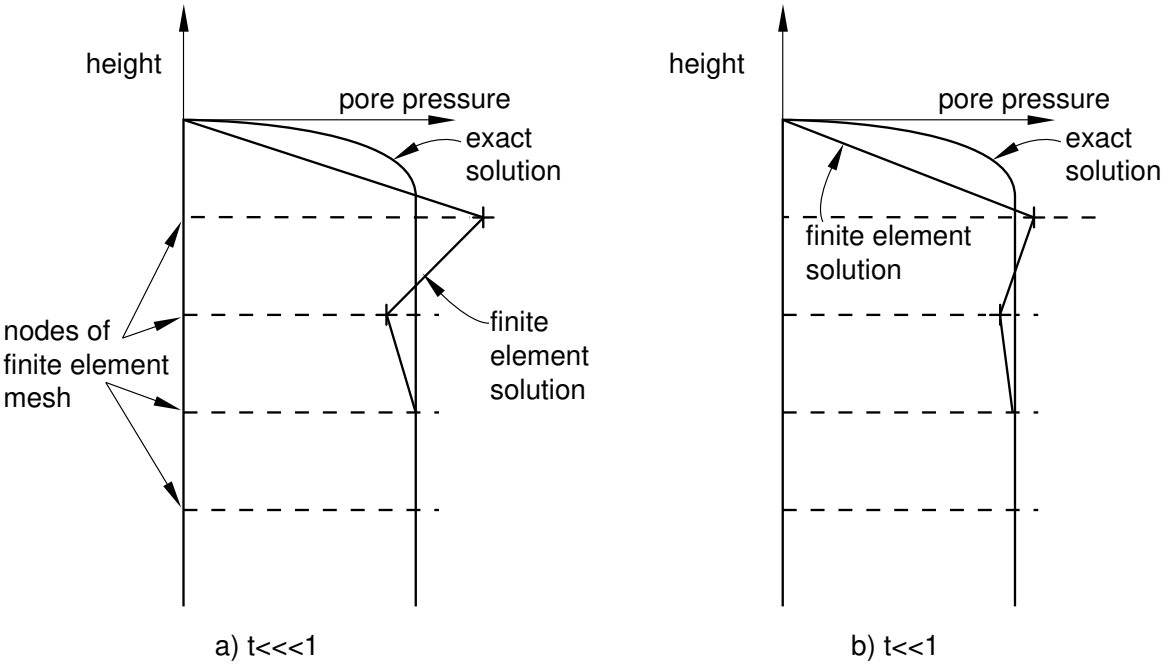


Figure 2: Solutions at very early times.

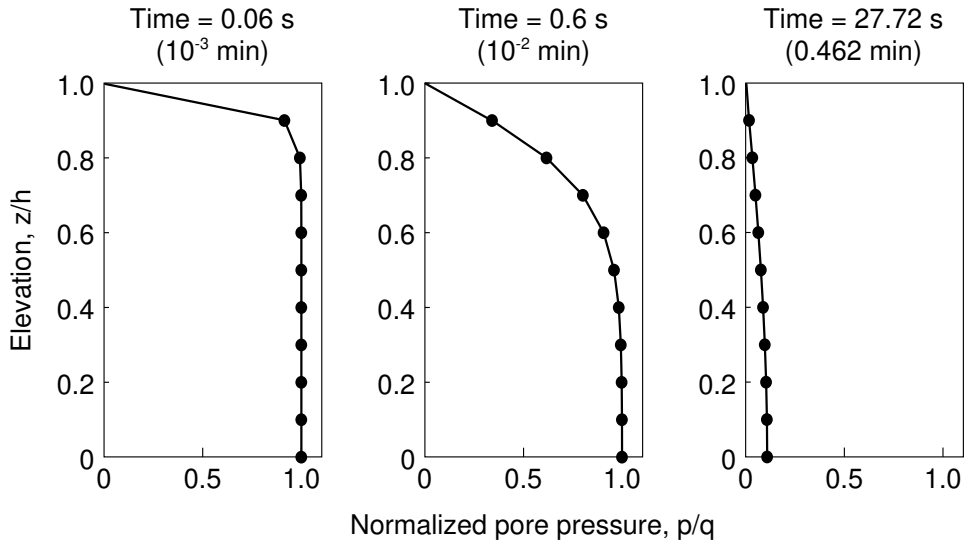


Figure 3: Pore pressure at various times.

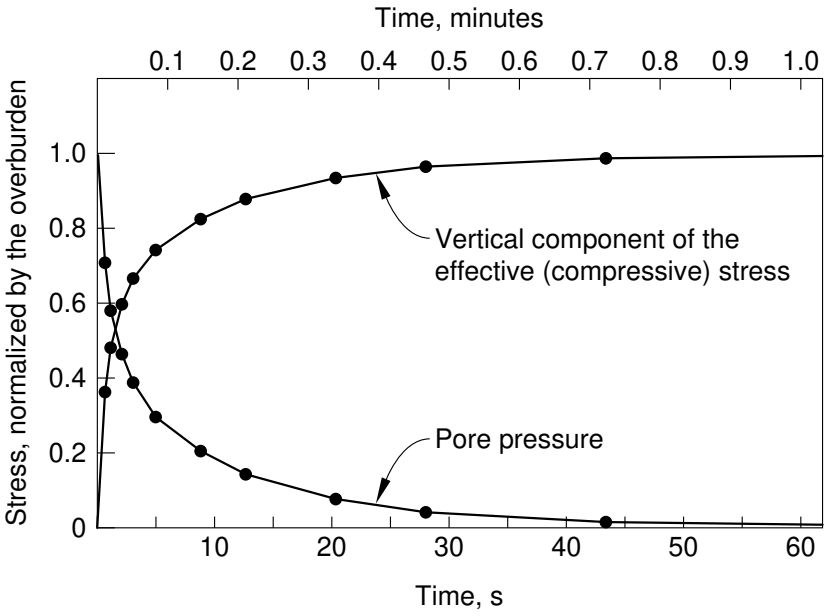


Figure 4: Pore pressure and effective stress at elevation 1.905 m (75 in).

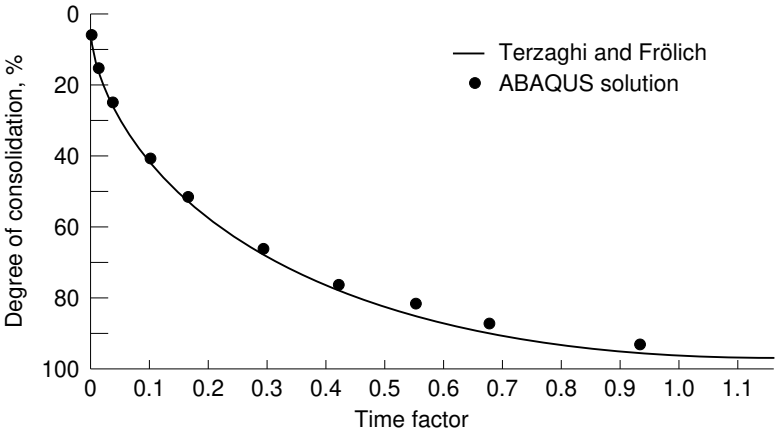


Figure 5: Degree of consolidation versus time factor.

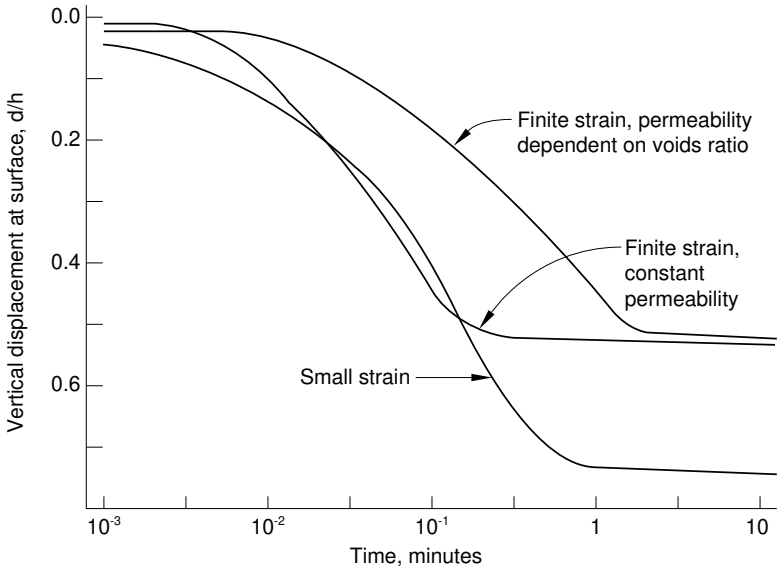


Figure 6: Comparisons of finite- and small-strain solutions to the Terzaghi consolidation problems.

Consolidation of a triaxial test specimen

Products: Abaqus/Standard

Overview

This example illustrates inelastic deformation of a soil specimen whose constitutive behavior is modeled with modified Cam-clay plasticity.
The elastic part of the behavior is modeled with both the linear elastic and porous elastic models. The Cam-clay plasticity theory, which is one of the critical state plasticity theories developed by Roscoe and his colleagues at Cambridge, is described in *Plasticity for nonmetals*. Verification of the model is provided by *Triaxial tests on a saturated clay*.

The geometric configuration is one of the most common soils tests: a triaxial specimen, confined by an enclosing membrane, being squeezed axially between platens (see *Figure 1*). Perfectly smooth and perfectly rough platens are both considered. The platen motion is assumed to be very slow compared to characteristic diffusion times in the soil, and the platen is assumed to provide perfect drainage, so that the pore pressures throughout the soil specimen are always essentially zero. Pore fluid diffusion is, thus, not a significant effect in this case. See *The Terzaghi consolidation problem* and *Plane strain consolidation* for cases where transient effects in the pore fluid diffusion are an important aspect of the overall response.

As the specimen is compressed, the elastic-plastic response of the specimen consists of two competing effects. Elastically, the increased compressive hydrostatic effective stress on the soil skeleton causes a stiffening of the response. When the soil yields, inelastic deformation results in softer behavior. Eventually the stress state in some region of the specimen reaches critical state, where the soil skeleton response is perfectly plastic. When this region is sufficiently developed, a limit state is attained, and the specimen's resistance to further compression no longer increases. The analysis is intended to track the response from the initial loading to this limit state.

Problem description

The soil sample is an axisymmetric cylinder, as shown in *Figure 1*. The model takes advantage of symmetry about the midplane, as well as the axisymmetry of the configuration. The specimen has a length to diameter ratio of 3. Two cases are considered: one in which the platen is assumed to be perfectly smooth, so that the stress state in the specimen will be homogeneous, and one in which the platen is assumed to be completely rough, so the soil in contact with the platen cannot move with respect to the platen. This latter case results in an nonhomogeneous stress state, as the specimen bulges during compression. The eight element mesh shown in *Figure 1* is not expected to capture this nonhomogeneous state accurately but should suffice for the present demonstration purposes.

The material properties of the soil are based on the example used by Zienkiewicz and Naylor (1972). The properties for the Cam-clay model with porous elasticity are shown in *Figure 1*. The Cam-clay model with linear elasticity uses a Young's modulus of 15 GPa. This value is based on the elastic stiffness (at the end of the loading step) of the examples that use porous elasticity.

Initial conditions

The Cam-clay model assumes that the soil has no stiffness at zero stress, so that some initial (compressive) stress state must be defined for the material. In this case we assume that the soil sample is under an initial hydrostatic pressure of 0.1 MPa (14.5 lb/in^2), and this confining pressure remains constant throughout the test. Since the soil may drain through the platen, this pressure is carried as an effective stress in the soil skeleton. This initial stress state is defined using initial conditions. In this particular example it is trivial to see that this initial stress state is in equilibrium with the external distributed pressure of the same magnitude. In more complex cases it

may not be so simple to ensure that the discrete, finite element model is in equilibrium with the geostatic loading. Accordingly, the first step of any analysis involving an initial stress state should be a geostatic step. In that step the geostatic external loads (in this case the pressure on the specimen) should be specified. Abaqus will then check whether the initial stress state is in equilibrium with these loads. If it is not, Abaqus will iterate and attempt to establish an equilibrium stress field that balances the prescribed tractions. Such iteration does not occur in this case since the prescribed initial stress is in equilibrium with the applied tractions.

Loading

The specimen is compressed to 40% of its initial height over 34.56×10^6 sec (400 days). Although this represents a large strain of the specimen, geometric nonlinearity is ignored in this example because we wish to examine the effects of the material nonlinearity, and we only report the stress-strain response at points, rather than overall load-deflection response that will be predicted quite inaccurately unless geometric nonlinearity is included. The loading is applied in a transient soils consolidation step specifying the time period, with an associated boundary condition prescribing the travel of the platen during that time. The platen is assumed to drain freely throughout the analysis. This is specified by a boundary condition, fixing the pore pressure at zero on the top edge of the mesh. The loading is intended to represent very slow compression, sufficiently slow that the pore pressures never rise to any significant values. We can obtain a rough idea of this time scale by noting that a characteristic time for pore pressure dissipation is $T = H^2\gamma_\omega / kE$, where H is a typical dimension from the draining surface (60 mm, 2.362 in, in this case); γ_ω is the specific weight of the pore fluid (1.0×10^4 N/m^3, 0.0369 lb/in^3); k is the permeability of the soil (0.1728 mm/day, 6.803×10^{-3} in/day); and E is a typical soil modulus, which we compute as $E = p/\kappa$, where κ is the logarithmic bulk modulus and p is a typical mean normal effective stress. T is, thus, estimated as 0.05 days. This is about the time it takes for pore pressures to drop to 5% of their initial values, following sudden application of a load (see Terzaghi and Peck, 1967). Since the time scale chosen for the loading of the test specimen in this example is very long compared to this value, no significant pore pressures should ever arise in the analysis.

The same analysis could be performed by using a static procedure, in which case the coupled, effective stress formulation element type could be replaced with an element that models soil deformation only. We choose to use the coupled element type and the soils consolidation procedure to exercise these features.

The accuracy of the equilibrium solution within a time increment is controlled by iterating until the out-of-balance forces reduce to a small fraction of an average force magnitude calculated internally by Abaqus. The rough platen causes an nonhomogeneous stress state, which tends to cause an underestimation of this average force magnitude since stresses are locally higher in the region of the mesh near the platen and the reference force magnitude is averaged over the entire mesh. To avoid iterating to excessive accuracy, we have overridden the default calculation of the average force magnitude and have defined that typical actual nodal forces will be of the order 100 N (22.52 lb). This is done using solution controls. The increment size choice is automatic, determined by allowing a maximum pore pressure change (Δu_w^{max}) of 0.16 KPa (.023 lb/in^2) per increment, which should give sufficient definition of the solution.

Results and discussion

Figure 2 shows results for the rough platen case, when the stress field is nonhomogeneous, and shows results corresponding to point *A* in *Figure 1*: this is the stress output point at the centroid of the element shown. *Figure 3* is for the smooth platen case, when the stress field is homogeneous. The top section of each figure shows the p–q plane. Here p is the equivalent effective pressure stress, defined by

$$p = -\frac{1}{3}\mathrm{trace}\,(\sigma)\,,$$

and q is the equivalent deviatoric stress (the Mises equivalent stress) defined by

$$q = \sqrt{\tfrac{3}{2}\mathbf{S} : \mathbf{S}},$$

where

$$\mathbf{S} = \boldsymbol{\sigma} + p\mathbf{I}$$

is the deviatoric stress (here $\mathbf{I}$ is a unit matrix).

The $(p - q)$ plot in each case shows the critical state line, the initial yield surface, and the stress trajectory followed in the solution. The bottom section of each figure is a plot of the equivalent deviatoric stress, q, versus the vertical deflection of the platen. The behavior in both cases is as we would expect: a gradual softening of the specimen after yield, until critical state is reached, when the behavior becomes perfectly plastic. In the rough platen case, the response at the point plotted moves some way up the critical state line after it reaches that line: presumably this is because other points in the model have not yet reached the limit state.

Similar results are obtained for the Cam-clay model with linear elasticity.

Input files

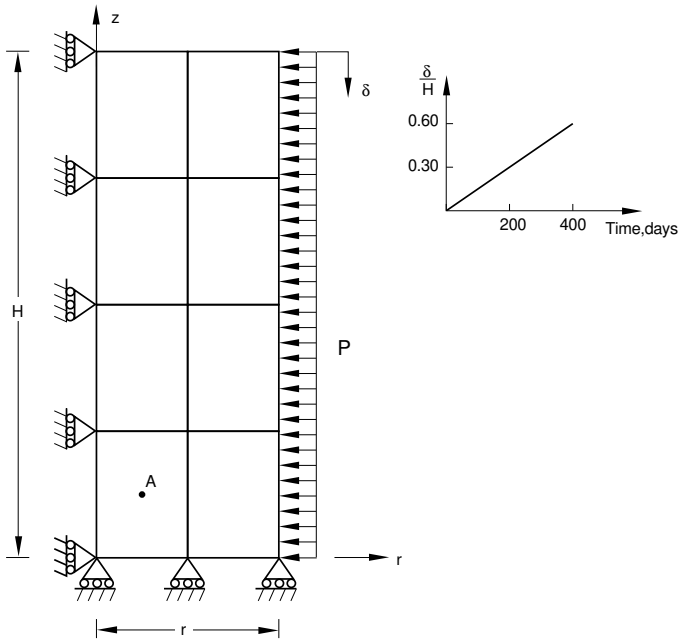
triaxconsolid_cax8rp_por.inp	Rough platen case using the porous elasticity model with CAX8RP elements.
triaxconsolid_caxa8rp1_por.inp	Rough platen case using the porous elasticity model with CAXA8RP1 elements. This analysis is done as basic verification of this element type.
triaxconsolid_cax8rp_lin.inp	Rough platen case using the linear elasticity model with CAX8RP elements. This analysis is done as basic verification of the Cam-clay model with linear elasticity.

The only change needed for the smooth platen case is to remove the boundary conditions in the radial direction at the top of the mesh.

References

- Terzaghi, K., and R. B. Peck, Soil Mechanics in Engineering Practice, John Wiley and Sons, New York, 2nd, 1967.
- Zienkiewicz, O. C., and D. J. Naylor, “The Adaptation of Critical State Soil Mechanics Theory for Use in Finite Elements,” *Stress-Strain Behavior of Soils*, edited by R. H. G. Parry, G. T. Foulis and Co., Ltd., London, 1972.

Figures



Material properties:

Elastic properties:

$\upsilon = 0.3$
$\kappa = 0.026$

Plastic constants:

$\lambda = 0.174$
$M = 1.0$
$a_0 = 58.3$ kPa

Soil permeability:

$k = 1.728 \times 10^{-4}$ m/day

Specific wt. of fluid:

$\gamma_w = 1.0 \times 10^4$ N/m^3

Initial voids ratio:

$e_0 = 1.08$

Geometry:

$H = 60$ mm
$r = 20$ mm

Loading:

$P = 100$ kPa .

Figure 1: Triaxial consolidation: geometry, properties, and loading.

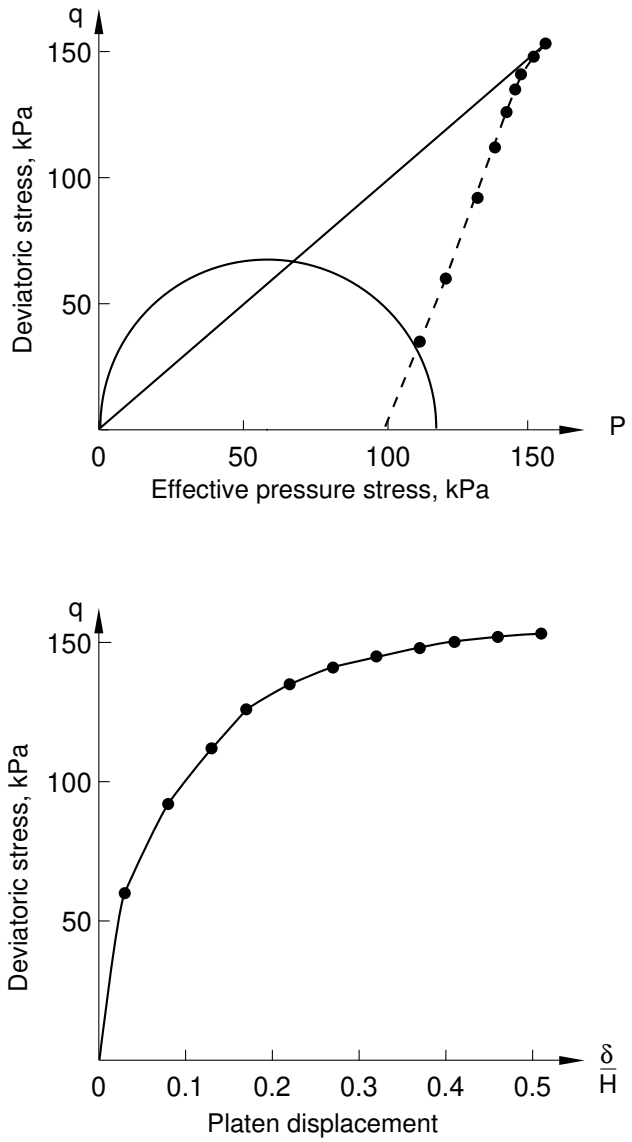


Figure 2: Shear stress versus mean normal stress and axial strain. Rough platen case.

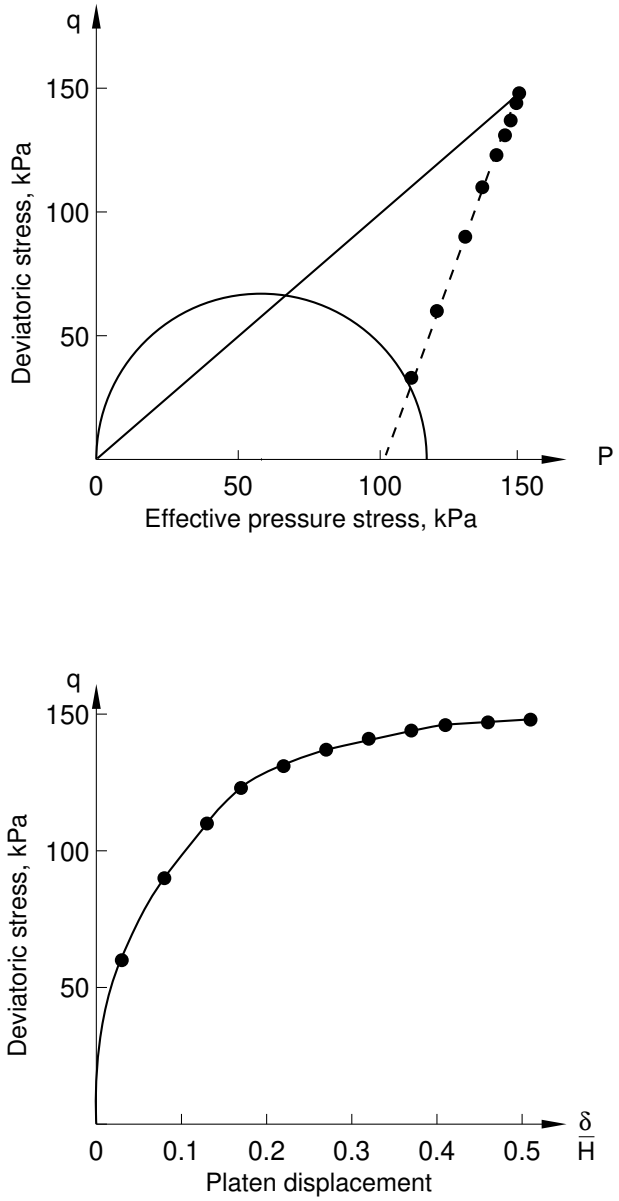


Figure 3: Shear stress versus mean normal stress and axial strain. Smooth platen case.

Finite-strain consolidation of a two-dimensional solid

Products: Abaqus/Standard

Overview

This example illustrates the large-scale consolidation of a two-dimensional solid.
Nonlinearities caused by the large geometry changes are considered, as well as the effects of the change in the void ratio on the permeability of the material. The material is assumed to be linear elastic. The example exhibits many features in common with the one-dimensional Terzaghi consolidation problem discussed in *The Terzaghi consolidation problem*, notably the reduced settlement magnitudes predicted by finite-strain analysis in comparison with the results provided by small-strain theory.

Problem description

The example considers a finite strip of soil, loaded over its central portion. Symmetry permits modeling half the strip, as shown in *Figure 1*: the half-width of the strip is equal to its height, and the ratio of the loaded portion of the strip to the width of the strip is 1:5. The finite element discretization used is also shown in *Figure 1*: 35 CPE8RP elements are used, and the mesh is graded in the vertical direction in length ratios 1:2:3:4:5 and horizontally in ratios 1:1:1:2:2:4:4. This is a coarse mesh, but it is expected to provide representative results. A similar mesh using CPE4P elements is included for verification purposes.

Figure 1 also summarizes the material properties and boundary conditions used. The ratio of the pressure load to the Young's modulus is 1:2, and the Poisson's ratio is specified as 0. The soil is assumed to have an initial void ratio of 1.5, and the permeability at this value of the void ratio is 0.508 μm/sec (2.0×10^{-5} in/sec). The permeability is assumed to be one order of magnitude smaller at a void ratio of 1.0. These low permeability values are representative of clays.

The strip of soil is assumed to lie on a rigid, impermeable, smooth base. No horizontal displacement or pore fluid flow is permitted along the vertical sides of the model. Free drainage is assumed along the top surface of the model.

Loading and time stepping

The analysis is performed using two transient soils consolidation steps. In the preliminary step the full load is applied over two equal fixed time increments. The load remains constant in the subsequent step during which the soil undergoes consolidation.

In the analysis accounting for finite-strain effects, the preliminary step requires six iterations for convergence of the first increment and seven iterations for convergence at full load. These relatively large numbers of iterations are due to the large geometry changes experienced by the soil. As shown in *Figure 2*, at full load the midpoint vertical deflection in this case is about 0.49 times the width of the strip that is loaded. The geometrically linear analysis predicts the midpoint vertical deflection to be approximately 0.52 times the width of the strip that is loaded.

Practical consolidation analyses require solutions across several orders of magnitude of time (see *Figure 2*, for example), and the automatic time stepping scheme is designed to generate cost-effective solutions for such cases. The algorithm is based on the user supplying a tolerance on the pore pressure change permitted in any increment, Δu_w^{max}. Abaqus uses this value in the following manner: if the maximum change in pore pressure at any node is greater than Δu_w^{max}, the increment is repeated with a proportionally reduced time step. If the maximum change in pore pressure at any node is consistently less than Δu_w^{max}, the time step is increased. In this case Δu_w^{max} is

set to 0.103 MPa (15 lb/in^2). This represents about 3% of the maximum pore pressure in the model following application of the load. With this value the first time increment is 7.2 seconds, and the final time increment is 1853 seconds. This is quite typical of diffusion processes: at early times the time rates of pore pressure are significant, and at later times these time rates are very low.

Results and discussion

The first analysis considers finite-strain effects, and the soil permeability varies with the void ratio. This change of permeability with the void ratio is physically realistic—as soil is compressed, it becomes harder for pore fluid to flow through it. A small-strain analysis is also run with constant permeability. The midpoint settlement versus time for the finite- and small-strain analyses are shown in *Figure 2*. The two analyses predict large differences in the final consolidation: the small-strain result shows about 40% more deformation than the finite-strain case. This is consistent with results from the one-dimensional Terzaghi consolidation solutions—(see *The Terzaghi consolidation problem*). Quite clearly, in cases where settlement magnitudes are significant, finite-strain effects are important.

The time scale in *Figure 2* spans five orders of magnitude, pointing to the importance of automatic time incrementation for cost-effective solutions.

Figure 3 shows time histories of pore pressure at two points in the model, points *a* and *b* in *Figure 1*. The pore pressure results are normalized by the value of pore pressure at these points at the end of the preliminary step and are taken from the finite-strain analysis. The increase in pore pressure shown at point ***b*** is evidence of the Mandel-Cryer effect (see Prevost, 1981 and Lambe and Whitman, 1969) and is typical of two- and three-dimensional consolidation analysis.

Input files

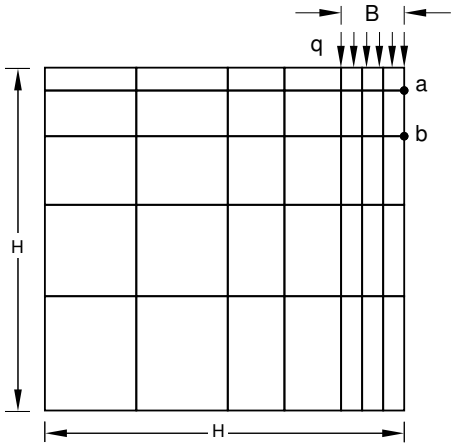
finstrainconsolid2d_node.f	Generates the nodal coordinates required by finstrainconsolid2d_cpe8rp.inp.
finstrainconsolid2d_cpe8rp.inp	Finite-strain analysis (element type CPE8RP).
finstrainconsolid2d_cpe4p.inp	Element type CPE4P.
finstrainconsolid2d_autostab.inp	Input data with automatic stabilization added.
finstrainconsolid2d_autostab_adap.inp	Input data with adaptive automatic stabilization added.

The small-strain case is obtained by removing the NLGEOM parameter on the **STEP* option.

References

- Lambe, T. W., and R. V. Whitman, Soil Mechanics, John Wiley and Sons, New York, 1969.
- Prevost, J. H., "Consolidation of an Elastic Porous Media," *Journal of the Engineering Mechanics Division, ASCE*, vol. 107, pp. 169–186, February 1981.

Figures



H = 1.524 m (60.0 in)

B = 304.8 mm (12.0 in)

Material:

Young's modulus = 6.895 MPa (1.0 x 10^3 lb/in^2)

Poisson's ratio = 0.0

Initial void ratio = 1.5

Permeability = 5.08 x 10^{-7} m/s (2.0 x 10^{-5} in/s)
at void ratio = 1.5

Permeability = 5.08 x 10^{-8} m/s (2.0 x 10^{-6} in/s)
at void ratio = 1.0

Loading:

Pressure = q = 3.4475 MPa (500.0 lb/in^2)

Boundary conditions:

Free drainage across top surface

Other surfaces impermeable and smooth

Figure 1: Two-dimensional elastic consolidation problem description.

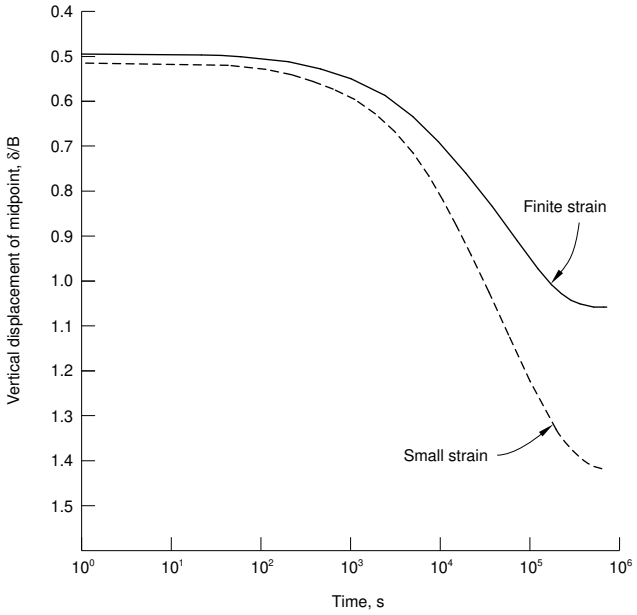


Figure 2: Midpoint settlement time history.

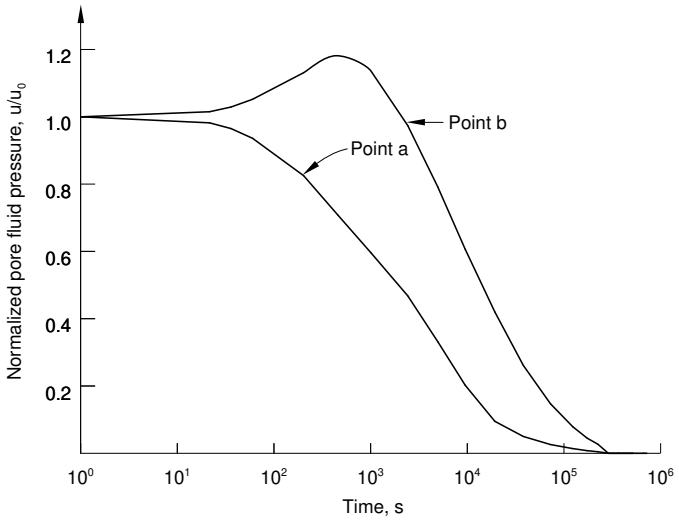


Figure 3: Pore pressure time history.

Limit load calculations with granular materials

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example shows solutions to limit load calculations for a strip of sand loaded by a rigid, perfectly rough footing. The foundation material is defined as a Mohr-Coulomb material; therefore, we show results obtained with the Mohr-Coulomb plasticity model available in Abaqus. The example also shows results obtained with different parameters used in the modified Drucker-Prager model in Abaqus, with and without a cap, matched to the classical Mohr-Coulomb yield model.

The classical failure model for granular materials is the Mohr-Coulomb model, which can be written as

$$(\sigma_1 - \sigma_3) + (\sigma_1 + \sigma_3)\sin\phi - 2c\cos\phi = 0,$$

where σ_1 and σ_3 are the maximum and minimum principal stresses (positive in tension), ϕ is the friction angle, and c is the cohesion. The intermediate principal stress has no effect on yield in this model. Experimental evidence suggests that the intermediate principal stress does have an effect on yield; nonetheless, laboratory data characterizing granular materials are often presented as values of ϕ and c.

Abaqus offers a Mohr-Coulomb model for modeling this class of material behavior. This model uses the classical Mohr-Coulomb yield criterion: a straight line in the meridional plane and a six-sided polygon in the deviatoric plane. However, the Abaqus Mohr-Coulomb model has a completely smooth flow potential instead of the classical hexagonal pyramid: the flow potential is a hyperbola in the meridional plane, and it uses the smooth deviatoric section proposed by Menetrey and Willam (1995). The Abaqus Mohr-Coulomb model is described in *Mohr-Coulomb Plasticity*.

Abaqus also offers two Drucker-Prager models, with and without a compression cap, to model this class of material behavior. The Abaqus Drucker-Prager model without a cap provides a choice of three yield criteria. The differences are based on the shape of the yield surface in the meridional plane, which can be a linear form, a hyperbolic form, or a general exponent form (as described in *Extended Drucker-Prager Models*). The linear form is used here to make direct comparisons with the classical linear Mohr-Coulomb model. In addition, the hyperbolic and exponential forms are also verified in this example by using parameters that reduce them to equivalent linear forms.

This section also illustrates how to match the parameters of a corresponding linear Drucker-Prager model, β and d, to the Mohr-Coulomb parameters, ϕ and c, under plane strain conditions.

The Abaqus Drucker-Prager and Mohr-Coulomb models restrict possible flow patterns when the stress point is at a vertex of the Mohr-Coulomb yield surface. Thus, the models will not reproduce some localization effects exhibited by real materials, which are assumed to behave more according to a vertex model than with a smooth model when the plastic flow direction wants to change rapidly with load. Either model must be used with nonassociated flow to avoid excessive dilatation in modeling real materials.

The Drucker-Prager/Cap model adds a cap yield surface to the modified Drucker-Prager model. The cap surface serves two main purposes: it bounds the yield surface in hydrostatic compression, thus providing an inelastic hardening mechanism to represent plastic compaction; and it helps control volume dilatancy when the material yields in shear by providing softening as a function of the inelastic volume increase created as the material yields on the Drucker-Prager shear failure and transition yield surfaces. The model uses associated flow in the cap region and a particular choice of nonassociated flow in the shear failure and transition regions.

Problem description

The plane strain model analyzed is shown in *Figure 1*. The strip of sand is 3.66 m (12 ft) deep and of infinite horizontal extent. The footing is rigid and perfectly rough and spans a central portion 3.05 m (10 ft) wide. The model assumes symmetry about a center plane, and the region modeled with finite elements extends 8.84 m (29 ft) to the right of the center plane. In Abaqus/Standard reduced-integration, second-order, plane strain quadrilaterals (element type CPE8R) are used for the finite region, and infinite elements (element type CINPE5R) are used beyond this line to simulate the rest of the strip. In Abaqus/Explicit these elements are substituted with their linear counterparts (element type CPE4R and element type CINPE4, respectively). In Abaqus the infinite elements are always assumed to have linear elastic behavior; therefore, they are used beyond the region where plastic deformation takes place. The base of the strip is fixed in both the horizontal and vertical directions. The mesh is shown in *Figure 1*. No mesh convergence studies have been performed.

Material

The material's elastic response is assumed to be linear and isotropic, with a Young's modulus 207 MPa (30×10^3 lb/in^2) and a Poisson's ratio 0.3. Yield is assumed to be governed by the Mohr-Coulomb surface, with a friction angle $\phi = 20°$ and cohesion, c, of 0.069 MPa (10 lb/in^2). These constants can be used directly in the Abaqus Mohr-Coulomb model.

Extended Drucker-Prager Models describes the method for converting these Mohr-Coulomb parameters to Drucker-Prager parameters in plane strain. Applying the formulae given in *Extended Drucker-Prager Models* provides $\tan\beta = 0.581$ ($\beta = 30.16°$) and $\sigma_c^0 = 0.137$ MPa (19.8 lb/in^2) for associated flow and $\tan\beta = 0.592$ ($\beta = 30.64°$) and $\sigma_c^0 = 0.140$ MPa (20.2 lb/in^2) for nondilatant flow. The example is run using the associated flow parameters together with $\psi = \beta$ and using the nondilatant flow parameters together with $\psi = 0$.

The Drucker-Prager/Cap model is run using the same plane strain matching of the Mohr-Coulomb parameters. The cap eccentricity parameter is chosen as $R = 0.1$. The initial cap position (which measures the initial consolidation of the specimen) is taken as $\varepsilon_{\text{vol}(0)}^{pl} = 0.00041$, and the cap hardening curve is as shown in *Figure 2*. The transition surface parameter $\alpha = 0.01$ is used.

For verification of the hyperbolic and exponent forms of the yield criteria, input files have been included that correspond to the dilatant linear Drucker-Prager model. Reducing the hyperbolic yield function into a linear form requires that $p_t|_0 = d/\tan\beta$. Reducing the exponent yield function into a linear form requires that $b = 1.0$ and that $a = (\tan\beta)^{-1}$.

Loading and controls

We are mainly interested in obtaining the limit footing pressure and in estimating the vertical displacement under the footing as a function of load.

A convenient way of defining a rigid and perfectly rough footing is to use an equation constraint to constrain all of the nodes under the footing to have the same displacement, which is done by retaining the central top node (node 801) to represent the footing. A vertical displacement is then applied to this node while its horizontal displacement is constrained to be zero. The total footing load is obtained as the vertical reaction force at node 801. The average footing pressure is this vertical load divided by the width of the footing.

For the nonassociated flow cases unsymmetric matrix storage and solution is used: this is essential to obtain an acceptable convergence rate since nonassociated flow plasticity results in unsymmetric stiffness matrices.

Results and discussion

The load-displacement responses are shown in *Figure 3*, where we also show the limit analysis (slip line) Prandtl and Terzaghi solutions, as given by Chen (1975). The nondilatant Drucker-Prager and Mohr-Coulomb models give a softer response and a lower limit load than the corresponding dilatant versions. The cap model provides a response that is comparable to the corresponding Drucker-Prager nondilatant response. This response is due to the addition of the cap and the nonassociated flow in the failure region, which combine to reduce the dilation in the model and—therefore—approximate the Drucker-Prager nondilatant flow model.

The results obtained for the nondilatant Drucker-Prager model and the corresponding cap model match closely those obtained with the nondilatant Mohr-Coulomb model. They provide almost identical limit loads, which lie between the Prandtl and Terzaghi solutions. This conclusion can be extended to general geotechnical problems that are analyzed under plane strain or axisymmetric assumptions.

Input files

Abaqus/Standard input files

granularlimitload_mc_nondilat.inp	Mohr-Coulomb nondilatant flow case.
granularlimitload_mc_dilat.inp	Mohr-Coulomb dilatant flow case.
granularlimitload_dp_nondilat.inp	Drucker-Prager nondilatant flow case.
granularlimitload_dp_dilat.inp	Drucker-Prager dilatant flow case.
granularlimitload_cap2.inp	Case with cap model.
granularlimitload_hyper_dilat.inp	Hyperbolic yield criterion, dilatant case.
granularlimitload_expo_dilat.inp	Exponential yield criterion, dilatant case.

Abaqus/Explicit input files

granularlimitload_mc_nondilat_xpl.inp	Mohr-Coulomb nondilatant flow case.
granularlimitload_mc_dilat_xpl.inp	Mohr-Coulomb dilatant flow case.

References

- Chen, W. F., Limit Analysis and Soil Plasticity, Elsevier, Amsterdam, 1975.
- Menétrey, Ph., and K. J. Willam, “Triaxial Failure Criterion for Concrete and its Generalization,” *ACI Structural Journal*, vol. 92, pp. 311–318, May/June 1995.

Figures

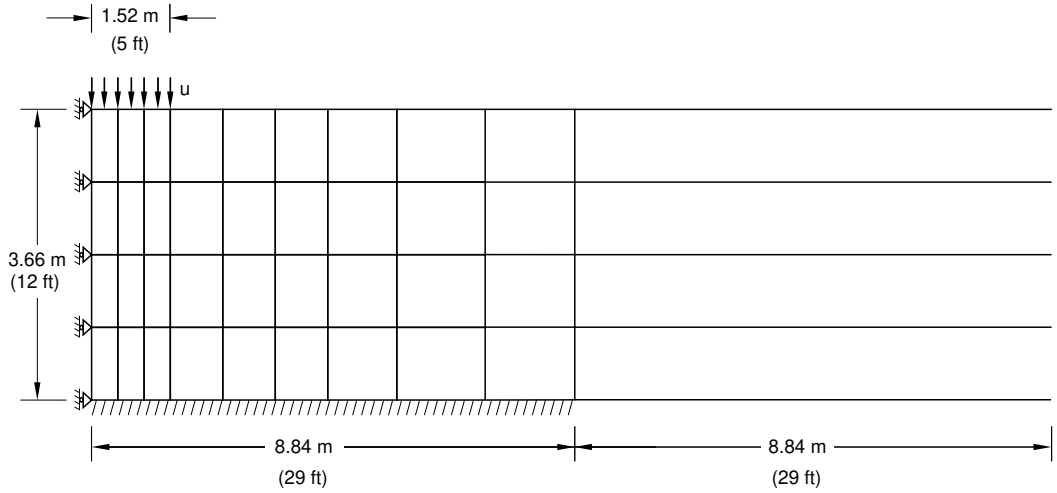


Figure 1: Model for limit load calculations on centrally loaded sand strip.

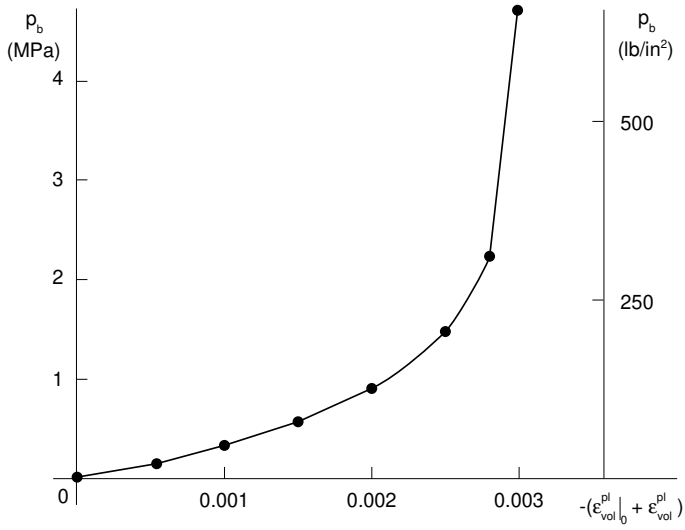


Figure 2: Cap hardening curve.

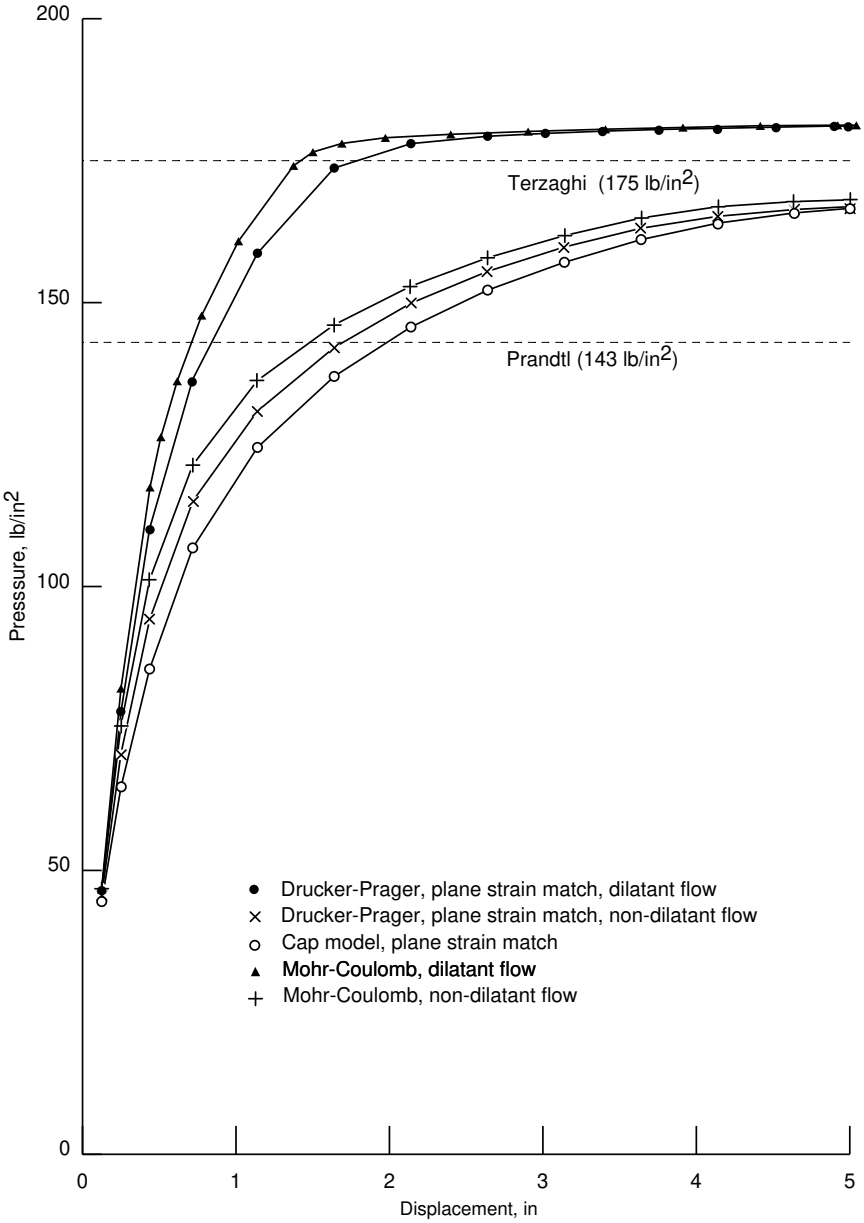


Figure 3: Limit load results.

Finite deformation of an elastic-plastic granular material

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example develops the homogeneous, finite-strain inelastic response of a granular material subject to uniform extension or compression in plane strain.
Results given by Carter et al. (1977) for these cases are used for comparison.

Problem description

The specimen is initially stress-free and is made of an elastic, perfectly plastic material. The elasticity is linear, with a Young's modulus of 30 MPa and a Poisson's ratio of 0.3. Carter et al. assume that the inelastic response is governed by a Mohr-Coulomb failure surface, defined by the friction angle of the Coulomb line ($\phi = 30°$) and the material's cohesion (c). They also assume that the cohesion is twice the Young's modulus for the extension test and 10% of the Young's modulus in the compression test. The above problem is solved using the Mohr-Coulomb plasticity model in Abaqus with the friction angle and the dilation angle equal to 30°. However, note that this Abaqus Mohr-Coulomb model is not identical to the classical Mohr-Coulomb model used by Carter because it uses a smooth flow potential.

An alternative solution is to use the associated linear Drucker-Prager surface in place of the Mohr-Coulomb surface. In this case it is necessary to relate ϕ and c to the material constants β and $\sigma_c^0 = d/\left(1 - \frac{1}{3}\tan\beta\right)$ that are used in the Drucker-Prager model. Matching procedures are discussed in *Extended Drucker-Prager Models*. In this case we select a match appropriate for plane strain conditions:

$$\tan\beta = \frac{3\sqrt{3}\tan\phi}{\sqrt{9+12\tan^2\phi}}$$

$$d = \frac{3\sqrt{3}c}{\sqrt{9+12\tan^2\phi}}.$$

The first equation gives $\beta = 40°$. Using the assumptions of Carter et al., the second equation gives d as 86.47 MPa (σ_c^0 = 120 MPa) for the extension case and d as 4.323 MPa (σ_c^0 = 6 MPa) for the compression case.

Uniform extension or compression of the soil sample is specified by displacement boundary conditions since the load-displacement response will be unstable for the extension case.

Results and discussion

The results are shown in *Figure 1* for extension and in *Figure 2* for compression. The solutions for Abaqus/Standard and Abaqus/Explicit are the same. The Drucker-Prager solutions agree well with the results given by Carter et al.; this is to be expected since the Drucker-Prager parameters are matched to the classical Mohr-Coulomb parameters under plane strain conditions. The differences between the Abaqus Mohr-Coulomb solutions and Carter's solutions are due to the fact that the Abaqus Mohr-Coulomb model uses a different flow potential. The Abaqus Mohr-Coulomb model uses a smooth flow potential that matches the classical Mohr-Coulomb surface only at the triaxial extension and compression meridians (not in plane strain).

However, one can also obtain Abaqus Mohr-Coulomb solutions that match Carter's plane strain solutions exactly. As discussed earlier, the classical Mohr-Coloumb model can be matched under plane strain conditions to an associated linear Drucker-Prager model with the flow potential

$$G = q - p\tan\beta.$$

This match implies that under plane strain conditions the flow direction of the classical Mohr-Coulomb model can be alternatively calculated by the corresponding flow direction of the Drucker-Prager model with the dilation angle β as computed before. Therefore, we can match the flow potential of the Abaqus Mohr-Coulomb model to that of the Drucker-Prager model. Matching between these two forms of flow potential assumes $e = 1$ and results in

$$\tan\psi = R_{mc}\left(\frac{\pi}{3}, \phi\right)\tan\beta = \frac{3-\sin\phi}{6\cos\phi}\tan\beta,$$

which gives $\psi = 22°$ in the Abaqus Mohr-Coulomb model. These Abaqus Mohr-Coulomb solutions are shown in *Figure 1* and *Figure 2* and match Carter's solutions exactly.

Input files

Abaqus/Standard input files

deformgranularmat_mc3030.inp	Extension case with the Mohr-Coulomb plasticity model ($\phi = 30°$ and $\psi = 30°$) and CPE4 elements.
deformgranularmat_dp.inp	Extension and compression cases with the linear Drucker-Prager plasticity model and CPE4 elements.
deformgranularmat_cpe4i_dp.inp	Extension case with the linear Drucker-Prager plasticity model and CPE4I incompatible mode elements.
deformgranularmat_mc3030_comp.inp	Compression case with the Mohr-Coulomb plasticity model ($\phi = 30°$ and $\psi = 30°$) and CPE4 elements.
deformgranularmat_dp_comp.inp	Compression case with the linear Drucker-Prager plasticity model and CPE4 elements.
deformgranularmat_mc3022.inp	Extension case with the Mohr-Coulomb plasticity model ($\phi = 30°$ and $\psi = 22°$) and CPE4 elements.
deformgranularmat_mc3022_comp.inp	Compression case with the Mohr-Coulomb plasticity model ($\phi = 30°$ and $\psi = 22°$) and CPE4 elements.

Abaqus/Explicit input files

granular.inp	Extension and compression cases with the linear Drucker-Prager plasticity model and CPE4R elements.
deformgranularmat_mc3030_xpl.inp	Extension case with the Mohr-Coulomb plasticity model ($\phi = 30°$ and $\psi = 30°$) and CPE4R elements.
deformgranularmat_mc3030_comp_xpl.inp	Compression case with the Mohr-Coulomb plasticity model ($\phi = 30°$ and $\psi = 30°$) and CPE4R elements.
deformgranularmat_mc3022_xpl.inp	Extension case with the Mohr-Coulomb plasticity model ($\phi = 30°$ and $\psi = 22°$) and CPE4R elements.

deformgranularmat_mc3022_comp_xpl.inp	Compression case with the Mohr-Coulomb plasticity model ($\phi = 30°$ and $\psi = 22°$) and CPE4R elements.

References

- Carter, J. P., J. R. Booker, and E. H. Davis, "Finite Deformation of an Elasto-Plastic Soil," *International Journal for Numerical and Analytical Methods in Geomechanics*, vol. 1, pp. 25–43, 1977.

Figures

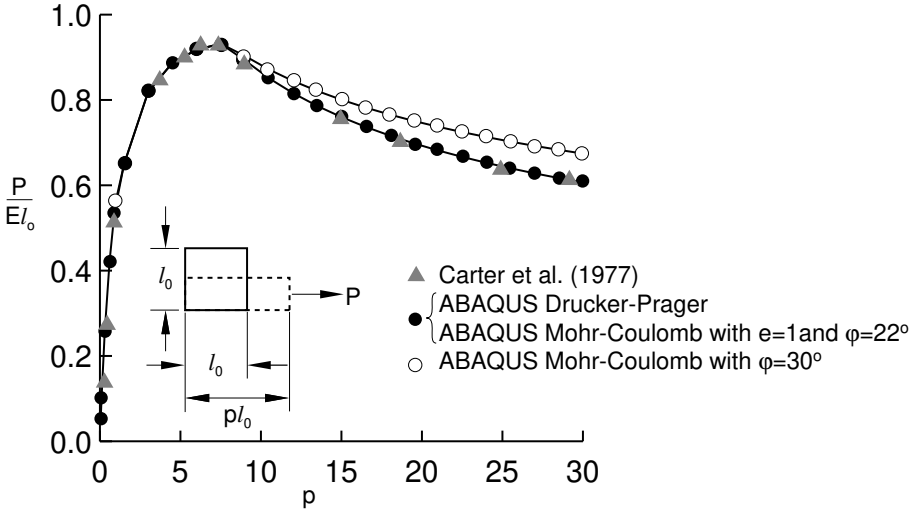


Figure 1: Load-displacement results for uniform extension.

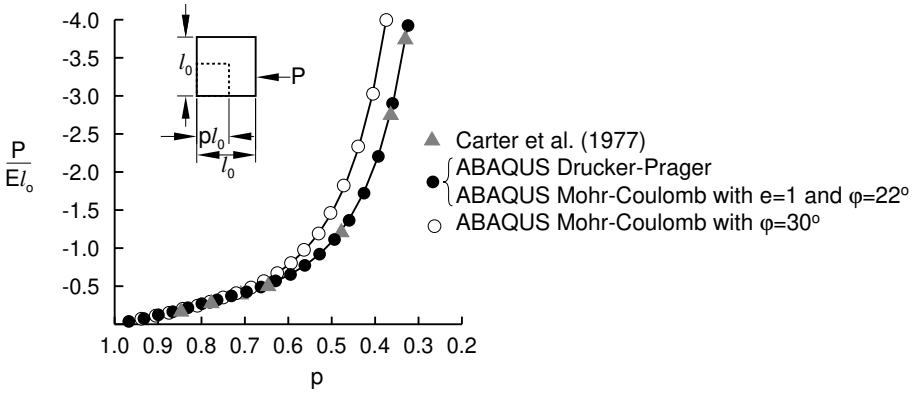


Figure 2: Load-displacement results for uniform compression.

The one-dimensional thermal consolidation problem

Products: Abaqus/Standard

Overview

This example illustrates the use of Abaqus/Standard to model one-dimensional thermal consolidation. Consolidation behavior of a one-dimensional column of fully saturated soil subjected to constant surface loads and constant surface temperature is studied, and the results obtained are compared to those obtained by Aboustit et al. (1985).

When soil is subjected to loads and temperature variation, a coupled system of equations that describe the deformation, pore fluid flow, and heat transfer through the soil must be solved to accurately predict the consolidation behavior.

Problem description

This problem can be considered as the thermal counterpart to *The Terzaghi consolidation problem*. The discussion presented in that section is equally applicable to this problem and is not repeated here. *Figure 1* shows one-dimensional thermoelastic consolidation of a linear elastic soil column under constant surface pressure and constant surface temperature. The column is 7 units high and 2 units wide. The bottom of the column is restrained, and all sides of the column are impermeable except for the top surface where free flow is allowed. The top surface is subjected to a constant pressure of 1 unit and a constant temperature of 50 units. The soil is assumed to be fully saturated. Gravity is neglected. The material properties reported by Aboustit et al. (1985) are used. The soil is elastic with a modulus of 6000 units and Poisson's ratio of 0.4. The permeability of the soil is 4×10^{-6} units with a specific weight of 1 unit. Since Aboustit et al. (1985) used only one set of thermal properties, identical thermal properties for the solid and the pore fluid are used. The specific heat is 40 units, and the density is 1 unit. The conductivity of the soil as well as the pore fluid is 0.2 units, and the coefficient of thermal expansion is 0.3×10^{-6}.

All displacements perpendicular to the sides are restrained to enforce one-dimensional behavior. The consolidation analysis is performed using a transient soils consolidation step with automatic time stepping. The time stepping for this problem is controlled by two parameters: one that controls the accuracy of time integration for the temperature field, and one that controls the accuracy of time integration for the pore fluid flow. The stability limit for the pore fluid solution is given by

$$\Delta t \geq \frac{\gamma_w}{6Ek}(\Delta h)^2,$$

which dictates the minimum time increment. Variables used in this equation are defined in *Coupled Pore Fluid Diffusion and Stress Analysis*. The mesh used is identical to the one used by Aboustit et al. (1985), which led to a minimum time increment of 0.1. Because of the applied surface load, the elements near the surface immediately acquire a pore pressure equal to the applied load; hence, a maximum pore pressure change per increment of 1.1 with an initial time increment of 0.1 is used. This ensures that time steps smaller than 0.1 are not used in the analysis to satisfy time integration accuracy for pore fluid flow. The value for the maximum allowable temperature change in an increment was chosen as 3 to avoid having to use time increments that were smaller than what the stability limit for pore fluid required. The value for the maximum allowable temperature change was obtained by first running the problem using only the value for the maximum pore pressure change and determining the incremental temperature change. The parameter values listed above result in a moderately accurate solution. If a more accurate solution is desired, a more refined mesh should be used.

Nonlinear geometric effects are not important in this problem due to small load magnitudes. Similarly due to very small fluid velocities, heat convection effects due to pore fluid flow are not dominant enough to necessitate the inclusion of unsymmetric stiffness. However, for completeness, we have chosen to activate geometric

nonlinear analysis as well as unsymmetric stiffness. Results of small-strain analysis with symmetric stiffness are indistinguishable from the results presented. The time period for the step is 21.1, corresponding to the time at which the Abaqus/Standard results are compared to the reference solution.

Results and discussion

At the start of the analysis there is zero temperature throughout the domain except at the top surface, and the pore pressure equals the applied surface load as all of the load is carried by the pore fluid. As time progresses, the temperature front progresses from the top to the bottom and the applied surface load is transferred from the pore fluid to the soil skeleton as pore fluid exits at the top, reducing the pore pressure in the domain. In the steady-state limit, all of the domain has zero pore pressure and constant temperature equal to the applied surface temperature. The Abaqus/Standard solution for pore pressure and temperature are compared at time 21.1, when the temperature front has progressed some distance from the top surface and there is partial reduction of the pore pressure. The results are shown in *Figure 2*. The temperature and pore pressure values are normalized using the applied temperature and the applied surface pressure. The ordinate is normalized using the height of the soil column. Abaqus/Standard results compare well with those presented by Aboustit et al. (1985).

Input files

unidircon_c3d4pt.inp	Mesh with C3D4PT elements.
unidircon_c3d4pht.inp	Mesh with C3D4PHT elements.
unidircon_c3d4ph.inp	Mesh with C3D4PH elements.
unidircon_c3d8pt.inp	Mesh with C3D8PT elements.
unidircon_c3d8rpt.inp	Mesh with C3D8RPT elements.
unidircon_c3d8pht.inp	Mesh with C3D8PHT elements.
unidircon_c3d8rpht.inp	Mesh with C3D8RPHT elements.
unidircon_c3d10mpt.inp	Mesh with C3D10MPT elements.
unidircon_c3d10pt.inp	Mesh with C3D10PT elements.
unidircon_c3d10p.inp	Mesh with C3D10P elements.
unidircon_c3d10ph.inp	Mesh with C3D10PH elements.
unidircon_c3d10pht.inp	Mesh with C3D10PHT elements.
unidircon_cax4pt.inp	Mesh with CAX4PT elements.
unidircon_cax4rpt.inp	Mesh with CAX4RPT elements.
unidircon_cax4rpht.inp	Mesh with CAX4RPHT elements.

References

- Aboustit, B. L., S. H. Advani, and J. K. Lee, “Variational Principles and Finite Element Simulations for Thermo-Elastic Consolidation,” *International Journal for Numerical and Analytical Methods in Geomechanics*, vol. 9, pp. 49–69, 1985.

Figures

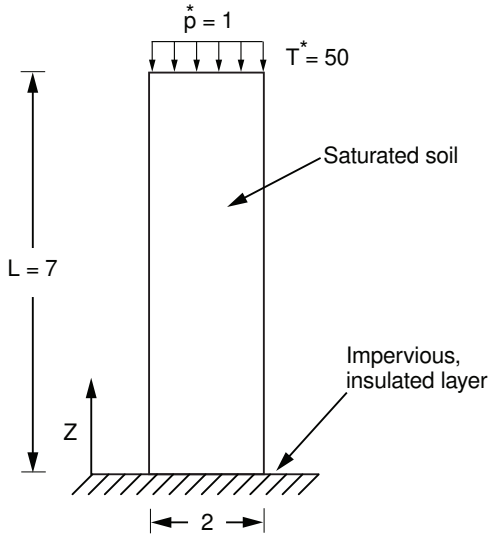


Figure 1: One-dimensional thermal consolidation model.

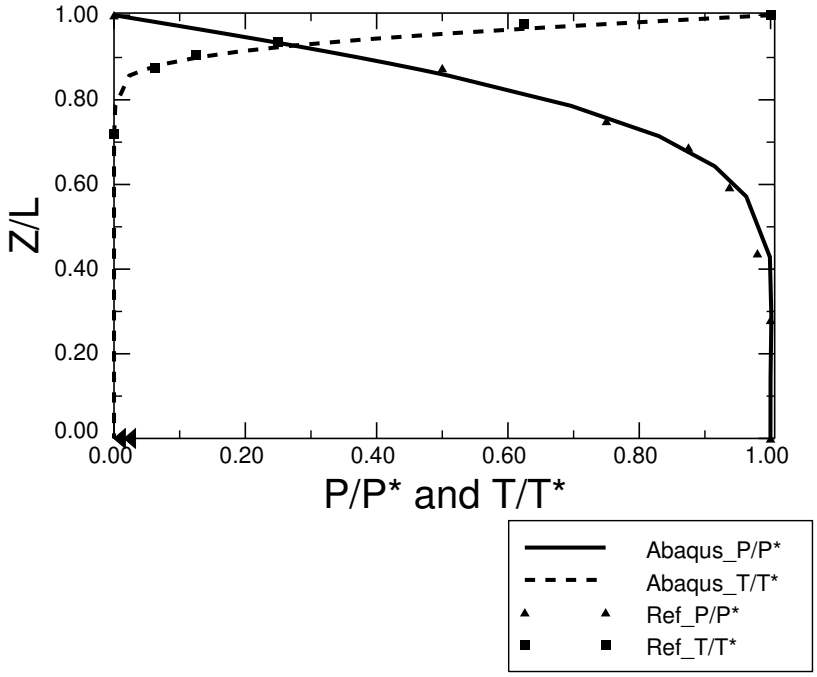


Figure 2: Normalized temperature and pore pressure along the z-direction at time 21.1.

Consolidation around a cylindrical heat source

Products: Abaqus/Standard

Overview

This problem shows the solution of consolidation in saturated soil around a cylindrical heat source.

The problem has been studied by Booker and Savvidou (1985) and represents an idealization of the problem of a canister of radioactive waste buried in saturated soil. The temperature changes that occur due to radiation of heat from the canister cause the pore water to expand a greater amount than the pores in the soil, resulting in an increase in the pore pressure around the heat source. The resulting pore pressure gradient drives pore fluid away from the heat source, resulting in the dissipation of the pore pressure with time. Booker and Savvidou developed an analytical solution for the fundamental problem of a point heat source buried deep in saturated soil. They subsequently used this analytical solution to derive an approximate solution to the problem of consolidation around a cylindrical heat source. This problem provides a verification for the coupled thermal consolidation capability in Abaqus. The analysis of saturated soils requires solution of coupled stress-diffusion equations, and the formulation used in Abaqus is described in detail in *Analysis of porous media*. The thermal consolidation capability also solves the heat transfer equation (considering both conductive and convective effects) in a fully coupled manner with the stress-diffusion equations and, thereby, models the influence of the pore pressure on the temperature field in the pore fluid and the soil and vice versa.

Numerical values for the parameters that define the geometry and the material properties are based on the details presented in a parametric study of this problem by Lewis and Schrefler (2000).

Problem description

The problem setup is shown in *Figure 1*. A cylindrical heat source of radius 0.1604 m and height 2.5 m is embedded within a cylindrical volume of soil with radius and height both equal to 10 m. The cylindrical volume of soil, in effect, represents an infinite medium surrounding the heat source. Gravity is neglected. Because of the boundary conditions (discussed below in detail), the problem is essentially one-dimensional, the only gradient being in the radial direction. The purpose of the analysis is to predict the evolution of pore pressure and temperature throughout the soil mass, especially in the neighborhood of the heat source, as a function of time.

Geometry and models

Only half of the problem is modeled, taking advantage of the symmetry in the vertical direction. This problem is solved using both three-dimensional and axisymmetric coupled temperature–pore pressure elements. For the purpose of presenting the results, three-dimensional element type C3D8RPT is chosen. Both the three-dimensional and axisymmetric analyses are carried out using different variants (such as reduced-integration and hybrid) of the basic three-dimensional 8-node or axisymmetric 4-node elements, as well as the modified tetrahedral elements.

The response of the soil is assumed to be linear elastic, with a Young's modulus of 60.0 MPa and a Poisson's ratio of 0.4. The specific weight of the pore fluid is assumed to be 9800.0 N/m^3 (1 lb/in^3). The permeability is assumed to be constant, with a value of 4.63×10^{-8} m/sec. The thermal expansion coefficients of the soil and pore fluid are assumed to be 0.3×10^{-6} per °C and 0.21×10^{-5} per °C, respectively. The density, specific heat, and conductivity of the soil and the pore fluid are assumed to be the same, with values of 1000 kg/m^3, 40.0 cal/(kg°C), and 11.9 W/cal/(m°C), respectively. The void ratio is assumed to be 1.0 initially throughout the soil volume. The initial temperature and pore pressure are assumed to be zero everywhere. It is also assumed that the pores are fully saturated with pore fluid.

Boundary conditions

The normal (vertical) component of displacement is constrained at the base of the soil, while rigid body motion in the two lateral directions is prevented by constraining a set of points on the outer boundary. The pore pressure and the temperature are assumed to be zero at all points on the outer boundary of the soil volume. Thus, the outer boundary is assumed to be connected to a heat and fluid reservoir (as represented by the soil that surrounds the volume considered for this model) that allows transfer of heat and pore fluid such that the boundary temperature and pore pressure are maintained at the specified values. The heat source is specified as heat body flux per unit volume with a magnitude of 11.58.

Analytical solution

There are two distinct time scales associated with this problem: one for each of the two diffusive mechanisms that are operational. The first time scale is associated with diffusive heat transfer and is given by $T_1 = r_0^2/\kappa$, where r_0 is the radius of the cylindrical heat source, while κ represents the thermal diffusivity of the surrounding medium and is given by $\kappa = K/\rho C_p$. In the preceding expression, ρ, C_p, and K represent the density, specific heat capacity, and conductivity, respectively, of the surrounding medium. In general, the thermal properties used in this expression need to be weighted average quantities based on the volume fraction of the pores. However, such averaging is not necessary in this example as the thermal properties of the soil and the pore fluid are assumed to be the same. The second time scale is associated with diffusive flow of the pore fluid and is given by $T_2 = r_0^2/c$. In the preceding expression, the quantity c represents the consolidation coefficient that is defined as $c = k(\lambda + 2G)/\gamma_w$, where k is the permeability of the porous medium, λ and G are elastic constants, and γ_w is the specific weight of the pore fluid. The choice of the different parameters for the problem is such that the ratio $T_1/T_2 = c/\kappa$ is approximately equal to 2.

Booker and Savvidou obtained an analytical solution for consolidation around a point heat source in an otherwise infinite medium and utilized this analytical solution to approximate the solution to the problem of consolidation around a cylindrical heat source. The latter was accomplished by simply integrating the point source solution throughout the cylindrical volume. The expressions for the temperature and pore pressure fields, as given in the above reference, are reproduced below. These expressions are used to obtain the analytical solutions for comparison with the numerical results. The value of the temperature at the point (x_F, y_F, z_F) and at time t is given by

$$\theta(x_F, y_F, z_F, t) = \int\int_{V_s}\int \frac{q_v}{4\pi KR} f\left(\frac{\kappa t}{R^2}\right) dV_s,$$

where q_v is the strength of the heat source (heat energy radiated from the source per unit volume per unit time), $R^2 = (x_F - x_S)^2 + (y_F - y_S)^2 + (z_F - z_S)^2$, and ($x_S$, y_S, z_S) represent the coordinates of a point source within the cylindrical volume V_s. The function $f\left(\frac{\kappa t}{R^2}\right)$ is expressed in terms of the complementary error function **erfc** as

$$f\left(\frac{\kappa t}{R^2}\right) = \text{erfc}\left(\frac{R}{2\sqrt{\kappa t}}\right).$$

Likewise, under the assumption that $c/\kappa = 2$, the pore pressure field can be expressed as

$$p(x_F, y_F, z_F, t) = \int\int_{V_s}\int \frac{-Xq_v}{4\pi KR}\left[f\left(\frac{\kappa t}{R^2}\right) - f\left(\frac{ct}{R^2}\right)\right] dV_s,$$

where the quantity X depends upon the elastic properties of the soil skeleton and the (volumetric) thermal expansion coefficients of both the soil skeleton and the pore fluid and is given by

$X = [(1 - n)\ \beta_s + n\beta_w]\ (\lambda + 2G) - (\lambda + 2G/3)\ \beta_s$. Booker and Savvidou note that the temperature reaches a maximum value of $\theta_N = 1.45\ \frac{q_v\, r_0^2}{K}$ at the midpoint of the cylindrical source. If the soil were to be impermeable ($c = 0$), the pore pressure would reach a maximum value of $p_N = X\,\theta_N$ at the same point.

The expression for pore pressure given in the above paragraph clearly suggests that the effect of pore fluid flow is to reduce (with time) the pore pressure at a given point. For the special case of an impermeable fluid, the pore pressure will simply build up over time and never reduce. On the other hand, if $c \approx \kappa$, the fluid diffuses at the same rate as the heat and, hence, the pore pressure never builds up.

Time stepping

The accuracy of the time integration for the transient soils consolidation procedure that also models heat transfer is controlled by the maximum allowable pore pressure and temperature changes per time step. Even in a linear problem these values control the accuracy of the solution because the time integration operators for the consolidation and heat transfer problems are not exact (the backward difference rule is used in both cases). In this case the allowable pore pressure change per time step is chosen as 0.5 Pa, which is a relatively large value compared to p_N. Simulations with smaller values, such as 0.1 Pa and 0.075 Pa, produce essentially the same results, although the analysis takes progressively more increments to complete with lower values. The allowable temperature change per time step is chosen as 0.003°C, which is approximately $0.1\theta_N$. A value of 0.0003°C (along with a value of 0.075 Pa) produced essentially the same results, although the analysis took a significantly larger number of increments to complete.

The analysis is continued for a time period of approximately 1000 T_1.

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.

For the pore fluid flow equations, the simulations also use a nondefault and a relatively large value of the ratio of the largest residual flux to the average flux, which sets the convergence criterion for an increment. This setting is helpful as the fluid velocity in this problem is very small and ensures that the pore pressure increment is not considered converged without at least one correction (iteration). There is not much advantage in reducing the tolerance further as the flow residuals are already sufficiently small, and any further reduction in the residual does not make any difference to the overall solution.

Results and discussion

The automatic time incrementation capabilities of Abaqus/Standard were used for all the simulations. As discussed earlier, the total number of increments to complete the analysis depends strongly on the choices of the maximum allowable pore pressure and temperature changes per time step for the transient soils consolidation procedure. For relatively loose tolerances, the time increment size increases by a factor of approximately 25,000 from the beginning to the end of the analysis, while for relatively tight tolerances the factor reduces to approximately 20,000. These very large changes in the time increment size are typical of problems that are diffusion dominated and highlight the value of using automatic time stepping with an unconditionally stable integration operators for such problems.

Figure 2 and *Figure 3* show the variations of temperature and pore pressure, respectively, with time at three different radii from the cylindrical heat source. These locations correspond to the outer surface of the heat source ($r/r_0 = 1$), and distances of 2 ($r/r_0 = 2$) and 5 ($r/r_0 = 5$) times, respectively, of the radius of the heat source, along $z = 0$. The axes for temperature, pore pressure, and time (the latter shown on a logarithmic scale) are normalized by the quantities θ_N, p_N, and T_1, respectively.

The analytical solutions to the problem, based on the equations presented earlier, are also shown in the figures using data points (as opposed to continuous curves) and are labeled `Analytical-1`, `Analytical-2`, and `Analytical-5`, respectively, and correspond to the three different radii discussed above. The analytical solutions were obtained by evaluating the integrals numerically.

The temperature results suggest that at the surface of the heat source, the temperature approaches θ_N with time, while at distances further away from the heat source the temperature increases with time at a slower rate. The results agree well with the analytical results, especially relatively early in the analysis. The mesh used for the simulations is relatively coarse with 2718 elements. The agreement between the finite element predictions and the analytical solutions is much better at all times if the analysis is carried out with a more refined mesh consisting of 9816 elements.

There is an initial elevation, followed by a reduction in the pore pressure with time. The initial increase in pore pressure is due to the relatively higher volumetric expansion of the pore fluid compared to the pores. A gradient in the pore pressure field is necessary to drive pore fluid flow. The results suggest that at relatively early times, the diffusion of the pore fluid away from a material point is not strong enough to offset the increase in volume associated with an increase in temperature. Hence, the pore pressure increases with time. However, with the passage of time the rate of increase of temperature at a material point slows down, and the diffusion of pore fluid picks up such that any further increase in temperature (and the associated volume change) does not result in additional increase in pore pressure, and the pore pressure decays with time.

Figure 4 and *Figure 5* show the contour plot of pore pressure and a vector plot of the magnitude of the fluid velocity, respectively, at some intermediate time (approximately 5700 seconds) during the analysis. The distribution of pore pressure is approximately axisymmetric, with higher pore pressures closer to the central heat source. The radial gradient in the pore pressure drives the pore fluid flow, resulting in pore fluid velocity vectors that point approximately in the radial direction. The mesh itself is not axisymmetric, which results in small variations in the solution from a purely axisymmetric state.

While this problem illustrates the coupled nature of the physical problem of a heat source embedded in soil, the coupling is of a relatively weak nature. Thus, while the pore fluid flow field is mainly driven by the relative thermal volumetric expansions of the pore fluid and the pores and, hence, depends directly on the temperature field, the heat transfer problem is insensitive to the pore fluid flow. A stronger coupling could be included, for example, by considering convective heat transfer where the rate of transfer of heat is directly influenced by the pore fluid velocities. Additional potential sources of coupling include the dependence of permeability on the void ratio, which can depend on the level of straining (including thermal expansion) in the material. Although such effects are accounted for in the formulation in Abaqus/Standard, they are neglected in the present problem.

Input files

pointheatsrcconsl_c3d8pt.inp	Consolidation analysis with heat transfer using element type C3D8PT.
pointheatsrcconsl_c3d8pht.inp	Consolidation analysis with heat transfer using element type C3D8PHT.
pointheatsrcconsl_c3d8rpt.inp	Consolidation analysis with heat transfer using element type C3D8RPT.
pointheatsrcconsl_c3d8rpht.inp	Consolidation analysis with heat transfer using element type C3D8RPHT.

pointheatsrcconsl_cax4pt.inp	Consolidation analysis with heat transfer using element type CAX4PT.
pointheatsrcconsl_cax4rpt.inp	Consolidation analysis with heat transfer using element type CAX4RPT.
pointheatsrcconsl_cax4rpht.inp	Consolidation analysis with heat transfer using element type CAX4RPHT.
pointheatsrcconsl_c3d10mpt.inp	Consolidation analysis with heat transfer using element type C3D10MPT.

References

- Booker, J. R., and C. Savvidou, "Consolidation Around a Point Heat Source," *International Journal for Numerical and Analytical Methods in Geomechanics*, vol. 9, pp. 173–184, 1985.
- Lewis, R. W., and B. A. Schrefler, The Finite Element Method in the Static and Dynamic Deformation and Consolidation of Porous Media, John Wiley & Sons Ltd., 1998.

Figures

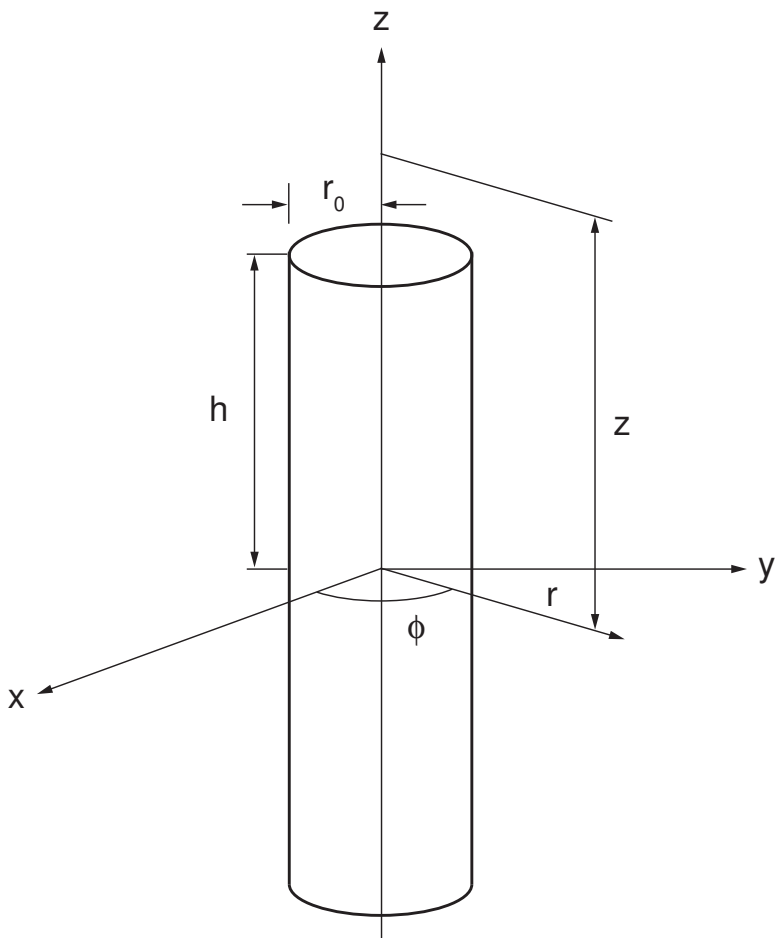


Figure 1: Geometry of the heat source that is embedded in an infinite soil medium (modeled as a cylindrical domain with finite dimensions).

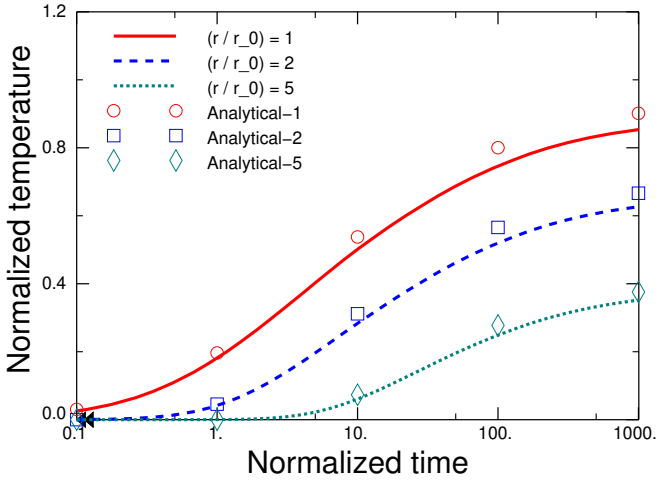


Figure 2: Variation of normalized temperature with normalized time at three different radii.

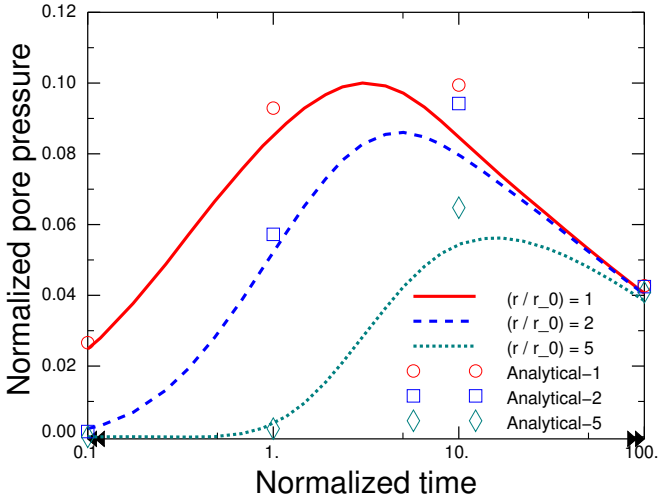


Figure 3: Variation of normalized pore pressure with normalized time at three different radii.

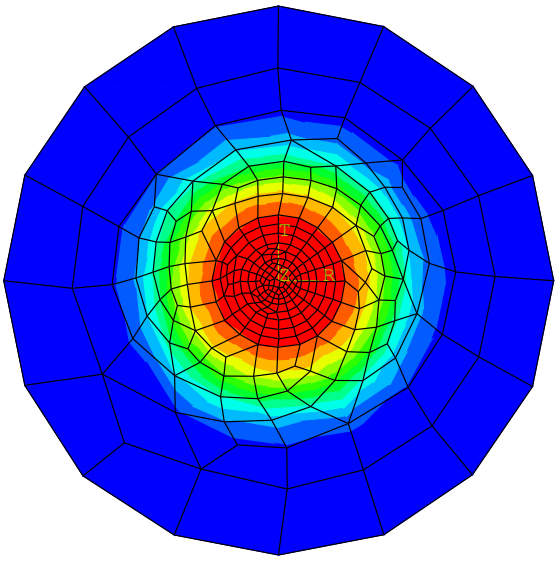


Figure 4: Contour plot of pore pressure at an intermediate time.

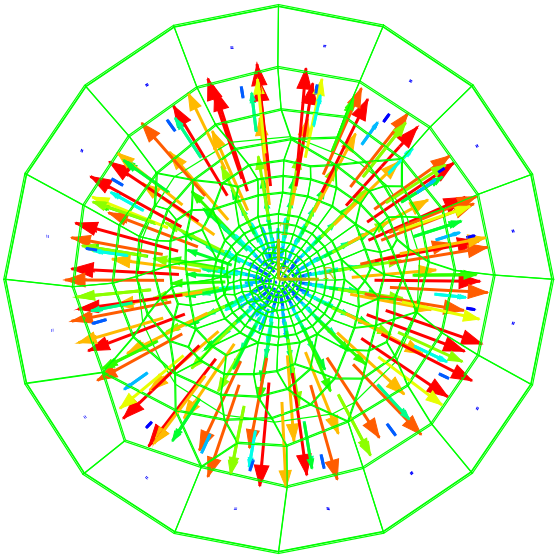


Figure 5: Vector plot of pore fluid velocity at an intermediate time.

Fracture mechanics

In this section:

- *Contour integral evaluation: two-dimensional case*
- *Contour integral evaluation: three-dimensional case*
- *Center slant cracked plate under tension*
- *A penny-shaped crack under concentrated forces*
- *Fully plastic J-integral evaluation*
- *Ct-integral evaluation*
- *Nonuniform crack-face loading and J-integrals*
- *Single-edged notched specimen under a thermal load*

Contour integral evaluation: two-dimensional case

Products: Abaqus/Standard

Overview

This example verifies the evaluation of the *J*-integral, stress intensity factors, and *T*-stress, based on either the conventional finite element method or the extended finite element method.

The *J*-integral, stress intensity factors, and *T*-stress are widely used in fracture mechanics; and their accurate estimation for postulated flaws under given load conditions is an important aspect of the use of fracture mechanics in design. The domain integral method of Shih et al. (1986) provides a useful method for numerically evaluating contour integrals for the *J*-integral, stress intensity factors, and *T*-stress. This method provides high accuracy with rather coarse models in two dimensions; in three dimensions coarse meshes still give reasonably accurate values. It adds only a small increment to the cost of the stress analysis and can be specified easily. Abaqus offers the evaluation of these parameters for fracture mechanics studies based on either the conventional finite element method or the extended finite element method (XFEM). Contour integral evaluation is available in Abaqus for any loading (including thermal loading: see *Single-edged notched specimen under a thermal load*) and for elastic, elastic-plastic, and viscoplastic (creep) behaviors, the latter two cases being based on the equivalent hypoelastic material concept. The evaluation of the contour integral in three-dimensional cases is also often of interest: see *Contour integral evaluation: three-dimensional case*.

Problem description

Four examples are presented here for verification purposes. The first is a linear elastic, plane strain, double-edged notch specimen under Mode I loading, for which Bowie (1964) has provided a series solution for the stress intensity factor, K_I; the second is an axisymmetric specimen with a penny-shaped crack; the third is a single-edged notch specimen under Mode I loading; and the fourth is a bimaterial specimen with an interface crack lying along the interface between the two materials. For the plane strain case a three-dimensional model is also used, with one layer of elements in the thickness direction, to verify the capability for evaluating the *J*-integral as a function of position along the crack front. In this case the *J*-integral should be constant along the crack front—the same value as is obtained from the two-dimensional plane strain analysis. The submodeling technique is used to demonstrate how to obtain more accurate results around the crack tip. All four examples are studied based on the conventional finite element method. In addition, the first and the third examples are also studied based on the extended finite element method.

Geometry and model

The geometry of the first example is shown in *Figure 1*. The plane strain structure is a section of a plate with symmetric edge cracks at its centerline, leaving an uncracked ligament of half the plate's width. The specimen is loaded in Mode I by uniform tension applied to its top and bottom surfaces. When the conventional finite element method is used, symmetry about $x = 0$ and about $y = 0$ can be used to model only the top right-hand quadrant of the plate. The mesh used for the quarter model is shown in *Figure 2*. Second-order elements (8-node quadrilaterals, 20-node bricks) are used. In the case of the full-model seam crack, left and right contour integrals are defined in Abaqus/CAE, as shown in *Figure 3*. Either the normal to the crack extension or the q vectors can be used to define the crack extension direction.

One advantage of using second-order elements with the conventional finite element method is that they can be used to model the desired singularity at the crack tip. To obtain a $1/\sqrt{r}$ singularity term, the following conditions must be met:

1. The elements around the crack tip must be focused on the crack tip. One edge of each element must be collapsed to zero length (as shown in *Figure 2*) so that the nodes of this zero length edge are located at the crack tip.
2. The "midside" nodes of the edges radiating out from the crack tip of each of the elements attached to the crack tip must be placed at one-quarter of the distance from the crack tip to the other node of the edge.

The region around the crack tip can be partitioned as shown in *Figure 4* and swept meshed with elements having a quad-dominated shape. The node connectivity table is adjusted internally by Abaqus/CAE to create the degenerate quadrilateral elements.

If the coincident nodes at the crack tip are constrained to displace together, the only singularity term in strain is $1/\sqrt{r}$; if the crack-tip nodes are free to displace independently, the crack-tip singularity in strain includes a $1/r$ term in addition to the $1/\sqrt{r}$ term. These methods of creating singularities in standard isoparametric elements are explained in detail by Barsoum (1976).

The material type determines the singularity at the crack tip. A linear elastic material exhibits a $1/\sqrt{r}$ singularity in strain at a sharp crack tip, whereas a perfectly plastic material exhibits a $1/r$ singularity in strain. The singularity in strain for a plastic hardening material lies somewhere between $1/\sqrt{r}$ and $1/r$.

Frequently the need for meshing simplicity with a preprocessor is greater than the need for extreme accuracy of contour integral results. The contour integral results often will be adequate as long as some singularity is included.

For example, a $1/r$ singularity is introduced if the element edges are collapsed, the nodes at the crack tip are free to displace independently, and the midside nodes are not moved to the quarter points (they remain at the midside points). This singularity is often quite adequate for elastic-plastic problems.

The model in this example problem uses a linear elastic material and, thus, should be modeled with only a $1/\sqrt{r}$ singularity term.

For the quarter model of the double-edged notch specimen, symmetry is used for calculating the contour integral results. Thus, the results for the contour integrals are multiplied by two before being output. The three-dimensional model uses the same mesh with one layer of 20-node bricks, as shown in *Figure 5*. The loading applied is either a uniform edge load in two dimensions or a uniform surface pressure (of negative magnitude) in three dimensions.

When the extended finite element method is used, the mesh is not required to match the cracked geometry. The presence of a crack is ensured by the special enriched functions in conjunction with additional degrees of freedom. This approach also removes the requirement to explicitly define the crack front or to specify the virtual crack extension direction when evaluating the contour integral. The data required for the contour integral will be determined automatically based on the level set signed distance functions at the nodes in an element. First-order brick elements with both full integration and reduced integration are used for this example. The mesh is shown in *Figure 6*.

The axisymmetric model corresponds to a penny-shaped crack in a round bar. The model is shown in *Figure 7* and is loaded in Mode I by uniform tension applied to the top and bottom surfaces. Symmetry about $r = 0$ and $z = 0$ allows you to model only the top right-hand quadrant with a mesh, as shown in *Figure 8*. Second-order elements (CAX8 and CAX8R) are used.

The third example is a single-edged notch specimen under Mode I tension, as shown in *Figure 9*. This specimen contains a crack in the symmetry plane in a homogeneous linear elastic material. This example has been studied using both the conventional finite element method and the extended finite element method. Due to symmetry only the top half-plane is modeled for the specimen when used with the conventional finite element method. The complete body is modeled when used with the extended finite element method. Second-order elements CPE8 and CPS8 are used with the conventional finite element method, while first-order brick elements C3D8 and first-order tetrahedral elements C3D4 are used with the extended finite element method.

The last example is also a single-edged notch specimen under Mode I tension, as shown in *Figure 9*. This specimen contains a crack lying along the interface between two dissimilar elastic materials. The complete body is modeled for the specimen with an interface crack. Second-order elements CPE8 and CPS8 are used for this example.

The *J*-integral, stress intensity factors, and *T*-stress should be path independent, and Abaqus provides for its evaluation on as many contours as you request. The first contour is normally at the crack tip, and subsequent contours are generated automatically as contours passing through the nearest neighboring elements, moving out from the crack tip. The mesh used in this case has several rings of elements surrounding the crack tip and as many contours as the number of rings that can be requested. The contour integral should be path independent, so the variation of values between contours can be taken as an indicator of the quality of the mesh for determining the fracture parameters. Path independence of the contour integral values is sufficient to indicate mesh convergence for stress, strain, or displacements.

Results and discussion

The plane strain solutions obtained with full- and reduced-integration elements (CPE8 and CPE8R) are compared with Bowie's approximate solution in *Table 1* for the *J*-integral. The Abaqus results are very close to Bowie's approximation, and path independence is well preserved. The *J*-integral is also calculated automatically based on the stress intensity factors if the latter are requested by the contour integral evaluation of Abaqus. These *J* values show very good agreement with those presented in *Table 1*.

The three-dimensional solutions of the *J*-integral for the 20-node brick mesh are shown in *Table 2*. The *J* values provided by the fully integrated 20-node brick model show some oscillation along the crack front for the first contour. In the 20-node brick models contours involving midside nodes have fewer nodes being perturbed than contours involving corner nodes of such elements, which results in differences in strain energy calculations and, hence, in the integral values. The effect is not large and is generally apparent only for the first contour; it becomes smaller as the mesh is refined. Since the *J* values for this contour are also probably the least accurate regardless of the element type that is used, they are frequently ignored; and only the *J* values for contours two and higher are used in estimating *J*. The oscillation of the *J*-integral values with crack-front position is not evident in the reduced-integration models for the 20-node brick case. The stress intensity factors and the *T*-stress are also calculated for the same three-dimensional model. They have the same features as described above for the contour integral evaluation of *J*.

The three-dimensional solutions of the *J*-integral for the 8-node brick mesh using the extended finite element method are shown in *Table 3*.

The axisymmetric solutions of the *J*-integral are shown in *Table 4*, where they can be compared to an approximate solution from Tada et al. (1973). Path independence is well preserved in the mesh. The numerical results are again slightly higher than the approximate solution from Tada et al. (1973). However, the stress intensity factors and *T*-stress show evidence of contour dependence in this mesh. The crack tip is so close to the symmetry axis that the auxiliary plane strain crack-tip field cannot be used satisfactorily in the interaction method to extract their values. Using more concentrated, refined meshes around the crack tip eliminates the problem.

Both the stress intensity factor, $\boldsymbol{K_I}$, and the *T*-stress are calculated for the single-edged notch specimen. The results are compared with the $\boldsymbol{K_I}$ values presented in Tada et al. (1973) in *Table 5* and with the *T*-stress values presented in Nakamura and Parks (1991) in *Table 6*. The comparisons show good agreement. In addition, the results obtained using the extended finite element method at the midsurface of a three-dimensional single-edged notch specimen are presented in *Table 7*.

For the interface crack model, the calculated results for the stress intensity factors and *J*-integrals are presented in *Table 8*. It can be seen from *Table 8* that, although the specimen is subjected to a pure Mode *I* loading, the value of $\boldsymbol{K_{II}}$ is nonzero—a typical feature of an interfacial crack. *Table 8* also indicates that *J* calculated from the stress intensity factors agrees very well with *J* calculated directly by Abaqus.

Submodeling around the crack tip

The submodeling technique is capable of providing a more accurate analysis of the stresses around the crack tip. The global model has a coarse mesh, while the submodel has a refined mesh. For the double-edged notch specimen and the single-edged notch specimen, the submodel region is a semicircular region of radius 127 mm (5 inches). Thus, the submodel boundary is the same as the partition around the crack tip in the global model. The submodel uses a focused mesh with six rows of elements around the crack tip. For the axisymmetric penny crack specimen, the submodel region is a semicircular region of radius 114.3 mm (4.5 inches) and coincides with the outer partition around the crack tip in the global model. The global mesh in all three problems 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 meshed global model with the boundary of the submodel (in dashed lines) is shown on the left, and on the right an enlarged view of the submodel is shown in *Figure 10* and *Figure 11* for the double-edged notch specimen and the axisymmetric penny crack specimen, respectively.

Contours of the vertical displacement field in the submodel and the global model are shown for a double-edged notch specimen in *Figure 12*. The continuity of the contour lines verifies that the proper displacement values are prescribed on the submodel boundary. Contour integral values are calculated using five contours. The results of the *J*-integral are listed in *Table 9*. The *J*-integral results obtained with the global mesh are quite accurate; hence, only minor improvements in *J*-integral values are expected. The same trend also prevails for the calculated stress intensity factor K_I and the *T*-stress. The agreement with Bowie's approximate solution is indeed slightly better, and a somewhat better path independence can be observed as well. A submodel analysis is also carried out for the axisymmetric model. The calculated *J*-integral values for the submodel analysis of the axisymmetric penny crack are listed in *Table 10*.

Python scripts

Full two-dimensional double-edged notch specimen meshed using fully integrated plane strain elements

Run the *2DDoubleEdgedNotchCPE8_model.py* script to create the model. Then run the *2DDoubleEdgedNotchCPE8_job.py* script to analyze the model.

Full two-dimensional double-edged notch specimen meshed using reduced-integration plane strain elements

Run the *2DDoubleEdgedNotchCPE8R_model.py* script to create the model. Then run the *2DDoubleEdgedNotchCPE8R_job.py* script to analyze the model.

Symmetric two-dimensional double-edged notch specimen meshed using fully integrated plane strain elements

Run the *2DDoubleEdSymmCPE8_model.py* script to create the model. Then run the *2DDoubleEdSymmCPE8_job.py* script to analyze the model.

Symmetric two-dimensional double-edged notch specimen meshed using reduced-integration plane strain elements

Run the *2DDoubleEdSymmCPE8R_model.py* script to create the model. Then run the *2DDoubleEdSymmCPE8R_job.py* script to analyze the model.

Submodel analysis of a symmetric two-dimensional double-edged notch specimen meshed using fully integrated plane strain elements

The analysis is done in two stages:

1. Run the *2DDoubleEdSymmGlCPE8_model.py* script to create the global model. Then run the *2DDoubleEdSymmGlCPE8_job.py* script to analyze the global model and to create the output database (`.odb`) file that will drive the submodel.
2. Run the *2DDoubleEdSymmSubCPE8_model.py* script to create the submodel. Then run the *2DDoubleEdSymmSubCPE8_job.py* script to analyze the submodel using the output database file from the global model to drive it.

Submodel analysis of a symmetric two-dimensional double-edged notch specimen meshed using reduced-integration plane strain elements

The analysis is done in two stages:

1. Run the *2DDoubleEdSymmGlCPE8R_model.py* script to create the global model. Then run the *2DDoubleEdSymmGlCPE8R_job.py* script to analyze the global model and to create the output database (`.odb`) file that will drive the submodel.
2. Run the *2DDoubleEdSymmSubCPE8R_model.py* script to create the submodel. Then run the *2DDoubleEdSymmSubCPE8R_job.py* script to analyze the submodel using the output database file from the global model to drive it.

Symmetric two-dimensional single-edged notch specimen meshed using fully integrated plane strain elements

Run the *2DSingleEdgedSymmCPE8_model.py* script to create the model. Then run the *2DSingleEdgedSymmCPE8_job.py* script to analyze the model.

Symmetric two-dimensional single-edged notch specimen meshed using fully integrated plane stress elements

Run the *2DSingleEdgedSymmCPS8_model.py* script to create the model. Then run the *2DSingleEdgedSymmCPS8_job.py* script to analyze the model.

Submodel analysis of a symmetric two-dimensional single-edged notch specimen meshed using fully integrated plane strain elements

The analysis is done in two stages:

1. Run the *2DSingleEdSymmGlCPE8_model.py* script to create the global model. Then run the *2DSingleEdSymmGlCPE8_job.py* script to analyze the global model and to create the output database (`.odb`) file that will drive the submodel.
2. Run the *2DSingleEdSymmSubCPE8_model.py* script to create the submodel. Then run the *2DSingleEdSymmSubCPE8_job.py* script to analyze the submodel using the output database file from the global model to drive it.

Submodel analysis of a symmetric two-dimensional single-edged notch specimen meshed using fully integrated plane stress elements

The analysis is done in two stages:

1. Run the *2DSingleEdSymmGlCPS8_model.py* script to create the global model. Then run the *2DSingleEdSymmGlCPS8_job.py* script to analyze the global model and to create the output database (`.odb`) file that will drive the submodel.
2. Run the *2DSingleEdSymmSubCPS8_model.py* script to create the submodel. Then run the *2DSingleEdSymmSubCPS8_job.py* script to analyze the submodel using the output database file from the global model to drive it.

Axisymmetric penny-shaped crack specimen meshed using fully integrated axisymmetric elements

Run the *2DAxPennyCrackCAX8_model.py* script to create the model. Then run the *2DAxPennyCrackCAX8_job.py* script to analyze the model.

Axisymmetric penny-shaped crack specimen meshed using reduced-integration axisymmetric elements

Run the *2DAxPennyCrackCAX8R_model.py* script to create the model. Then run the *2DAxPennyCrackCAX8R_job.py* script to analyze the model.

Submodel analysis of an axisymmetric penny-shaped crack specimen meshed using fully integrated axisymmetric elements

The analysis is done in two stages:

1. Run the *2DAxPennyCrackGlCAX8_model.py* script to create the global model. Then run the *2DAxPennyCrackGlCAX8_job.py* script to analyze the global model and to create the output database (`.odb`) file that will drive the submodel.
2. Run the *2DAxPennyCrackSubCAX8_model.py* script to create the submodel. Then run the *2DAxPennyCrackSubCAX8_job.py* script to analyze the submodel using the output database file from the global model to drive it.

Submodel analysis of an axisymmetric penny-shaped crack specimen meshed using reduced-integration axisymmetric elements

The analysis is done in two stages:

1. Run the *2DAxPennyCrackGlCAX8R_model.py* script to create the global model. Then run the *2DAxPennyCrackGlCAX8R_job.py* script to analyze the global model and to create the output database (`.odb`) file that will drive the submodel.
2. Run the *2DAxPennyCrackSubCAX8R_model.py* script to create the submodel. Then run the *2DAxPennyCrackSubCAX8R_job.py* script to analyze the submodel using the output database file from the global model to drive it.

Symmetric three-dimensional double-edged notch specimen meshed using fully integrated plane continuum elements

Run the *3DDoubleEdgedNotchC3D20_model.py* script to create the model. Then run the *3DDoubleEdgedNotchC3D20_job.py* script to analyze the model.

Symmetric three-dimensional double-edged notch specimen meshed using reduced-integration continuum elements

Run the *3DDoubleEdgedNotchC3D20R_model.py* script to create the model. Then run the *3DDoubleEdgedNotchC3D20R_job.py* script to analyze the model.

Symmetric two-dimensional plane strain single-edged notch specimen containing an interface crack

Run the *2DDissimilarMaterialsCPE8_model.py* script to create the model. Then run the *2DDissimilarMaterialsCPE8_job.py* script to analyze the model.

Input files

The input files listed below are provided for users who prefer to use the Abaqus keyword interface instead of Abaqus/CAE. The meshes created in these input files are different from those created by using the Python scripts; however, the results are of the same accuracy.

jintegral2d_cpe8.inp	Two-dimensional plane strain model with full integration.
jintegral2d_cpe8r.inp	Two-dimensional plane strain model with reduced integration.
jintegral2d_cpe4_residual.inp	Two-dimensional first-order plane strain model with full integration and the effect of a residual stress field.
jintegral2d_cpe8_submodel.inp	Two-dimensional plane strain submodel with full integration.
jintegral2d_c3d20.inp	20-node brick three-dimensional model with full integration.
jintegral2d_postoutput.inp	**POST OUTPUT* analysis of jintegral2d_c3d20.inp.
jintegral2d_c3d20r.inp	20-node brick three-dimensional model with reduced integration.
jintegral2d_c3d27.inp	27-node brick three-dimensional model with full integration.
jintegral2d_c3d27r.inp	27-node brick three-dimensional model with reduced integration.
jintegral2d_cax8.inp	Axisymmetric model with full integration.
jintegral2d_cax8r.inp	Axisymmetric model with reduced integration.
jintegral2d_cax8_submodel.inp	Axisymmetric submodel with full integration.
jintegral2d_3daxi.inp	20-node brick three-dimensional model of the axisymmetric problem with reduced integration.
cintegral2d_1edge_cpe8.inp	Two-dimensional plane strain model for single-edged notch specimen.
cintegral2d_1edge_cps8.inp	Two-dimensional plane stress model for single-edged notch specimen.
cintegral2d_1edge_intf_cpe8.inp	Two-dimensional plane strain model for single-edged notch specimen containing an interface crack.
contourintegral_den_xfem_c3d8.inp	8-node brick three-dimensional model with full integration for double-edged notch specimen with extended finite element method.
contourintegral_den_xfem_c3d8r.inp	8-node brick three-dimensional model with reduced integration for double-edged notch specimen with extended finite element method.

contourintegral_sen_xfem_c3d8.inp	8-node brick three-dimensional model with full integration for single-edged notch specimen with extended finite element method.
contourintegral_sen_xfem_c3d8_base.inp	Same as contourintegral_sen_xfem_c3d8.inp but including the plasticity to generate a residual stress field.
contourintegral_sen_xfem_c3d8_residual.inp	Same as contourintegral_sen_xfem_c3d8.inp but including the effect of a residual stress field that was generated from the analysis of contourintegral_sen_xfem_c3d8_base.
contourintegral_sen_xfem_c3d4.inp	4-node tetrahedron three-dimensional model for single-edged notch specimen with extended finite element method.

References

- Barsoum, R. S., "On the Use of Isoparametric Finite Elements in Linear Fracture Mechanics," *International Journal for Numerical Methods in Engineering*, vol. 10, pp. 25–37, 1976.
- Bowie, O. L., "Rectangular Tensile Sheet With Symmetric Edge Cracks," *Journal of Applied Mechanics*, vol. 31, pp. 208–212, 1964.
- Nakamura, T., and D. M. Parks, "Determination of Elastic *T*-Stress along Three-Dimensional Crack Fronts Using an Interaction Integral," *International Journal of Solids and Structures*, vol. 28, pp. 1597–1611, 1991.
- 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.
- Tada, H., P. C. Paris, and G. R. Irwin, The Stress Analysis of Cracks Handbook, Del Research Corporation, Hellertown, Pennsylvania, 1973.

Tables

Table 1: *J*-integral values: two-dimensional symmetric double-edged notch specimen modeled using plane strain elements. Bowie's approximate solution: *J* = 2.245 N/m (0.0128 lb/in).

Contour	Full integration		Reduced integration	
	N/m	lb/in	N/m	lb/in
1	2.284	0.01303	2.285	0.01304
2	2.282	0.01302	2.280	0.01301
3	2.282	0.01302	2.282	0.01302
4	2.282	0.01302	2.282	0.01302
5	2.282	0.01302	2.282	0.01302

Table 2: *J*-integral values: three-dimensional symmetric double-edged notch specimen modeled using continuum elements.

Full integration						
Contour	Front face		Middle surface		Back face	
	N/m	lb/in	N/m	lb/in	N/m	lb/in
1	2.212	0.01262	2.306	0.01316	2.212	0.01262
2	2.277	0.01299	2.277	0.01299	2.277	0.01299
3	2.280	0.01301	2.280	0.01301	2.280	0.01301
Reduced integration						

Contour	Front face		Middle surface		Back face	
	N/m	lb/in	N/m	lb/in	N/m	lb/in
1	2.273	0.01297	2.284	0.01303	2.273	0.01297
2	2.277	0.01299	2.277	0.01299	2.277	0.01299
3	2.280	0.01301	2.280	0.01301	2.280	0.01301

Table 3: J-integral values: three-dimensional double-edged notch specimen modeled using continuum elements with the extended finite element method.

Full integration				
Contour	Front face		Back face	
	N/m	lb/in	N/m	lb/in
3	2.874	0.01641	2.874	0.01641
4	2.795	0.01596	2.795	0.01596
5	3.108	0.01775	3.108	0.01775
6	2.559	0.01461	2.559	0.01461
7	2.343	0.01338	2.343	0.01338
8	2.485	0.01419	2.485	0.01419
Reduced integration				
Contour	Front face		Back face	
	N/m	lb/in	N/m	lb/in
3	2.884	0.01647	2.884	0.01647
4	2.799	0.01598	2.799	0.01598
5	3.112	0.01777	3.112	0.01777
6	2.567	0.01466	2.567	0.01466
7	2.361	0.01348	2.361	0.01348
8	2.494	0.01424	2.494	0.01424

Table 4: J-integral values: axisymmetric penny-shaped crack specimen. Tada et al. approximate solution: J = 0.7635 N/m (0.00436 lb/in).

Contour	Full integration		Reduced integration	
	N/m	lb/in	N/m	lb/in
1	0.7870	0.00449	0.7853	0.00448
2	0.7818	0.00446	0.7853	0.00448
3	0.7835	0.00447	0.7870	0.00449
4	0.7835	0.00447	0.7870	0.00449
5	0.7835	0.00447	0.7870	0.00449

Table 5: Nondimensional stress intensity factor $K_I/\sigma^\infty\sqrt{\pi a}$ for two-dimensional symmetric single-edged notch specimen. Tada et al. approximate solution: $K_I/\sigma^\infty\sqrt{\pi a} = 2.826$.

Contour	CPE8	CPS8
1	2.8250	2.8249
2	2.8230	2.8231
3	2.8237	2.8238
4	2.8238	2.8239
5	2.8238	2.8239

Table 6: Nondimensional T-stress T/σ^∞ for single-edged notch specimen. Nakamura and Parks approximate solution: $T/\sigma^\infty = -0.43$.

Contour	CPE8	CPS8
1	–0.4307	–0.4298
2	–0.4204	–0.4202
3	–0.4226	–0.4224
4	–0.4226	–0.4224
5	–0.4225	–0.4223

Table 7: Nondimensional stress intensity factor $K_I/\sigma^\infty\sqrt{\pi a}$ at the midsurface for three-dimensional single-edged notch specimen with extended finite element method. Tada et al. approximate solution: $K_I/\sigma^\infty\sqrt{\pi a} = 2.826$.

Contour	C3D8	C3D4
2	2.8537	2.8871
3	2.9643	2.8675
4	3.0027	2.8675
5	2.9696	2.9014

Table 8: Nondimensional $K_I/\sigma^\infty\sqrt{\pi a}$, $K_{II}/\sigma^\infty\sqrt{\pi a}$, and $JE/\sigma^{\infty 2}a$ values of an interface crack.

Contour	*J*-integral value estimated by the stress intensity factors			*J*-integral value estimated directly
	$K_I/\sigma^\infty\sqrt{\pi a}$	$K_{II}/\sigma^\infty\sqrt{\pi a}$	$JE/\sigma^{\infty 2}a$ from Ks	$JE/\sigma^{\infty 2}a$
1	2.8245	0.0121	17.36	17.30
2	2.8226	0.0127	17.33	17.30
3	2.8232	0.0127	17.34	17.30
4	2.8233	0.0127	17.34	17.30
5	2.8232	0.0127	17.34	17.30

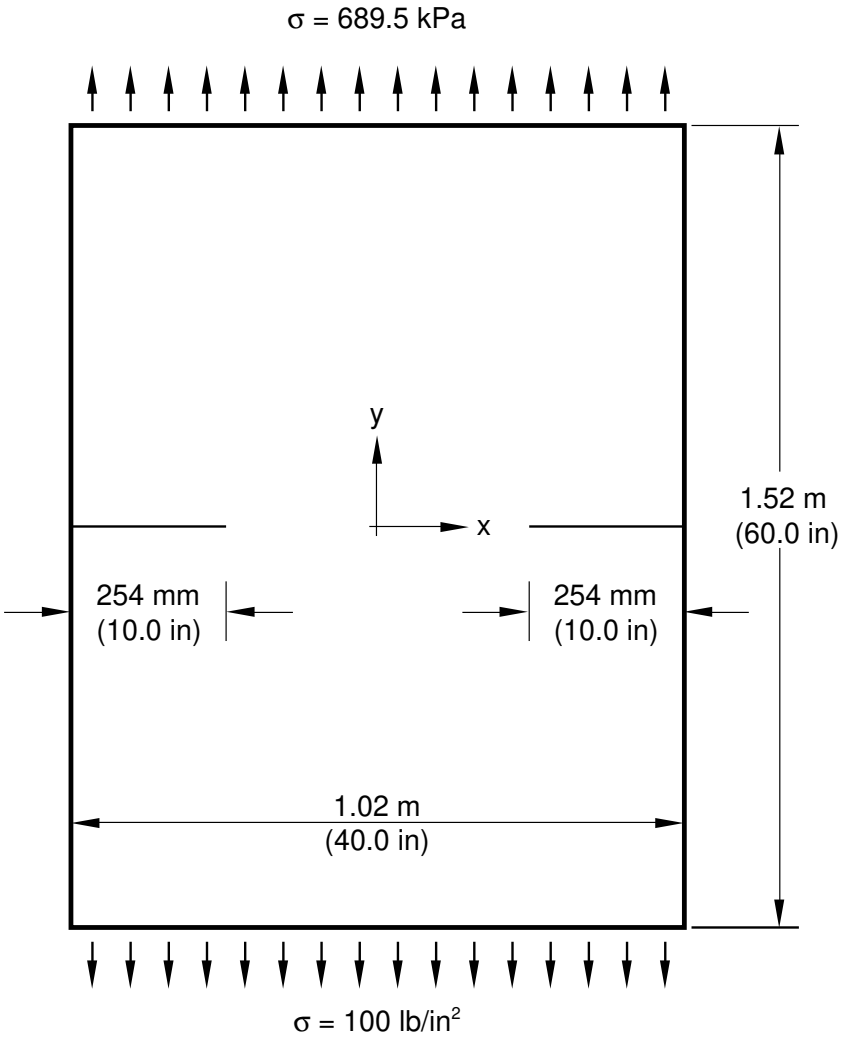
Table 9: J-integral values: two-dimensional submodel analysis of a double-edged notch specimen using plane strain elements.

Contour	Full integration		Reduced integration	
	N/m	lb/in	N/m	lb/in
1	2.282	0.01302	2.285	0.01304
2	2.278	0.013	2.280	0.01301
3	2.280	0.01301	2.282	0.01302
4	2.280	0.01301	2.282	0.01302
5	2.280	0.01301	2.282	0.01302

Table 10: J-integral values: submodel analysis of an axisymmetric penny-shaped crack specimen.

Contour	Full integration		Reduced integration	
	N/m	lb/in	N/m	lb/in
1	0.7765	0.00443	0.7853	0.00448
2	0.7748	0.00442	0.7835	0.00447
3	0.7765	0.00443	0.7835	0.00447
4	0.7765	0.00443	0.7835	0.00447
5	0.7765	0.00443	0.7835	0.00447

Figures



Material: linear elastic, E = 206.8 GPa (30.0 x 10^6 lb/in^2)
υ = 0.3

Figure 1: Double-edged notch example.

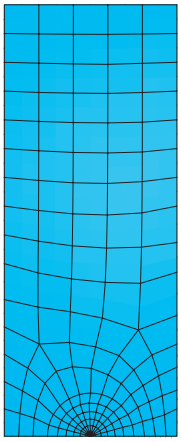


Figure 2: Symmetric finite element model of double-edged notch specimen.

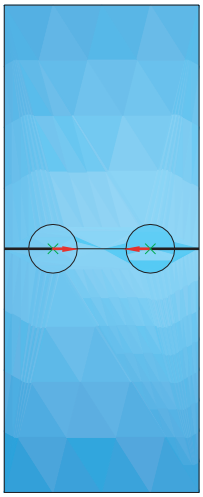


Figure 3: Model showing the seam cracks defined in bold and the q vectors defined at the left and right crack tips.

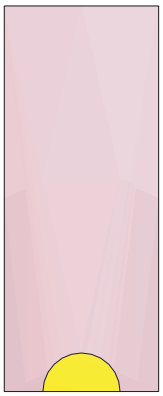


Figure 4: Two-dimensional double-edged notch specimen showing the partitions created around the crack tip. The circular partitioned region is meshed using the sweep meshing technique with quad-dominated elements.

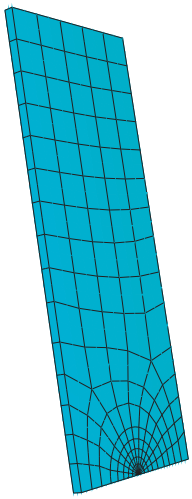


Figure 5: Finite element model of a three-dimensional quarter model of the double-edged notch specimen meshed with one layer of C3D20 elements.

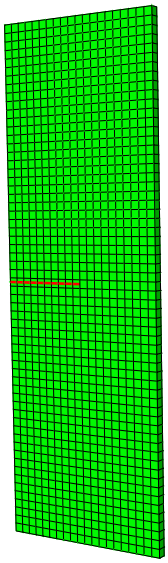
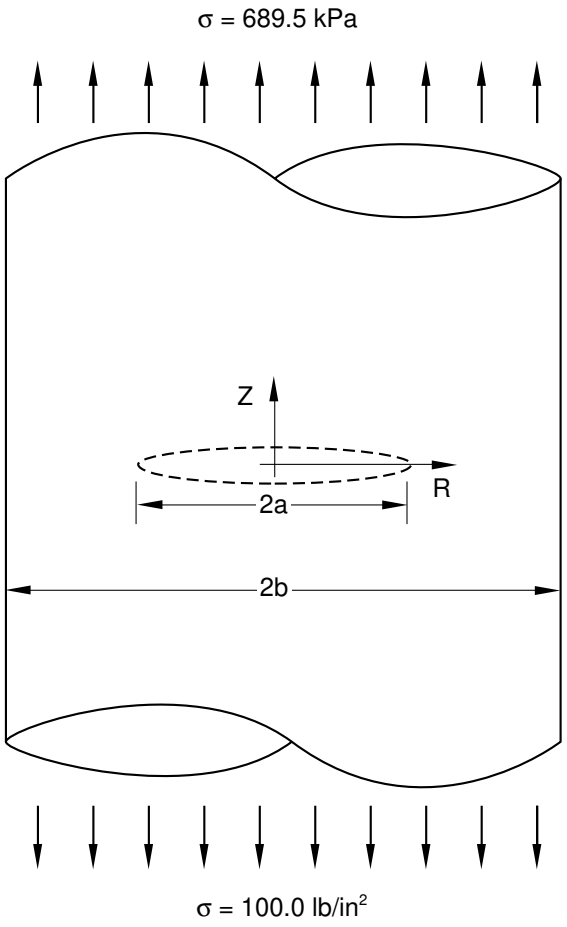


Figure 6: A three-dimensional finite element model of the double-edged notch specimen meshed with one layer of first-order brick elements using the extended finite element method.



Material: linear elastic, E = 206.8 GPa (30.0 x 10^6 lb/in^2)
υ = 0.3

Figure 7: Penny-shaped crack in round bar.

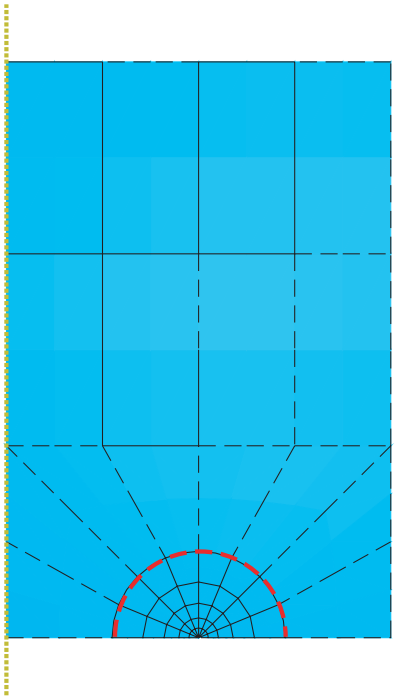


Figure 8: Axisymmetric finite element model of penny-shaped crack in round bar.

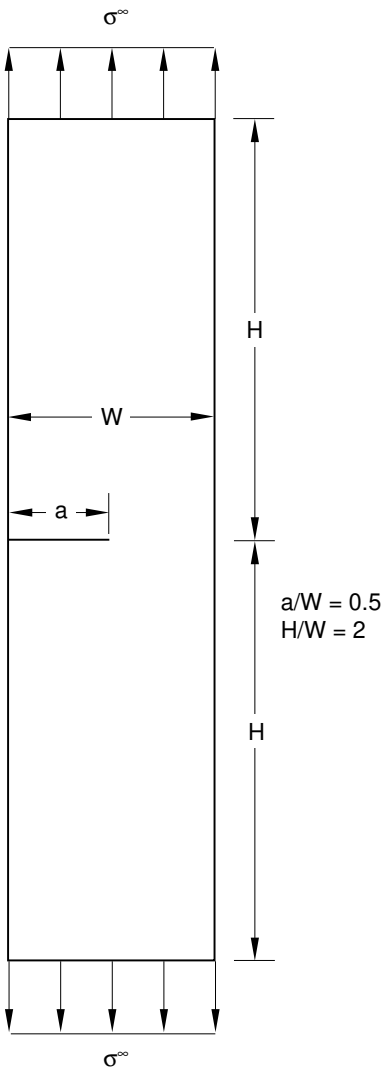


Figure 9: Single-edged notch specimen.

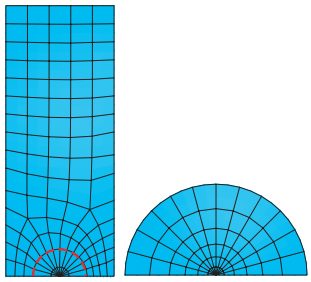


Figure 10: Left: meshed global model of two-dimensional double-edged notch specimen with boundary of submodel shown with dashed lines, only two elements in submodel region. Right: enlarged view of submodel, refined mesh with six rows of elements.

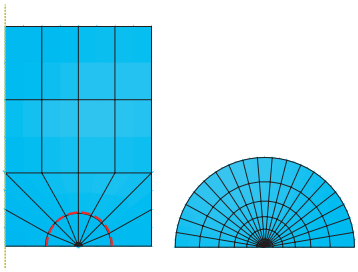


Figure 11: Left: meshed global model of axisymmetric penny-shaped crack specimen with boundary of submodel shown with dashed lines, only two elements in submodel region. Right: enlarged view of submodel, refined mesh with six rows of elements.

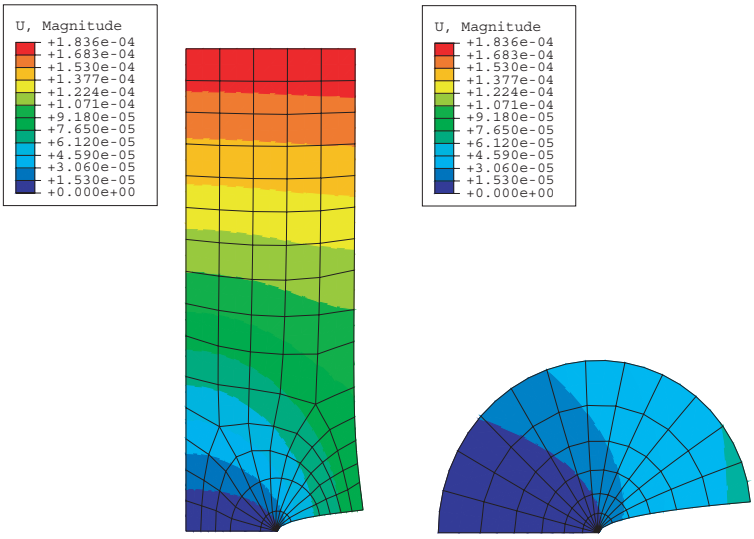


Figure 12: Displacement contours of the global model and the submodel for a two-dimensional double-edged notch specimen.

Contour integral evaluation: three-dimensional case

Products: Abaqus/Standard

Overview

This example illustrates contour integral evaluation in a fully three-dimensional crack configuration.
The example provides validation of the method (for linear elastic response), because comparative results are available for this geometry.

Abaqus provides values of the J-integral; stress intensity factor, K_I; and T-stress as a function of position along a crack front in three-dimensional geometries. Several contours can be used; and, since the integral should be path independent, the scatter in the values obtained with different contours can be used as an indicator of the quality of the results. The domain integral method used to calculate the contour integral in Abaqus generally gives accurate results even with rather coarse models, as is shown in this case. Abaqus offers the evaluation of these parameters for fracture mechanics studies based on either the conventional finite element method or the extended finite element method (XFEM).

Problem description

Two geometries are analyzed in this example. In addition, both the conventional finite element method and the extended finite element method are used.

Semi-elliptic crack in a half-space

The first geometry analyzed is a semi-elliptic crack in a half-space and is shown in *Figure 1*. The crack is loaded in Mode I by far-field tension. When used with the conventional finite element method, because of symmetry, only one-quarter of the body needs to be analyzed. The mesh is shown in *Figure 2*. Reduced-integration elements (C3D20R) are used, with the midside nodes moved to the quarter-point position on those element edges that focus onto the crack tip nodes. This quarter-point method provides a strain singularity and, thus, improves the modeling of the strain field adjacent to the crack tip (see *Contour integral evaluation: two-dimensional case* for a discussion of this technique).

The normal to the crack front is used to specify the crack extension direction, as shown in *Figure 3*. The mesh extends out far enough to cause the boundary conditions on the far faces of the model to have negligible effect on the solution. Three rings of elements surrounding the crack tip are used to evaluate the contour integrals.

A model meshed with second-order tetrahedral elements (C3D10) is also created using the conventional finite element method. This model is shown in *Figure 4* with a mesh that is unstructured (as is typical for tetrahedral meshes) and, therefore, is not focused about the crack front. This mesh does not have midside nodes adjacent to the crack front moved to the quarter-point position. A refined mesh around the crack front is required to obtain accurate results from contour integral evaluations in these conditions.

When used with the extended finite element method, the mesh is not required to match the cracked geometry. The presence of a crack is ensured by the special enriched functions in conjunction with additional degrees of freedom. This approach also removes the requirement to define the crack front explicitly or to specify the virtual crack extension direction when evaluating the contour integral. The data required for the contour integral are determined automatically based on the level set signed distance functions at the nodes in an element. The mesh with first-order brick elements (C3D8) for the first geometry is shown in *Figure 5*; and the crack front, which is represented by the level set contour plot of output variable PSILSM, is shown in *Figure 6*.

Semi-elliptic crack in a rectangular plate

The second geometry analyzed is a semi-elliptic crack in a rectangular plate, as shown in *Figure 7*. The plate is subjected to uniform tension. Due to symmetry only one-quarter of the body needs to be analyzed when used with the conventional finite element method. The dimensions of the plate relative to the plate thickness, *t*, are as follows: the half-height $H/t =$ 16; the half-width $W/t =$ 8; the midplane crack depth $a/t =$ 0.6; and the surface crack half-length $c/t =$ 2.5, which results in a surface crack aspect ratio of $a/c =$ 0.24.

The mesh for the plate using the conventional finite element method is shown in *Figure 8*, with its profile on the crack plane for brick elements and tetrahedral elements shown in *Figure 9* and *Figure 10*, respectively. A more refined mesh is required around the crack front when using tetrahedral elements. The model uses C3D8 and C3D10 elements for the conventional finite element method and both C3D8 and C3D4 elements for the extended finite element method.

Results and discussion

The results are presented for each of the geometries.

Semi-elliptic crack in a half-space

The *J*-integral values computed by Abaqus using the conventional finite element method for the first geometry are given in *Table 1* as functions of angular position ϕ along the crack front, where ϕ is defined by $x/c = \cos\phi$, $z/a = -\sin\phi$. The values show a rather smooth variation along the crack front and are reasonably path independent; that is, the values provided by the three contours are almost the same. There is some loss of path independence and, hence, presumably, of accuracy as the crack approaches the free surface (at $\phi = 0$). This accuracy loss is assumed to be attributable to the coarse and rather distorted mesh in that region.

The stress intensity factors K_I obtained using the conventional finite element method along the crack line are compared in *Table 2* and in *Figure 11* with those obtained by Newman and Raju (1979), who used a nodal force method to compute K_I from a finite element model that had 3078 degrees of freedom. The *J*-integral values calculated by Abaqus are converted to K_I using

$$J = \frac{1-\nu^2}{E} K_I^2,$$

where ν is the Poisson's ratio and *E* is the Young's modulus. The fourth column of *Table 2* presents the stress intensity factors, K_I, that are calculated directly by Abaqus using the conventional finite element method. For comparison, the stress intensity factors K_I obtained using the extended finite element method are also included in *Figure 11*. The comparisons show good agreement with the results published by Newman and Raju (1979).

Semi-elliptic crack in a rectangular plate

The stress intensity factor values, K_I, computed by Abaqus based on the conventional finite element method for the second geometry are compared with the K_I results calculated by Nakamura and Parks (1991) in *Figure 12*, and the agreement between them is excellent. The difference is less than 4% when comparing the results obtained with brick elements with those obtained with second-order tetrahedral elements using the conventional finite element method. The results obtained using the extended finite element method with linear brick or linear tetrahedral elements are also included in *Figure 12* for comparison.

Figure 13 presents the *T*-stresses calculated by Abaqus and those obtained by Nakamura and Parks (1991) and Wang and Parks (1992). Comparison shows good agreement between them, especially near the middle of the crack line.

Python scripts

Semi-elliptic crack in a half-space

Run the *3DEllipticCrackC3D20R_model.py* script to create the model. Then run the *3DEllipticCrackC3D20R_job.py* script to analyze the model.

Semi-elliptic crack in a rectangular plate

Run the *RectBlEllipticCrC3D8_model.py* script to create the model. Then run the *RectBlEllipticCrC3D8_job.py* script to analyze the model.

Input files

The input files below create models with different meshes than the Abaqus/CAE models created by the Python scripts above. The results are identical in both cases.

jintegral3d.inp	First model with brick elements using the conventional finite element method.
jintegral3d_node.inp	Nodal coordinates for the first model. These have been generated by a special-purpose program.
jintegral3d_line_c3d10_half.inp	First model with second-order tetrahedral elements using the conventional finite element method.
contourintegral_ellip_xfem_c3d8.inp	First model using the extended finite element method.
jktintegral3d.inp	Second model with brick elements using the conventional finite element method.
jktintegral3d_node.inp	Node definitions for jktintegral3d.inp.
jktintegral3d_element.inp	Element definitions for jktintegral3d.inp.
jktintegral3d_line_c3d10_half.inp	Second model with second-order tetrahedral elements using the conventional finite element method.
contourintegral_ellip_plate_xfem_c3d8.inp	Second model with linear brick elements using the extended finite element method.
contourintegral_ellip_plate_xfem_c3d4.inp	Second model with linear tetrahedral elements using the extended finite element method.

References

- Nakamura, T., and D. M. Parks, "Determination of Elastic *T*-Stress along Three-Dimensional Crack Fronts Using an Interaction Integral," *International Journal of Solids and Structures*, vol. 28, pp. 1597–1611, 1991.
- Newman, J. C., and I. S. Raju, "Stress-Intensity Factors for a Wide Range of Semi-Elliptical Surface Cracks in Finite Thickness Plates," *Engineering Fracture Mechanics*, vol. 11, pp. 817–829, 1979.
- Wang, Y-Y., and D. M. Parks, "Evaluation of the Elastic *T*-Stress in Surface-Cracked Plate Using the Line-Spring Method," *International Journal of Fracture*, vol. 56, pp. 25–40, 1992.

Tables

Table 1: J-integral estimates for semi-elliptic crack using the conventional method with brick elements (× 10^{-3} N/mm (top); × 10^{-3} lb/in (bottom)).

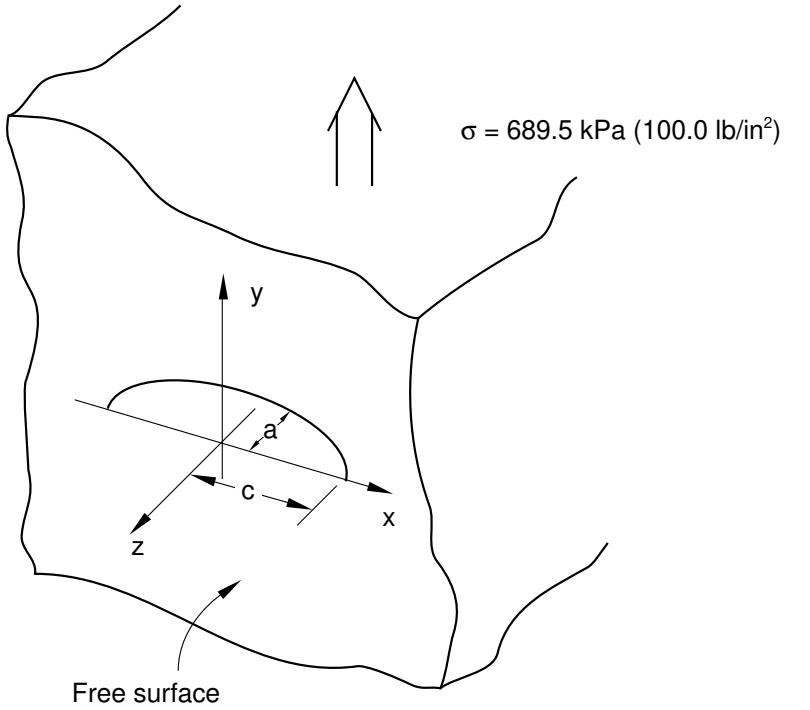
Crack Front Location, ϕ (deg)	Contour			Average Value
	1	2	3	
0.00	0.8006	0.7838	0.7862	0.7902
	4.5716	4.4757	4.4896	4.5123
11.25	0.7804	0.8014	0.8073	0.7964
	4.4562	4.5759	4.6098	4.5473
22.50	0.8726	0.8629	0.8558	0.8638
	4.9827	4.9271	4.8867	4.9322
33.75	1.0617	1.0643	1.0673	1.0644
	6.0623	6.0771	6.0945	6.0780
45.00	1.2199	1.2178	1.2159	1.2179
	6.9658	6.9540	6.9430	6.9543
56.25	1.3701	1.3698	1.3686	1.3695
	7.8232	7.8220	7.8151	7.8201
67.50	1.4975	1.4965	1.4965	1.4968
	8.5512	8.5455	8.5450	8.5472
78.75	1.5503	1.5499	1.5496	1.5499
	8.8525	8.8504	8.8484	8.8504
90.00	1.6007	1.5996	1.5986	1.5996
	9.1400	9.1337	9.1283	9.1340

Table 2: Comparison of computed Mode I stress intensity factors using the conventional method with brick elements (N/mm^2–mm$^{1/2}$ (top); lb/in^2–in$^{1/2}$ (bottom)).

Crack Front Location, ϕ (deg)	Newman and Raju	Average Value with Conventional Method (calculated from J-integral)	Average Value with Conventional Method (calculated directly by Abaqus)
0.00	12.99	13.41	13.14
	373.60	385.69	377.90
11.25	13.23	13.46	13.45
	380.50	387.19	386.77
22.50	14.26	14.02	13.99
	410.20	403.24	402.37
33.75	15.63	15.56	15.55
	449.60	447.64	447.37
45.00	16.90	16.65	16.65

Crack Front Location, ϕ (deg)	Newman and Raju	Average Value with Conventional Method (calculated from J-integral)	Average Value with Conventional Method (calculated directly by Abaqus)
	486.20	478.82	478.90
56.25	17.92	17.65	17.66
	515.40	507.75	507.77
67.50	18.68	18.46	18.45
	537.30	530.83	530.74
78.75	19.12	18.78	18.79
	554.40	540.16	540.33
90.00	19.27	19.08	19.07
	554.40	548.75	548.60

Figures



a = 254 mm (10.0 in)
c = 635 mm (25.0 in)

Elastic material: Young's modulus = 206.8 GPa (30.0 x 10^6 lb/in^2)
Poisson's ratio = 0.3

Figure 1: Semi-elliptic surface crack in a half-space.

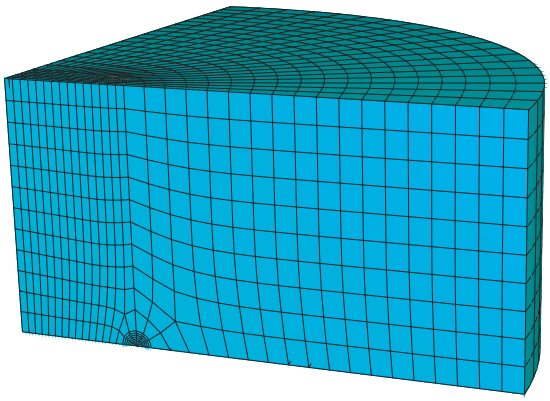


Figure 2: Mesh for semi-elliptic surface crack problem with brick elements using the conventional finite element method.

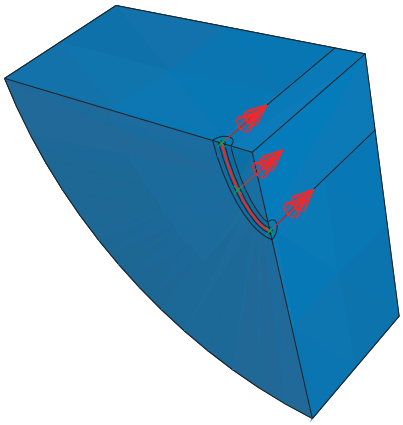


Figure 3: Normal to the crack front is used to define the crack extension direction.

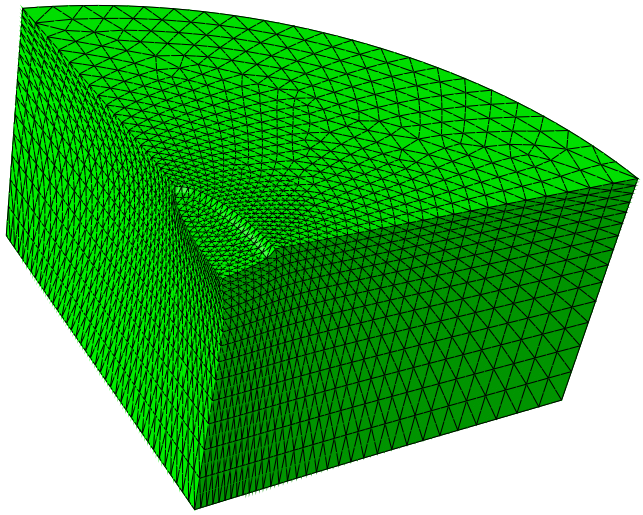


Figure 4: Mesh for semi-elliptic surface crack problem with tetrahedral elements using the conventional finite element method.

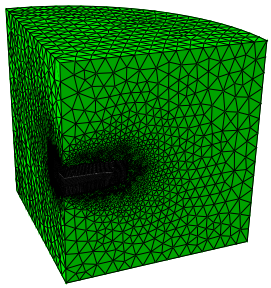


Figure 5: Mesh for semi-elliptic surface crack problem using the extended finite element method.

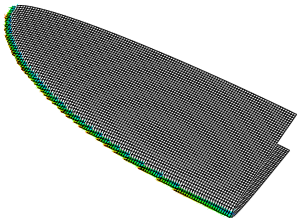


Figure 6: Crack front for the semi-elliptic surface crack problem using the extended finite element method.

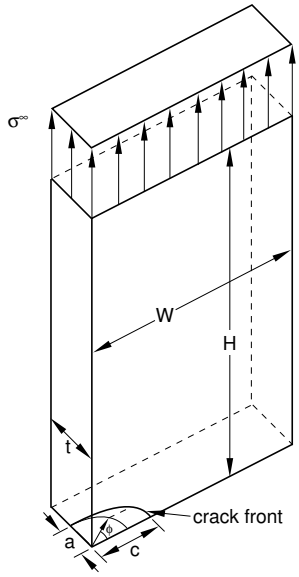


Figure 7: Semi-elliptic surface crack in a rectangular plate.

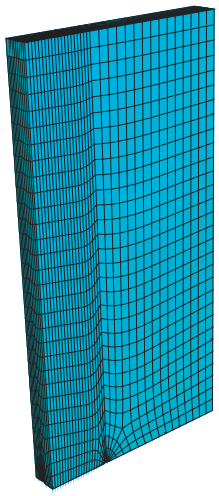


Figure 8: Mesh for semi-elliptic surface crack in a rectangular plate using the conventional finite element method.

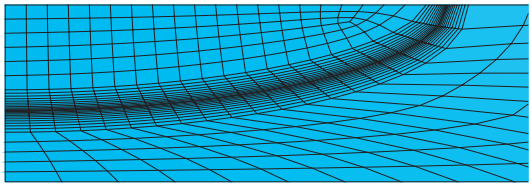


Figure 9: Mesh profile on the crack surface of brick elements using the conventional finite element method.

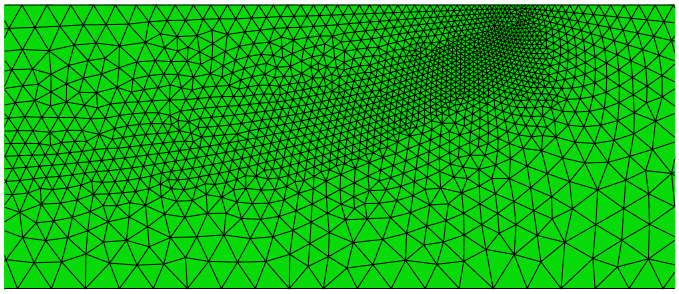


Figure 10: Mesh profile on the crack surface of tetrahedral elements using the conventional finite element method.

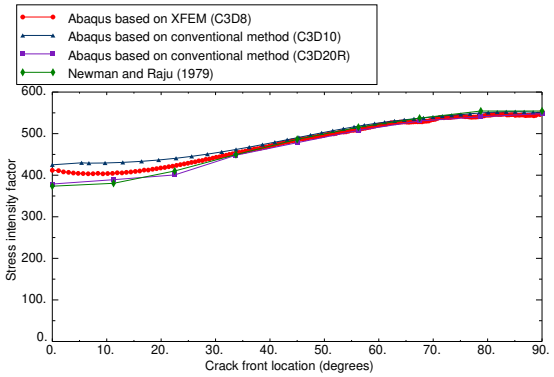


Figure 11: Stress intensity factors computed for a semi-elliptic crack.

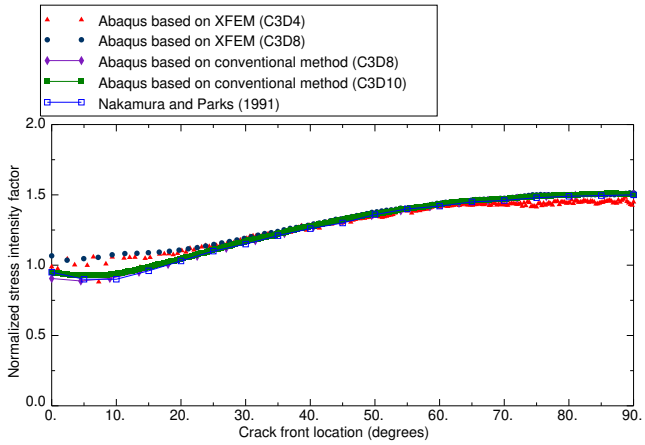


Figure 12: Stress intensity factors computed for a semi-elliptic crack in a rectangular plate.

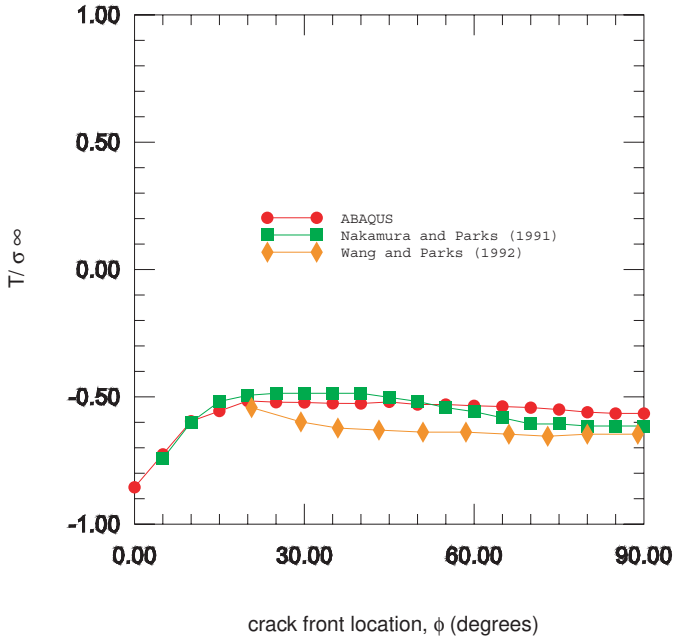


Figure 13: T-stresses computed for a semi-elliptic crack in a rectangular plate.

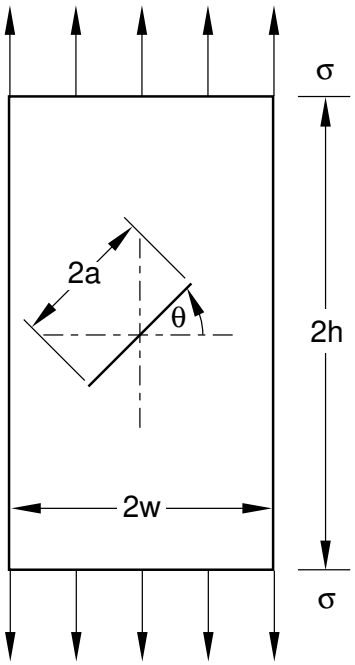
Center slant cracked plate under tension

Products: Abaqus/Standard

Overview

The example predicts the propagation of a center slant crack subjected to far-field tension.

Problem description



The model consists of a rectangular plate with a center slant crack subjected to far-field tension. The calculation is carried out using CPE8 elements with a linear elastic material. The geometrical ratios chosen for the plate are $a/w = 0.4$ and $h/w = 2$. The slant angle is $\theta = 45°$. Displacement boundary conditions are prescribed on the top and bottom plate surfaces to apply tension to the plate so that consistent solutions can be obtained around the two crack tips. The total remote tensile load is obtained by summing the y-direction nodal reaction forces on every node in the top or bottom plane; and the remote stress, σ, is defined as the total tensile force divided by the cross-sectional area of the plate.

Results and discussion

The calculated stress intensity factors are compared with the solutions taken from Page 909 of the *Stress Intensity Factors Handbook*, edited by Y. Murakami.

Table 1: Nondimensional stress intensity factor results for isotropic elasticity. Contour 1 is omitted from the calculation.

	$K_I/\sigma\sqrt{\pi a}$	$K_{II}/\sigma\sqrt{\pi a}$
Reference solution	0.5719	0.5290
Abaqus	0.5540	0.5289

Based on the stress intensity factors K_I and K_{II}, Abaqus can automatically predict the crack propagation direction, which is an angle $\hat{\theta}$ measured with respect to the crack plane. For example, $\hat{\theta} = -52.41°$ if the maximum tangential stress criterion is used, $\hat{\theta} = -55.76°$ if the maximum energy release rate criterion is used, and $\hat{\theta} = -56.12°$ if the $K_{II} = 0$ criterion is used.

Abaqus also outputs the *J*-integral value estimated by the stress intensity factors, $J/(\sigma a) = 3.4517 \times 10^{-3}$, which agrees very well with the *J*-integral value estimated directly, $J/(\sigma a) = 3.4515 \times 10^{-3}$.

In addition, the stress intensity factors and the *J*-integral are evaluated for the same plate using four different anisotropic, linear elastic materials: orthotropic elasticity specified by the engineering constants (denoted by ENGC), orthotropic elasticity specified by the stiffness parameters (ORTH), fully anisotropic elasticity (ANIS), and lamina elasticity (LAMI). The model with lamina elasticity is meshed using plane stress CPS8 elements. The results are summarized in *Table 2*. Though no published solutions are available for comparison, the *J*-integrals from the stress intensity factors are in very good agreement with the *J*-integrals evaluated directly.

Table 2: Nondimensional $K_I/\sigma\sqrt{\pi a}$, $K_{II}/\sigma\sqrt{\pi a}$, and $J/(\sigma a)$ values for a slant crack. Contour 1 is omitted from the average value calculation.

Elasticity	*J*-integral value estimated by the stress intensity factors			*J*-integral value estimated directly
	$K_I/\sigma\sqrt{\pi a}$	$K_{II}/\sigma\sqrt{\pi a}$	$J/(\sigma a) \times 10^3$	$J/(\sigma a) \times 10^3$
ENGC	0.5717	0.5299	2.8740	2.8750
ORTH	0.599	0.5429	2.1931	2.1930
ANIS	0.5388	0.5260	4.3790	4.3798
LAMI	0.5391	0.5223	4.7639	4.7637

Python scripts

Two-dimensional model for isotropic elasticity

Run the *SlantCrackElasCPE8_model.py* script to create the model. Then run the *SlantCrackElasCPE8_job.py* script to analyze the model.

Two-dimensional model for orthotropic elasticity specified by the engineering constants

Run the *SlantCrackOrthEcCPE8_model.py* script to create the model. Then run the *SlantCrackOrthEcCPE8_job.py* script to analyze the model.

Two-dimensional model for orthotropic elasticity specified by stiffness parameters

Run the *SlantCrackOrthStCPE8_model.py* script to create the model. Then run the *SlantCrackOrthStCPE8_job.py* script to analyze the model.

Two-dimensional model for anisotropic elasticity

Run the *SlantCrackAnisoCPE8_model.py* script to create the model. Then run the *SlantCrackAnisoCPE8_job.py* script to analyze the model.

Two-dimensional model for lamina elasticity

Run the *SlantCrackLamCPS8_model.py* script to create the model. Then run the *SlantCrackLamCPS8_job.py* script to analyze the model.

Input files

The following input files create models that have different meshes from the Abaqus/CAE models created through the above Python scripts. The results are identical in both the cases.

psptskf2d.inp	Two-dimensional model for isotropic elasticity.
psptskf2d_node.inp	Node definitions.
psptskf2d_element.inp	Element definitions.
psptskf2d_engc.inp	Two-dimensional model for orthotropic elasticity specified by the engineering constants.
psptskf2d_orth.inp	Two-dimensional model for orthotropic elasticity specified by stiffness parameters.
psptskf2d_anis.inp	Two-dimensional model for anisotropic elasticity.
psptskf2d_lami.inp	Two-dimensional model for lamina elasticity.

A penny-shaped crack under concentrated forces

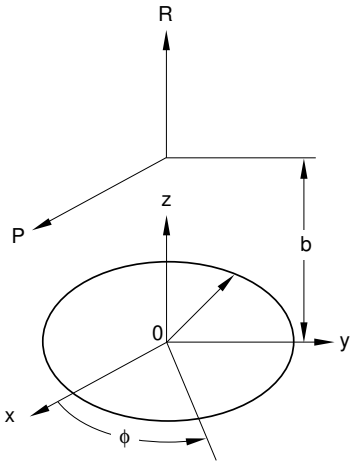
Products: Abaqus/Standard

Overview

The example illustrates the analysis of a penny-shaped crack in an infinite body, subjected to concentrated forces.

The analytical solutions for the stress intensity factors can be found on pages 672–673 of the *Stress Intensity Factors Handbook* edited by Y. Murakami.

Problem description



The geometry analyzed is a penny-shaped crack in an infinite body, subjected to concentrated forces P and R, as shown in the above figure. The size of the body is chosen big enough relative to the crack radius, a, such that an infinite body containing a small penny crack can be modeled. The ratio of lengths $b/a =$ 0.66. The material is linear elastic, with Young's modulus = 200 GPa and Poisson's ratio = 0.3. Two remote points on the z-axis in the body are fixed to prevent rigid body motion. The loading is $P = R =$ 400 MN.

Results and discussion

Due to symmetry only the results from $\phi =$ 0° to $\phi =$ 180° are presented, although the calculations are carried out for a full solid body. The calculated stress intensity factors are compared with the analytical ones in *Figure 1*.

Abaqus outputs the J-integral values estimated by the stress intensity factors when the latter are requested. They are compared with the J-integral values calculated directly and also compared with the analytical J-integral solutions in *Figure 2*.

The stress intensity factors and the J-integral values are also evaluated using the extended finite element method (XFEM). The values are compared with the analytical solutions in *Figure 3* and *Figure 4*.

Input files

ppennycrack.inp	Three-dimensional model with conventional finite element method.
ppennycrack_node.inp	Node definitions for ppennycrack.inp.
ppennycrack_element.inp	Element definitions for ppennycrack.inp.
contourintegral_penny_xfem_c3d8.inp	Three-dimensional model with extended finite element method.

Figures

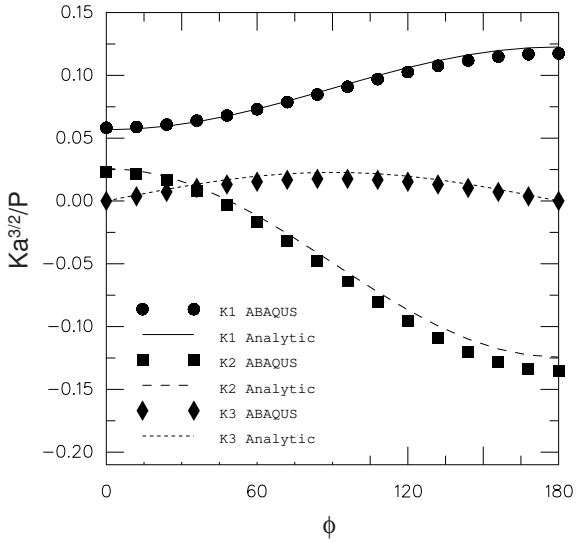


Figure 1: Variation of the stress intensity factors with ϕ.

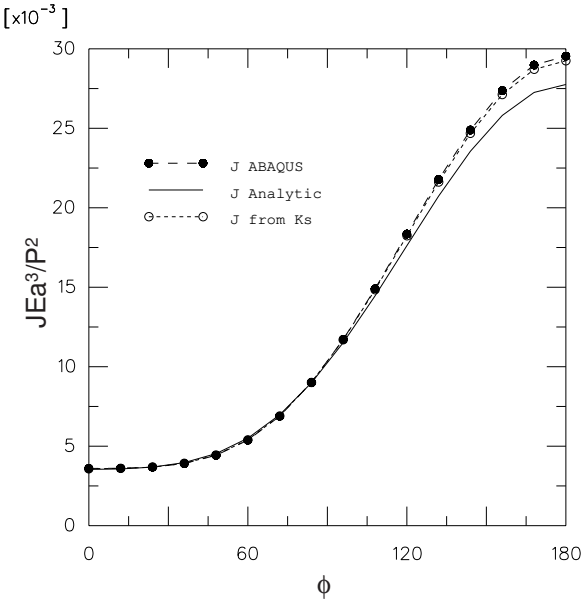


Figure 2: Variation of the J-integral with ϕ.

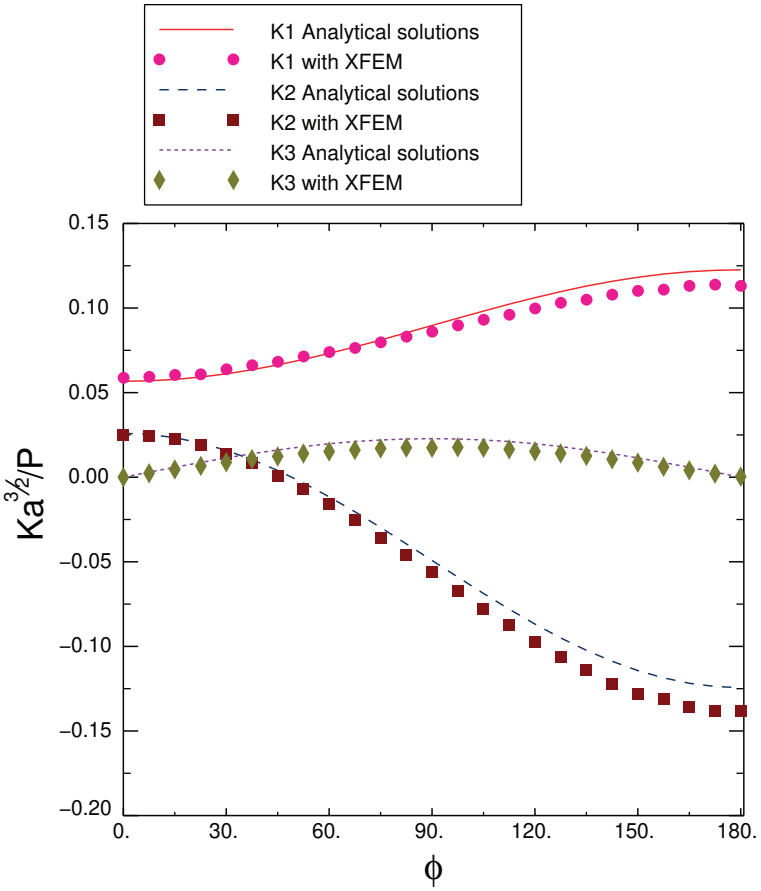


Figure 3: Variation of the stress intensity factors with ϕ obtained using the extended finite element method.

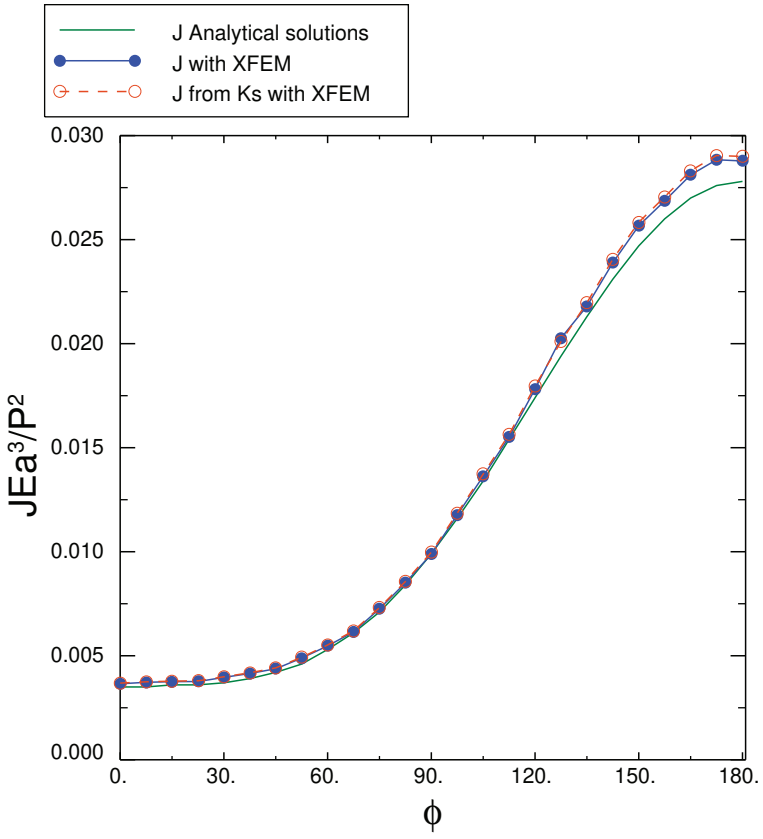


Figure 4: Variation of the J-integral with ϕ obtained using the extended finite element method.

Fully plastic *J*-integral evaluation

Products: Abaqus/Standard

Overview

This example illustrates fully plastic *J*-integral evaluation using deformation theory plasticity, as is used in the "engineering fracture mechanics" methodology developed by Kumar, et al. (1981).

In this type of analysis elastic and fully plastic *J*-integral values are first obtained for the geometry of concern and are then combined, using a simple formula, to obtain approximate values of the *J*-integral at all load levels up to the limit load. The method offers a simple technique for flaw evaluation, provided the fully plastic *J*-integral values are readily available. Abaqus contains a Ramberg-Osgood deformation plasticity theory model for this purpose. This example demonstrates the standard method provided in Abaqus to obtain such fully plastic results.

In many cases the user may prefer to evaluate the *J*-integral at each load level using incremental or deformation theory, thus providing a direct computation of the *J*-integral value at each load level. The "engineering fracture mechanics" approach used in this example is generally used when tabulations of values are required for standard geometries, loadings, and materials.

Problem description

The example uses the same double-edged notch specimen geometry used in *Contour integral evaluation: two-dimensional case* (where linear elastic *J*-integral evaluation is illustrated), except that the length of the specimen has been extended somewhat to ensure that the results are effective for an infinitely long plate. Plane stress and plane strain cases are both analyzed. Results for these cases are available in Kumar, et al. (1981), so that the example provides verification of the fully plastic *J*-integral results.

The geometry is shown in *Figure 1*. The specimen half-length has been extended to 2.54 m (100 in) to ensure that the far-field tension load is applied at a sufficient distance from the crack tip. The meshes for the 1/4 model are shown in *Figure 2*. Both a coarse mesh and a fine mesh are used. The fine mesh is similar to the coarse mesh, but with more elements. This mesh is used only in the plane strain case, because the incompressibility assumption in the material model makes that case more difficult. Shih and Needleman (1984) discuss this issue and point out that it is essential that the mesh should be able to model the fully plastic flow field accurately. For this reason mesh convergence studies are essential in such applications—see the discussion in the "Results" section below. Second-order elements are used. For the plane stress case the element type is CPS8R (the reduced integration, 8-node quadrilateral element). For the plane strain case element type CPE8RH is used; this element is a "hybrid" (mixed) formulation element and is used in this case because the material behavior is fully incompressible at the limit load and the mixed method can handle the incompressibility constraint. Acceptable results can also be obtained by using element type CPE8R, since the use of reduced integration avoids excessive constraint with incompressible response.

Material

The material model is the deformation theory, Ramberg-Osgood model provided in Abaqus for such applications. This plasticity model is nonlinear at all stress levels, although the initial response up to the reference stress and strain values is almost linear. Various hardening exponents are of practical interest, the most commonly needed values being from 3 to 10. For this reason several different values are studied in this example.

Loading

The load is far-field tension applied to the top edge of the model. This is accomplished by applying negative pressure to the edges of the elements along the top of the model.

Solution development

The deformation theory solutions are not path dependent (the deformation theory plasticity model used here is entirely equivalent to a nonlinear elasticity model), so any technique that will provide the fully plastic solution in a numerically efficient manner is satisfactory. The most effective approach in Abaqus for this purpose is usually the standard technique of incrementation and iteration, gradually increasing the load magnitude until the fully plastic solution is obtained. A general static analysis is done. Simultaneously, a region is monitored to become fully plastic, thus monitoring the progress of such a deformation theory solution. In this problem a set named `Monitor` is created that contains all of the elements in the focused part of the mesh and the first layer of elements above that region. In Abaqus/CAE such a region is created by partitioning. Abaqus will stop incrementing the load when all points in all elements in the specified set `Monitor` are in the fully plastic range (defined by the equivalent plastic strain being 10 times the offset yield strain), at which time the desired solution has been obtained.

Automatic incrementation is used, so the only control value that is needed is the suggested initial increment size. This can be estimated from knowledge of the limit load for the problem (available in Kumar, et al., 1981). The initial increment is suggested as 40% of the limit load value. This choice is not very critical in this case since the automatic incrementation algorithm will quickly find a suitable increment size, provided the suggestion is not grossly wrong.

Results and discussion

Kumar, et al. (1981) provide tables of values of the nondimensional parameter h_1, which defines the fully plastic J-integral for the geometry as

$$J = \alpha\sigma_0 e_0 c\left(\frac{P}{P_0}\right)^{n+1} h_1,$$

where σ_0, α, and n are the material parameters in the Ramberg-Osgood model; $e_0 = \sigma_0/E$; c is the half-width of the remaining ligament; b is the half-width of the specimen; P is the total load per unit thickness on the specimen; and P_0 is the limit load per unit thickness. For plane strain

$$P_0 = \left(0.72 + 1.82\frac{c}{b}\right)\sigma_0 b,$$

and for plane stress

$$P_0 = \frac{4}{\sqrt{3}} c\sigma_0.$$

Table 1 and *Table 2* summarize the values of h_1 obtained in this example (calculated from the J-integral values provided in the Abaqus output, using the equation above) and compare them to the values published by Kumar, et al. (1981). The plane stress case causes little difficulty, and the differences between J-integral values calculated on different contours are small, indicating that the results are fairly accurate. The agreement between these results and the values published by Kumar, et al. (1981) is quite good. In the plane strain case *Table 2* shows considerable scatter in the results obtained with the coarse mesh, indicating inaccuracy. The finer mesh results show only a small scatter between the different contours (six contours are available in this mesh, and *Table 2* shows the minimum and maximum values obtained). These finer mesh values are all close to the values obtained with the coarse mesh. These observations suggest that the finer mesh results are reliable. However, they do not agree closely with those tabulated by Kumar, et al. (1981). It has been established that some of the plane strain results

presented by Kumar, et al. (1981) are inaccurate; Shih and Needleman (1984) reanalyzed the single-edge cracked specimen for this reason. They point out the need for fine, carefully designed meshes to obtain accurate and reliable *J*-integral values, especially in such cases where incompressibility constrains the deformation. They also discuss consistency checks. One of these is the comparison of numerical values of the *J*-integral obtained from different contours around the crack tip. The *J*-integral should be path independent; therefore, any variation in *J*-integral values calculated on different contours implies inaccuracy. *Table 2* shows the range of *J*-integral values obtained in this example; as mentioned above, there is very little scatter in the values calculated with the fine mesh, so they satisfy this consistency check. The other consistency check discussed by Shih and Needleman (1984) requires the evaluation of *J*-integral values at different crack depths so that the slope of the *J*-versus-crack-depth variation can be calculated. In this example only one value of crack depth has been investigated, so this check cannot be applied. The discrepancy between the values reported here and those tabulated by Kumar, et al. (1981) must remain unexplained until further analysis, including the second consistency check, is done.

Submodeling of the crack tip

In *Contour integral evaluation: two-dimensional case* the submodeling capability is used to obtain more accurate near-tip stress fields in the linear elastic problem. In this example the submodeling capability is used to analyze the crack-tip region when the material is elastic-plastic. When small-scale yielding conditions exist, the far-field elastic region is not affected by the plastic zone around the crack tip. This will be true if the plastic zone size is less than about 10% of any characteristic length in the problem. The crack length serves as the characteristic length in this case. The loads in the problem are chosen so that the plastic zone is sufficiently small.

The problem is first solved with a relatively coarse mesh, as an elastic problem. The boundary of the submodel is chosen sufficiently far away from the crack tip so that the displacements on the boundary will not be affected by the plastic zone. The coarse mesh used is shown in *Figure 2* (left). Plane strain conditions are modeled with CPE8RH elements, and a focused mesh is used (see *jintegralplastic_global.inp*). The value of the far-field loading for the global problem is chosen so that the small-scale yielding conditions at the crack-tip field are met in the elastic-plastic material case. A region of 508 mm (20 in) by 254 mm (10 in) is used for the submodel. The driven boundary is sufficiently far from the crack tip so the stress field near this boundary is not influenced by the plastic zone. The submodel has six rings of CPE8RH elements around the crack tip. The elastic-perfectly plastic material properties can be found in the corresponding files for the submodel. *Figure 3* shows the geometry for the double-edged notch submodel and its deformed shape with a magnification factor of 169.

The *J*-integral values for the submodel should match the *J*-values for the global elastic mesh provided small-scale yielding conditions are met. Results are given in *Table 3* for an analysis in which the plastic zone is entirely contained within the first two rings of elements surrounding the crack tip. The corresponding Mises stress contours are shown in *Figure 4*.

Submodeling could equally be used with a Ramberg-Osgood deformation plasticity model.

Python scripts

Symmetric two-dimensional double-edged notch specimen coarsely meshed using reduced-integration plane stress elements

Run the *2DDoubleEdPlCoarseCPS8R_model.py* script to create the model. Then run the *2DDoubleEdPlCoarseCPS8R_job.py* script to analyze the model. In this model a hardening exponent of 5 is used. The other coarse mesh cases reported in the tables are available by changing the hardening exponent in the material definition.

Symmetric two-dimensional double-edged notch specimen coarsely meshed using hybrid reduced-integration plane strain elements

Run the *2DDoubleEdPlCoarseCPE8RH_model.py* script to create the model. Then run the *2DDoubleEdPlCoarseCPE8RH_job.py* script to analyze the model. In this model a hardening exponent of 5 is used. The other coarse mesh cases reported in the tables are available by changing the hardening exponent in the material definition.

Symmetric two-dimensional double-edged notch specimen finely meshed using hybrid reduced-integration plane strain elements

Run the *2DDoubleEdPlFineCPE8RH_model.py* script to create the model. Then run the *2DDoubleEdPlFineCPE8RH_job.py* script to analyze the model. In this model a hardening exponent of 5 is used. The other fine mesh cases reported in the tables are available by changing the hardening exponent in the material definition.

Submodel analysis of a symmetric two-dimensional double-edged notch specimen using an elastic-perfectly plastic material and meshed using hybrid reduced-integration plane strain elements

The analysis is done in two stages:

1. Run the *2DDoubleEdPlasGlCPE8RH_model.py* script to create the global model. Then run the *2DDoubleEdPlasGlCPE8RH_job.py* script to analyze the model and to create the output database (`.odb`) file that will drive the submodel.
2. Run the *2DDoubleEdPlasSubCPE8RH_model.py* script to create the submodel. Then run the *2DDoubleEdPlasSubCPE8RH_job.py* script to analyze the submodel using the output database file from the global model to drive it.

Input files

The input files listed below are provided for users who prefer to use the Abaqus keyword interface instead of Abaqus/CAE. The meshes created in these input files are different from those created by using the Python scripts; however, the results are of similar accuracy.

jintegralplastic_cps8r_coarse.inp	Typical input data for one case (plane stress, $n = 5$). The other coarse mesh cases reported in the tables are available by changing the element type for plane strain and/or by changing the exponent n in the material definition.
jintegralplastic_cpe8rh_fine.inp	Typical input data for a finer mesh study in plane strain; the other cases are available by changing the value of n.
jintegralplastic_submodel.inp	Submodel data for an elastic-perfectly plastic material. The file jintegralplastic_global.inp contains the global model used.
jintegralplastic_submodel_sb.inp	Submodel data for an elastic-perfectly plastic material, using the stress-based submodeling technique. The file jintegralplastic_global.inp contains the global model used.
jintegralplastic_global.inp	Coarse mesh in plane strain, global model used for submodeling.

References

- Kumar, V., M. D. German, and C. F. Shih, "An Engineering Approach for Elastic-Plastic Fracture Analysis," *Report NP–1931, Electric Power Research Institute, Palo Alto, California*, 1981.
- Shih, C. F., and A. Needleman, "Fully Plastic Crack Problems," *Parts I and II, ASME Journal of Applied Mechanics*, vol. 51, pp. 48–64, 1984.

Tables

Table 1: Fully plastic results for double-edged cracked plate in plane stress. h_1 values for double-edged cracked plate in tension; c/b (crack depth/half ligament) = 0.5.

Hardening exponent	h_1	
(n)	Abaqus	Kumar, et al. (1981)
3	1.37–1.38	1.38
5	1.17–1.18	1.17
7	1.01	1.01
9	0.90	not given
10	0.85	0.845

Table 2: Fully plastic results for double-edged cracked plate in plane strain. h_1 values for double-edged cracked plate in tension; c/b (crack depth/half ligament) = 0.5.

Hardening exponent (n)	h_1		
	Abaqus		Kumar, et al. (1981)
	Coarse mesh	Finer mesh	
3	2.55–2.59	2.55–2.58	2.48
5	2.59–2.62	2.58–2.59	2.43
7	2.55–2.58	2.55–2.56	2.32
10	2.39–2.43	2.46–2.47	2.12

Table 3: J-integral comparison for global and submodel analyses.

Contour	Global elastic analysis	Submodel elastic-plastic analysis
1	294.783	281.524
2	293.176	284.562
3	293.177	288.742
4	292.966	288.782
5	–	288.797
6	–	288.790

Figures

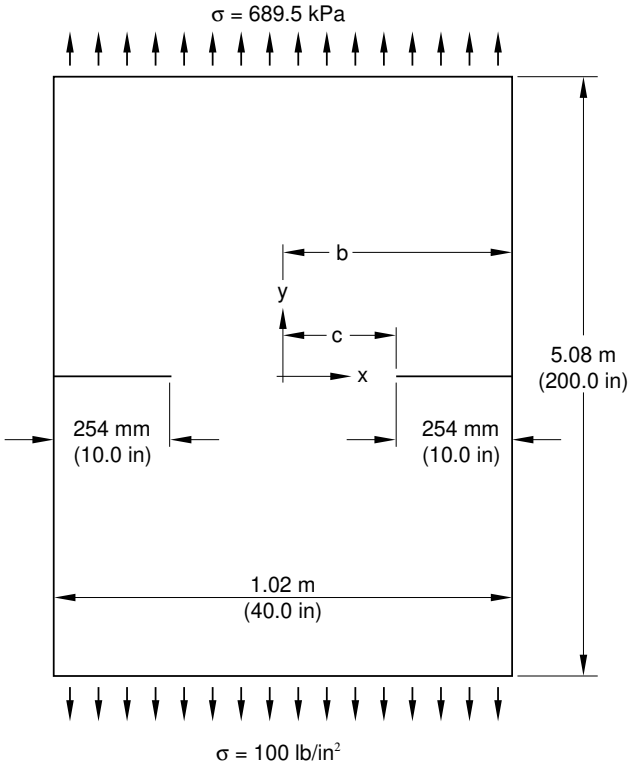


Figure 1: Geometry for double-edged notch specimen.

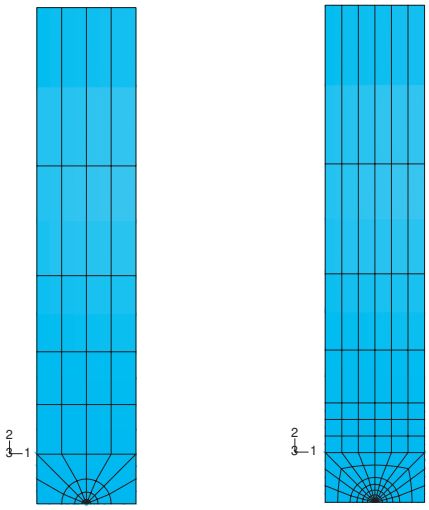


Figure 2: Coarse (left) and finer (right) meshes for double-edged notch specimen.

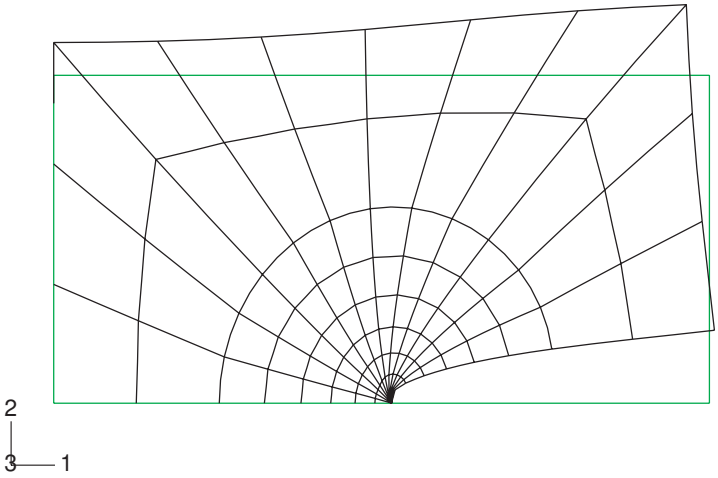


Figure 3: Geometry for double-edged notch submodel and its deformed shape shown with magnification factor of 169.

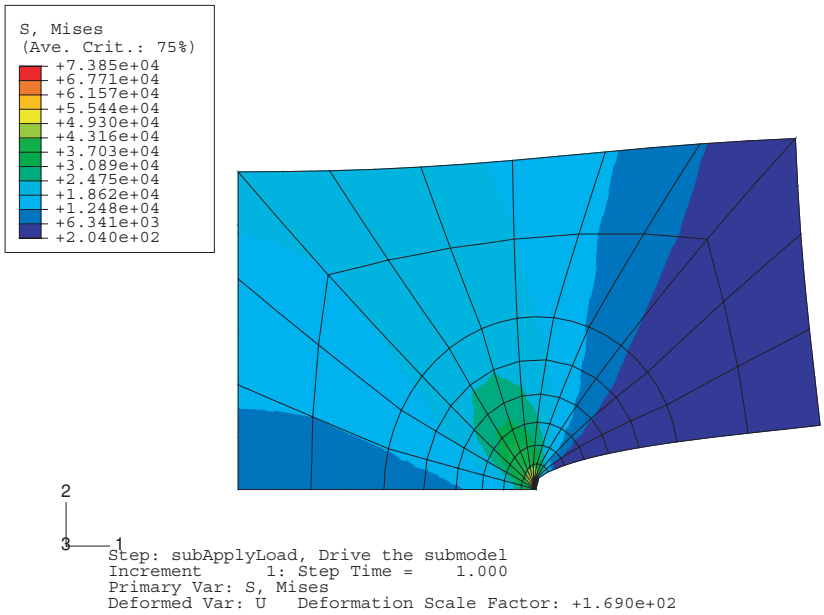


Figure 4: Contour of the Mises stress around the crack tip of the elastic-plastic submodel.

C_t-integral evaluation

Products: Abaqus/Standard

Overview

This example illustrates the evaluation of the C_t-integral as a function of time for a stationary crack under secondary power law creep conditions.

Because of the time-dependent effects of creep deformation, there is no one parameter that characterizes the stress state around the crack tip for all circumstances. The appropriate parameter to use depends on the details of the constitutive law (whether the law describes primary, secondary, or tertiary creep) and on the stage of deformation of the material around the crack tip. In addition, creep deformation can occur in either an initially elastic or an initially plastic stress field. Riedel (1981) discusses which parameters are correct for different circumstances. When the initial response of the material is linear elastic and secondary creep dominates the creep behavior, the stress intensity factor, K_I, and the path independent integral, C^*, are the relative loading parameters. For small-scale creep (that is, when elastic strains dominate almost everywhere in the specimen except in a small zone that grows around the crack tip), K_I governs crack growth initiation. If, however, the creep zone becomes large compared to the specimen size and the elastic strains small compared to the creep strains, C^* is the appropriate fracture parameter.

The C_t fracture mechanics parameter offered by Abaqus characterizes crack growth behavior for a wide range of creep conditions. For stationary cracks C_t characterizes the rate of growth of the crack-tip creep zone under small-scale creep conditions and is also related to the stress intensity factor K_I. Under extensive secondary creep conditions, $C_t \rightarrow C^*$ and is path independent throughout the extensive creep region.

Problem description

A shallow edge-crack in a half space in plane strain, subjected to constant Mode I far-field tensile loading, is considered. The geometry is shown in *Figure 1*. Initially the edge crack is stress free. At time $t = 0^+$ a tensile stress, σ_N, is applied suddenly on the circular boundary and held constant thereafter. The instantaneous response is purely elastic since creep deformation develops over a period of time. The crack-tip stress field is, thus, initially characterized by linear elastic fracture mechanics. With the load held constant, subsequent creep deformation causes a relaxation of the crack-tip stresses until, as $t \rightarrow \infty$, a steady-state stress distribution is reached.

The results for this problem are documented by Bassani and McClintock (1981).

The crack length, a, is 10 mm. As a result of symmetry, only one-half of the body needs to be analyzed. The mesh is comprised of CPE8R elements. Eleven rings of elements are focused radially at the crack tip, with 12 element divisions in the 180° modeled circumferentially. This focused portion of the mesh is connected to the outer portion shown in *Figure 2*.

The innermost ring of elements at the crack tip are degenerated into triangles. The three nodes along one side of the 8-node element are defined such that they share the same location; the other side nodes remain at the midpoints. Each of the three collapsed nodes can displace independently, so the interpolation function exhibits a r^{-1} singularity in displacement derivatives.

The reduced-integration (CPE8R) element is chosen since it does not “lock” during the incompressible creep deformation. The incompressibility constraint can also be modeled successfully with hybrid (mixed) formulation elements. The results obtained with both elements are in close agreement. A file with input data using hybrid elements is provided with the Abaqus release.

No mesh convergence studies have been performed.

Material

The material behavior includes linear elasticity and secondary creep response. The material is assumed to be isotropic elastic, with a Young's modulus of $E =$ 200 GPa and a Poisson's ratio of 0.3, and with uniaxial creep behavior defined by

$$\dot{\varepsilon}^{cr} = A\sigma^n,$$

where A is 5.0×10^{-12} per hour (stress in MPa) and $n = 3$.

Loading

The load is a constant far-field tension of $\sigma_N = E/2000$ applied to the circular boundary edge of the model by applying concentrated loads (equivalent to a pressure of 100 MPa) to the nodes on the circumference. The load is applied instantaneously and then held constant until steady-state creep conditions are reached. The initial application of the load is assumed to occur so quickly that it involves purely elastic response. This behavior is obtained by using the static procedure. The creep response is then developed in a second step, using the quasi-static procedure.

During the quasi-static step a user-specified tolerance is required to control the time increment choice and, hence, the accuracy of the transient creep solution. A maximum elastic principal stress of 2800 MPa occurs at the crack tip; therefore, errors in stress of about 20 MPa will make a small difference to the creep strain added within an increment. Converting this stress error to a strain error by dividing it by the elastic modulus gives a value for the accuracy tolerance of 1×10^{-4}. If only an estimate of C^* is required, a high value for the tolerance can be used. This allows Abaqus to use the largest possible time increments that result in a C_t value of low accuracy during the transient but reach the steady-state C^* value at minimum cost. 1000 hours of response are requested, which is sufficient to reach steady-state conditions.

Results and discussion

Figure 3 shows the displaced shape of the mesh near the crack tip at steady state.

The C_t-integral is only path independent in the limiting case when steady-state conditions are reached. The path dependence of C_t during transient creep is shown in *Figure 4*. The figure shows the variation of C_t with the radius of the contour, r, measured at different times during the early part of the creep history (before the transition time, t_T, defined at the end of this discussion, is reached). We define the radius for the nth contour, r, as the distance from the crack tip to the outer ring of nodes of the nth ring of elements surrounding the crack tip. Path independence is reached as time increases. The difference between the second and fifth contour values is less than 4% at t_T. This difference decreases further until the values become path independent at steady state.

Figure 5 compares the C_t values predicted by Abaqus (Line 4) with the steady-state C^* value, as well as with two approximate models available in literature. Since C_t is defined only on a contour of infinitesimal dimension around the crack tip at early times (before steady-state conditions apply), it is necessary to estimate its value by extrapolating the values provided by the Abaqus contour integral to a contour with zero radius. As explained in *Contour Integral Evaluation*, each contour integral evaluation in Abaqus is made by applying a uniform virtual perturbation to the nodes within a ring of elements surrounding the crack tip. Therefore, in a plane strain case such as this, the only contribution from the nth contour integral comes from the nth ring of elements, counting outward from the crack tip. For the purpose of this extrapolation we use the same definition of radius for the nth contour integral as described in the previous paragraph. We ignore the first contour in the extrapolation, since

experience has shown that the first contour is of significantly lower accuracy than the other contours. The C_t values shown in *Figure 5* are then based on a least-squares fit of a second-order polynomial

$$C_t = a_0 + a_1 r + a_2 r^2$$

to the four remaining contour integral values provided by Abaqus at each time point and recording the value provided by this curve fit at $r = 0$. We have not investigated the accuracy of this extrapolation technique.

The C^* value shown in *Figure 5* is also obtained by using another technique in Abaqus. By interpreting strains and displacements as their rate counterparts and J as C^*, the fully plastic solutions obtained from a power law hardening material can be applied directly to find C^* values for an equivalent power law creep model. In other words, power law creep is analogous to the fully plastic limit of power law hardening plasticity, so

$$C^* = \dot{\varepsilon}_0 \sigma_0 h_1 \left(\frac{P}{P_0}\right)^{n+1},$$

where σ_0 is a reference stress; $\dot{\varepsilon}_0$ is the creep strain rate at the reference stress; h_1 is a dimensionless function of the power law exponent, n, and of geometric parameters; P is the loading; and P_0 is the limit load. For this particular example, however, the applied far-field tension, σ_N, is insufficient to cause a fully plastic zone to develop; therefore, the J value cannot be directly interpreted as C^*. To obtain the C^* value from an equivalent Ramberg-Osgood material model in such a case, the following procedure is followed: The structure is loaded until a fully plastic zone exists in a zone surrounding the crack tip. For this purpose a fully plastic static analysis is used to monitor the progress of the solution in the element set containing the 11 rings of radially focused elements (*Figure 3*). A fully plastic solution is obtained at a load of 66.3 σ_N. The J value obtained at this load is then used to evaluate the calibration function h_1, which, in turn, is used to obtain C^* at a load of σ_N. This value is shown as Line 1 in *Figure 5*.

Figure 5 also shows the Riedel and Rice (1980) approximation. They proposed the following relation between C_t and the stress intensity factor K_I for small-scale creep:

$$C_t = \frac{1}{(n+1)t}\left(1-\nu^2\right)\frac{K_I^2}{E},$$

where n is the power law constant and E and ν are elastic constants. This approximation is shown as Line 2 in *Figure 5*.

The remaining curve in *Figure 5*, Line 3, represents the interpolation between short and long time behavior proposed by Riedel (1981):

$$C_t = C^*\left(\frac{t_T}{t}+1\right),$$

where the transition time from small-scale creep to extensive creep is given by

$$t_T = \frac{1}{(n+1)}\frac{1}{C^*}\frac{(1-\nu^2)}{E}K_I^2 = 22.5 \text{ hours.}$$

Input files

ctintegral_cpe8r.inp	CPE8R elements.
ctintegral_cstar.inp	Used to obtain the C^* value. The model uses the Ramberg-Osgood deformation plasticity law.
ctintegral_cpe8rh.inp	CPE8RH elements.
ctintegral_cpe8r_systemc.inp	Same as ctintegral_cpe8r.inp except that the **NGEN* option with the SYSTEM=C parameter is included.

References

- Bassani, J. L., and F. A. McClintock, "Creep Relaxation of Stress Around a Crack Tip," *International Journal of Solids and Structures*, vol. 17, pp. 479–492, 1981.
- Riedel, H., "Creep Deformation at Crack Tips in Elastic-Viscoplastic Solids," *Journal of Mechanics and Physics of Solids*, vol. 29, pp. 35–50, 1981.
- Riedel, H., and J. R. Rice, "Tensile Cracks in Creeping Solids," *Fracture Mechanics: Twelfth conference, ASTM STP 700, American Society for Testing of Materials*, pp. 112–130, 1980.

Figures

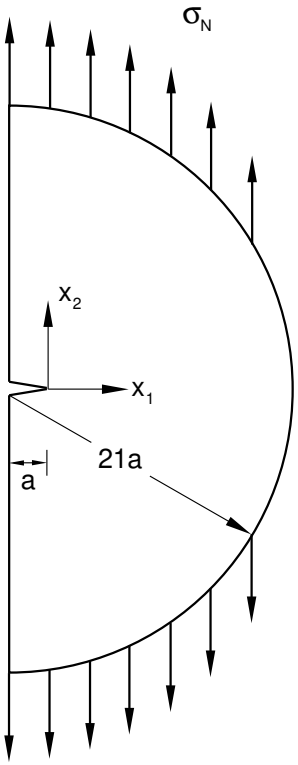


Figure 1: Geometry for edge-crack specimen under plane strain and constant nominal stress conditions.

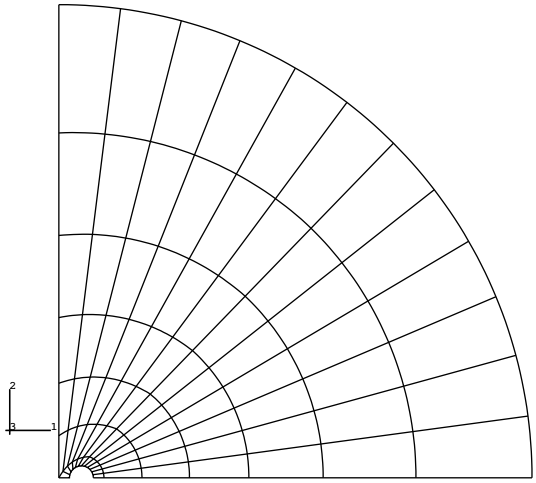


Figure 2: Mesh for portion of model outside the 11 rings of radially focused elements.

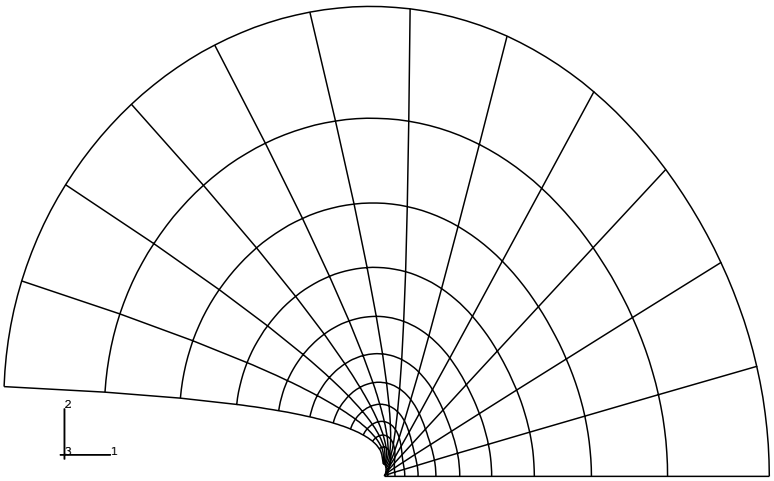


Figure 3: Displaced shape showing the 11 rings of focused elements near the crack tip.

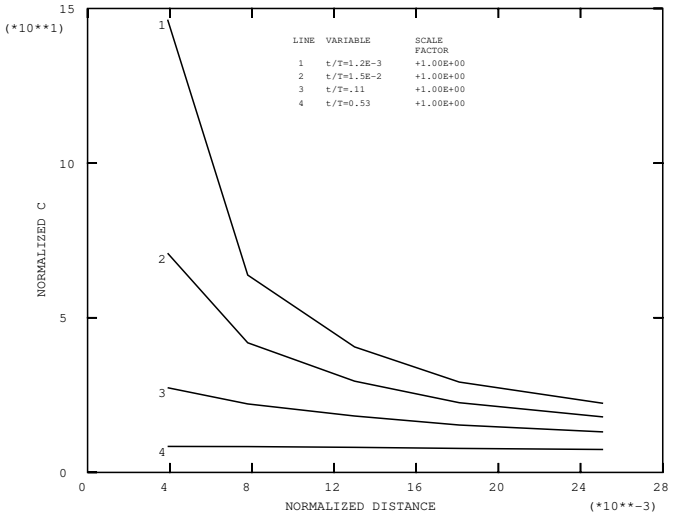


Figure 4: Normalized C_t values ($C_t/\sigma_N\dot{\varepsilon}_N a$) versus normalized distance from crack face (r/a).

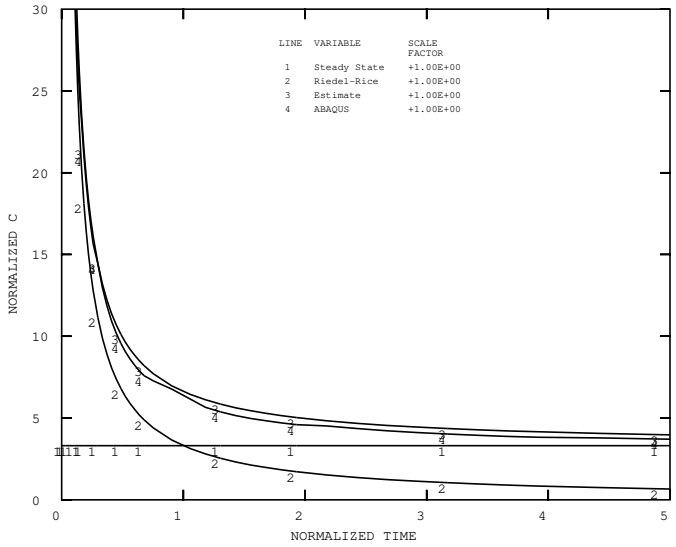


Figure 5: Normalized C_t values ($C_t/\sigma_N\dot{\varepsilon}_N a$) versus normalized time (t/t_T).

Nonuniform crack-face loading and *J*-integrals

Products: Abaqus/Standard

Overview

This example verifies nonuniform crack-face loading in two-dimensional and three-dimensional analyses.

Problem description

For the two-dimensional case an edge crack of length 1 m is modeled in a linear elastic specimen. The results are effectively for an infinitely long plate. The geometry is symmetric about the crack line, so only the top half is modeled. The geometry is meshed using CPE8R elements. The crack faces are loaded in five steps. In the first step a load of constant magnitude 1 MPa is applied. In all subsequent steps the load is zero at the surface of the specimen and has magnitude 1 MPa at the crack tip. The load varies linearly in Step 2, quadratically in Step 3, cubically in Step 4, and quartically in Step 5.

For the three-dimensional case the model from *3DDoubleEdgedNotchC3D20_model.py* in *Contour integral evaluation: two-dimensional case*, is modified to apply a uniform crack-face loading via user subroutine *DLOAD*.

Results and discussion

Results for the two-dimensional and three-dimensional analyses are discussed in the following sections.

Two-dimensional results

Abaqus results are compared with the results taken from page 8.8 of *The Stress Analysis of Cracks Handbook* by H. Tada, P. C. Paris, and G. R. Irwin. The crack-face loading is given by $p = x^{\gamma}$ MPa. Results for the *J*-integral in Pa are presented in *Table 1*.

Three-dimensional results

The results should be the same as those shown in *Contour integral evaluation: two-dimensional case*. The results are within a difference of 0.1%.

Python scripts

Nonuniform crack face loading on a two-dimensional model

Run the *2DEdgeCrackCPE8R_model.py* script to create the model. Then run the *2DEdgeCrackCPE8R_job.py* script to analyze the model. User subroutine *DLOAD* (*2DEdgeCrackCPE8R.for*) is used to apply the nonuniform loading on the crack face.

Uniform crack face loading on a three-dimensional model via user subroutine *DLOAD*

Run the *3DCrackC3D20_model.py* script to create the model. Then run the *3DCrackC3D20_job.py* script to analyze the model. User subroutine *DLOAD* (*3DCrackC3D20.for*) is used to apply the uniform loading on the crack face.

Input files

The input files listed below are provided for users who prefer to use the Abaqus keyword interface instead of Abaqus/CAE. The meshes created in these input files are different from those created by using the Python scripts; however, the results are of the same accuracy.

pjinnu2d.inp	Checks the nonuniform loads applied to plane strain elements via user subroutine *DLOAD*.
pjinnu2d.f	User subroutine *DLOAD* used in pjinnu2d.inp.
pjinnu3d.inp	Uses subroutine *DLOAD* and, therefore, "nonuniform" load types, to apply a uniform load to the crack faces.
pjinnu3d.f	User subroutine *DLOAD* used in pjinnu3d.inp.

Tables

Table 1: *J*-integral results in Pa.

γ	0	1	2	3	4
Tada et al.	17.98	6.67	3.94	2.78	2.27
Abaqus	18.57	6.81	4.01	2.81	2.16

Figures

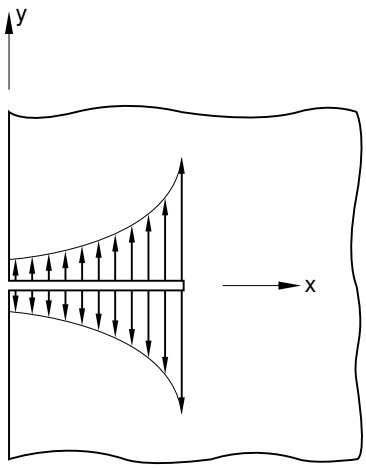


Figure 1: Crack model.

Single-edged notched specimen under a thermal load

Products: Abaqus/Standard

Overview

This example illustrates analysis procedures for evaluating the *J*-integral in flawed structures subjected to severe thermal events, such as thermal shocks.

Thermal loading is of major concern in many components, especially in process and power plants and high-performance engines. This example shows how Abaqus can be used to generate temperature solutions that can then be used for *J*-integral analysis.

Problem description

The problem is the case for which Shih et al. (1986) provide results. The example is a single-edged notched specimen, restrained against axial motion at its ends and subjected to a linear temperature variation across its width. The particular geometry studied is cracked halfway through its thickness; the total length of the specimen ($2L$) is 2.032 m (80 in), and the total width is 508 mm (20 in). The geometry and thermal loading are shown in *Figure 1*.

The material is isotropic and linear elastic, with a Young's modulus 207 GPa (30×10^6 lb/in^2), a Poisson's ratio 0.3, and a thermal expansion coefficient 1.35×10^{-5} per °C (7.5×10^{-6} per °F).

The temperature gradient is given about a mean temperature change of zero. If the mean temperature change were nonzero, the fixed axial end restraints and, in the plane strain case, the thickness direction constraint would induce severe thermal straining.

The two focused meshes that are used are shown in *Figure 2*. Although using elements with curved edges is not recommended in *J*-integral calculations, the error for this fine a mesh is negligible. The coarser mesh does not contain any curved elements. Due to symmetry, only the top half of the specimen is modeled. In the coarse mesh model there are six rings of elements around the crack tip; thus, *J*-integral values on at most six contours can be obtained. In the fine mesh model with nine rings of elements around the crack tip, *J*-integral output can be obtained for nine contours.

The relative path independence of the *J*-integral values does not prove that the mesh is adequate. In a linear problem approximate path independence of *J*-integral values can be achieved before the requirements of mesh convergence are met. Mesh convergence is best checked by reanalysis with a finer mesh.

The temperature distribution is obtained by solving a steady-state heat transfer problem, using a mesh of equivalent heat transfer elements, and prescribing the surface temperatures on the two vertical edges of the specimen. Abaqus makes it straightforward to read such temperature solutions into stress analysis models: this technique of thermal-stress analysis is illustrated in *Quenching of an infinite plate*. The procedure shown in that case could be used to analyze this specimen for the case of a thermal shock, if required.

The heat transfer and thermal-stress analysis can also be done simultaneously with the thermally coupled analysis procedure. This is convenient in the sense that only a single run needs to be made but has the disadvantage that the analysis cost of a thermally coupled analysis can be considerably higher than the cost of sequentially running a heat transfer and thermal-stress analysis, particularly for more complex problems.

Results and discussion

Plane stress and plane strain solutions are obtained using both meshes. The plane stress solutions are calculated with element type CPS8R. The results for mesh 1 are duplicated using element type S8R5 to verify the thermal

J-integral capability for shells. Results are also obtained for a single layer of three-dimensional elements (type C3D20R) using mesh 2.

Table 1 shows the values predicted for the *J*-integral, normalized by $\sigma_\theta^2 a/E$, where

$$\sigma_\theta = \widehat{E}\alpha\theta_e.$$

Here $\widehat{E}$ is $E/\left(1-\nu^2\right)$ for the plane stress case, where E is Young's modulus and ν is Poisson's ratio, or is $E/\left(1-\nu\right)$ for plane strain; α is the coefficient of thermal expansion; and θ_e is the temperature at the edge of the specimen. In both cases the results from the two meshes used are quite similar, although the finer model shows a higher degree of path independence for the values of the *J*-integral. The results of Shih et al. are given in the last two columns of the table. There is reasonable agreement between all of the solutions.

In addition to the *J*-integral, the stress intensity factors and the *T*-stress can be determined.

Python scripts

Heat transfer analysis using mesh 1

Run the *SingleEdgedThermMesh1_model.py* script to create the model. Then run the *SingleEdgedThermMesh1_job.py* script to analyze the model.

Stress analysis with reduced-integration plane strain elements using mesh 1

Run the *SingleEdThermMesh1CPE8R_model.py* script to create the model. Then run the *SingleEdThermMesh1CPE8R_job.py* script to analyze the model. This script uses the `.odb` file from the analysis of SingleEdgedThermMesh1_model.py as input for the temperature field.

Stress analysis with reduced-integration plane stress elements using mesh 1

Run the *SingleEdThermMesh1CPS8R_model.py* script to create the model. Then run the *SingleEdThermMesh1CPS8R_job.py* script to analyze the model. This script uses the `.odb` file from the analysis of SingleEdgedThermMesh1_model.py as input for the temperature field.

Stress analysis with reduced-integration shell elements using mesh 1

Run the *SingleEdThermMesh1S8R5_model.py* script to create the model. Then run the *SingleEdThermMesh1S8R5_job.py* script to analyze the model. This script uses the `.odb` file from the analysis of SingleEdgedThermMesh1_model.py as input for the temperature field.

Stress analysis with reduced-integration plane stress elements using mesh 1 and a temperature-dependent expansion coefficient

Run the *SingleEdTempDMesh1CPS8R_model.py* script to create the model. Then run the *SingleEdTempDMesh1CPS8R_job.py* script to analyze the model. This script uses the `.odb` file from the analysis of SingleEdgedThermMesh1_model.py as input for the temperature field.

Heat transfer analysis using mesh 2

Run the *SingleEdgedThermMesh2_model.py* script to create the model. Then run the *SingleEdgedThermMesh2_job.py* script to analyze the model.

Stress analysis with reduced-integration plane strain elements using mesh 2

Run the *SingleEdThermMesh2CPE8R_model.py* script to create the model. Then run the *SingleEdThermMesh2CPE8R_job.py* script to analyze the model. This script uses the `.odb` file from the analysis of SingleEdgedThermMesh2_model.py as input for the temperature field.

Coupled temperature-displacement analysis with reduced-integration plane strain elements using mesh 2

Run the *SingleEdCoupledTDCPE8R_model.py* script to create the model. Then run the *SingleEdCoupledTDCPE8R_job.py* script to analyze the model. This script does not require the results of a heat transfer analysis.

Three-dimensional version of the heat transfer analysis using mesh 2

Run the *3DSingleEdgedThermMesh2_model.py* script to create the model. Then run the *3DSingleEdgedThermMesh2_job.py* script to analyze the model.

Three-dimensional version of the stress analysis with reduced-integration continuum elements using mesh 2

Run the *SingleEdThMesh2C3D20R_model.py* script to create the model. Then run the *SingleEdThMesh2C3D20R_job.py* script to analyze the model. This script uses the `.odb` file from the analysis of *3DSingleEdgedThermMesh2_model.py* as input for the temperature field.

Input files

The input files listed below are provided for users who prefer to use the Abaqus keywords interface instead of Abaqus/CAE. The meshes created in these input files are different from those created by using the Python scripts; however, the results are of the same accuracy.

jintegraltherm_heatmesh1.inp	Heat transfer analysis using mesh 1.
jintegraltherm_cpe8rmesh1.inp	Stress analysis with plane strain elements using mesh 1. This file uses the results file from jintegraltherm_heatmesh1.inp as input for the temperature field.
jintegraltherm_cps8rmesh1.inp	Stress analysis with plane stress elements. This file uses the results file from jintegraltherm_heatmesh1.inp as input for the temperature field.
jintegraltherm_s8r5mesh1.inp	Stress analysis with shell elements. This file uses the results file from jintegraltherm_heatmesh1.inp as input for the temperature field.
jintegraltherm_cps8rmesh1_exp.inp	Plane stress model including temperature dependence of the expansion coefficient. This job uses the results file from jintegraltherm_heatmesh1.inp as input for the temperature field.
jintegraltherm_heatmesh2.inp	Heat transfer analysis using mesh 2.
jintegraltherm_cpe8r.inp	Stress analysis with plane strain elements using mesh 2. This job requires the results file from jintegraltherm_heatmesh2.inp as input for the temperature field. A plane stress model for mesh 2 is obtained by changing the element type in this data file.
jintegraltherm_cpe8rt.inp	Analysis with coupled temperature-displacement plane strain elements using mesh 2. This job does not require the results of a heat transfer analysis.
jintegraltherm_3dheat.inp	Three-dimensional version of the heat transfer analysis using mesh 2.

jintegraltherm_c3d20r.inp	Three-dimensional version of the stress analysis using mesh 2. This file uses the results file from jintegraltherm_3dheat.inp as input for the temperature field.

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: Comparison of results for $J/\left(\sigma_\theta^2 a/E\right)$ for models with reduced-integration plane strain elements.

Contour	Mesh 1	Mesh 2	Shih et al.	Shih et al.*
1	0.723	0.726	0.6366	0.6408
2	0.702	0.702	0.6618	0.6900
3	0.706	0.706	0.6712	0.6923
4	0.706	0.707	0.6759	0.6954
5	0.706	0.707	0.6800	0.7004
6	0.706	0.707	0.6827	0.7033
7	0.706		0.6848	0.7113
8	0.706		0.6869	0.7253
9	0.706		0.6899	0.7503
Average	0.708	0.709	0.6744	0.7010
*Values obtained by analyzing equivalent problem with crack face traction.				

Figures

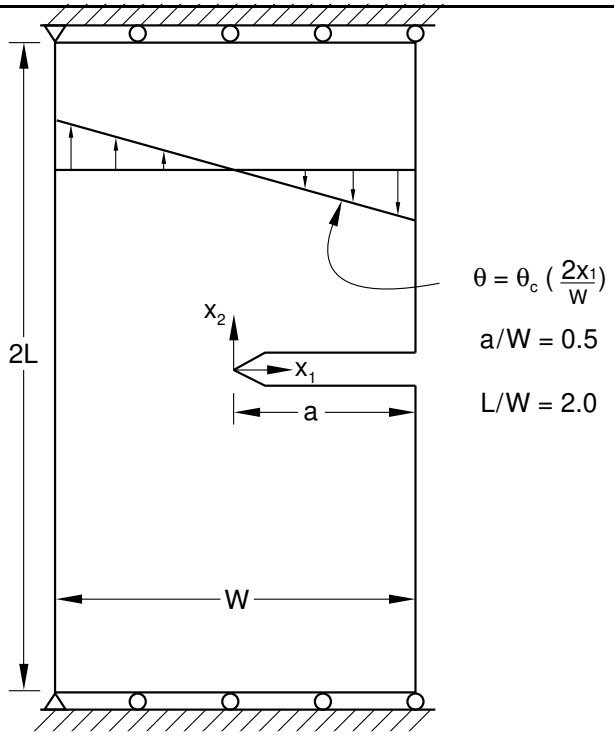


Figure 1: Geometry of cracked specimen.

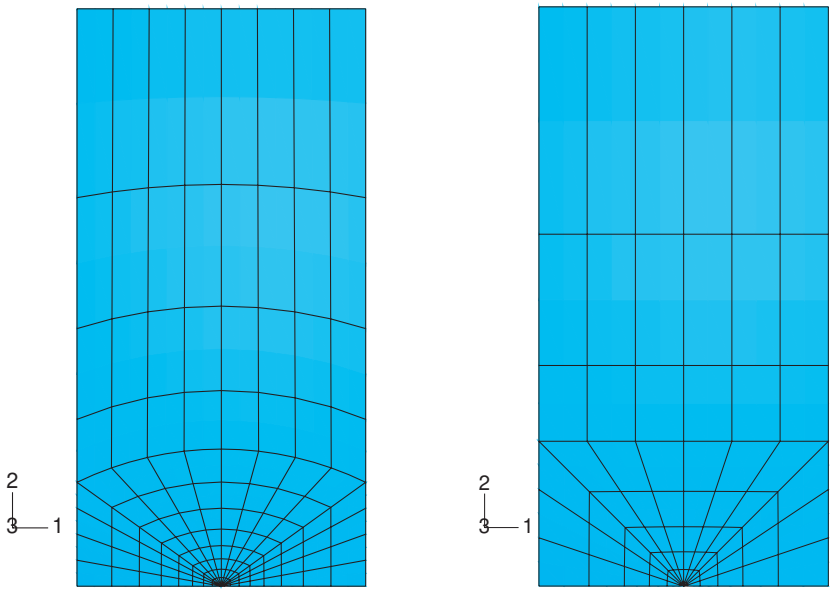


Figure 2: Mesh 1 (left) and mesh 2 (right).

Design sensitivity analysis

In this section:

- *Design sensitivity analysis for cantilever beam*
- *Sensitivity of the stress concentration factor around a circular hole in a plate under uniaxial tension*
- *Sensitivity analysis of modified NAFEMS problem 3DNLG-1: Large deflection of Z-shaped cantilever under an end load*

Design sensitivity analysis for cantilever beam

Products: Abaqus/Standard Abaqus/Design

Overview

This example demonstrates the effectiveness of the default perturbation sizing algorithm used by Abaqus/Design in obtaining accurate tip displacement sensitivities.
In addition, a sensitivity analysis is carried out to obtain the sensitivities of natural frequencies.

Design sensitivity analysis in Abaqus is performed using the semianalytical method. The issue of obtaining accurate sensitivities with respect to design shape parameters using this method has been discussed extensively in the literature (for example, Pedersen et al., 1989; Barthelemy and Haftka, 1990; Fenyes and Lust, 1991; and Van Keulen and De Boer, 1998). The difficulty is that the accuracy of the sensitivities can depend on the number of elements. This dependency is not seen with either analytical sensitivity analysis or with the overall finite difference method. A canonical example is a cantilever beam with an applied tip load, where the sensitivity of the tip displacement to the length of the beam is sought.

Problem description

A cantilever beam 100 units long and 2 units deep is modeled using CPS8, S4R, and B31 elements. A resultant tip load of 1 unit is applied to the free end to simulate a shear load. The material is linear elastic, and a linear static analysis is carried out with Abaqus/Standard. Four different mesh densities are used to study the effect of mesh refinement. The mesh densities chosen ensure that the error in the tip deflection compared to the Euler-Bernoulli solution is less than 0.05%. The coarsest mesh for each element type contains 50 elements along the length. The mesh density is doubled for each level of refinement, so the most refined mesh for each element type has 400 elements along the length. For the CPS8 element meshes, the coarsest mesh (50 elements along the length) contains one element through the depth and the most refined mesh (400 elements along the length) contains 8 elements through the depth. Only the axial coordinates of the nodes at the tip of the beam are assumed to depend on the length, so the gradients specified for the parameter shape variation are unity (that is, only a boundary perturbation is applied).

Subsequent to the static analysis, a frequency analysis is performed including design sensitivity analysis. The sensitivities of the first three eigenvalues and corresponding natural frequencies with respect to the beam length are computed.

Results and discussion

Based on Euler-Bernoulli beam theory the sensitivity of the tip displacement, v, with respect to the length, L, of the beam for a cantilever with end load P is given by

$$\left(\frac{dv}{dL}\right)_{ex} = \frac{PL^2}{EI},$$

where E is the modulus of elasticity and I is the moment of inertia.

The semianalytical method is based on perturbing the design parameter, L, and using a differencing technique to approximate the sensitivities. For this problem the size of the perturbation in L yielding the most accurate tip displacement sensitivity dv/dL can be determined by computing the error

$$e = \frac{\left(\frac{dv}{dL} - \left(\frac{dv}{dL}\right)_{ex}\right)}{\left(\frac{dv}{dL}\right)_{ex}}$$

over a range of perturbation sizes. The results for such an exercise using forward differencing and central differencing are plotted in *Figure 1* for the coarsest meshes. The behavior of these results is typical: to the left of the "optimum" perturbation size the error increases due to round off, or cancellation error; to the right of the optimum perturbation size the error again increases but this time due to the truncation of higher-order terms in the differencing formulae, known as truncation error.

To demonstrate the growth in truncation error with the number of elements, perturbation sizes are chosen from the truncation region of the plot shown in *Figure 1* such that the errors in the computed sensitivities for the most coarse meshes are acceptable (0.1%). This leads to perturbation sizes of 5×10^{-4}, 1×10^{-3}, and 1×10^{-3} for central differencing and 2×10^{-7}, 6×10^{-7}, and 9×10^{-7} for forward differencing for the CPS8, S4R, and B31 elements, respectively. These same perturbation sizes are then used in sensitivity analyses for the refined meshes. *Figure 2* shows the growth in the percentage error in the computed sensitivities as the number of elements along the length is increased. It is clear from these results that the perturbation sizes yielding accurate results for a coarse mesh may yield poor results for a more refined mesh because the truncation error grows with mesh refinement. The truncation error can be controlled by proper choice of the perturbation size. Indeed, if one chooses a perturbation size of 1×10^{-9}, it can be seen from *Figure 3* that the growth in the error for both central differencing and forward differencing is insignificant for all element types.

The default perturbation sizing algorithm in Abaqus/Design determines the perturbation sizes for each element, which are then used in a central difference scheme to compute the sensitivities. *Table 1* shows the perturbation sizes chosen for the element that has the dominant influence on the tip displacement sensitivity at various levels of mesh refinement and the percentage error in the Abaqus/Design sensitivity solution. For each of the coarsest meshes the perturbation size chosen by Abaqus/Design is in good agreement with the optimum for central differencing based on *Figure 1*.

It can be shown that the sensitivities of the eigenvalues (λ) and natural frequencies (f, in cycles/time) for a cantilever beam are given analytically by $d\lambda/dL = -4\lambda/L$ and $df/dL = -2f/L$, respectively. The frequency analysis is performed for the coarsest mesh, and the default perturbation sizing algorithm is used for the sensitivity analysis. By default, the algorithm determines the perturbation size based on the first mode, and the same perturbation size is then reused for the remaining modes. To force Abaqus/Design to obtain a new perturbation size for each mode, the DSA solution controls can be used in the frequency step with the sizing algorithm executed at every increment. The sensitivities of the eigenvalues and eigenfrequencies for the first three bending modes obtained with Abaqus/Design are compared to those obtained analytically and using the overall central differencing method in *Table 2*, *Table 3*, and *Table 4* for all element types. An optimum perturbation size of $1.0 \times 10^{-4}L$ was obtained by trial and error for the overall differencing method. An excellent agreement between Abaqus and overall central differencing method is seen. The advantages of using the design sensitivity analysis capability in Abaqus/Design over the overall finite difference method are: (1) automatic detemination of the perturbation size, and (2) reduced computational effort since the sensitivities are computed in the same analysis as the natural frequency extraction. The effort expended on recalculating the perturbation size for each mode by selecting a sizing frequency of 1 produces virtually no difference in the sensitivities, which are quite accurate with the default setting.

Input files

dsacantcps850.inp	Cantilever beam modeled with 50 CPS8 elements, including frequency step.
dsacantcps8100.inp	Cantilever beam modeled with 100 CPS8 elements.
dsacantcps8200.inp	Cantilever beam modeled with 200 CPS8 elements.
dsacantcps8400.inp	Cantilever beam modeled with 400 CPS8 elements.

dsacants4r850.inp	Cantilever beam modeled with 50 S4R elements, including frequency step.
dsacants4r8100.inp	Cantilever beam modeled with 100 S4R elements.
dsacants4r8200.inp	Cantilever beam modeled with 200 S4R elements.
dsacants4r8400.inp	Cantilever beam modeled with 400 S4R elements.
dsacantb31850.inp	Cantilever beam modeled with 50 B31 elements, including frequency step.
dsacantb318100.inp	Cantilever beam modeled with 100 B31 elements.
dsacantb318200.inp	Cantilever beam modeled with 200 B31 elements.
dsacantb318400.inp	Cantilever beam modeled with 400 B31 elements.

References

- Pedersen, P., G. Cheng, and J. Rasmussen, "On Accuracy Problems for Semi-Analytical Sensitivity Analysis," *Mechanics of Structures and Machines*, vol. 17, pp. 373–384, 1989.
- Barthelemy, B., and R. T. Haftka, "Accuracy Analysis of the Semi-Analytic Method for Shape Sensitivity Calculation," *Mechanics of Structures and Machines*, vol. 18, pp. 407–432, 1990.
- Fenyes, P. A., and R. V. Lust, "Error Analysis of Semianalytic Displacement Derivatives for Shape and Sizing Variables," *AIAA Journal*, vol. 29, pp. 271–279, 1991.
- Van Keulen, F., and H. De Boer, "Rigorous Improvement of Semi-Analytical Design Sensitivities by Exact Differentiation of Rigid Body Motions," *International Journal for Numerical Methods in Engineering*, vol. 42, pp. 71–91, 1998.

Tables

Table 1: Abaqus tip displacement sensitivity results.

Number of elements along the length	Perturbation size chosen by Abaqus for the dominant element			Percentage error		
	CPS8	S4R	B31	CPS8	S4R	B31
50	1.5e–06	1.5e–06	1.5e–08	–0.004	–0.002	–0.002
100	1.5e–06	1.5e–07	1.5e–08	0.008	–0.002	–0.002
200	1.5e–06	1.5e–07	1.5e–08	0.009	–0.002	–0.002
400	1.5e–07	1.5e–07	1.5e–08	0.009	–0.002	–0.002

Table 2: Comparison of eigenvalue and frequency sensitivities obtained with Abaqus and other methods for CPS8 element.

Bending Mode	Mode Number	Sensitivity w.r.t. beam length	Abaqus (default)	Abaqus (sizing every increment)	Overall central differencing scheme	Analytic
1	1	Eigenvalue	–3.460e–03	–3.460e–03	–3.460e–03	–3.461e–03
		Frequency	–9.362e–04	–9.362e–04	–9.350e–04	–9.363e–04
2	2	Eigenvalue	–1.355e–01	–1.355e–01	–1.353e–01	–1.353e–01
		Frequency	–5.860e–03	–5.860e–03	–5.856e–03	–5.859e–03
3	3	Eigenvalue	–1.057e–00	–1.057e–00	–1.052e–00	–1.058e–00

Bending Mode	Mode Number	Sensitivity w.r.t. beam length	Abaqus (default)	Abaqus (sizing every increment)	Overall central differencing scheme	Analytic
		Frequency	−1.630e−02	−1.636e−02	−1.630e−02	−1.637e−02

Table 3: Comparison of eigenvalue and frequency sensitivities obtained with Abaqus and other methods for S4R element.

Bending Mode	Mode Number	Sensitivity w.r.t. beam length	Abaqus (default)	Abaqus (sizing every increment)	Overall central differencing scheme	Analytic
1	4	Eigenvalue	−3.460e−03	−3.460e−03	−3.460e−03	−3.460e−03
		Frequency	−9.362e−04	−9.362e−04	−9.362e−04	−9.362e−04
2	10	Eigenvalue	−1.357e−01	−1.357e−01	−1.355e−01	−1.353e−01
		Frequency	−5.860e−03	−5.860e−03	−5.856e−03	−5.859e−03
3	17	Eigenvalue	−1.062e−00	−1.062e−00	−1.060e−00	−1.058e−00
		Frequency	−1.640e−02	−1.640e−02	−1.640e−02	−1.637e−02

Table 4: Comparison of eigenvalue and frequency sensitivities obtained with Abaqus and other methods for B31 element.

Bending Mode	Mode Number	Sensitivity w.r.t. beam length	Abaqus (default)	Abaqus (sizing every increment)	Overall central differencing scheme	Analytic
1	2	Eigenvalue	−3.460e−03	−3.460e−03	−3.457e−03	−3.461e−03
		Frequency	−9.366e−04	−9.360e−04	−9.370e−04	−9.363e−04
2	4	Eigenvalue	−1.353e−01	−1.353e−01	−1.354e−01	−1.353e−01
		Frequency	−5.854e−03	−5.854e−03	−5.850e−03	−5.859e−03
3	7	Eigenvalue	−1.055e−00	−1.055e−00	−1.058e−00	−1.058e−00
		Frequency	−1.634e−02	−1.634e−02	−1.639e−02	−1.637e−02

Figures

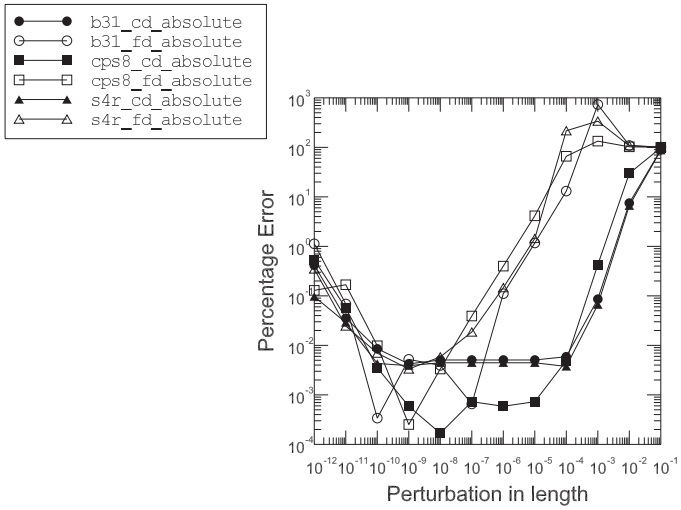


Figure 1: Variation of error in tip displacement sensitivity with respect to the perturbation size.

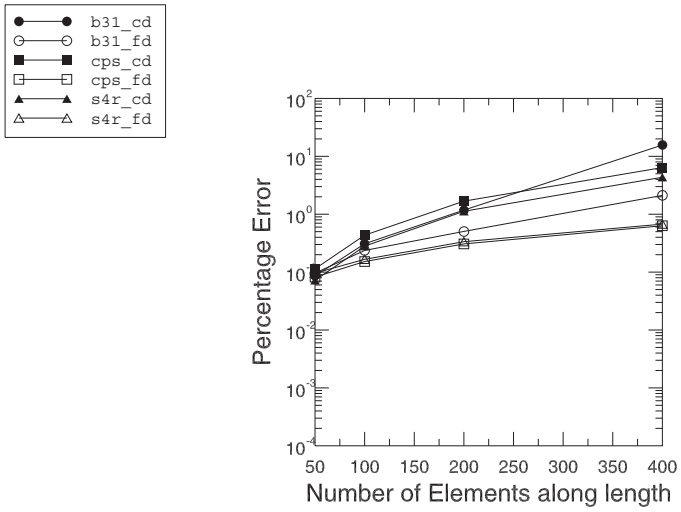


Figure 2: Variation of error in tip displacement sensitivity with mesh refinement for a perturbation size giving 0.1% error in dv_{tip}/dL for coarsest meshes.

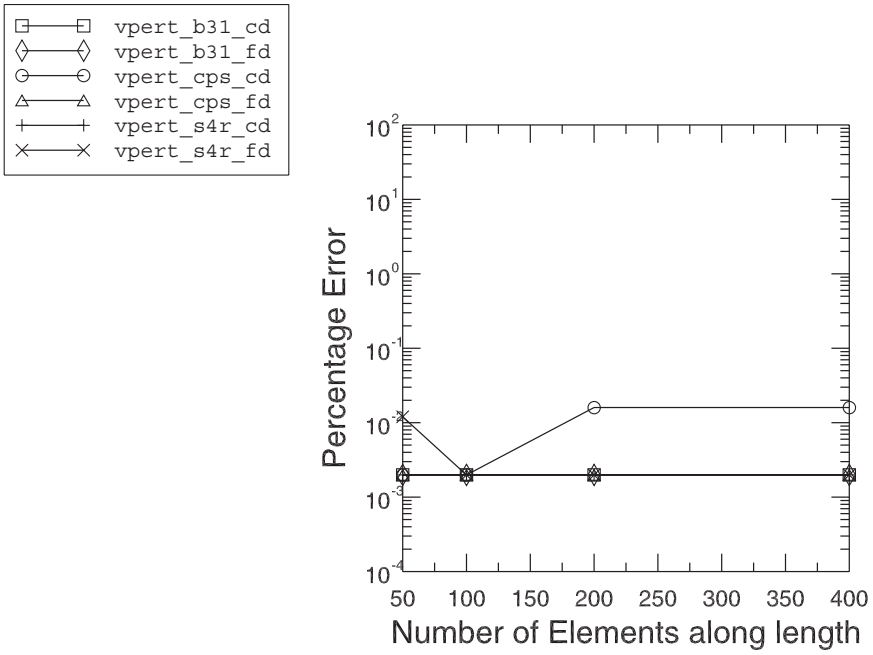


Figure 3: Variation of error in tip displacement sensitivity dv_{tip}/dL with mesh refinement for a perturbation of 1×10^{-9}.

Sensitivity of the stress concentration factor around a circular hole in a plate under uniaxial tension

Products: Abaqus/Standard Abaqus/Design

Overview

This example illustrates the design sensitivity analysis capability in Abaqus for a uniaxially loaded plate with a hole. In particular, the sensitivity of the stress concentration to a change in the shape of the hole from circular to elliptical is studied. Since an analytical solution exists, the results are easily verified.

Problem description

Taking advantage of symmetry, a quarter of a linear elastic square plate with a circular hole is modeled with CPS4 elements. The radius of the hole is 10 units. To simulate an infinite sheet, the plate width is set to 40 times the radius of the hole. A uniaxial tensile pressure P equal to 100 units is applied to the plate in the 2-direction during a linear static step in Abaqus/Standard. The finite element model is shown in *Figure 1*. The linear static step is followed by a static perturbation step where the load is perturbed by 50 units. Since this problem is linear, the perturbation should produce a sensitivity that is scaled by half.

We wish to study the effect of changing the shape of the hole from circular to elliptical on the maximum absolute stress, $\sigma_{\max}$; therefore, the nodal coordinates need to be related to the shape of the hole. Based on experience, the effect of a perturbation in the shape of the hole on the nodal coordinates lying outside the shaded region (shown in *Figure 1*) can be neglected. The circular hole can be regarded as a special case of an ellipse with its major axis lying along the x-axis. Let a and b represent the semi-major and semi-minor axes, respectively; and consider a point with coordinates (x, y) lying on a circle concentric to the hole. The coordinates of this point can be parameterized as $x = (a + c_1)\cos\theta$ and $y = (b + c_2)\sin\theta$, where c_1 and c_2 are constants and θ represents the angle (measured from the positive x-axis) of the position vector to the point from the center of the circle. If the point lies on the boundary of the hole, $c_1 = c_2 = 0$. Positive values of c_1 and c_2 move the point into the interior of the plate. The dependence of the nodal coordinates on the parameter a is specified by providing the gradients of x and y with respect to a for the parameter shape variation. The gradients are given as

$$\frac{dy}{da} = 0,$$

and

$$\frac{dx}{da} = \cos\theta.$$

Results and discussion

The maximum absolute stress, $\sigma_{\max}$, in an infinite plate with an elliptical hole subjected to uniaxial tension P perpendicular to the major axis is given by $\sigma_{\max} = P(1 + 2a/b)$ and is the 22-component at the end of the major axis of the ellipse (point M in *Figure 2*). The contour of σ_{22} in the vicinity of the hole is shown in *Figure 3* (shaded elements in *Figure 1*). The finite element model gives a maximum stress of 299.3 units, which is close to the expected value of 300 units.

The sensitivity of σ_{22} at M with respect to a is given by $d\sigma_{22}^{\mathrm{M}}/da = 2P/b$. The value of $d\sigma_{22}^{\mathrm{M}}/da$ obtained from Abaqus/Design is 19.99 units and compares well with the analytical result of 20 units. *Figure 4* shows the

contours of $d\sigma_{22}/da$ near the hole. As expected, σ_{22} is most sensitive to the variation in the semi-major axis at points nearest to the region of stress concentration. The value of $d\sigma_{22}^{\mathrm{M}}/da$ in the perturbation step is 9.995 units, exactly half the value in the linear static step as expected.

Input files

dsastressconcsens.inp	Quarter plate model using CPS4 elements.

Figures

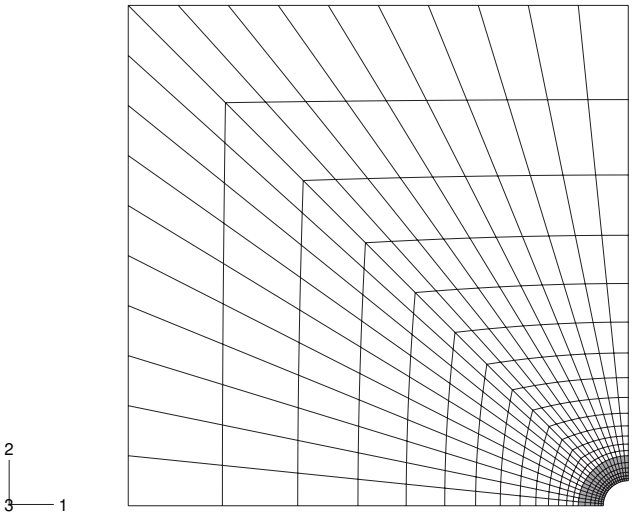


Figure 1: Quarter model using CPS4 elements.

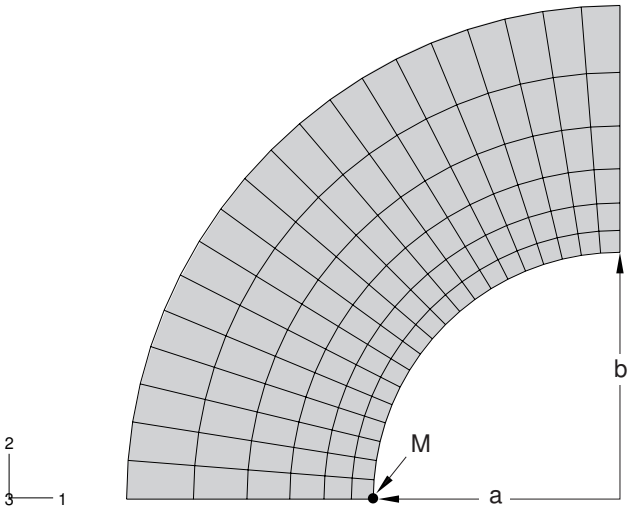


Figure 2: Mesh details near the hole.

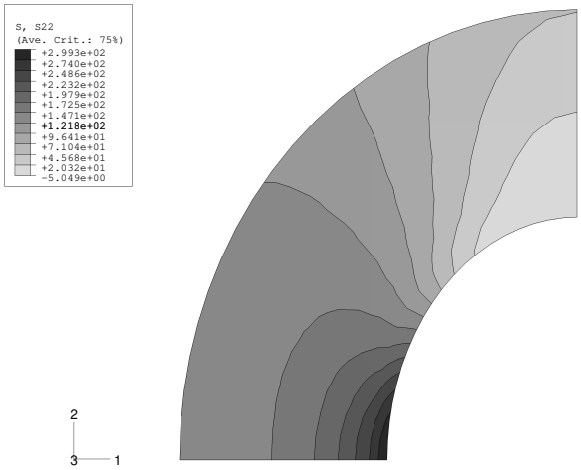


Figure 3: Variation of σ_{22} near the circular hole.

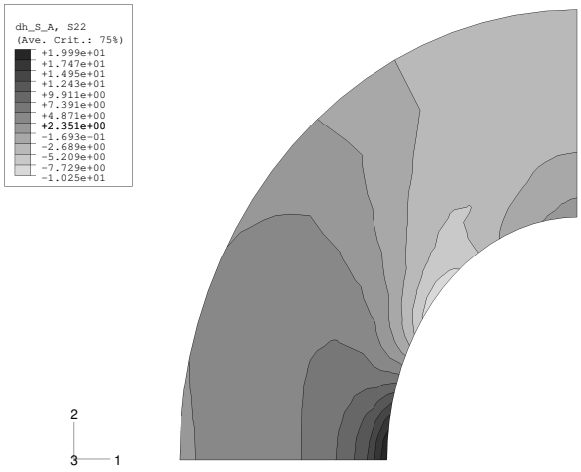


Figure 4: The variation of the sensitivity $d\sigma_{22}/da$ near the circular hole.

Sensitivity analysis of modified NAFEMS problem 3DNLG-1: Large deflection of Z-shaped cantilever under an end load

Products: Abaqus/Standard Abaqus/Design

Overview

This benchmark problem verifies the sensitivity results obtained using Abaqus/Design for the modified NAFEMS problem 3DNLG-1.
Results for the incremental and total DSA formulations are given. The results also serve to demonstrate the limitations of the total DSA formulation for history-dependent problems.

Problem description

The geometry of the problem is the same as that of the NAFEMS benchmark problem *3DNLG-1: Elastic large deflection response of a Z-shaped cantilever under an end load*. The problem is modified by adding plasticity to the material definition and by changing the direction of the application of the tip load to act along the axis of the beam as shown in *Figure 1* so as to cause extensive plastic yielding. All the problems use 72 elements to model the beam.

Initial yield occurs when the stress reaches 4000. The strain hardening increases the yield stress to 8000 at the plastic strain of 0.05 and to 9000 at the plastic strain of 0.10.

A total load of 50000 is applied in the global x- and y-directions such that the resultant load of 70710.6 acts along the axis of the beam. Of this, 25% is applied in the first step and the remainder is applied in the second step. In Step 3 75% of the load is removed, and the complete load removal is finished by the end of Step 4. The step time for each of the steps is unity.

The thickness of the beam, T; the applied load, P; and the angle, $\boldsymbol{\theta}$, of the inclined member of the Z-cantilever are chosen as the design parameters. The sensitivity is studied in detail for the same two output variables used in the original NAFEMS problem: the tip deflection of the cantilever, U, and the section moment, M, at point A, as shown in *Figure 1*.

Results and discussion

Tabulated sensitivity results are given at the end of the first, second, and third steps. The tables compare the sensitivities of the output variables with respect to the chosen design parameters obtained using the incremental and total DSA formulations in Abaqus to those obtained using the overall finite difference method (OFD). The OFD results are given for every element tested to provide a proper basis for comparison to the Abaqus sensitivity results, since every element provides slightly different sensitivity values.

Table 1 compares the sensitivity results obtained using the total and incremental formulations in Abaqus to the OFD results for element type B31. Similar results for element types B33, S4R, and S4 are shown in *Table 2*, *Table 3*, and *Table 4*, respectively. Since Abaqus outputs the sensitivity only at the integration points, the sensitivity for the moment refers to the integration point closest to the point A. The sensitivity results for shell elements have been multiplied by the width of the beam so that they can be compared with the results for beam elements. By the end of the second step (time = 2) the beam has straightened out, with the loads causing axial stretching and no further tip deflection. Consequently, at the end of Step 2 the tip deflection is insensitive to the design parameters P and T. The incremental formulation yields sensitivity results that are in close agreement with the OFD results. The total formulation shows deviations from the OFD results during the loading (up to time = 2) that are consistent with the approximations inherent in the total formulation. The deviations are large during unloading, when the problem becomes strongly history dependent (for times > 2) and the underlying

assumptions of the total formulation are no longer valid. *Figure 2* shows the comparison of the sensitivity results obtained using the total and incremental formulations for moment M with respect to *T*.

To compare the elements throughout the duration of the analysis, sensitivities obtained for B31 and S4R elements using the incremental formulation are plotted. The sensitivities of moment with respect to load, thickness, and incline angle are plotted in *Figure 3*, *Figure 4*, and *Figure 5*, respectively; while the sensitivities of the tip displacement with respect to load, thickness, and incline angle are plotted in *Figure 6*, *Figure 7*, and *Figure 8*, respectively. These plots show good agreement.

Input files

dsazcantb31.inp	Cantilever modeled using B31 elements.
dsazcantb33.inp	Cantilever modeled using B33 elements.
dsazcants4r.inp	Cantilever modeled using S4R elements.
dsazcants4.inp	Cantilever modeled using S4 elements.

Tables

Table 1: Sensitivity results for B31 elements.

Response	Design Parameter	Total Time	OFD	Incremental	Total
M	*P*	1.0	2.88E–01	2.88E–01	2.87E–01
		2.0	6.25E–02	6.19E–02	4.80E–02
		3.0	6.46E–02	6.52E–02	1.33E–01
	T	1.0	3.41E+04	3.40E+04	3.41E+04
		2.0	6.42E+04	6.42E+04	6.61E+04
		3.0	3.37E+04	3.37E+04	4.7E+04
	θ	1.0	6.72E+04	6.71E+04	5.7E+04
		2.0	7.53E+04	7.53E+04	8.24E+04
		3.0	4.35E+04	4.35E+04	–5.26E+04
U	*P*	1.0	–6.29E–07	–6.23E–07	–6.20E–07
		2.0	0.0	–9.45E–10	8.17E–11
		3.0	–5.72E–07	–5.63E–07	–5.07E–07
	T	1.0	5.49E–02	5.48E–02	5.47E–02
		2.0	0.0	8.1E–05	7.55E–06
		3.0	4.65E–02	4.65E–02	4.65E–02
	θ	1.0	–7.19E+01	–7.19E+01	–7.19E+01
		2.0	–7.20E+01	–7.20E+01	–7.19E+01
		3.0	–7.19E+01	–7.19E+01	–8.15E+01

Table 2: Sensitivity results for B33 elements.

Response	Design Parameter	Total Time	OFD	Incremental	Total
M	P	1.0	2.86E–01	2.86E–01	2.86E–01

		2.0	7.8E−02	7.70E−02	8.36E−01
		3.0	6.25E−02	6.21E−02	1.38E−01
	T	1.0	3.43E+04	3.43E+04	3.43E+04
		2.0	6.58E+04	6.58E+04	5.39+E04
		3.0	3.45E+04	3.45E+04	4.37E+04
	θ	1.0	6.62E+04	6.62E+04	6.60E+04
		2.0	7.78E+04	7.79E+04	7.77E+05
		3.0	4.77E+04	4.76E+04	−2.81E+04
U	*P*	1.0	−6.29E−07	−6.40E−07	−6.27E−07
		2.0	−1.35E−07	−7.08E−08	−3.63E−08
		3.0	−5.14E−07	−3.99E−07	−5.03E−07
	T	1.0	5.61E−02	5.66E−02	8.63E−02
		2.0	–6.17E−04	8.7E−04	2.40E−04
		3.0	4.20E−02	4.7E−02	8.06E−02
	θ	1.0	−7.19E+01	−7.19E+01	−7.17E+01
		2.0	−7.20E+01	−7.20E+01	−7.18E+01
		3.0	−7.19E+01	−7.19E+01	−8.14E+01

Table 3: Sensitivity results for S4R elements.

Response	Design Parameter	Total Time	OFD	Incremental	Total
M	*P*	1.0	2.92E−01	2.92E−01	2.90E−01
		2.0	9.74E−02	9.7E−02	6.84E−02
		3.0	7.58E−02	7.6E−02	1.53E−01
	T	1.0	3.38E+04	3.38E+04	3.38E+04
		2.0	6.44E+04	6.44E+04	6.48E+04
		3.0	3.38E+04	3.38E+04	4.56E+04
	θ	1.0	6.76E+04	6.76E+04	5.66E+04
		2.0	7.96E+04	7.94E+04	5.64E+04
		3.0	4.52E+04	4.52E+04	–4.0E+04
U	*P*	1.0	−6.11E−07	−6.22E−07	−6.22E−07
		2.0	0.0	−4.90E−09	−2.42E−09
		3.0	–5.72E−07	−5.67E−07	−5.16E−07
	T	1.0	5.49E−02	5.51E−02	5.42E−02
		2.0	5.60E−04	4.95E−04	7.34E−05
		3.0	4.71E−02	4.74E−02	4.65E−02
	θ	1.0	−7.19E+01	−7.19E+01	−7.19E+01
		2.0	−7.19E+01	−7.19E+01	−7.20E+01
		3.0	−7.19E+01	−7.19E+01	−7.99E+01

Table 4: Sensitivity results for S4 elements.

Response	Design Parameter	Total Time	OFD	Incremental	Total
M	*P*	1.0	2.92E−01	2.92E−01	2.90E−01
		2.0	1.07E−01	1.07E−01	7.58E−02
		3.0	8.10E−02	8.12E−02	1.53E−01
	T	1.0	3.38E+04	3.38E+04	3.38E+04
		2.0	6.44E+04	6.44E+04	6.48E+04
		3.0	3.38E+04	3.38E+04	4.56E+04
	θ	1.0	6.76E+04	6.76E+04	5.66E+04
		2.0	8.02E+04	8.02E+04	5.74E+03
		3.0	4.52E+04	4.54E+04	–4.30E+03
U	*P*	1.0	−6.1E−07	−6.22E−07	−6.22E−07
		2.0	0.0	−2.40E−09	−2.38E−09
		3.0	–5.72E−07	−5.69E−07	−5.19E−07
	T	1.0	5.49E−02	5.51E−02	5.42E−02
		2.0	5.60E−04	2.85E−04	7.74E−05
		3.0	4.76E−02	4.76E−02	4.6E−02
	θ	1.0	−7.19E+01	−7.19E+01	−7.19E+01
		2.0	−7.19E+01	−7.19E+01	−7.20E+01
		3.0	−7.19E+01	−7.19E+01	−7.98E+01

Figures

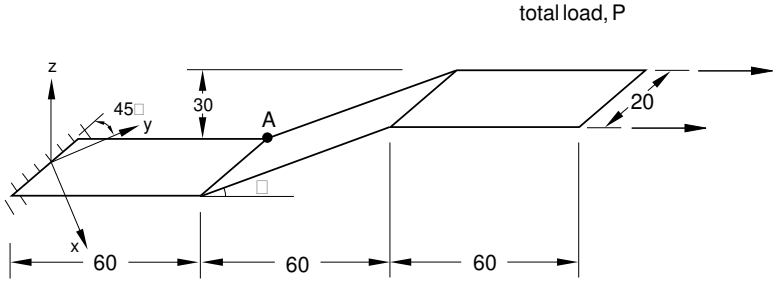


Figure 1: The geometry and modified loading for the Z-cantilever.

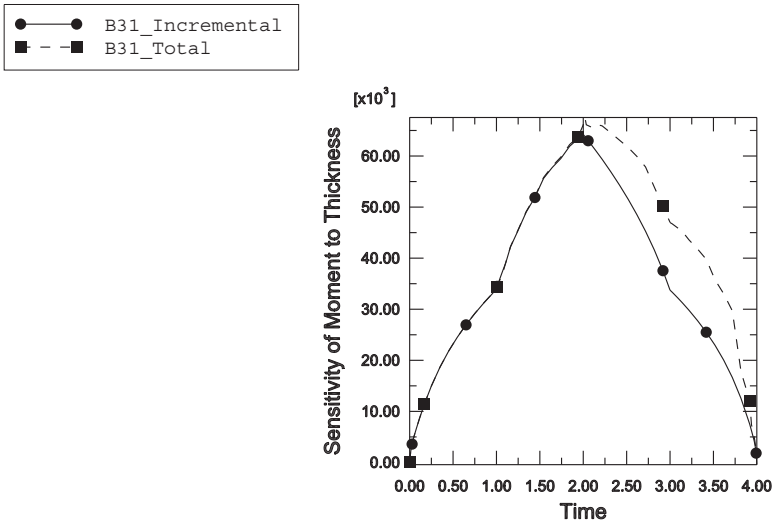


Figure 2: Comparison of the sensitivity of moment M to thickness T obtained using the total and incremental formulations.

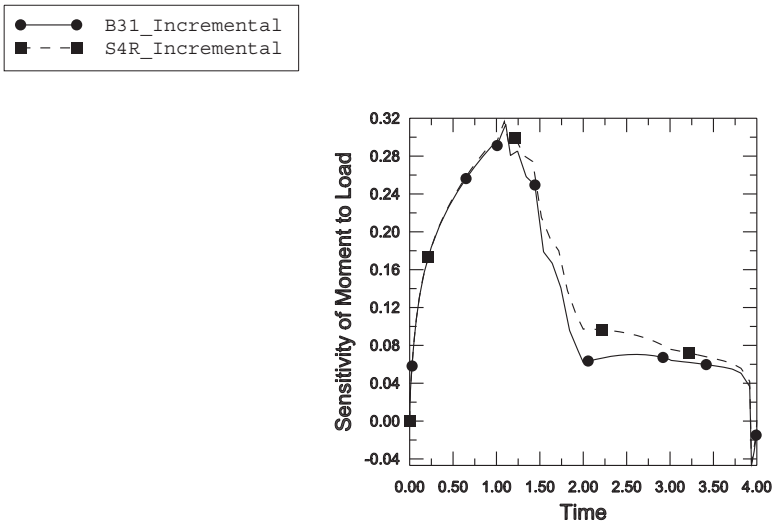


Figure 3: Sensitivity of the moment M at point A with respect to the applied load P.

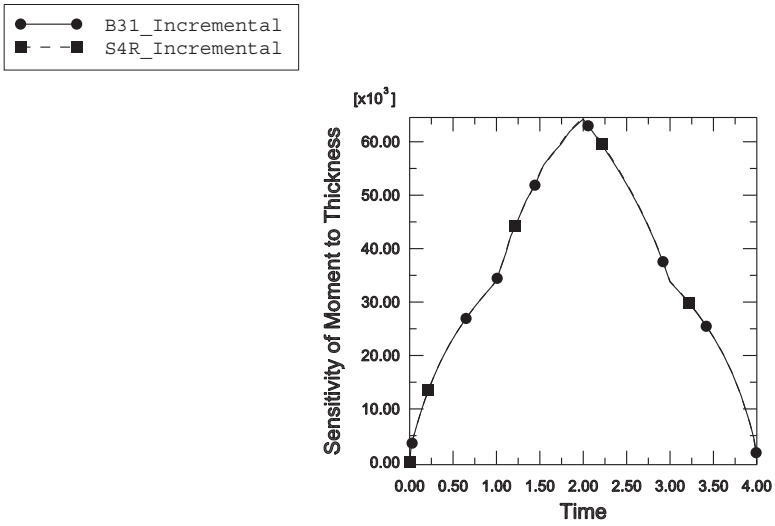


Figure 4: Sensitivity of the moment M at point A with respect to the thickness T.

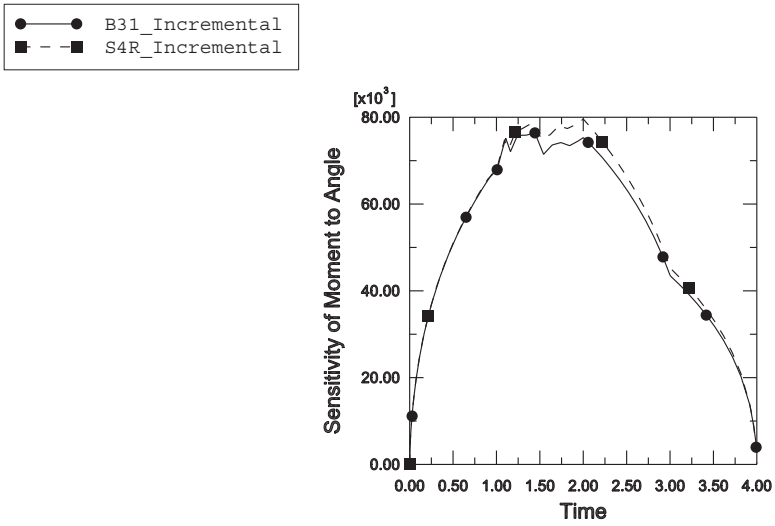


Figure 5: Sensitivity of the moment M at point A with respect to angle θ.

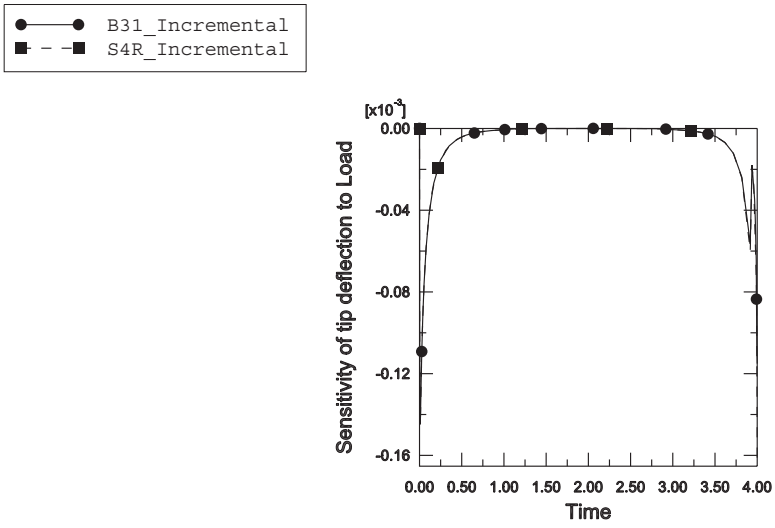


Figure 6: Sensitivity of the deflection U with respect to the applied load P.

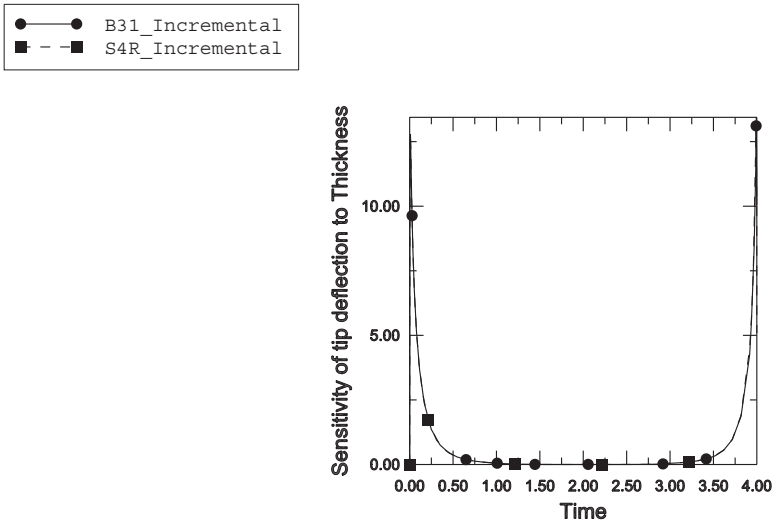


Figure 7: Sensitivity of the deflection U with respect to the thickness T.

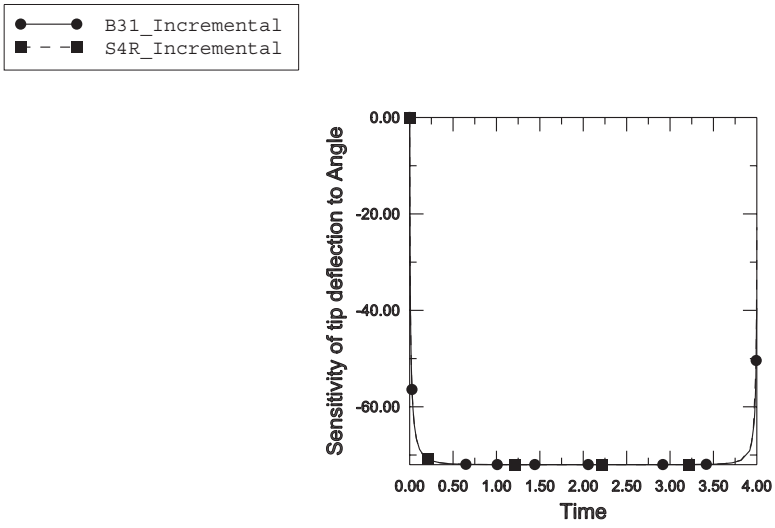


Figure 8: Sensitivity of the deflection U with respect to angle θ.

Modeling discontinuities using XFEM

In this section:

- *Crack propagation of a single-edge notch simulated using XFEM*
- *Crack propagation in a plate with a hole simulated using XFEM*
- *Crack propagation in a beam under impact loading simulated using XFEM*
- *Dynamic shear failure of a single-edge notch simulated using XFEM*
- *Propagation of hydraulically driven fracture using XFEM*

Crack propagation of a single-edge notch simulated using XFEM

Products: Abaqus/Standard Abaqus/CAE

Overview

This example verifies and illustrates the use of the extended finite element method (XFEM) in Abaqus/Standard to predict crack initiation and propagation of a single-edge notch in a specimen along an arbitrary path by modeling the crack as an enriched feature.

Both the XFEM-based cohesive segments method and the XFEM-based linear elastic fracture mechanics (LEFM) approach are used to analyze this problem. Both two- and three-dimensional models are studied. The specimen is subjected to loadings ranging from pure Mode I to pure Mode II to mixed-mode. In some cases distributed pressure loads are applied to the cracked element surfaces as the crack initiates and propagates in the specimen. The results presented are compared to the available analytical solutions and those obtained using cohesive elements. In addition, the same model is analyzed using the XFEM-based low-cycle fatigue criterion to assess the fatigue life when the model is subjected to subcritical cyclic loading.

The fatigue crack growth analysis subjected to pure mode II loading is also carried out using submodeling techniques. Both the traditional node-based submodeling approach and the field import–based submodeling approach are studied.

Problem description

Two single-edge notch specimens are studied. The first specimen is shown in *Figure 1* and has a length of 3 m, a thickness of 1 m, a width of 3 m, and an initial crack length of 0.3 m, loaded under pure Mode I loading. Equal and opposite displacements are applied at both ends in the longitudinal direction. The maximum displacement value is set equal to 0.001 m. In the low-cycle fatigue analysis, a cyclic displacement loading with a peak value of 8×10^{-5} m is specified. The second specimen has a length of 6 m, a thickness of 1 m, a width of 3 m, and an initial crack length of 1.5 m, loaded under pure Mode II or mixed-mode loading. Equal and opposite displacements are applied at both ends in the width direction under pure Mode II loading, while equal and opposite displacements are applied at both ends in both the longitudinal and width directions under mixed-mode loading. The maximum displacement value is set equal to 0.01 m. In the low-cycle fatigue analysis, a cyclic displacement loading with a peak value of 8×10^{-4} m is specified.

The material data for the bulk material properties in the enriched elements are $E = 210.0$ GPa and $\nu = 0.3$.

The response of cohesive behavior in the enriched elements in the model is specified. The maximum principal stress failure criterion is selected for damage initiation; and a mixed-mode, energy-based damage evolution law based on a power law criterion is selected for damage propagation. The relevant material data are as follows: $\sigma_{max} = 220$ MPa, $G_{1C} = 42.2 \times 10^3$ N/m, $G_{2C} = 42.2 \times 10^3$ N/m, $G_{3C} = 42.2 \times 10^3$ N/m, and $\alpha = 1.0$. The relevant material data defined above are also used in the model simulated using the XFEM-based LEFM approach.

When the fatigue crack growth analysis is performed, the additional relevant data for the fracture energy release rate–based Paris law are as follows: $c_1 = 0.5$, $c_2 = -0.1$, $c_3 = 4.88 \times 10^{-6}$, $c_4 = 1.15$, $\frac{G_{thresh}}{G_C} = 0.001$, and $\frac{G_{pl}}{G_C} = 0.85$. When the stress intensity factor–based Paris law is used, the equivalent constants are $c_3 = 4.17 \times 10^{-19}$, $c_4 = 2.3$.

Results and discussion

Figure 2 shows plots of the prescribed displacement versus the corresponding reaction force obtained using the XFEM method under the pure Mode I loading compared with the results obtained using cohesive elements. The results displayed are from the two-dimensional plane strain analyses. The results obtained using the XFEM method agree well with those obtained using cohesive elements. The results from the equivalent three-dimensional models show similar agreement.

Under the pure Mode II or mixed-mode loading, the crack will no longer propagate along a straight path and will instead propagate along a path based on the maximum tangential stress criterion according to *Erdogan and Sih (1963)*. The direction of crack propagation is given by

$$\hat{\theta} = \cos^{-1}\left(\frac{3K_{II}^2+\sqrt{K_I^4+8K_I^2K_{II}^2}}{K_I^2+9K_{II}^2}\right),$$

where the crack propagation angle, $\hat{\theta}$, is measured with respect to the crack plane. $\hat{\theta} = 0$ represents the crack propagation in the "straight-ahead" direction. $\hat{\theta} < 0$ if $K_{II} > 0$ while $\hat{\theta} > 0$ if $K_{II} < 0$. Under pure Mode II loading, the above equation predicts that the crack will propagate at an angle of 70° while the crack propagation angle predicted using XFEM is 66.5°, which is illustrated in *Figure 3*.

Similar results are obtained when comparing the crack profiles obtained using the traditional node-based submodeling approach with those obtained using the field import–based submodeling approach.

Input files

Pure Mode I loading

XFEM-based cohesive segments method:

crackprop_mode_I_xfem_cpe4r.inp	Two-dimensional plane strain model with reduced integration.
crackprop_mode_I_xfem_cpe4.inp	Two-dimensional plane strain model.
crackprop_mode_I_xfem_dload_cpe4.inp	Two-dimensional plane strain model with distributed pressure loads.
crackprop_mode_I_xfem_cps4r.inp	Two-dimensional plane stress model with reduced integration.
crackprop_mode_I_xfem_cps4.inp	Two-dimensional plane stress model.
crackprop_mode_I_xfem_dload_cps4.inp	Two-dimensional plane stress model with distributed pressure loads.
crackprop_mode_I_xfem_dload_cax4.inp	Axisymmetric model with distributed pressure loads.
crackprop_mode_I_xfem_c3d4.inp	Three-dimensional tetrahedron model.
crackprop_mode_I_xfem_dload_c3d4.inp	Three-dimensional tetrahedron model with distributed pressure loads.
crackprop_mode_I_xfem_c3d8r.inp	Three-dimensional brick model with reduced integration.
crackprop_mode_I_xfem_c3d8.inp	Three-dimensional brick model.
crackprop_mode_I_xfem_dload_c3d8.inp	Three-dimensional brick model with distributed pressure loads.
crackprop_mode_I_xfem_c3d10.inp	Three-dimensional second-order tetrahedron model.

XFEM-based LEFM approach:

crackprop_mode_I_lefm_xfem_cpe4r.inp	Two-dimensional plane strain model with reduced integration.
crackprop_mode_I_lefm_xfem_cpe4.inp	Two-dimensional plane strain model.
crackprop_mode_I_lefm_xfem_c3d4.inp	Three-dimensional tetrahedron model.
crackprop_mode_I_lefm_xfem_c3d8r.inp	Three-dimensional brick model with reduced integration.
crackprop_mode_I_lefm_xfem_c3d10.inp	Three-dimensional second-order tetrahedron model.

XFEM-based low-cycle fatigue analysis:

crackprop_mode_I_fatigue_xfem_cpe4.inp	Same as crackprop_mode_I_lefm_xfem_cpe4.inp but subjected to cyclic displacement loading.
crackprop_mode_I_fatigue_xfem_c3d8r.inp	Same as crackprop_mode_I_lefm_xfem_c3d8r.inp but subjected to cyclic displacement loading.
crackprop_mode_I_fatigue_xfem_c3d10.inp	Same as crackprop_mode_I_lefm_xfem_c3d10.inp but subjected to cyclic displacement loading.

Pure Mode II loading

XFEM-based cohesive segments method:

crackprop_mode_II_xfem_cpe4r.inp	Two-dimensional plane strain model with reduced integration.
crackprop_mode_II_xfem_cpe4.inp	Two-dimensional plane strain model.
crackprop_mode_II_xfem_cps4r.inp	Two-dimensional plane stress model with reduced integration.
crackprop_mode_II_xfem_cps4.inp	Two-dimensional plane stress model.
crackprop_mode_II_xfem_c3d4.inp	Three-dimensional tetrahedron model.
crackprop_mode_II_xfem_c3d8r.inp	Three-dimensional brick model with reduced integration.
crackprop_mode_II_xfem_c3d8.inp	Three-dimensional brick model.
crackprop_mode_II_xfem_c3d8r_user.inp	Same as crackprop_mode_II_xfem_c3d8r.inp but with user-defined damage initiation criterion.
crackprop_maxps_xfem_udmgini.f	Subroutine for user-defined damage initiation criterion.
crackprop_mode_II_xfem_c3d10.inp	Three-dimensional second-order tetrahedron model.

XFEM-based LEFM approach:

crackprop_mode_II_lefm_xfem_cpe4r.inp	Two-dimensional plane strain model with reduced integration.
crackprop_mode_II_lefm_xfem_cpe4.inp	Two-dimensional plane strain model.
crackprop_mode_II_lefm_xfem_c3d4.inp	Three-dimensional tetrahedron model.
crackprop_mode_II_lefm_xfem_c3d8r.inp	Three-dimensional brick model with reduced integration.
crackprop_mode_II_lefm_xfem_c3d10.inp	Three-dimensional second-order tetrahedron model.

XFEM-based low-cycle fatigue analysis:

crackprop_mode_II_fatigue_xfem_cpe4.inp	Same as crackprop_mode_II_lefm_xfem_cpe4.inp but subjected to cyclic displacement loading.
crackprop_mode_II_fatigue_xfem_c3d8r.inp	Same as crackprop_mode_II_lefm_xfem_c3d8r.inp but subjected to cyclic displacement loading using the direct cyclic approach.
crackprop_mode_II_fatigue_xfem_c3d8r_2.inp	Same as crackprop_mode_II_lefm_xfem_c3d8r.inp but subjected to cyclic displacement loading using the general fatigue crack growth approach.

crackprop_mode_II_fatigue_xfem_c3d8r_2_simp.inp	Same as crackprop_mode_II_lefm_xfem_c3d8r.inp but subjected to cyclic displacement loading using the simplified fatigue crack growth approach.
crackprop_mode_II_fatigue_xfem_c3d8r_2_global_onestep.inp	Global model subjected to a cyclic loading defined in a single static step.
crackprop_mode_II_fatigue_xfem_c3d8r_2_global.inp	Global model subjected to a cyclic loading defined in two static steps.
crackprop_mode_II_fatigue_xfem_c3d8r_2_submodel_onestep.inp	Same as `crackprop_mode_II_fatigue_xfem_c3d8r_2.inp` but uses a submodel driven from the displacements obtained from the global model `crackprop_mode_II_fatigue_xfem_c3d8r_2_global_onestep.inp` using the traditional node-based submodeling approach.
crackprop_mode_II_fatigue_xfem_c3d8r_2_submodel_onestep_importfield.inp	Same as `crackprop_mode_II_fatigue_xfem_c3d8r_2.inp` but uses a submodel driven from the displacements obtained from the global model `crackprop_mode_II_fatigue_xfem_c3d8r_2_global_onestep.inp` using the field import–based submodeling approach.
crackprop_mode_II_fatigue_xfem_c3d8r_2_submodel_importfield.inp	Same as `crackprop_mode_II_fatigue_xfem_c3d8r_2.inp` but uses a submodel driven from displacements obtained from the global model `crackprop_mode_II_fatigue_xfem_c3d8r_2_global.inp` using the field import–based submodeling approach.
crackprop_mode_II_fatigue_xfem_c3d8r_3_tabular.inp	Same as `crackprop_mode_II_fatigue_xfem_c3d8r_2.inp` but with tabular form to define fatigue crack growth.
crackprop_mode_II_fatigue_xfem_c3d8r_3_user.inp	Same as `crackprop_mode_II_fatigue_xfem_c3d8r_2.inp` but with user-defined subroutine to define fatigue crack growth.
crackprop_mode_II_fatigue_xfem_c3d8r_3_simp_tabular.inp	Same as `crackprop_mode_II_fatigue_xfem_c3d8r_2_simp.inp` but with tabular form to define fatigue crack growth.
crackprop_mode_II_fatigue_xfem_c3d8r_3_simp_user.inp	Same as `crackprop_mode_II_fatigue_xfem_c3d8r_2_simp.inp` but with user-defined subroutine to define fatigue crack growth.
crackprop_mode_II_fatigue_xfem_c3d10.inp	Same as `crackprop_mode_II_lefm_xfem_c3d10.inp` but subjected to cyclic displacement loading.
crackprop_mode_II_fatigue_xfem_c3d10_2.inp	Same as `crackprop_mode_II_lefm_xfem_c3d10.inp` but subjected to cyclic displacement loading using the general fatigue crack growth approach.
crackprop_mode_II_fatigue_xfem_c3d10_2_irwin.inp	Same as `crackprop_mode_II_fatigue_xfem_c3d10_2.inp` but with IRWIN law to define fatigue crack growth.
crackprop_mode_II_fatigue_xfem_c3d10_2_tabular_kbased.inp	Same as `crackprop_mode_II_fatigue_xfem_c3d10_2.inp` but with K-BASED tabular form to define fatigue crack growth.
crackprop_mode_II_fatigue_xfem_c3d10_2_user_kbased.inp	Same as `crackprop_mode_II_fatigue_xfem_c3d10_2.inp` but with K-BASED user-defined subroutine to define fatigue crack growth.
crackprop_fatigue_3d_3_user.f	User subroutine *UMIXMODEFATIGUE* used in `crackprop_mode_II_fatigue_xfem_c3d8r_3_user.inp` and `crackprop_mode_II_fatigue_xfem_c3d8r_3_simp_user.inp`.
crackprop_fatigue_3d_3_kbased_user.f	User subroutine *UMIXMODEFATIGUE* used in `crackprop_mode_II_fatigue_xfem_c3d10_2_user_kbased.inp`.

Mixed-mode loading

XFEM-based cohesive segments method:

crackprop_mixmode_xfem_cpe4r.inp	Two-dimensional plane strain model with reduced integration.
crackprop_mixmode_xfem_cpe4.inp	Two-dimensional plane strain model.
crackprop_mixmode_xfem_cps4r.inp	Two-dimensional plane stress model with reduced integration.
crackprop_mixmode_xfem_cps4.inp	Two-dimensional plane stress model.
crackprop_mixmode_xfem_c3d4.inp	Three-dimensional tetrahedron model.
crackprop_mixmode_xfem_c3d8r.inp	Three-dimensional brick model with reduced integration.
crackprop_mixmode_xfem_c3d8.inp	Three-dimensional brick model.
crackprop_mixmode_xfem_c3d10.inp	Three-dimensional second-order tetrahedron model.

Python scripts

Pure Mode II loading

XFEM-based LEFM approach:

crackprop_mode_II_lefm_xfem_cpe4.py	Script to generate the two-dimensional plane strain model in Abaqus/CAE.
crackprop_mode_II_lefm_xfem_c3d8r.py	Script to generate the three-dimensional brick model in Abaqus/CAE.

Mixed-mode loading

XFEM-based cohesive segments method:

crackprop_mixmode_xfem_cpe4.py	Script to generate the two-dimensional plane strain model in Abaqus/CAE.
crackprop_mixmode_xfem_c3d8.py	Script to generate the three-dimensional brick model in Abaqus/CAE.

References

- Erdogan, F., and G. C. Sih, “On the Crack Extension in Plates under Plane Loading and Transverse Shear,” *Journal of Basic Engineering*, vol. 85 519–527, 1963.

Figures

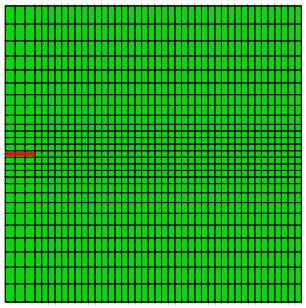


Figure 1: Model geometry for crack propagation in a single-edge notch specimen.

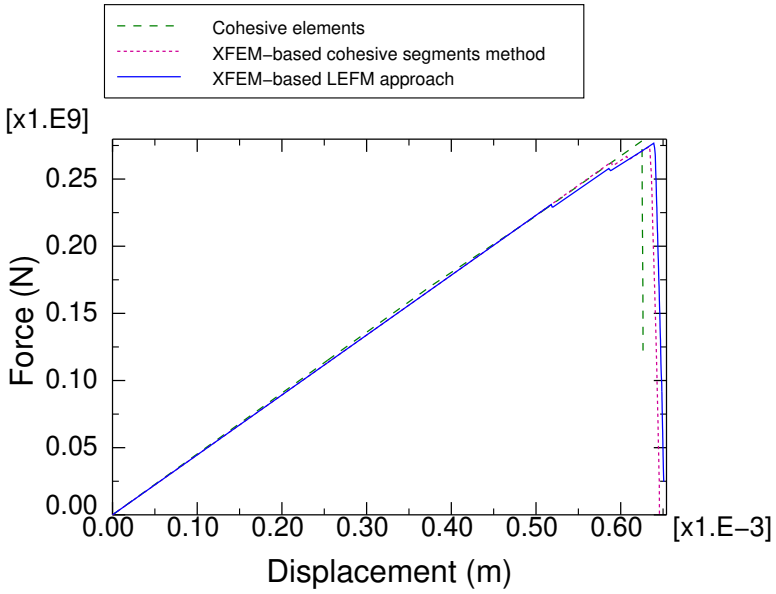


Figure 2: Reaction force versus prescribed displacement: XFEM and cohesive element results.

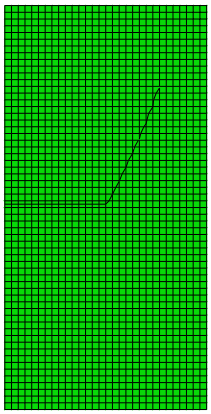


Figure 3: Crack path in a single-edge notch specimen under pure Mode II loading.

Crack propagation in a plate with a hole simulated using XFEM

Products: Abaqus/Standard Abaqus/CAE

Overview

This example verifies and illustrates the use of the extended finite element method (XFEM) in Abaqus/Standard to predict crack initiation and propagation due to stress concentration in a plate with a hole.

Both the XFEM-based cohesive segments method and the XFEM-based linear elastic fracture mechanics (LEFM) approach are used to analyze this problem. The specimen is subjected to pure Mode I loading. In some cases distributed pressure loads are applied to the cracked element surfaces as the crack initiates and propagates in the specimen. The results presented are compared to the available analytical solution. In addition, the same model is analyzed using the XFEM-based low-cycle fatigue (LCF) criterion to assess the fatigue life when the model is subjected to sub-critical cyclic loading.

Problem description

A plate with a circular hole is studied. The specimen, shown in *Figure 1*, has a length of 0.34 m, a thickness of 0.02 m, a width of 0.2 m, and a hole radius of 0.02 m, under pure Mode I loading. *Figure 1* defines the dimensions used to calculate the variation of crack length, $\frac{a+c}{b}$: a is the crack length, b is half the specimen width, and c is the hole radius.

Equal and opposite displacements are applied at both ends in the longitudinal direction. The maximum displacement value is set equal to 0.00055 m. To examine the mesh sensitivity, three different mesh discretizations of the same geometry are studied. Symmetry conditions reduce the specimen to a half model. The original mesh, as depicted in *Figure 2*, has 2060 plane strain elements. The second mesh has four times as many elements as the original one, while the third mesh has sixteen times as many elements as the original one.

In the low-cycle fatigue analysis, two steps are involved. A static step is used to nucleate a crack at the site of stress concentration prior to the low-cycle fatigue direct cyclic step, in which a cyclic distributed loading with a peak value of 1.25 MPa is specified. Three different mesh discretizations of the same geometry are studied. The second mesh has twice as many elements as the original mesh, while the third mesh has four times as many elements as the original mesh.

The material data for the bulk material properties in the enriched elements are $E = 3.24$ GPa and $\nu = 0.3$.

The response of cohesive behavior in the enriched elements in the model is specified. The maximum principal stress failure criterion is selected for damage initiation, and an energy-based damage evolution law based on a BK law criterion is selected for damage propagation. The relevant material data are as follows: $\sigma_{max} = 22.0$ MPa, $G_{1C} = 2.87 \times 10^3$ N/m, $G_{2C} = 2.87 \times 10^3$ N/m, and $\eta = 1.0$. The relevant material data defined above are also used in the model simulated using the XFEM-based LEFM approach. When the low-cycle fatigue analysis using the Paris law is performed, the additional relevant data are as follows: $c_1 = 0.5$, $c_2 = -0.1$, $c_3 = 4.88 \times 10^{-6}$, $c_4 = 1.15$, $\frac{G_{thresh}}{G_C} = 0.001$, and $\frac{G_{pl}}{G_C} = 0.85$.

Results and discussion

Figure 3 shows plots of the prescribed displacement versus the corresponding reaction force with different mesh discretizations when the XFEM-based cohesive segments method is used. The figure clearly illustrates the

convergence of the response to the same solution with mesh refinement. A plot of the applied stress versus the variation of crack length is presented in *Figure 4* and compared with the results obtained by using the XFEM-based LEFM approach as well as the analytical solution of *Tada et al. (1985)*. The agreement is better than 10% except when the crack length is small, in which case the stress singularity ahead of the crack is not considered by the XFEM approach. However, as indicated in this figure, the crack initiates (i.e., $a = 0$) when the applied stress, σ, reaches a level of 8.37 MPa, giving a ratio of $\frac{\sigma_{max}}{\sigma}$ equal to 2.63. This value is in close agreement with the stress concentration factor of 2.52 obtained analytically for the same geometry. In addition, 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 5*. Reasonably good agreement is obtained. Similarly, good results are obtained using the general and the simplified fatigue crack growth approaches.

Input files

crackprop_hole_xfem_cpe4.inp	Abaqus/Standard two-dimensional plane strain model with a hole under pure Mode I loading simulated using the XFEM-based cohesive segments method.
crackprop_hole_xfem_dload_cpe4.inp	Same as crackprop_hole_xfem_cpe4.inp but with distributed pressure loads.
crackprop_hole_xfem_dload_user_cpe4.inp	Same as crackprop_hole_xfem_cpe4.inp but with user-defined distributed pressure loads.
crackprop_hole_xfem_dload_user_cpe4.f	Subroutine for user-defined distributed pressure loads.
crackprop_hole_lefm_xfem_cpe4.inp	Abaqus/Standard two-dimensional plane strain model with a hole under pure Mode I loading simulated using the XFEM-based LEFM approach.
crackprop_hole_xfem_cpe4_user.inp	Same as crackprop_hole_xfem_cpe4.inp but with user-defined damage initiation criterion.
crackprop_maxps_quads_xfem_udmgini.f	Subroutine for a user-defined damage initiation criterion with two different failure mechanisms.
crackprop_hole_fatigue_xfem_cpe4.inp	Same as crackprop_hole_lefm_xfem_cpe4.inp but subjected to cyclic distributed loading using the direct cyclic approach.
crackprop_hole_fatigue_xfem_cpe4_2.inp	Same as crackprop_hole_fatigue_xfem_cpe4.inp but with twice as many elements using the direct cyclic approach.
crackprop_hole_fatigue_xfem_cpe4_3.inp	Same as crackprop_hole_fatigue_xfem_cpe4.inp but with four times as many elements using the direct cyclic approach.
crackprop_hole_fatigue_xfem_cpe4_4.inp	Same as crackprop_hole_lefm_xfem_cpe4.inp but subjected to cyclic distributed loading using the general fatigue crack growth approach.
crackprop_hole_fatigue_xfem_cpe4_4_simp.inp	Same as crackprop_hole_lefm_xfem_cpe4_4.inp but using the simplified fatigue crack growth approach.

Python scripts

crackprop_hole_xfem_cpe4.py	Script to generate the two-dimensional plane strain model with a hole under pure Mode I loading in Abaqus/CAE simulated using the XFEM-based cohesive segments method.
crackprop_hole_lefm_xfem_cpe4.py	Script to generate the two-dimensional plane strain model with a hole under pure Mode I loading in Abaqus/CAE simulated using the XFEM-based LEFM approach.

References

- Tada, H., P. C. Paris, and G. R. Irwin, "The Stress Analysis of Cracks Handbook, 2nd Edition," *Paris Productions Incorporated, 226 Woodbourne Drive, St. Louis, Missouri, 63105*, 1985.

Figures

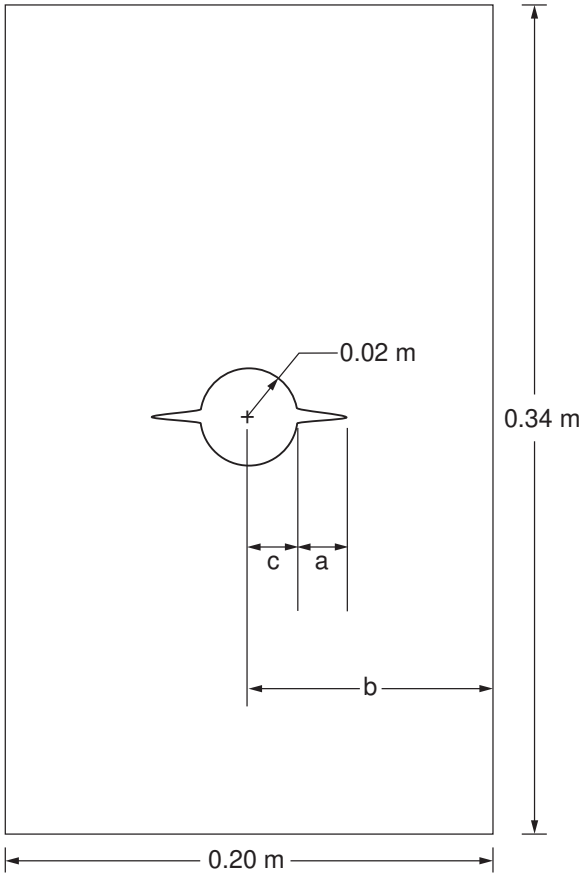


Figure 1: Model geometry of the plate with a hole specimen.

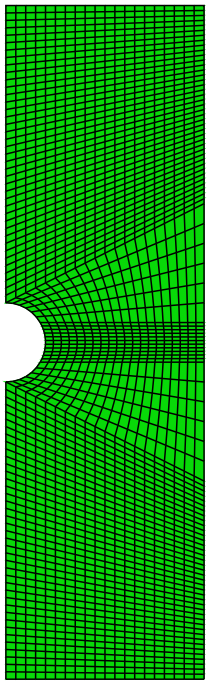


Figure 2: Original mesh of the half model for crack propagation in a plate with a hole.

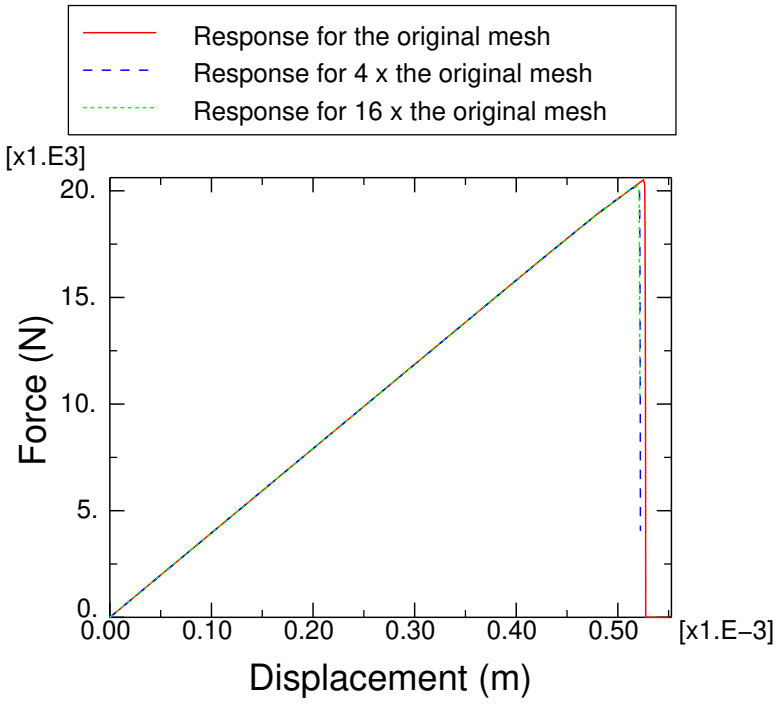


Figure 3: Reaction force versus prescribed displacement with different mesh discretizations (XFEM-based cohesive segments method).

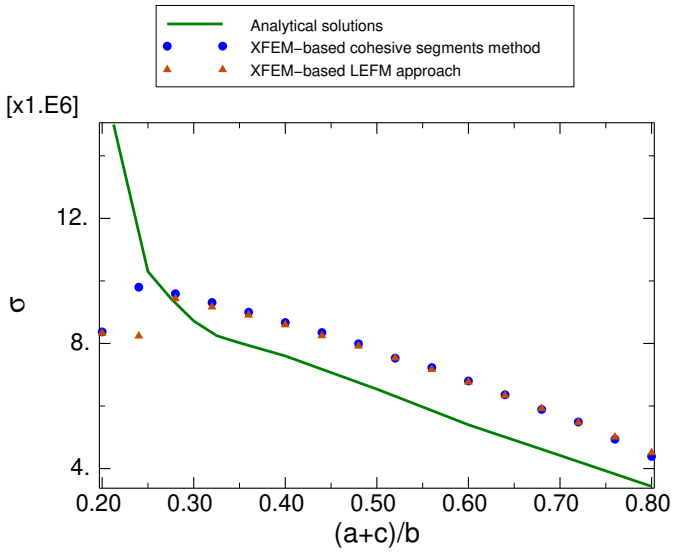


Figure 4: Applied stress versus variation of crack length: XFEM and analytical solution.

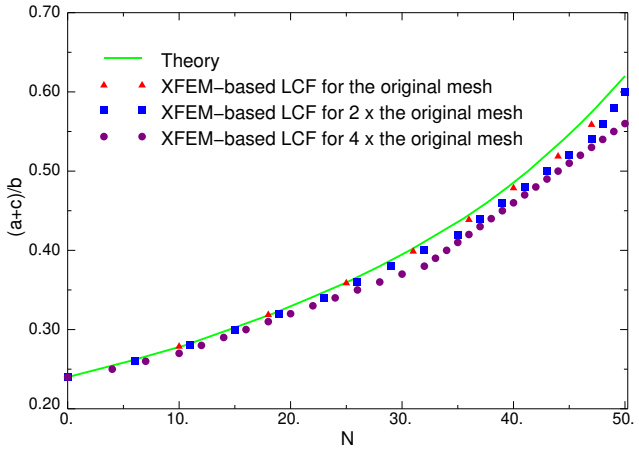


Figure 5: Crack length versus cycle number in a low-cycle fatigue analysis with different mesh densities.

Crack propagation in a beam under impact loading simulated using XFEM

Products: Abaqus/Standard

Overview

This example verifies and illustrates the use of the extended finite element method (XFEM) in Abaqus/Standard to predict dynamic crack propagation of a beam with an offset edge crack.
The specimen is subjected to a mixed-mode impact loading. Both two- and three-dimensional models are studied. The crack paths and crack initiation angles presented are compared to the experimental results of *John and Shah (1990)*.

Problem description

A beam with an offset edge crack is studied. The specimen, shown in *Figure 1*, has a length of 0.2286 m, a thickness of 0.0254 m, and a width of 0.0762 m. The distance between supports is 0.2032 m. To induce a mixed-mode fracture, an initial crack with a length of 0.019 m is made at an offset distance of 0.06604 m measured from the midspan in the specimen. A velocity boundary condition is imposed at the midspan of the specimen to simulate a dynamic impact:

$$v(t) = \begin{cases} \dfrac{v_0 t}{t_{ramp}}, & t_{ramp} \geq t \\ v_0, & \text{otherwise;} \end{cases}$$

where v_0 = 0.06 m/s and t_{ramp} = 1.96 × 10^{-4} s.

The material data for the bulk material properties in the enriched elements are E = 31.37 GPa, ρ = 2400 kg/m^3, and ν = 0.2.

The response of cohesive behavior in the enriched elements in the model is specified. The maximum principal stress failure criterion is selected for damage initiation, and an energy-based damage evolution law based on a power law fracture criterion is selected for damage propagation. The relevant material data are as follows: σ_{max} = 10.45 MPa, G_{1C} = 19.58 N/m, G_{IIC} = 19.58 N/m, G_{IIIC} = 19.58 N/m, a_m = 1.0, a_n = 1.0, and a_0 = 1.0.

Results and discussion

Figure 2 shows the crack profile when t = 1.3 × 10^{-3} s. The crack propagates at an angle of 58°, which is in reasonable agreement with the experimental result of 60°.

Input files

crackprop_tpb_xfem_2d_dyn.inp	Two-dimensional plane strain model under mixed-mode impact loading.
crackprop_tpb_xfem_c3d4_dyn.inp	Three-dimensional tetrahedron model under mixed-mode impact loading.
crackprop_tpb_xfem_c3d8r_dyn.inp	Three-dimensional brick model with reduced integration under mixed-mode impact loading.

Python scripts

crackprop_tpb_xfem_c3d8r_dyn.py	Script to generate the three-dimensional brick model with reduced integration under mixed-mode impact loading in Abaqus/CAE simulated using the XFEM-based cohesive segments approach.

References

- John, P., and S. P. Shah, "Mixed Mode Fracture of Concrete Subjected to Impact Loading," *Journal of Structural Engineering (ASCE)*, vol. 116 585–602, 1990.

Figures

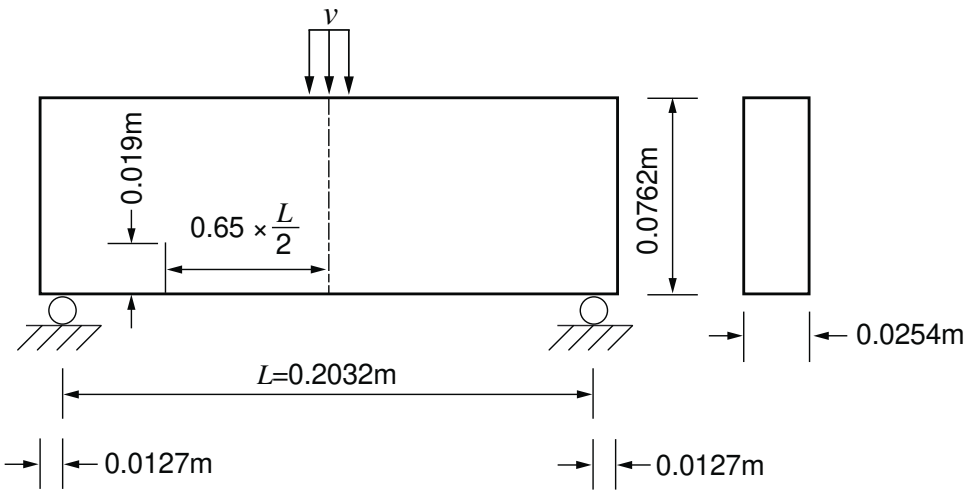


Figure 1: Model geometry of the beam with an offset edge crack (dimensions in m).

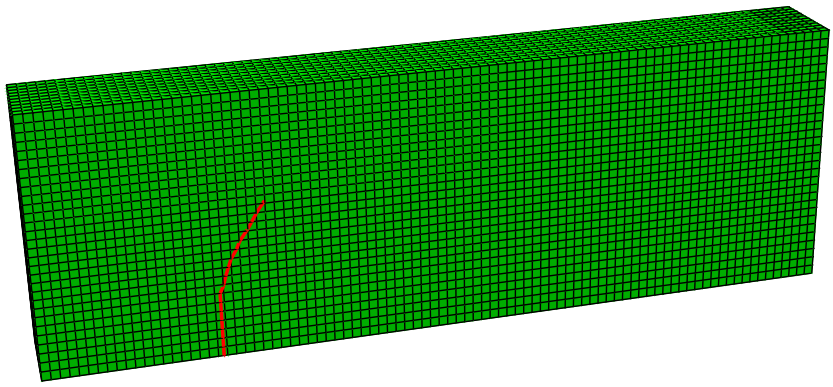


Figure 2: Crack profile at t = 1.3 × 10^{-3} s.

Dynamic shear failure of a single-edge notch simulated using XFEM

Products: Abaqus/Standard

Overview

This example verifies and illustrates the use of the extended finite element method (XFEM) in Abaqus/Standard to predict dynamic crack propagation of a plate with an edge crack.
The specimen is subjected to a high rate shear impact loading. The crack paths and crack initiation angles presented are compared to the experimental results of *Kalthoff and Winkler (1987)*.

Problem description

A plate with a single edge crack is studied. The specimen, shown in *Figure 1*, has dimensions L = 0.003 m and W = 0.0015 m and an initial crack with length a = 0.0015 m. The lower part of the specimen is subjected to an impulse load along the horizontal direction, which is modeled as a prescribed velocity:

$$v(t) = \begin{cases} \dfrac{v_0 t}{t_{ramp}}, & t_{ramp} \geq t \\ v_0, & \text{otherwise;} \end{cases}$$

where v_0 = 25 m/s and t_{ramp} = 1.0×10^{-7} s.

The material data for the bulk material properties in the enriched elements are E = 3.24 GPa, ρ = 1190 kg/m^3, and ν = 0.35.

The response of cohesive behavior in the enriched elements in the model is specified. The maximum principal stress failure criterion is selected for damage initiation, and an energy-based damage evolution law based on a power law fracture criterion is selected for damage propagation. The relevant material data are as follows: σ_{max} = 100.0 MPa, G_{1C} = 700 N/m, G_{IIC} = 700 N/m, G_{IIIC} = 700 N/m, a_m = 1.0, a_n = 1.0, and a_0 = 1.0.

Results and discussion

Figure 2 shows the crack profile when t = 6.0×10^{-6} s. The crack propagates at an angle of 62°, which is in reasonable agreement with the experimental result of 65°.

Input files

crackprop_shear_xfem_3d_dyn.inp	Three-dimensional brick model with reduced integration under shear impact loading.

Python scripts

crackprop_shear_xfem_3d_dyn.py	Script to generate the three-dimensional brick model with reduced integration under shear impact loading in Abaqus/CAE simulated using the XFEM-based cohesive segments approach.

References

- Kalthoff, J. K., and S. Winkler, "Failure Mode Transition at High Rates of Loading," *Proceedings of the International Conference on Impact Loading and Dynamic Behavior of Materials* 185–195, 1987.

Figures

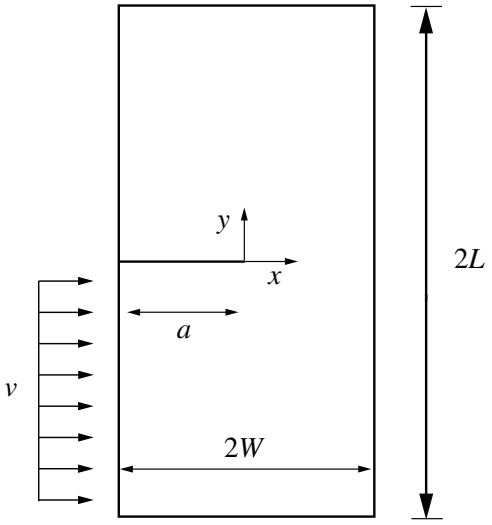


Figure 1: Model geometry of the plate with an edge crack subjected to shear impact loading.

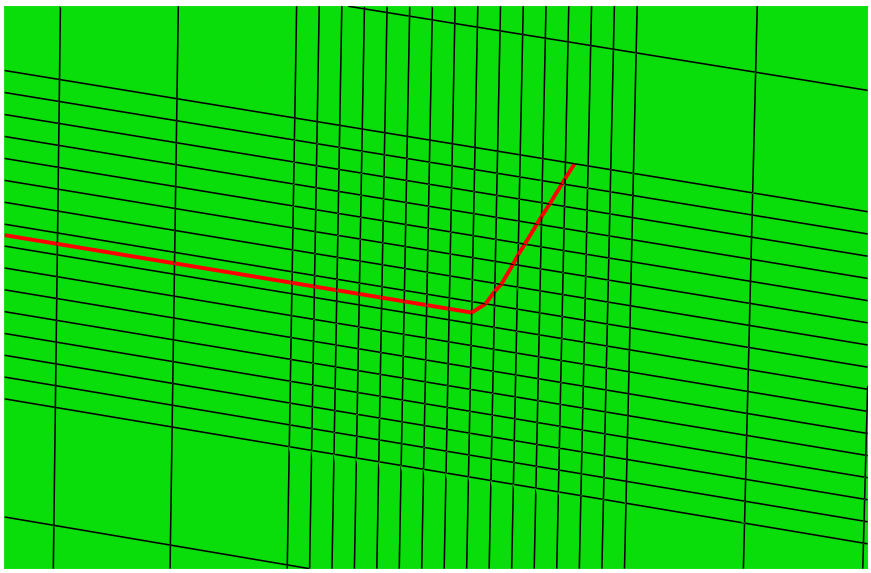


Figure 2: Crack profile at $t = 6.0 \times 10^{-6}$ s.

Propagation of hydraulically driven fracture using XFEM

Products: Abaqus/Standard

Overview

This example verifies and illustrates the use of the extended finite element method (XFEM) in Abaqus/Standard to model hydraulically driven crack propagation in a permeable porous medium.
The results presented are compared to those obtained using cohesive elements.

Problem description

The domain of the problem considered in this example is a 1 m thick circular slice of oil-bearing rock with a small initial crack modeled. The domain has a diameter of 160 m.

Due to symmetry only one-half of the domain is modeled. *Figure 1* shows the finite element model. The rock is modeled with two-dimensional CPE4P elements. Four different mesh discretizations of the same geometry are studied. The second mesh has two times as many elements as the original one, the third mesh has four times as many elements as the original one, while the fourth mesh has eight times as many elements as the original one. Similar analyses are performed using C3D8P, C3D8RP, C3D4P, CAX4P, CAX4RP, and CPE4RP elements.

A two-step analysis is performed, and crack propagation is simulated. 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 next step represents the hydraulic fracture stage where the main volume of fluid is being injected into the well. Flow at a rate of 1.0×10^{-3} m^3 per second is injected in the target formation in the model, and the enriched elements adjacent to the wellbore are defined as initially open to permit entry of fluid. The duration of this stage is 140 seconds.

The material data for the bulk material properties in the enriched elements are $\boldsymbol{E = 1.294}$ GPa and $\boldsymbol{\nu} = 0.25$.

The response of cohesive behavior in the enriched elements in the model is specified. The maximum principal stress failure criterion is selected for damage initiation, and an energy-based damage evolution law based on a BK law criterion is selected for damage propagation. The relevant material data are as follows: $\boldsymbol{\sigma_{max} = 0.32}$ MPa, $\boldsymbol{G_{IC} = 28} \times 10^3$ N/m, $\boldsymbol{G_{IIC} = 28} \times 10^3$ N/m, $\boldsymbol{G_{IIIC} = 28} \times 10^3$ N/m, and $\boldsymbol{\eta = 2.284}$.

Tangential and normal flow are both modeled in the fracture zone of the enriched elements. The following parameters are specified:

- Gap flow is specified as Newtonian with a viscosity of 1×10^{-6} kPa•s, roughly the viscosity of water.
- Fluid leakoff is specified as 5.879×10^{-10} m/(kPa•s).

Results and discussion

The flow injected during Step 2, the pumping stage, initiated and grew a crack extending outward from the wellbore. *Figure 2* shows the resulting geometry of the fracture at the end of the 140-second pumping period. *Figure 3* shows the pore pressure as a function of time at the crack mouth with different mesh discretizations obtained using the XFEM method compared with those obtained using cohesive elements. This figure clearly indicates that the fluid flow has stabilized. The results obtained using the XFEM method converge with mesh refinement and agree well with those obtained using cohesive elements. The differences are largely due to the fact that the pore pressure averaged over an element is output when the XFEM method is used, while the pore

pressure is output at the node when cohesive elements are used. A similar history of the leak-off flow rate at the crack mouth is illustrated in *Figure 4*.

Input files

hydrfract_xfem_cpe4p.inp	Abaqus/Standard two-dimensional plane strain model using CPE4P elements.
hydrfract_xfem_cpe4p_2.inp	Same as hydrfract_xfem_cpe4p.inp but with twice as many elements.
hydrfract_xfem_cpe4p_3.inp	Same as hydrfract_xfem_cpe4p.inp but with four times as many elements.
hydrfract_xfem_cpe4p_4.inp	Same as hydrfract_xfem_cpe4p.inp but with eight times as many elements.
hydrfract_xfem_cpe4rp.inp	Two-dimensional plane strain model using CPE4RP elements.
hydrfract_xfem_cax4p.inp	Axisymmetric model using CAX4P elements.
hydrfract_xfem_cax4rp.inp	Axisymmetric model using CAX4RP elements.
hydrfract_xfem_c3d8rp.inp	Three-dimensional model using C3D8RP elements.

Figures

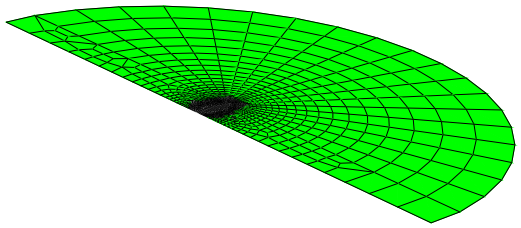


Figure 1: Near wellbore mesh.

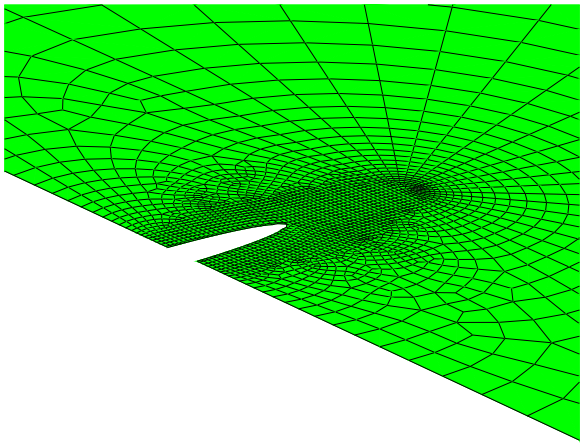


Figure 2: Fracture geometry following the injection stage.

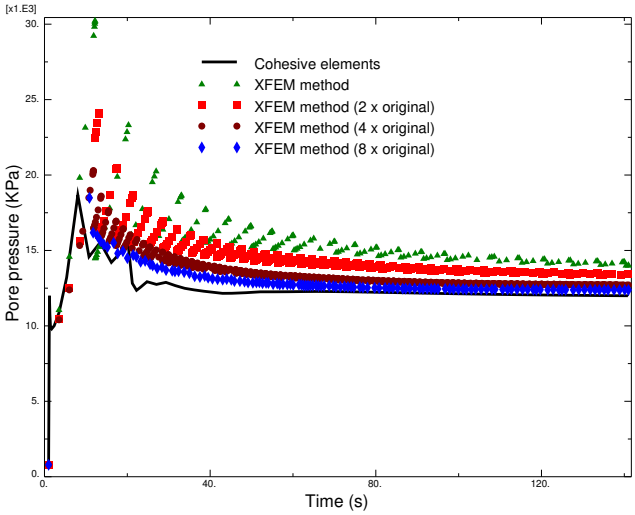


Figure 3: History of the pore pressure at the crack mouth.

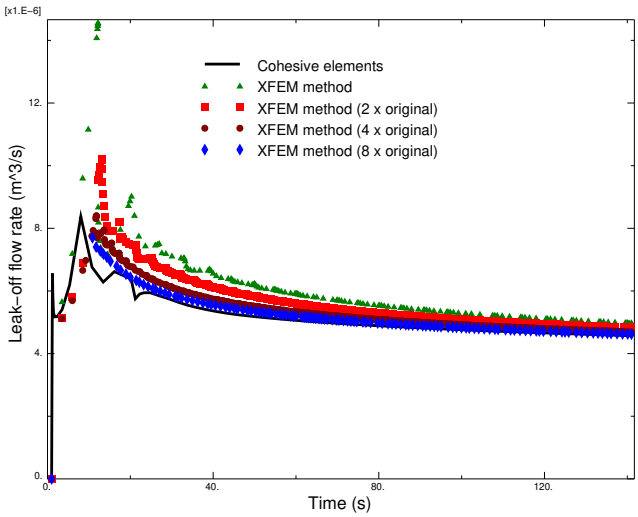


Figure 4: History of the leak-off flow rate at the crack mouth.

Element Tests

In this section:

- *Continuum elements*
- *Infinite elements*
- *Structural elements*
- *Connector elements*
- *Special-purpose elements*

Continuum elements

In this section:

- *Torsion of a hollow cylinder*
- *Geometrically nonlinear analysis of a cantilever beam*
- *Cantilever beam analyzed with CAXA and SAXA elements*
- *Two-point bending of a pipe due to self weight: CAXA and SAXA elements*
- *Cook's membrane problem*

Torsion of a hollow cylinder

Products: Abaqus/Standard

Overview

This problem verifies and illustrates the use of axisymmetric solid elements with twist in Abaqus.
An Airy stress function provided by Fung (1977) is used to obtain the stress components in the cylindrical coordinate system for this problem.

Problem description

The physical problem consists of a hollow cylinder fixed at one end in the translational degrees of freedom. Opposing twists, equal in magnitude, are applied to the inner and outer diameters of the hollow cylinder. *Figure 1* shows the model geometry used in this analysis. Most of the meshes used for this problem are uniform; however, since the stresses are a quadratic function of the radius, mesh refinement in regions of high strain gradients is suggested. Two input files have refined meshes near the inner diameter of the hollow cylinder with appropriate mesh refinement multi-point constraints. Some input files use a kinematic coupling to couple the boundary surfaces to reference nodes; the twists are applied to the reference nodes. All axisymmetric solid elements with twist are tested in this problem; coupled temperature-displacement elements are tested in a steady-state coupled temperature-displacement step with arbitrarily applied boundary temperatures and a coefficient of thermal expansion, $\alpha = 0$, to prevent coupling with the mechanical solution. Mesh convergence studies have not been performed. To test the use of substructures in problems involving axisymmetric elements with twist, the problem is also solved using substructuring with the entire model treated as a single substructure.

Results and discussion

The derived analytical solution for this problem, based on an Airy stress function provided by Fung (1977), is

$$\phi = (1+\nu)\, C_1/Er^2 - C_2,$$
$$\sigma_{r\theta} = -C_1/r^2,$$

where ϕ is the twist angle in radians, $C_1 = 2.05128 \times 10^4$, and $C_2 = -1.6667 \times 10^{-2}$.

Figure 2 shows the variation of the shear stress, $\sigma_{r\theta}$, with respect to the radius of the hollow cylinder for the CGAX8 element model (where $\sigma_{r\theta} = -$S13) compared to the analytical solution. The results for all elements agree well with the analytical solution. The results from the substructure analysis match the results that are obtained when substructures are not used.

Input files

torsholcyl_cgax3_couplingk.inp	CGAX3 model using the **COUPLING* and **KINEMATIC* options to impose the twist.
torsholcyl_cgax3_kincoupl.inp	CGAX3 model using the **KINEMATIC COUPLING* option to impose the twist.
torsholcyl_cgax3h.inp	CGAX3H model.
torsholcyl_cgax4.inp	CGAX4 model.
torsholcyl_cgax4h.inp	CGAX4H model.
torsholcyl_cgax4ht.inp	CGAX4HT model.

torsholcyl_cgax4r_meshrefine.inp	CGAX4R model with mesh refinement.
torsholcyl_cgax4rh.inp	CGAX4RH model.
torsholcyl_cgax4rh_eh.inp	CGAX4RH model with enhanced hourglass control.
torsholcyl_cgax4t.inp	CGAX4T model.
torsholcyl_cgax6.inp	CGAX6 model.
torsholcyl_cgax6h.inp	CGAX6H model.
torsholcyl_cgax6m.inp	CGAX6M model.
torsholcyl_cgax6mh.inp	CGAX6MH model.
torsholcyl_cgax8.inp	CGAX8 model.
torsholcyl_cgax8_substruct.inp	CGAX8 model using substructuring.
torsholcyl_cgax8_substruct_gen1.inp	Substructure generation referenced in the analysis torsholcyl_cgax8_substruct.inp.
torsholcyl_cgax8h.inp	CGAX8H model.
torsholcyl_cgax8h_neohook.inp	CGAX8H model with a neo-Hookean incompressible hyperelastic material.
torsholcyl_cgax8ht.inp	CGAX8HT model.
torsholcyl_cgax8r_meshrefine.inp	CGAX8R model with mesh refinement.
torsholcyl_cgax8rh.inp	CGAX8RH model.
torsholcyl_cgax8rht.inp	CGAX8RHT model.
torsholcyl_cgax8rt.inp	CGAX8RT model.
torsholcyl_cgax8t.inp	CGAX8T model.

References

- Fung, Y. C., Foundations of Solid Mechanics, Prentice-Hall Inc., New Jersey, 1977.

Figures

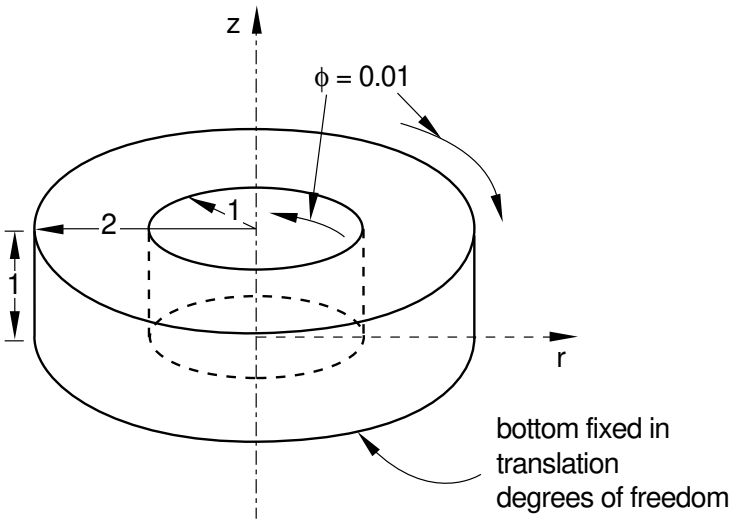


Figure 1: Torsion of a hollow cylinder.

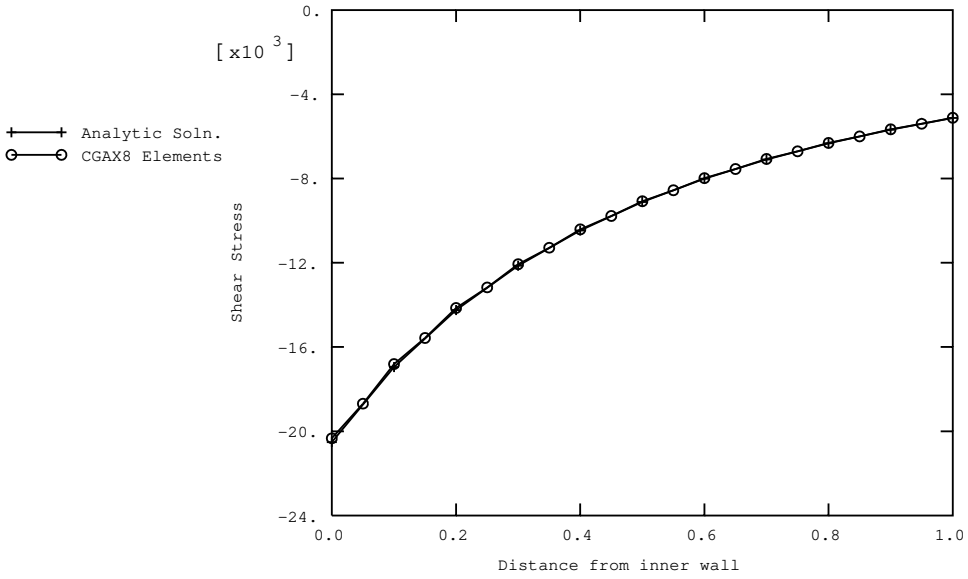


Figure 2: Variation of shear stress with radius.

Geometrically nonlinear analysis of a cantilever beam

Products: Abaqus/Standard

Overview

These examples verify the accuracy of several of the beam and continuum elements in Abaqus in large-displacement geometrically nonlinear analyses.
The first problem is a cantilever loaded at its tip by a load of constant vertical direction. The second is the problem of a cantilever with a tip moment.

Problem description

The cantilever is 10 m long. A circular pipe cross-section of outer radius 0.1 m and wall thickness 0.01 m is used for closed-section beam elements so that the beam is moderately slender ($L/R = 100$). This type of problem becomes considerably more difficult numerically as the slenderness ratio increases. The standard beam elements in Abaqus should have little difficulty up to slenderness ratios of 1000, but above that the hybrid elements are usually required. However, for this example we have elected to use hybrid elements throughout since they result in better convergence for problems where the beam remains inextensible—in this case, the moment load problem, and to a certain degree, the transverse load problem. Young's modulus is chosen as 100 MPa. The model consists of three elements for the transverse load case and either 10 or 20 elements for the moment load problem, depending on whether linear or quadratic elements are used.

Open-section beam elements of type B31OS, B32OS, B31OSH, and B32OSH can be used with ARBITRARY, GENERAL, and I section types. To test these elements, the cross-section is defined as an I-beam with a height of 200 mm, a width of 100 mm, and web and flange thicknesses of 10 mm. An additional degree of freedom (7) defines the amplitude of the cross-sectional warping. Since the problems considered here are two-dimensional, the torsion and warping considerations are irrelevant.

For the continuum elements the solid section thickness and height are chosen to be 100 mm and 147.8 mm, respectively, so that the section moment of inertia is identical to that of the pipe section defined above. The section moment of inertia determines the bending behavior, which is the dominant deformation for both cases since axial deformation is insignificant. We measure the performance of the incompatible mode elements, CPS4I and C3D8I; the second-order elements, CPS8, CPS8R, CPS6, CPS6M, C3D10, C3D10HS, and C3D10M; and the first-order elements, CPS4, CPS4R, C3D8, and C3D8R. The corresponding second-order three-dimensional elements, C3D20 and C3D20R, are not included but can be expected to provide about the same results as the CPS8 and CPS8R elements, respectively. A single layer of elements is used in all the meshes. The reduced-integration, linear elements, CPS4R and C3D8R, when used with enhanced hourglass control give good results that match the results obtained using incompatible mode elements since both are based on an assumed strain formulation. Two mesh types are used for the transverse load case: a coarse mesh of 1×10 for the first-order elements and 1×5 for the second-order elements and a fine mesh of 1×20 for the first-order elements and 1×10 for the second-order elements. For the moment load case a 1×40 mesh is used with the element types tested (CPS4I, CPS4R, CPS8R, CPS6, CPS6M, and C3D8R). A fine mesh is needed to converge to the correct result; for this loading case a 1×20 mesh results in a noticeably stiffer response.

For comparison purposes, reduced-integration linear membrane (M3D4R), shell (S4R), and continuum shell (SC8R) elements are used with enhanced hourglass control for the transverse load and moment load cases with the same meshes as for CPS4R elements. In addition, continuum shell meshes are provided for the transverse load and moment load cases with similar meshes as for C3D8R elements.

Loading

For the transverse load case a total vertical load of 269.35 N is applied at the tip of the cantilever, which causes the tip to deflect more than 8 m.

For the moment load problem two different methods of applying the load to the beam tip are used. In the first a moment of 3384.78 N-m is applied to the end of the closed-section pipe beam and a moment of 2873.09 N-m is applied to the open-section I-beam. In the second method, which is most commonly used for prescribing rotations of more than 2π radians, a constant angular velocity of 12.5664 rad/time is prescribed at the tip. Since the problem under consideration is a static analysis, Abaqus interprets the angular velocity in terms of the normalized time used for incrementation. An amplitude reference is used to keep the angular velocity constant. The beam will wind around itself twice with either the applied moment or the prescribed angular velocity.

Boundary Conditions provides details of the method illustrated here to prescribe rotations of more than 2π radians.

For the continuum elements the moment is applied through a distributing coupling constraint. The distributing coupling constraint is used to couple the nodes at the cantilever tip to a reference node placed at the tip. The moment is applied to this reference node resulting in a force-couple at the bottom and top nodes of the cantilever tip.

Results and discussion

Displacement plots for the transverse load case are shown for the coarse and fine meshes in *Figure 1* and *Figure 2*. *Figure 3* and *Figure 4* plot the displacement histories of the tip of the cantilever for the various elements. For both coarse and fine meshes using B22H, CPS8R, CPS4I, and C3D8I elements the displacements compare well with the exact solution for the inextensible beam, as given by Bisshopp and Drucker (1945). The displacements for CPS8, CPS6, CPS6M, C3D10, C3D10HS, and C3D10M improve significantly with the fine mesh, indicating good convergence. As expected, the linear full-integration elements, CPS4 and C3D8, give very stiff responses.

For the moment load problem the deformed shapes of the beams for elements B21H, CPS4I, CPS6, CPS6M, and CPS8R, at various increments throughout the step, are shown in *Figure 5* and *Figure 6*. The analytical solution for this problem is $ML/EI = 2\pi n$, where n is the number of times the rod will wind around itself. In our problem we have $n = 2$, as shown by the final deformed shapes. These analyses can be extended, by minor modifications to the data, to several other interesting large-displacement beam problems (see Love, 1944).

The reduced-integration, linear elements (CPS4R, C3D8R, M3D4R, S4R, and SC8R) with enhanced hourglass control give good results that match the results of incompatible mode elements for both load cases.

Input files

nlgeocantilever_b22h_tload.inp	Transverse load case using element B22H.
nlgeocantilever_b32h_tload.inp	Transverse load case using element B32H.
nlgeocantilever_b31osh_tload.inp	Transverse load case using element B31OSH.
nlgeocantilever_b21h_mload.inp	Moment load case using an applied moment and element B21H.
nlgeocantilever_b31h_mload.inp	Moment load case using a prescribed rotation and element B31H.
nlgeocantilever_b32osh_mload.inp	Moment load case using an applied moment and element B32OSH.
nlgeocantilever_c3d8_coarse.inp	Transverse load case using element C3D8; coarse mesh. The transverse load is applied via DCOUP3D elements instead of the **COUPLING* option.
nlgeocantilever_c3d8_fine.inp	Transverse load case using element C3D8; fine mesh.

nlgeocantilever_c3d8i_coarse.inp	Transverse load case using element C3D8I; coarse mesh. The transverse load is applied via DCOUP3D elements instead of the **COUPLING* option.
nlgeocantilever_c3d8i_fine.inp	Transverse load case using element C3D8I; fine mesh. The transverse load is applied via DCOUP3D elements instead of the **COUPLING* option.
nlgeocantilever_c3d8r_coarse_eh.inp	Transverse load case using element C3D8R; coarse mesh. The transverse load is applied via DCOUP3D elements instead of the **COUPLING* option.
nlgeocantilever_c3d8r_fine_eh.inp	Transverse load case using element C3D8R; fine mesh.
nlgeocantilever_c3d10_coarse.inp	Transverse load case using element C3D10; coarse mesh.
nlgeocantilever_c3d10_fine.inp	Transverse load case using element C3D10; fine mesh.
nlgeocantilever_c3d10hs_coarse.inp	Transverse load case using element C3D10HS; coarse mesh.
nlgeocantilever_c3d10hs_fine.inp	Transverse load case using element C3D10HS; fine mesh.
nlgeocantilever_c3d10m_coarse.inp	Transverse load case using element C3D10M; coarse mesh.
nlgeocantilever_c3d10m_fine.inp	Transverse load case using element C3D10M; fine mesh.
nlgeocantilever_cps4_coarse.inp	Transverse load case using element CPS4; coarse mesh. The transverse load is applied via DCOUP2D elements instead of the **COUPLING* option.
nlgeocantilever_cps4_fine.inp	Transverse load case using element CPS4; fine mesh.
nlgeocantilever_10cps4i_tload.inp	Transverse load case using element CPS4I.
nlgeocantilever_20cps4i_tload.inp	Transverse load case using element CPS4I.
nlgeocantilever_cps4r_coarse_eh.inp	Transverse load case using element CPS4R; coarse mesh. The transverse load is applied via DCOUP2D elements instead of the **COUPLING* option.
nlgeocantilever_cps4r_fine_eh.inp	Transverse load case using element CPS4R; fine mesh.
nlgeocantilever_cps6_coarse.inp	Transverse load case using element CPS6; coarse mesh.
nlgeocantilever_cps6_fine.inp	Transverse load case using element CPS6; fine mesh.
nlgeocantilever_cps6m_coarse.inp	Transverse load case using element CPS6M; coarse mesh. The transverse load is applied via DCOUP2D elements instead of the **COUPLING* option.
nlgeocantilever_cps6m_fine.inp	Transverse load case using element CPS6M; fine mesh.
nlgeocantilever_cps8_coarse.inp	Transverse load case using element CPS8; coarse mesh.
nlgeocantilever_cps8_fine.inp	Transverse load case using element CPS8; fine mesh.
nlgeocantilever_cps8r_coarse.inp	Transverse load case using element CPS8R; coarse mesh. The transverse load is applied via DCOUP2D elements instead of the **COUPLING* option.
nlgeocantilever_cps8r_fine.inp	Transverse load case using element CPS8R; fine mesh.
nlgeocantilever_m3d4r_coarse_eh.inp	Transverse load case using element M3D4R; coarse mesh. The transverse load is applied via DCOUP3D elements instead of the **COUPLING* option.
nlgeocantilever_m3d4r_fine_eh.inp	Transverse load case using element M3D4R; fine mesh.
nlgeocantilever_s4r_coarse_eh.inp	Transverse load case using element S4R; coarse mesh. The transverse load is applied via DCOUP3D elements instead of the **COUPLING* option.
nlgeocantilever_s4r_fine_eh.inp	Transverse load case using element S4R; fine mesh.

nlgeocantilever_sc6r_fine.inp	Transverse load case using element SC6R; fine mesh. The transverse load is applied via the **COUPLING* option.
nlgeocantilever_sc8r_fine.inp	Transverse load case using element SC8R; fine mesh. The transverse load is applied via the **COUPLING* option.
nlgeocantilever_cps4i_mload.inp	Moment load case using an applied moment and element CPS4I.
nlgeocantilever_cps4r_mload_eh.inp	Moment load case using an applied moment and element CPS4R.
nlgeocantilever_cps6_mload.inp	Moment load case using element CPS6. The moment is applied via DCOUP2D elements instead of the **COUPLING* option.
nlgeocantilever_cps6m_mload.inp	Moment load case using element CPS6M.
nlgeocantilever_cps8r_mload.inp	Moment load case using element CPS8R.
nlgeocantilever_c3d8r_mload_eh.inp	Moment load case using element C3D8R.
nlgeocantilever_m3d4r_mload_eh.inp	Moment load case using an applied moment and element M3D4R.
nlgeocantilever_s4r_mload_eh.inp	Moment load case using an applied moment and element S4R.
nlgeocantilever_sc6r_mload.inp	Moment load case using element SC6R. The moment is applied via the **COUPLING* option.
nlgeocantilever_sc8r_mload.inp	Moment load case using element SC8R. The moment is applied via the **COUPLING* option.
nlgeocantilever_sc8r_mload_eh.inp	Moment load case using element SC8R with enhanced hourglass control. The moment is applied via the **COUPLING* option.

References

- Bisshopp, R. E., and D. C. Drucker, "Large Deflection of Cantilever Beams," *Quarterly of Applied Mathematics*, vol. 3, no. 1, 1945.
- Love, A. E. H., A Treatise on the Mathematical Theory of Elasticity, Dover Publications, New York, 1944.

Figures

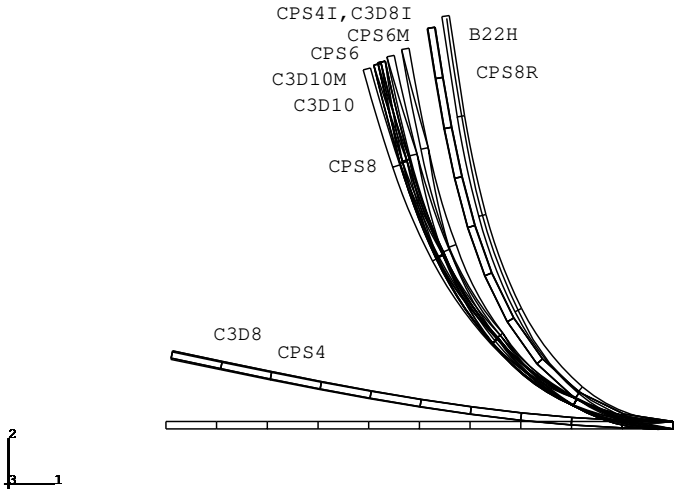


Figure 1: Displacement plots for coarse mesh of cantilever beam with transverse loading.

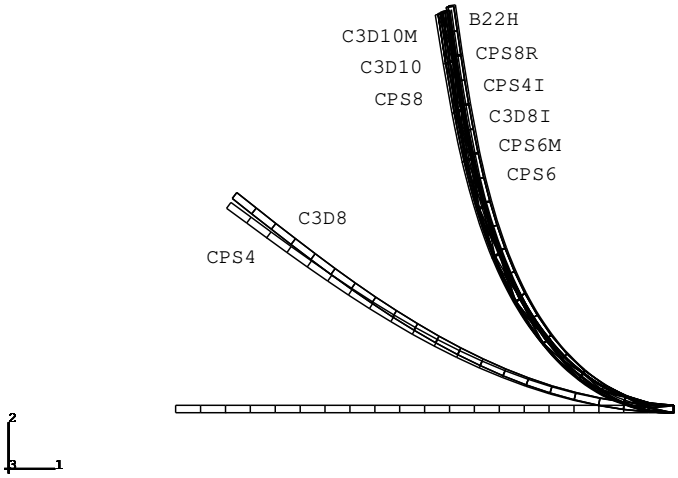


Figure 2: Displacement plots for fine mesh of cantilever beam with transverse loading.

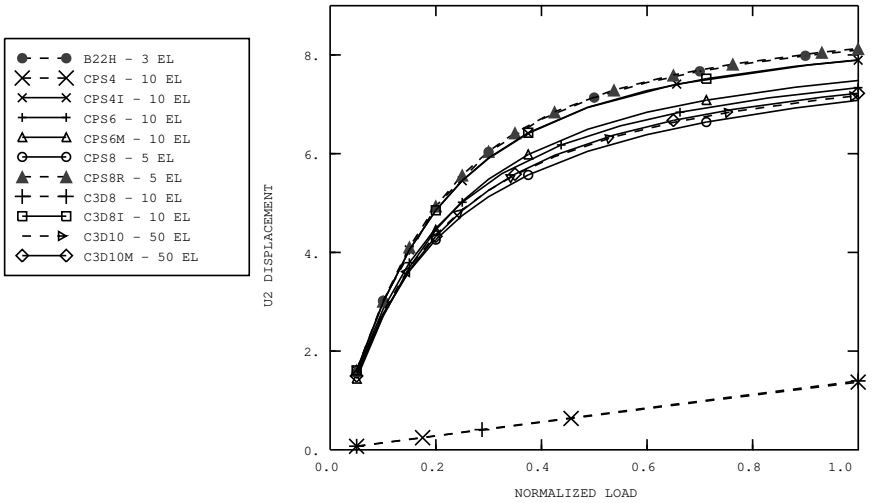


Figure 3: Displacement history of tip of coarse mesh of cantilever beam with transverse loading.

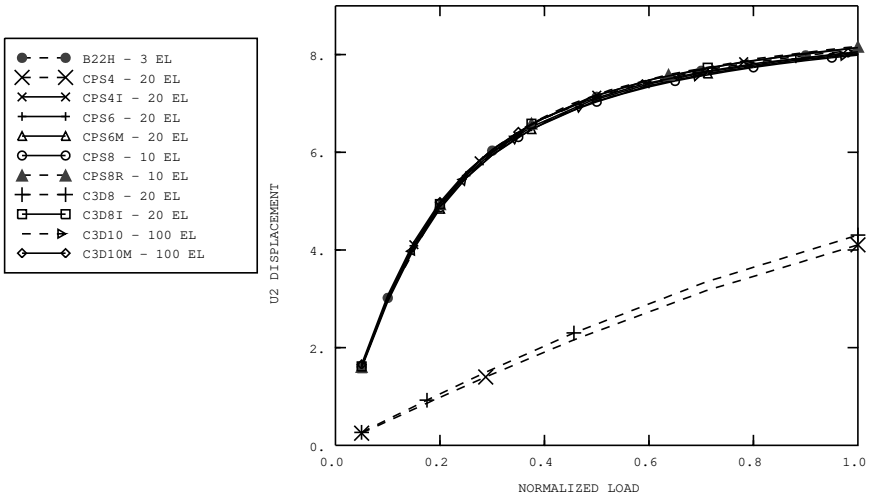
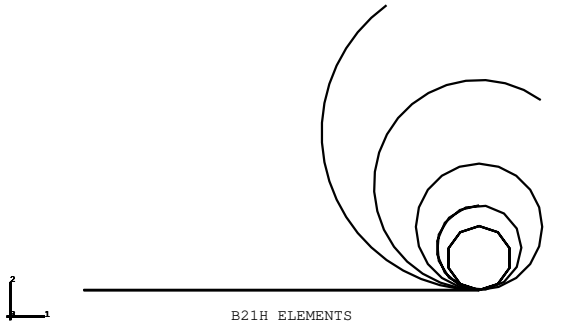
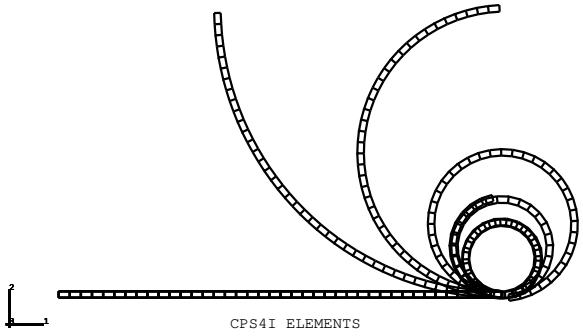


Figure 4: Displacement history of tip of fine mesh of cantilever beam with transverse loading.





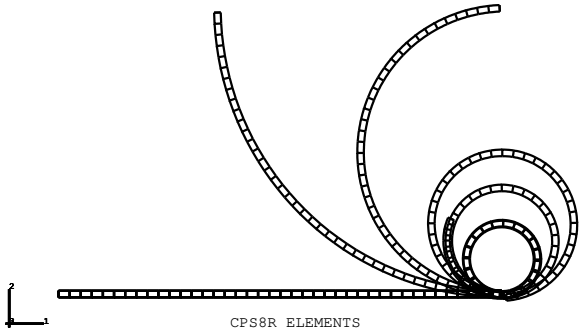


Figure 5: Displacement plots of cantilever with moment loading.

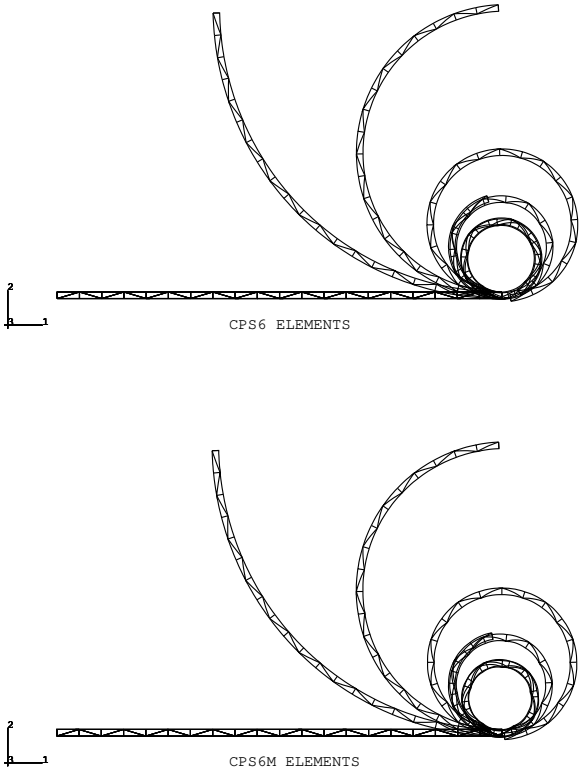


Figure 6: Displacement plots of cantilever with moment loading.

Cantilever beam analyzed with CAXA and SAXA elements

Products: Abaqus/Standard

Overview

This example verifies the accuracy of the second-order (8-node for CAXA and 3-node for SAXA) elements and the first-order (4-node for CAXA and 2-node for SAXA) elements in a single large-displacement case.
It also analyzes the frequency response of the tip-loaded cantilever beam modeled with CAXA and SAXA elements. In both cases the results are compared to those obtained with beam theory.

Abaqus provides a family of elements that are intended for the nonlinear analysis of structures that are initially axisymmetric but undergo nonlinear, nonaxisymmetric deformation. These elements, continuum elements named CAXA and shell elements named SAXA, are often used to model cylindrical or pipe structures in which the deformation is assumed to be symmetric with respect to $\theta = 0°$ and the bending of the structure occurs about the $\theta = 90°$-axis. The elements are written for arbitrarily large deformation in geometrically nonlinear analysis. The nonlinear capability is particularly useful for slender structures. The elements use standard isoparametric interpolation within the r–z plane, combined with Fourier interpolation with respect to θ. Up to four Fourier modes are allowed. As a simple large-deformation demonstration problem, the cantilever problem in *Geometrically nonlinear analysis of a cantilever beam* is solved with both CAXA and SAXA elements. The cantilever is loaded at its tip by a load of constant direction.

Problem description

The cantilever, a pipe 100 units long, has a cross-section with outer radius 1.2675 and wall thickness 0.2. This pipe is moderately slender ($L/R = 78.9$). This type of problem becomes considerably more difficult numerically as the slenderness ratio increases. Young's modulus is chosen as 30×10^6, and Poisson's ratio is 0.3. The motion of the pipe axis is entirely in a plane, so any of the CAXA or SAXA elements would be suitable except for those elements using only one Fourier mode in the θ-direction. (Due to the finite rotation of the pipe, the projection of the cross-section on the r–z plane becomes an ellipse.) Since the Fourier modes are defined in a fixed r–z system, the use of second-order Fourier expansion (including ovalization) is the minimum required. The finite element model uses the second-order elements with 10 elements along the length (z-direction) of the pipe and one element in the r-direction for the CAXA model. The first-order SAXA model uses 20 elements. A finite element model using the first-order CAXA4n (n=2, 3, or 4) elements is expected to give a stiffer response as a result of shear locking. However, a model using the first-order elements with reduced integration and hourglass control, CAXA4Rn (n= 2, 3, or 4), is capable of giving a much more accurate response. Without any mesh convergence study, we solve the problem by using a 2×20 mesh of fully integrated first-order elements and a 4×40 mesh with the corresponding reduced-integration elements with hourglass control.

Loading and boundary conditions

The load on the tip of the cantilever is increased to a value of 20000, which causes the tip to deflect more than 75 units. CAXA elements have rigid body modes in both the global x- and z-directions. The rigid body mode in the z-direction is removed by fixing the z-displacement of node set `BASE` at the fixed end of the pipe. The rigid body mode in the x-direction is eliminated by fixing the r-displacements at the midside nodes located at the fixed end of the pipe. The ovalization of the fixed end is also restricted by these boundary conditions. All other cross-sectional planes can ovalize. The concentrated load is split in two, with half applied to midside nodes in each of the $\theta = 0°$ and $\theta = 180°$ planes on the loaded end of the pipe. To avoid any deformation through the wall thickness in the CAXA model due to the application of concentrated loads on the loaded end, the radial

displacements at the midside nodes are constrained to be equal to the average radial motion of the nodes at the inside and outside radii. This is accomplished with an equation constraint (*Linear Constraint Equations*).

The general loading step forms the base state for the frequency analysis step that follows. In the frequency analysis step the load and boundary conditions are maintained as defined in the previous step.

Results and discussion

Figure 1 shows the progressive deformation of the pipe modeled with element type CAXA82. The results for the CAXA elements, in terms of the motion of the tip of the cantilever, are shown in *Figure 2*, where they are compared to the beam solution obtained with the B22 beam elements. It is apparent that the displacement solutions with the CAXA8*n*, CAXA8R*n*, and CAXA4R*n* (*n*=2, 3, or 4) elements predict almost precisely the results obtained by the model using the B22 beam elements. As expected, the fully integrated, first-order CAXA models have a much stiffer response, while the counterpart elements with reduced integration and hourglass control give more accurate results. The results for the SAXA elements are shown in *Figure 3*, where once again the results are compared to the B22 solution. For clarity, only the SAXA22 results are plotted since all elements SAXA1*n* and SAXA2*n* (*n*=2, 3, or 4) produce nearly identical results.

The frequency response of the tip-loaded cantilever beam modeled with CAXA8*n*, CAXA8R*n*, and CAXA4R*n* (*n*=2, 3, or 4) elements is very close to that obtained from the model using B22 beam elements. The natural frequencies for the fully integrated, first-order CAXA elements are higher because of the stiffer response of these elements.

Input files

cantilevercaxasaxa_caxa42.inp	CAXA42 element model.
cantilevercaxasaxa_caxa43.inp	CAXA43 element model.
cantilevercaxasaxa_caxa44.inp	CAXA44 element model.
cantilevercaxasaxa_caxa4r2.inp	CAXA4R2 element model.
cantilevercaxasaxa_caxa4r3.inp	CAXA4R3 element model.
cantilevercaxasaxa_caxa4r4.inp	CAXA4R4 element model.
cantilevercaxasaxa_caxa82.inp	CAXA82 element model.
cantilevercaxasaxa_caxa83.inp	CAXA83 element model.
cantilevercaxasaxa_caxa84.inp	CAXA84 element model.
cantilevercaxasaxa_caxa8r2.inp	CAXA8R2 element model.
cantilevercaxasaxa_caxa8r3.inp	CAXA8R3 element model.
cantilevercaxasaxa_caxa8r4.inp	CAXA8R4 element model.
cantilevercaxasaxa_saxa12.inp	SAXA12 element model.
cantilevercaxasaxa_saxa13.inp	SAXA13 element model.
cantilevercaxasaxa_saxa14.inp	SAXA14 element model.
cantilevercaxasaxa_saxa22.inp	SAXA22 element model.
cantilevercaxasaxa_saxa23.inp	SAXA23 element model.
cantilevercaxasaxa_saxa24.inp	SAXA24 element model.

Figures

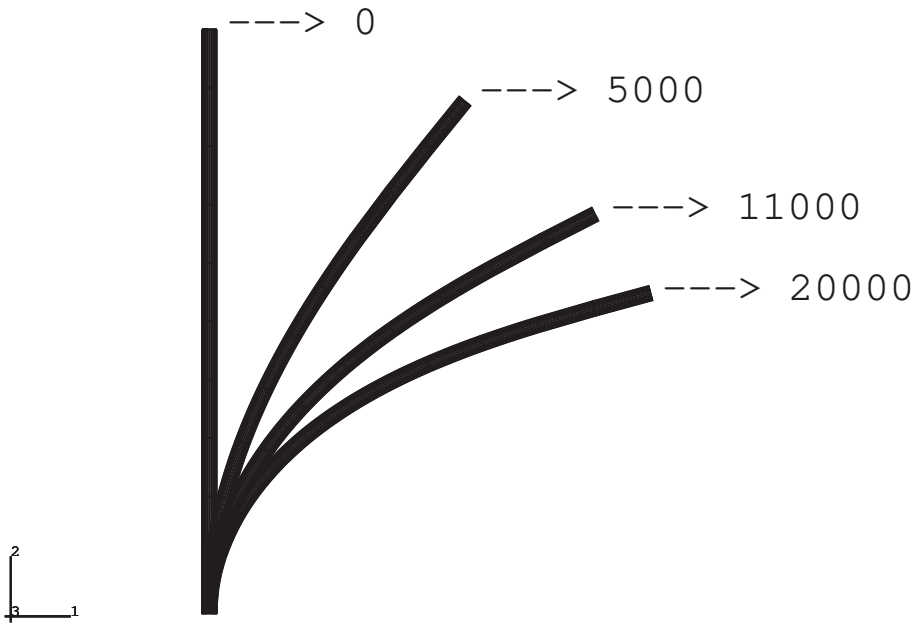


Figure 1: Progressive deformation of pipe.

LINE	ABSCISSA VARIABLE	ORDINATE VARIABLE
1	ux-b22	p-b22
	(*+1.0E+00)	(*+1.0E+00)
2	ux-caxa42	p-caxa42
	(*+1.0E+00)	(*+1.0E+00)
3	ux-caxa43	p-caxa43
	(*+1.0E+00)	(*+1.0E+00)
4	ux-caxa44	p-caxa44
	(*+1.0E+00)	(*+1.0E+00)
5	ux-caxa4r2	p-caxa4r2
	(*+1.0E+00)	(*+1.0E+00)
6	ux-caxa4r3	p-caxa4r3
	(*+1.0E+00)	(*+1.0E+00)
7	ux-caxa4r4	p-caxa4r4
	(*+1.0E+00)	(*+1.0E+00)
8	ux-caxa82	p-caxa82
	(*+1.0E+00)	(*+1.0E+00)
9	ux-caxa83	p-caxa83
	(*+1.0E+00)	(*+1.0E+00)
10	ux-caxa84	p-caxa84
	(*+1.0E+00)	(*+1.0E+00)
11	ux-caxa8r2	p-caxa8r2
	(*+1.0E+00)	(*+1.0E+00)
12	ux-caxa8r3	p-caxa8r3
	(*+1.0E+00)	(*+1.0E+00)
13	ux-caxa8r4	p-caxa8r4
	(*+1.0E+00)	(*+1.0E+00)

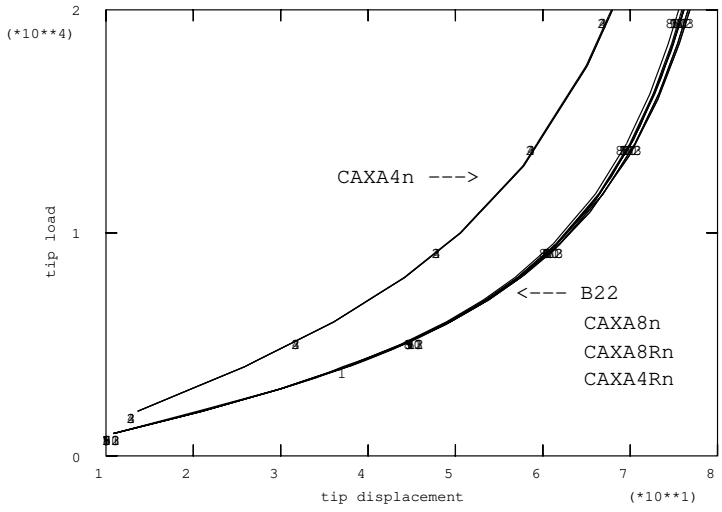


Figure 2: Comparison of load-displacement curves for CAXA elements.

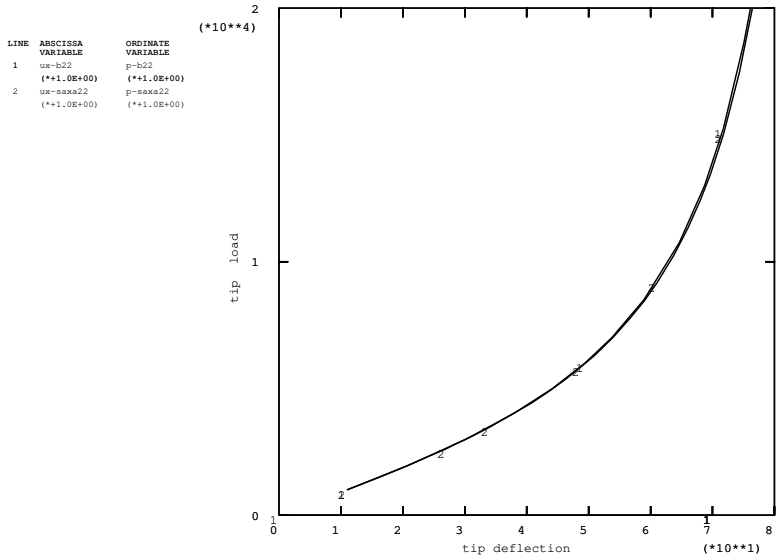


Figure 3: Comparison of load-displacement curves for SAXA elements. (Only SAXA22 is shown since all elements SAXA1*n* and SAXA2*n* for *n*=2, 3, or 4 give identical results.)

Two-point bending of a pipe due to self weight: CAXA and SAXA elements

Products: Abaqus/Standard

Overview

This example verifies the two-point bending of a pipe due to self weight for CAXA and SAXA elements.

Problem description

The material is linear elastic with a Young's modulus of 207 GPa, a Poisson's ratio of 0.3, and a weight density of 0.15 MN/m^3. $u_z = 0$ on the $z = 0$ plane; $u_r = 0$ at point B. A gravity force is considered to be acting in the positive *x*-direction.

CAXA mesh

In the analyses that test the CAXA elements, only the symmetric half of the structure is considered. The models using full-integration and reduced-integration second-order elements employ one element through the thickness and seven elements along the pipe, with two equal-sized elements along CB and five equal-sized elements along BA. The models using the lower-order fully integrated elements employ twice as many elements in both the radial and axial directions. The models using the lower-order reduced-integration elements employ four times as many elements in both the radial and axial directions.

SAXA mesh

In the analyses that test the SAXA elements, only the symmetric half of the structure is considered. Second-order element models use seven elements along the pipe, with two equal-sized elements along CB and five equal-sized elements along BA. First-order element models use 14 elements along the pipe, with four equal-sized elements along CB and 10 equal-sized elements along BA.

Reference solution

This problem provides a test on the body force type BX for the axisymmetric solid elements with nonlinear, asymmetric deformation. The reference solution is obtained from the analysis of an equivalent three-dimensional model using the 20-node brick element C3D20. The three-dimensional mesh employs one element in the radial direction, six elements in the circumferential direction, and seven elements along the pipe. The input file for the reference solution is *eref3ksg.inp*.

Results and discussion

The solutions are linear, small-displacement solutions and are in good agreement with the reference solution.

Input files

ecnssfsg.inp	CAXA41 elements.
ecnssrsg.inp	CAXA4R1 elements.
ecnsshsg.inp	CAXA4H1 elements.
ecnssysg.inp	CAXA4RH1 elements.

ecntsfsg.inp	CAXA42 elements.
ecntsrsg.inp	CAXA4R2 elements.
ecntshsg.inp	CAXA4H2 elements.
ecntsysg.inp	CAXA4RH2 elements.
ecnusfsg.inp	CAXA43 elements.
ecnusrsg.inp	CAXA4R3 elements.
ecnushsg.inp	CAXA4H3 elements.
ecnusysg.inp	CAXA4RH3 elements.
ecnvsfsg.inp	CAXA44 elements.
ecnvsrsg.inp	CAXA4R4 elements.
ecnvshsg.inp	CAXA4H4 elements.
ecnvsysg.inp	CAXA4RH4 elements.
ecnwsfsg.inp	CAXA81 elements.
ecnwsrsg.inp	CAXA8R1 elements.
ecnwshsg.inp	CAXA8H1 elements.
ecnwsysg.inp	CAXA8RH1 elements.
ecnxsfsg.inp	CAXA82 elements.
ecnxsrsg.inp	CAXA8R2 elements.
ecnxshsg.inp	CAXA8H2 elements.
ecnxsysg.inp	CAXA8RH2 elements.
ecnysfsg.inp	CAXA83 elements.
ecnyshsg.inp	CAXA8H3 elements.
ecnwpfsg.inp	CAXA8P1 elements.
ecnwprsg.inp	CAXA8RP1 elements.
ecnxpfsg.inp	CAXA8P2 elements.
ecnxprsg.inp	CAXA8RP2 elements.
ecnypfsg.inp	CAXA8P3 elements.
ecnyprsg.inp	CAXA8RP3 elements.
esnssxsg.inp	SAXA11 elements.
esntsxsg.inp	SAXA12 elements.
esnusxsg.inp	SAXA13 elements.
esnvsxsg.inp	SAXA14 elements.
esnwsxsg.inp	SAXA21 elements.
esnxsxsg.inp	SAXA22 elements.
esnysxsg.inp	SAXA23 elements.
esnzsxsg.inp	SAXA24 elements.

Figures

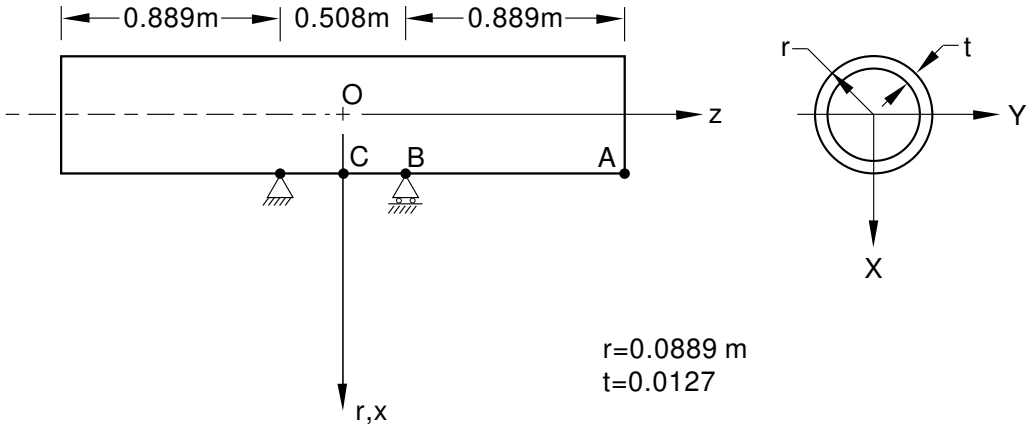


Figure 1: Two-point bending of a pipe.

Cook's membrane problem

Products: Abaqus/Standard

Overview

"Cook's membrane problem" verifies combined bending and shear response with moderate distortion.

The problem consists of a tapered panel of nearly incompressible hyperelastic material clamped on one side while a shearing load is applied on the opposite side. The problem is solved with both plane strain and three-dimensional hybrid elements. A mesh convergence study is performed.

Problem description

The panel measures 44 mm on the left-hand side and 16 mm on the right-hand side. The two sides are parallel and 48 mm apart. The top right-hand corner is initially 16 mm above the top left-hand corner, as shown in *Figure 1*. The panel is clamped on its left edge and loaded in shear with a 1.0 N force on its right edge. It is composed of nearly incompressible neo-Hookean material with parameters C_{10}=0.4 and D_1=2.5 × 10^{-4} to enable comparison with the results of Simo and Armero (1992) and Brink and Stein (1996). When solving the problem with three-dimensional elements, a thickness of 5 mm is assumed and symmetry boundary conditions are applied on the *X*–*Y* plane faces.

Hybrid elements are used due to the nearly incompressible nature of the material. The plane strain problem is solved with meshes composed of triangular (CPE3H, CPE6H, and CPE6MH) and quadrilateral (CPE4H, CPE4IH, and CPE4RH) elements. The three-dimensional problem is solved with meshes composed of tetrahedral (C3D4H, C3D10H, and C3D10MH) and hexahedral (C3D8H, C3D8IH, and C3D8RH) elements. A mesh convergence study is performed: for each element type the problem is solved with increasingly refined meshes of 4, 8, 16, and 32 elements on each side of the panel. For three-dimensional elements, a fixed number of two elements through the thickness is considered.

Loading

The concentrated load of 1.0 N is applied through a distributing coupling constraint. The distributing coupling constraint is used to couple the nodes on the right edge of the panel to a reference node where the load is applied.

Results and discussion

Figure 2 shows the deformed configuration corresponding to a mesh consisting of 32 C3D4H elements per side. *Figure 3* shows the vertical displacement of the top right-edge node versus the number of elements per side for the plane strain elements considered, while *Figure 4* shows a similar plot for the three-dimensional elements. The CPE3H element results are particularly stiff due to their poor bending and nearly incompressible behavior. All other elements converge to the same result as the meshes are refined; however, we note the stiff response in coarse meshes exhibited by the full-integration quadrilateral (CPE4H) elements, the full-integration hexahedral (C3D8H) elements, and the linear tetrahedral (C3D4H) elements. Reduced-integration quadrilateral (CPE4RH) and hexadedral (C3D8RH) elements show good agreement with the converged solution even for coarse meshes; however, the predicted stress in the reduced-integration coarse meshes is compromised by the reduced number of sampling points. When an accurate stress distribution is required, quadratic or incompatible mode elements should be used.

Input files

cook_2d.inp	Solution using plane strain elements.
cook_3d.inp	Solution using three-dimensional elements.

References

- Simo, J. C., and F. Armero, "Geometrically Nonlinear Enhanced Strain Mixed Methods and the Method of Incompatible Modes," *International Journal for Numerical Methods in Engineering*, vol. 33, pp. 1413–1449, 1992.
- Brink, U., and E. Stein, "On some Mixed Finite Element Methods for Incompressible and Nearly Incompressible Finite Elasticity," *Computational Mechanics*, vol. 19, pp. 105–119, 1996.

Figures

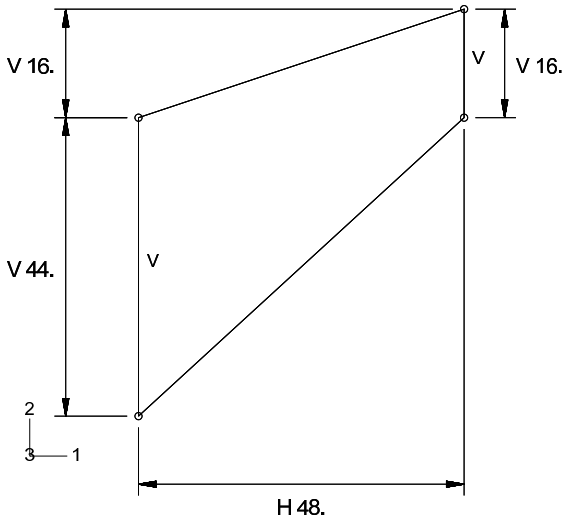


Figure 1: Cook's membrane problem: initial geometry.

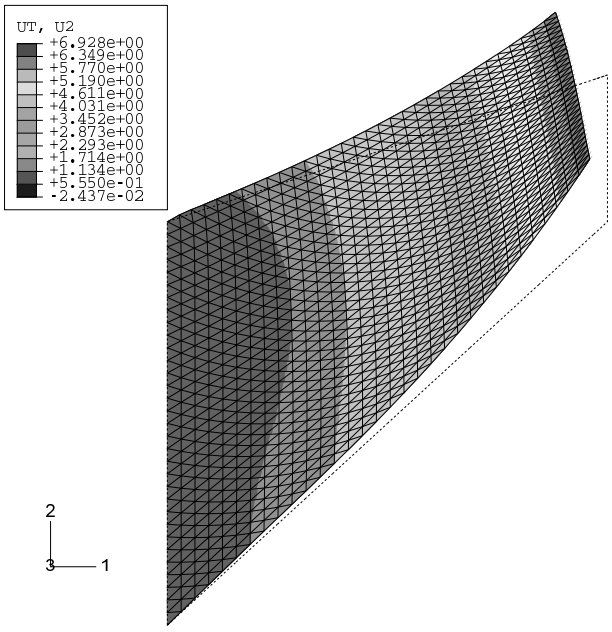


Figure 2: Final deformed configuration (C3D4H elements, 32 elements per side).

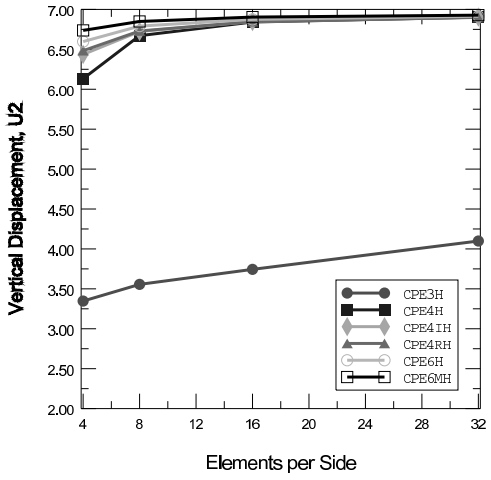


Figure 3: Mesh convergence study for plane strain elements.

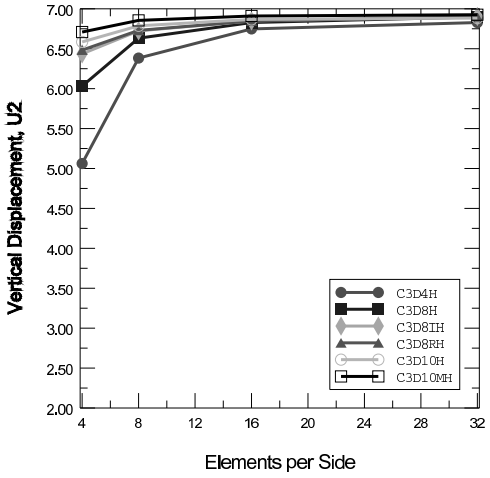


Figure 4: Mesh convergence study for three-dimensional elements.

Infinite elements

In this section:

- *Wave propagation in an infinite medium*
- *Infinite elements: the Boussinesq and Flamant problems*
- *Infinite elements: circular load on half-space*
- *Spherical cavity in an infinite medium*

Wave propagation in an infinite medium

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example verifies the effectiveness of the infinite element (quiet boundary) formulation in dynamic applications. The problem is similar to that analyzed by Cohen and Jennings (1983).

Problem description

The problem is an infinite half-space (plane strain is assumed) subjected to a vertical pulse line load (see *Figure 1*). A vertical plane of symmetry is used so that only half the configuration is meshed. Two load cases are considered: a vertical pulse load with a triangular amplitude variation (see *Figure 2*) and a vertical pulse load in the form of a 10 MHz raised-cosine function, $1 - \sin(\omega t/3)\cos(\omega t)$, with an amplitude of 1 GPa and a period of 0.3μs (see *Figure 9*). A raised-cosine function was chosen because its frequency content has a Gaussian distribution about its center frequency, ω.

Three meshes are used for load case 1: a small finite/infinite element (quiet boundary) mesh of 16 × 16 CPE4R finite elements plus 32 CINPE4 infinite elements, as shown in *Figure 3*; a small finite element mesh of 16 × 16 CPE4R elements, as shown in *Figure 4*; and an extended finite element mesh of 48 × 48 CPE4R elements, as shown in *Figure 5*. The results obtained using the small mesh including infinite element quiet boundaries are compared with those obtained using the extended mesh of finite elements only. Results obtained using the small mesh without the infinite element quiet boundaries are also given to show how the solution is affected by the reflection of the propagating waves. The mesh used for load case 2 consists of 180 × 107 CPE4R finite elements and 287 CINPE4 infinite elements. The finite element meshes are assumed to have free boundaries at the far field and will reflect the propagating waves, while the finite/infinite element meshes model the infinite domain and provide quiet boundaries that minimize reflection of propagating waves back into the mesh. Geometric nonlinearities are not significant in this problem and are ignored.

The material is assumed to be elastic with the following properties:

Property	Value
Young's modulus	73 GPa
Poisson's ratio	0.33
density	2842 kg/m^3

Material damping and artificial bulk viscosity are not included in the analyses. Based on these material properties, the speed of propagation of longitudinal waves in the material is approximately 6169.1 m/s, and the speed of propagation of shear waves is approximately 3107.5 m/s (see *Solid infinite elements*). Therefore, the longitudinal waves, which are predominant with the vertical pulse excitation, should reach the boundary of the small mesh used for load case 1 in about 0.324μs, the boundary of the extended mesh in about 0.97μs, and the boundary of the mesh used for load case 2 in about 0.77μs. Load case 1 analyses are run for 1.5μs, so that the waves are allowed to reflect into the finite element meshes that do not have quiet boundaries.

All analyses are performed with both Abaqus/Standard and Abaqus/Explicit.

Results and discussion

The results of load case 1 are shown in the form of time histories of vertical displacements at nodes 13, 103, and 601, as indicated on the meshes. *Figure 6* (node 13), *Figure 7* (node 103), and *Figure 8* (node 601) show the displacement responses. The wave reflection caused by the free boundaries in the small finite element mesh is evident, while the small finite/infinite element quiet boundary mesh largely succeeds in eliminating this reflection. The results obtained with Abaqus/Standard and Abaqus/Explicit agree well.

The wave pattern produced by a distributed load on an infinite half-space is shown in *Figure 1*. The majority of the energy introduced into the system by the loading is contained in the straight section of the longitudinal wave. The curved wave fronts and the surface waves are produced by the discontinuity at the edge of the distributed load. The same wave pattern can be identified in the deformed configurations of both load cases. In particular, the deformed configuration for load case 2 just prior to the longitudinal wave leaving the lower boundary of the mesh is shown in *Figure 10*. The contour plots of the vertical and horizontal displacements for load case 2 (see *Figure 11* and *Figure 12*, respectively) show the lower energy shear waves emanating from the edge of the distributed load more clearly.

The time histories of the whole model energies for load case 2 are shown in *Figure 13*. It can be seen that the kinetic and internal energies remain constant until the longitudinal wave reaches the infinite elements at the mesh boundary. The viscous dissipation time history represents the energy absorbed by the infinite elements. At 1.07μs, when the trailing end of the longitudinal pulse reaches the mesh boundary, most of the energy has been absorbed by the infinite elements. The last wave to exit the mesh should be the shear wave generated by the discontinuity at the edge of the distributed load that travels toward the symmetry axis. It will be reflected by the symmetry axes and travel toward the lower-right corner of the mesh. This should occur at 3.92μs. Any waves remaining in the mesh after this time are due to spurious wave reflection at the infinite boundaries. The kinetic energy associated with these waves is less than 0.2% of the total kinetic energy created by the pulse.

Figure 14 shows the vertical displacement responses of a node positioned 2 mm below the edge of the distributed load. The initial longitudinal pulse reaches the node in 0.324μs. Because the wave is not completely absorbed by the infinite elements, its reflection can be seen on its way back to the surface and again on its return from the surface after it is has been reflected. The response after 2.4μs is due to the shear wave reflecting off the symmetry axis. The shear wave traveling directly downward from the edge of the distributed load does not appear in this plot because its motion is completely in the horizontal direction.

Figure 15 shows the horizontal displacement responses of a node positioned 3.2 mm below the edge of the distributed load. The horizontal component of the longitudinal wave reaches the node in 0.519μs, while the slower traveling shear wave reaches the node at a time of 1.03μs. The response after approximately 1.7μs is due to spurious reflections of the longitudinal and shear waves from the lower boundary as well as the shear wave reflected from the symmetry axes. Again, the results obtained with Abaqus/Standard and Abaqus/Explicit agree well.

Input files

Abaqus/Standard input files

waveprop_fininfmesh.inp	Small finite/infinite element (quiet boundary) mesh of load case 1.
waveprop_smallfinmesh.inp	Small finite element mesh of load case 1.
waveprop_extdfinmesh.inp	Extended finite element mesh of load case 1.
waveprop_3d_fininfmesh.inp	Small finite/infinite element (quiet boundary) mesh of load case 1 in three dimensions.
waveprop_3d_smallfinmesh.inp	Small finite element mesh of load case 1 in three dimensions.
waveprop_3d_extdfinmesh.inp	Extended finite element mesh of load case 1 in three dimensions.

waveprop_prestatic.inp	Contains the analysis in waveprop_fininfmesh.inp preceded by a static step, which is used to verify statics followed by dynamics when using infinite elements.
waveprop_verssd.inp	Contains the analysis to verify **STEADY STATE DYNAMICS*, DIRECT when using infinite elements.
waveprop_cpe4r_std.inp	CPE4R mesh of load case 2.
waveprop_cpe3_std.inp	The same model meshed with CPE3 elements.
waveprop_ef1.inp	A file that contains the amplitude data for the 10 MHz raised cosine function. This file is read by the two input files listed above.

Abaqus/Explicit input files

waveprop_fininfmesh_exp.inp	Small finite/infinite element (quiet boundary) mesh of load case 1.
waveprop_smallfinmesh_exp.inp	Small finite element mesh of load case 1.
waveprop_extdfinmesh_exp.inp	Extended finite element mesh of load case 1.
waveprop_3d_fininfmesh_exp.inp	Small finite/infinite element (quiet boundary) mesh of load case 1 in three dimensions.
waveprop_3d_smallfinmesh_exp.inp	Small finite element mesh of load case 1 in three dimensions.
waveprop_3d_extdfinmesh_exp.inp	Extended finite element mesh of load case 1 in three dimensions.
waveprop_cpe4r.inp	CPE4R mesh of load case 2.
waveprop_cpe3.inp	The same model meshed with CPE3 elements.
waveprop_ef1.inp	A file that contains the amplitude data for the 10 MHz raised cosine function. This file is read by the two input files listed above.

References

- Cohen, M., and P. C. Jennings, “Silent Boundary Methods for Transient Analysis,” *Computational Methods for Transient Analysis*, Ed. T. Belytschko and T. R. J. Hughes, Elsevier, 1983.

Figures

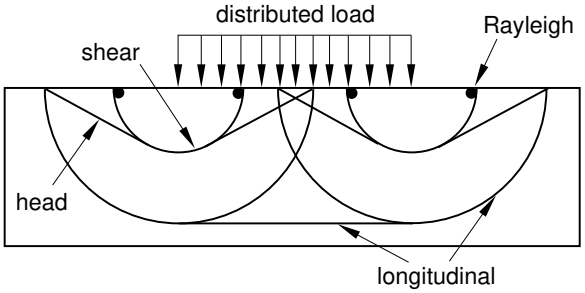


Figure 1: Wave pattern caused by a distributed load on an infinite half-space.

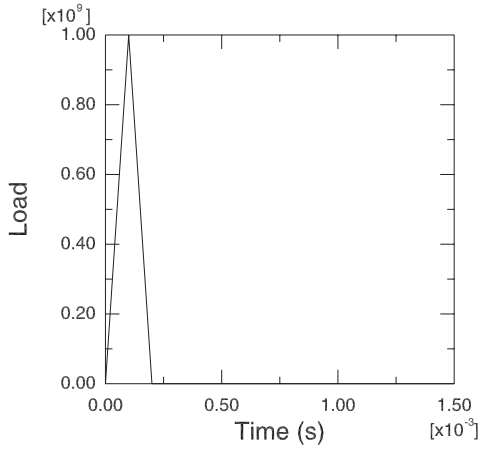


Figure 2: Triangular amplitude variation of load case 1.

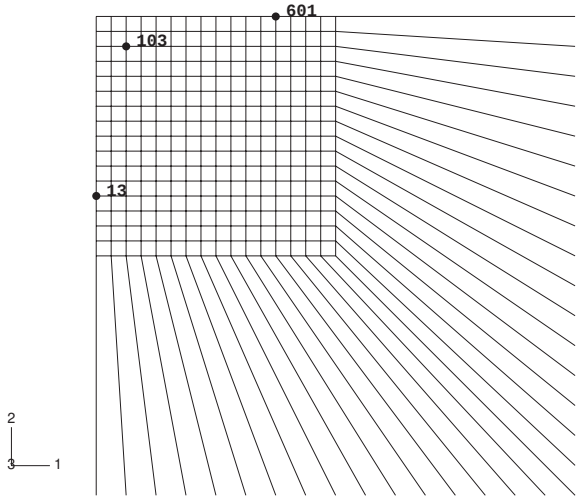


Figure 3: Small finite/infinite element mesh (quiet boundaries) of load case 1.

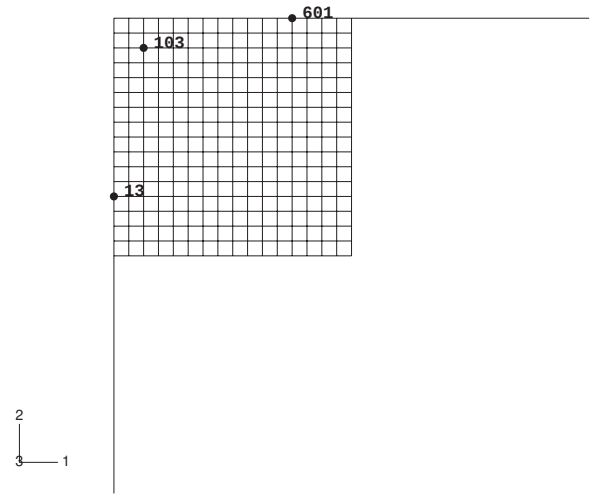


Figure 4: Small finite element mesh of load case 1.

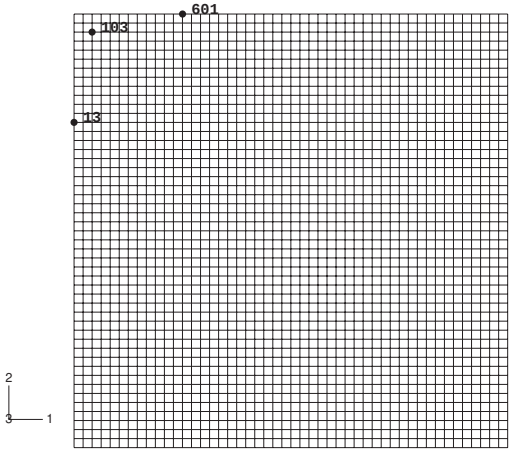


Figure 5: Extended finite element mesh of load case 1.

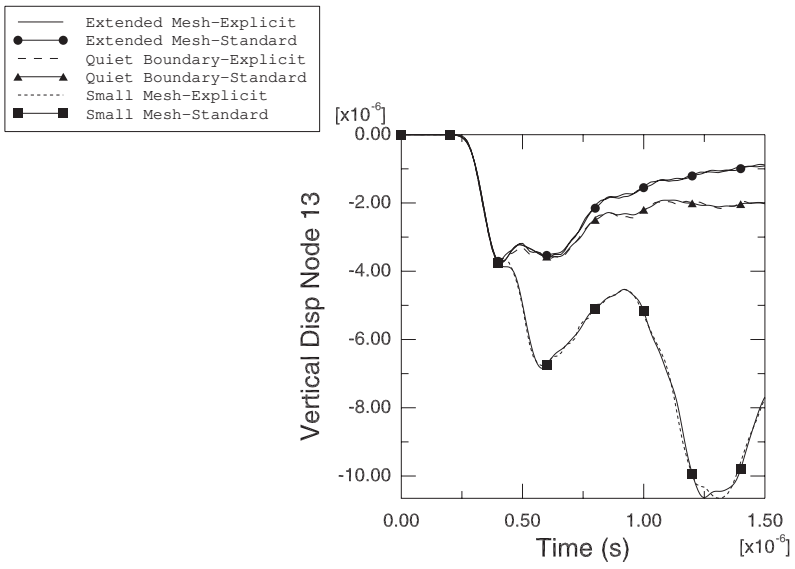


Figure 6: Vertical displacement responses at node 13 (load case 1).

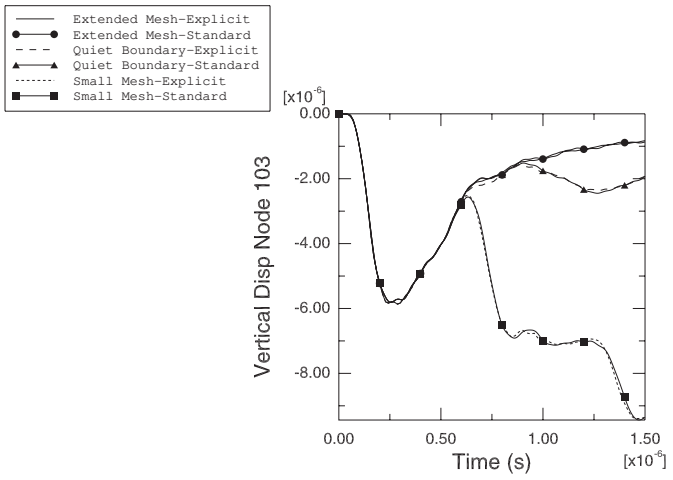


Figure 7: Vertical displacement responses at node 103 (load case 1).

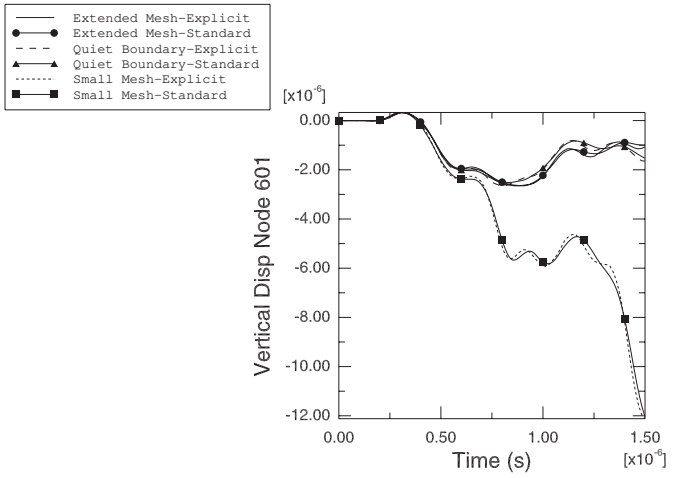


Figure 8: Vertical displacement responses at node 601 (load case 1).

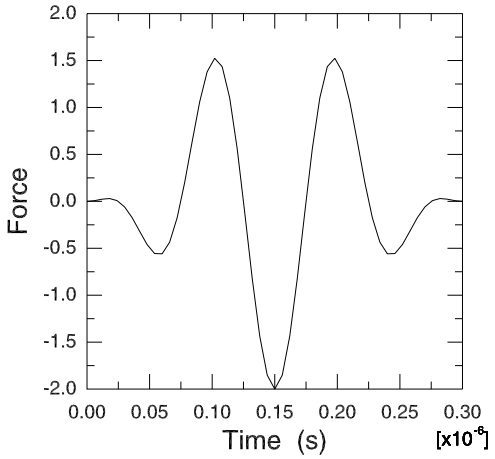


Figure 9: 10 MHz raised-cosine function used for load case 2.

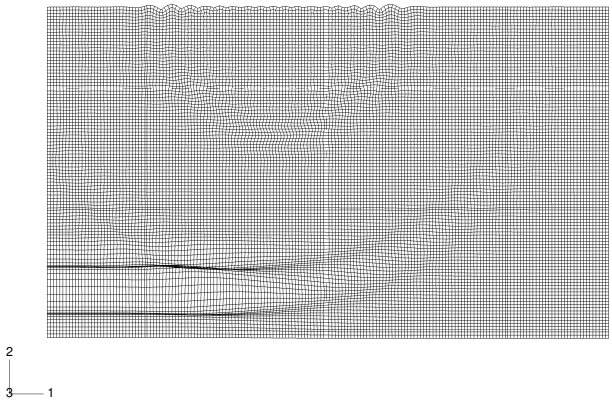


Figure 10: Deformed configuration prior to waves leaving the mesh boundary (load case 2, 0.81μs, displacement magnified by 75%).



Figure 11: Vertical displacement contour at 0.81μs (load case 2).

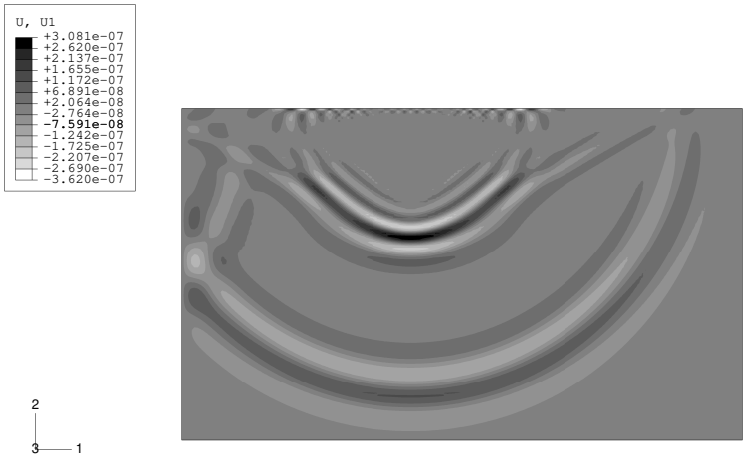


Figure 12: Horizontal displacement contour at 0.81μs (load case 2).

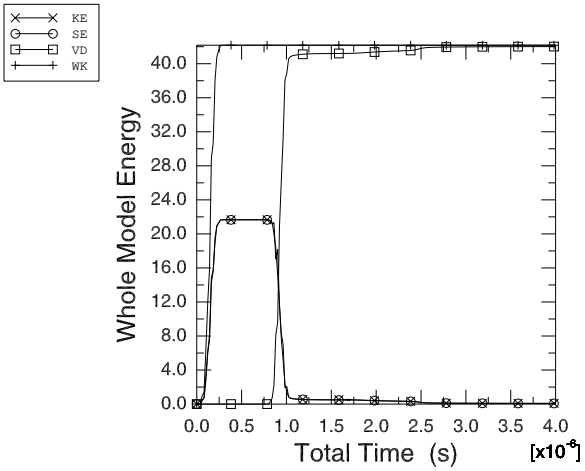


Figure 13: Whole model energy histories (load case 2).

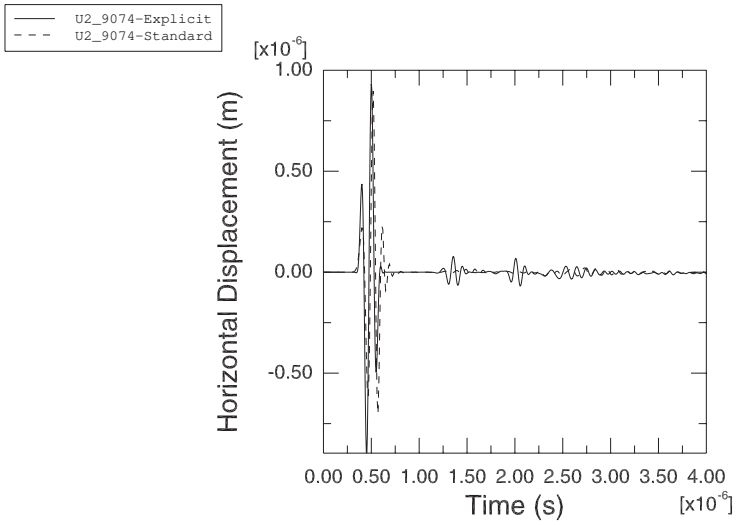


Figure 14: Vertical displacement response 2 mm below the edge of the load (load case 2).

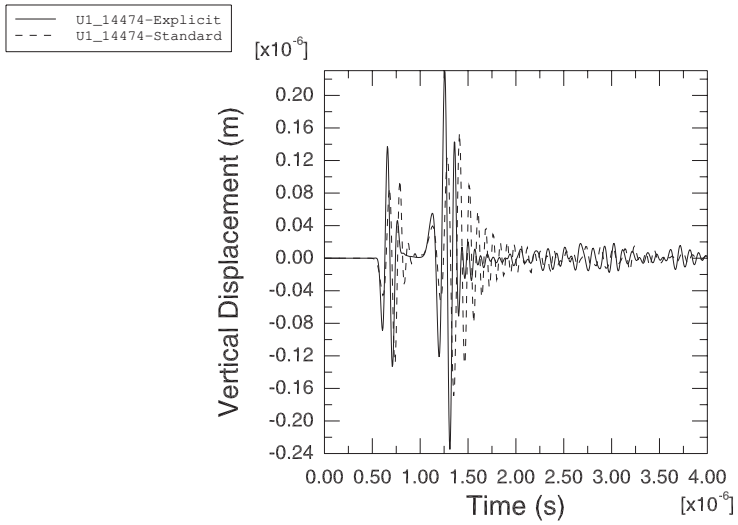


Figure 15: Horizontal displacement response 3.2 mm below the edge of the load (load case 2).

Infinite elements: the Boussinesq and Flamant problems

Products: Abaqus/Standard

Overview

This example verifies the performance of infinite elements in modeling the far-field domain.
The results from the problem of a point load on a half-space and a line load on a half-space are compared with the analytical solutions due to Boussinesq and Flamant (Timoshenko and Goodier, 1970), respectively. For comparison purposes results obtained using only finite elements are also given.

Problem description

For the Boussinesq problem of a point load on a half-space two mesh configurations are used. The infinite element mesh, *Figure 1*, is composed of 12 finite elements extending to a radius of 4.0, with 4 infinite elements modeling the far-field domain. The finite element mesh, *Figure 2*, is made up of 16 finite elements, truncated at a radius of 5.0, where fully fixed boundary conditions are applied.

The positioning of the second node in the infinite direction in the infinite elements is such that the first node is equidistant between the source of loading and the second node.

For the Flamant problem of a line load on a half-space, the same mesh configurations are used. In this case they are in plane strain, and a vertical plane of symmetry is used.

The material is linear elastic, with Young's modulus $E = 1.0$ and Poisson's ratio $\nu = 0.1$. A unit load is applied in both problems.

Results and discussion

Boussinesq's analytical solution for the problem of a point load on a half-space gives the vertical displacement as

$$w = \frac{P}{2\pi E}\left[(1+\nu)\,z^2\left(r^2+z^2\right)^{-\frac{3}{2}} + 2\left(1-\nu^2\right)\left(r^2+z^2\right)^{-\frac{1}{2}}\right],$$

where r and z are the radial and vertical distance from the point load, respectively. This equation clearly shows the $1/r$ singularity at the point of application of the load ($r = 0$). Here we compare the displacement variation along a vertical line beneath the point load where, for the given elastic properties,

$$w = \frac{0.49}{z}.$$

This analytical result is plotted in *Figure 3*, together with results obtained with the finite and infinite element models.

It is clear that the results obtained with the infinite element meshes show a significant improvement over the finite element meshes with the same number of elements, and that the infinite elements provide reasonable accuracy even with such relatively coarse modeling. In this case the load is a point load, so that the infinite elements can be focused on the pole of the solution. *Infinite elements: circular load on half-space* considers a distributed load, for which the infinite element mesh design is not as obvious.

Flamant's analytical solution for the problem of a line load on a half-space gives the vertical displacement along a vertical line beneath the line load as

$$w = \frac{2P}{\pi E} ln\left(\frac{d}{z}\right),$$

where d is an arbitrary large distance at which the displacement is assumed to be zero (see the discussion in *Infinite Elements*). Here, we have chosen to fix the far-field nodes on the infinite elements so that $d = 8.0$. This analytical result is plotted in *Figure 4*. The results obtained with the finite and infinite element models are also shown in this figure. Even though the infinite elements contain displacement interpolations in the infinite direction with terms of order $1/z, 1/z^2$ while the analytical solution is of a $ln(z)$ nature, they provide a significant improvement over the solutions obtained with finite elements only.

Input files

bousflamant_bous_cax4_cinax4.inp	First-order coupled finite/infinite element axisymmetric mesh.
bousflamant_bous_cax4_cinax4_po.inp	**POST OUTPUT* analysis.
bousflamant_bous_cax8r_cinax5r.inp	Second-order coupled finite/infinite element axisymmetric mesh.
bousflamant_flam_cpe4_cinpe4.inp	Plane strain Flamant problem; first-order coupled finite/infinite element axisymmetric mesh.
bousflamant_flam_cpe8r_cinpe5r.inp	Plane strain Flamant problem; second-order coupled finite/infinite element axisymmetric mesh.

References

- Timoshenko, S. P., and J. N. Goodier, Theory of Elasticity, McGraw-Hill, New York, 1970.

Figures

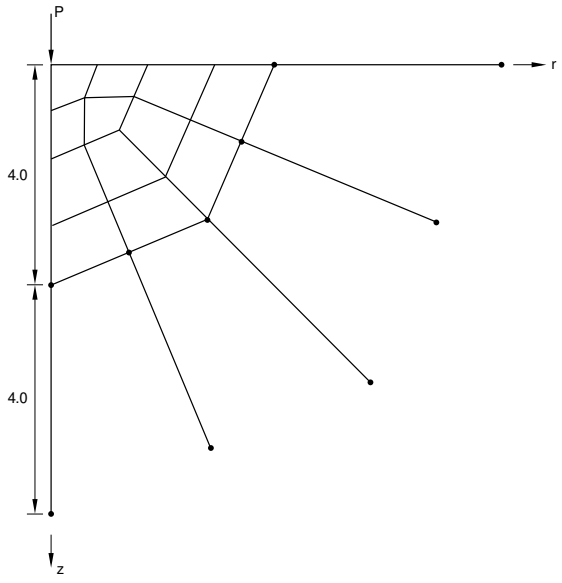


Figure 1: Finite/infinite element mesh.

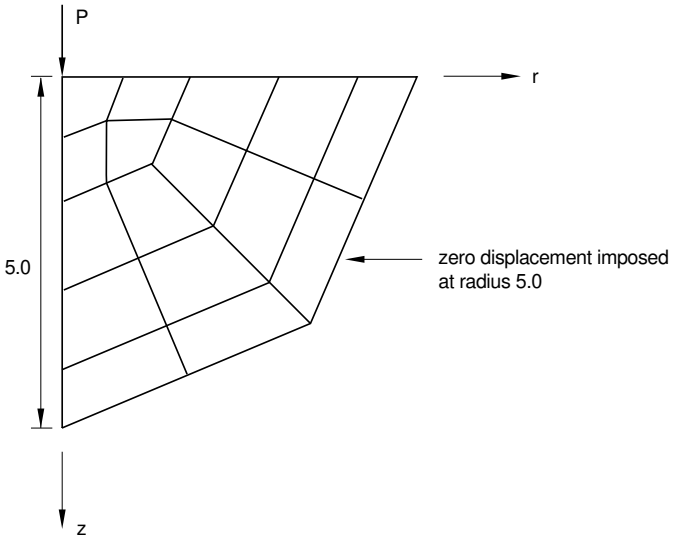


Figure 2: Mesh of finite elements only.

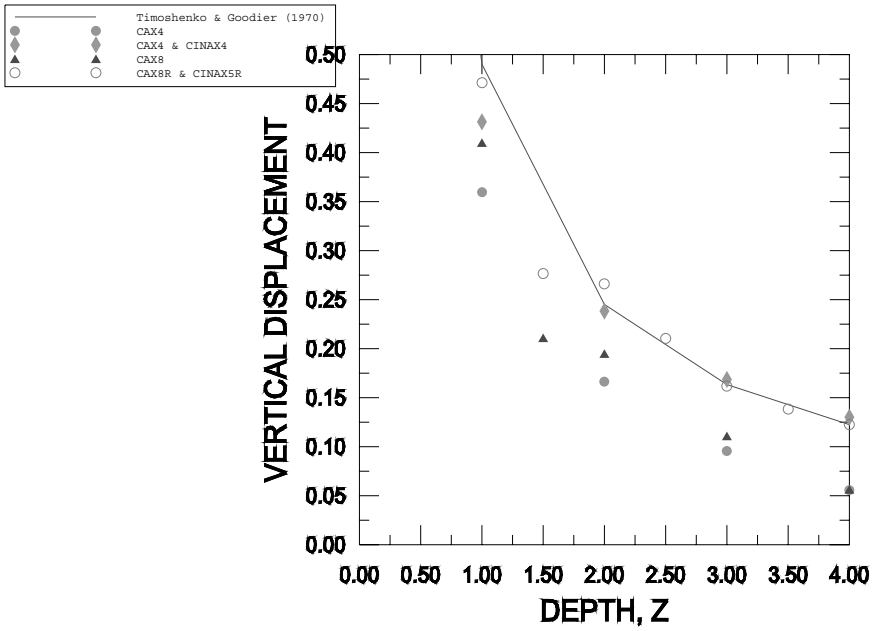


Figure 3: The Boussinesq problem: displacement results.

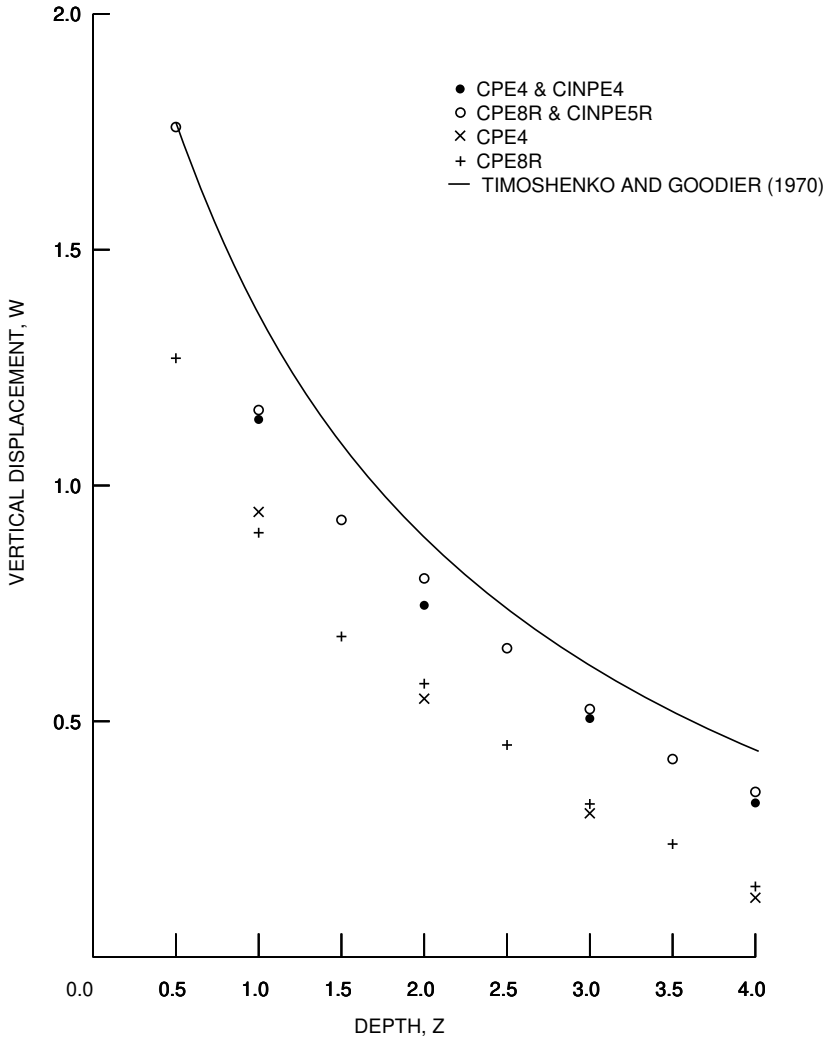


Figure 4: The Flamant problem: displacement results.

Infinite elements: circular load on half-space

Products: Abaqus/Standard

Overview

This example of an elastic half-space subjected to a uniform pressure load was suggested by Lynn and Hadid (1981) and examines the performance of different coupled finite/infinite element meshes. The results are compared to the analytical solution given by Timoshenko and Goodier (1970).
For comparison purposes a mesh of finite elements only is also used.

Problem description

The two finite/infinite element meshes used are shown in *Figure 1* and *Figure 2*. The mesh shown in *Figure 1* uses a radial configuration, with the finite elements of the type CAX8R. It extends to a radius of 3 m (10 ft), twice the extent of the load. The far field is modeled with four CINAX5R infinite elements. The second mesh is rectangular, and the finite element part (16 CAX8R elements) also extends to a radius of 3 m (10 ft). Eight CINAX5R elements model the far field. A third mesh of finite elements only is shown in *Figure 3*: this mesh is identical to the one in *Figure 2* with the exception that the outer layer of infinite elements is replaced with a layer of finite elements extending to a distance of 9 m (30 ft), where the normal component of displacement is fixed.

The material is isotropic, linear elastic, with Young's modulus 4.788 MPa (10^5 lb/ft^2) and Poisson's ratio 0.3. The elastic half-space is subjected to uniform pressure load of intensity 4788 Pa (100 lb/ft^2) within a radius of 1.5 m (5 ft).

Results and discussion

The analytical solution for this problem is given by Timoshenko and Goodier (1970) and is plotted in *Figure 4* and *Figure 5*. *Figure 4* shows the surface deflection as a function of radius, while *Figure 5* shows the distribution of vertical stress along a vertical line beneath the center of the load. The displacement results for the meshes with infinite elements show almost exact agreement with the theory, while those obtained with the pure finite elements mesh are correct in form but have an offset from the exact result. The stress results shown in *Figure 5* are all in close agreement with the theory.

Input files

infelemcircular_radial_cinax5r.inp	Radial coupled finite/infinite element mesh.
infelemcircular_rect_cinax5r.inp	Rectangular coupled finite/infinite element mesh.
infelemcircular_rect_cax8r.inp	Mesh composed of finite elements only.

References

- Lynn, P. P., and H. A. Hadid, "Infinite Elements with $1/r^n$ Type Decay," *International Journal of Numerical Methods in Engineering*, vol. 17, no. 3, pp. 347–355, 1981.
- Timoshenko, S. P., and J. N. Goodier, Theory of Elasticity, McGraw-Hill, New York, 1970.

Figures

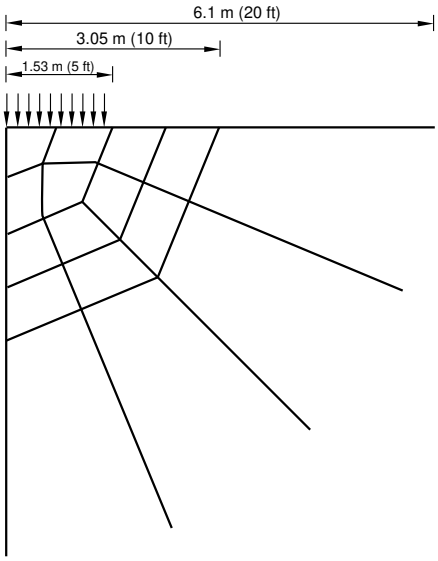


Figure 1: Radial finite/infinite element mesh.

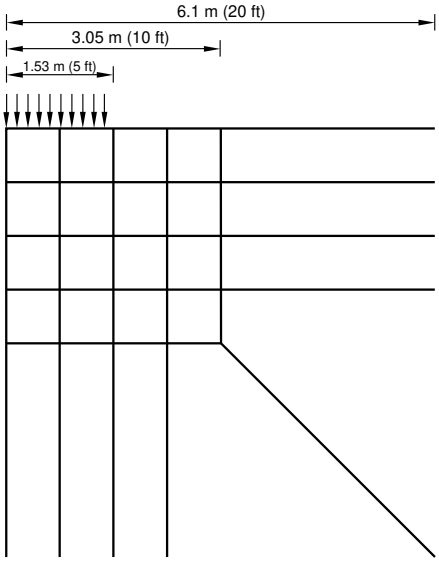


Figure 2: Rectangular finite/infinite element mesh.

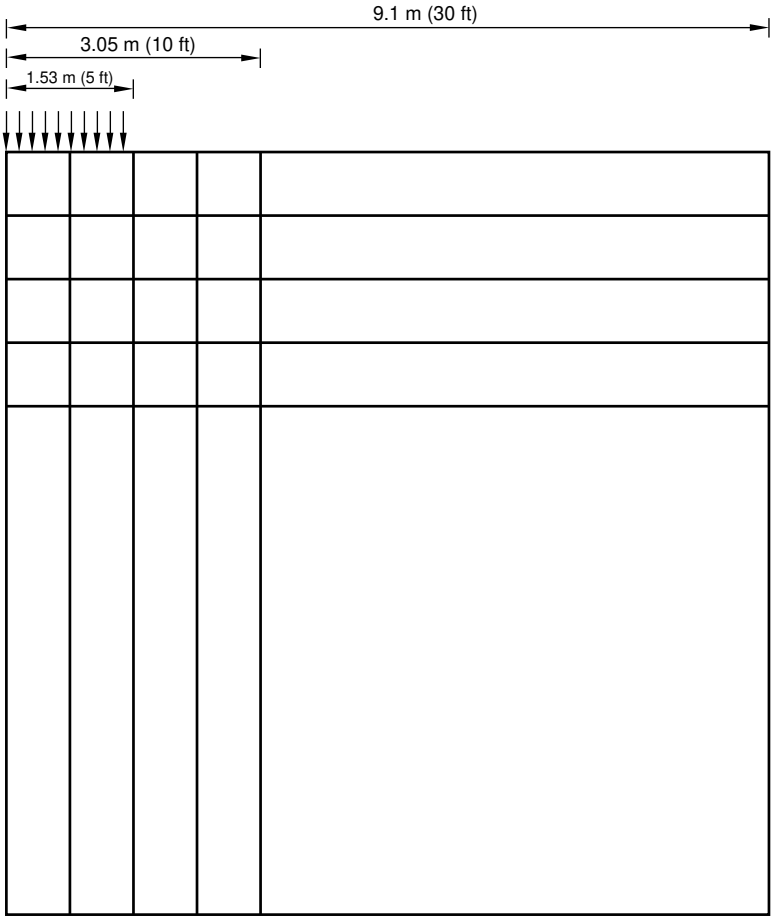


Figure 3: Mesh of finite elements only.

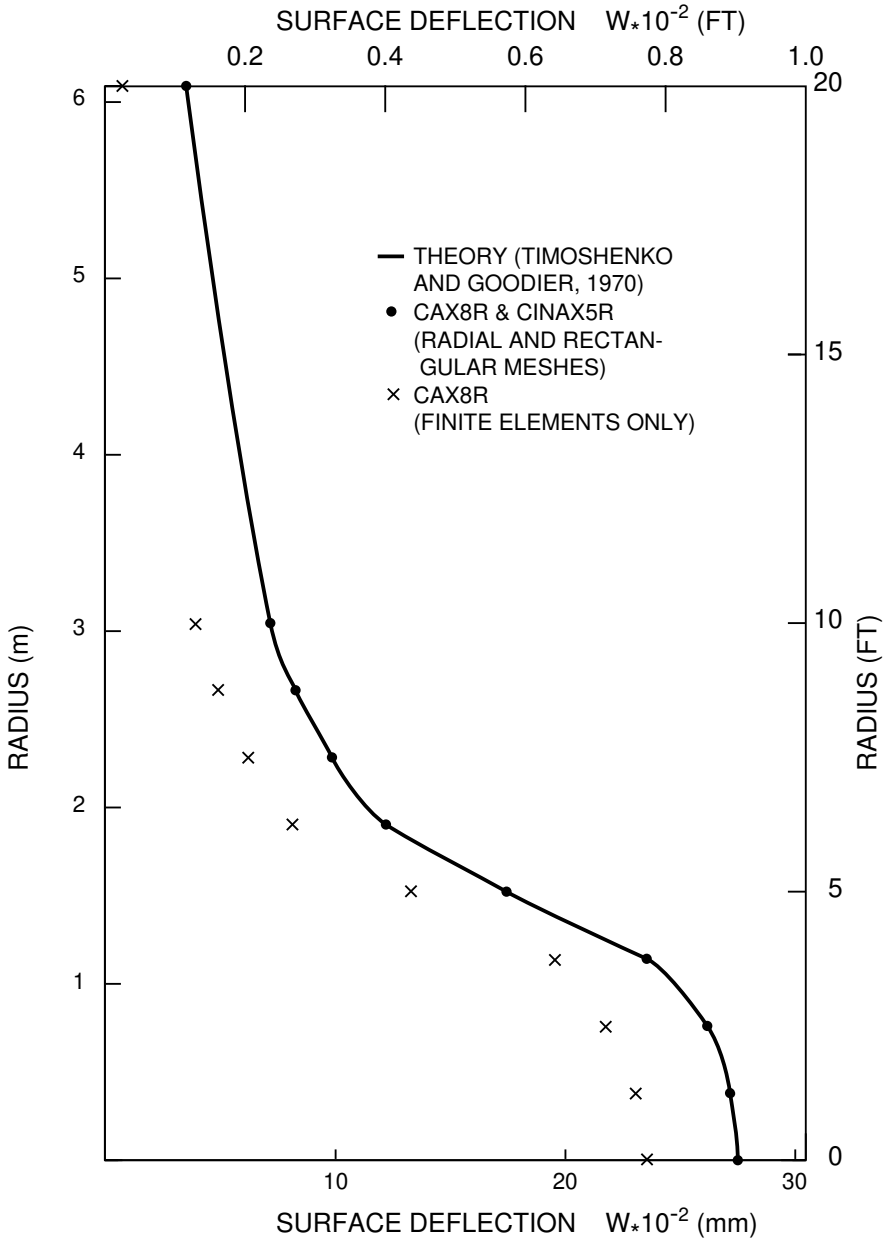


Figure 4: Surface deflection results.

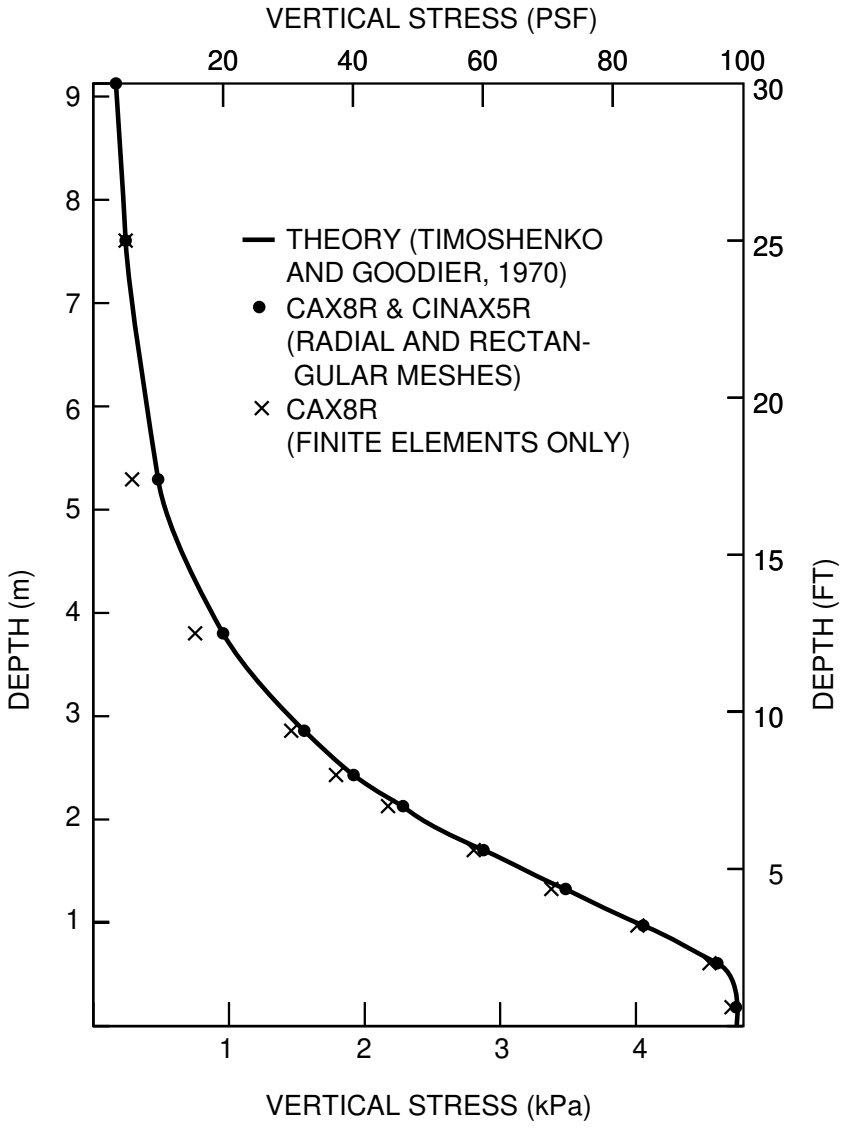


Figure 5: Vertical stress distribution.

Spherical cavity in an infinite medium

Products: Abaqus/Standard

Overview

This example of an internally pressurized spherical cavity in an infinite medium was analyzed by Marques and Owen (1983) and compares results obtained using infinite elements to results obtained with finite elements only, assuming fixed or free boundary conditions at the truncated end of the mesh. In addition, it investigates the performance of meshes with different degrees of refinement compared to the analytical solution provided by Hill (1950).

Problem description

In consistent units the radius of the cavity is 1.0, and an internal pressure of 750 is applied. The material is isotropic, linear elastic, with Young's modulus 10^7 and Poisson's ratio 0.33.

Using three orthogonal planes of symmetry, only one-eighth of the configuration needs to be analyzed. The finite element meshes are made up of layers of 12 C3D20R elements, as shown in *Figure 1*. Two basic meshes of three layers of elements are used: one where the layers are defined by the radial positions 1, 2, 3, 4 and the other where the layers are defined by the radial positions 1, 2, 4, 8. In each case the outermost nodes are considered to be either fixed or free. This results in four different finite element meshes that we label as follows:

- F4FX – outer radius = 4, outer nodes fixed
- F4FR – outer radius = 4, outer nodes free
- F8FX – outer radius = 8, outer nodes fixed
- F8FR – outer radius = 8, outer nodes free

Two coupled finite/infinite element meshes are used. First, we take an F8 mesh and replace the outer layer of finite elements by a layer of CIN3D12R elements. We label this mesh I4, the digit 4 indicating the radius at which the finite and infinite elements are coupled. Second, we use a mesh with a single layer of finite elements between radial values 1 and 2 and coupled to it a layer of CIN3D12R elements. This mesh is labeled I2. To test the use of substructures in problems involving coupled finite/infinite element meshes, the I2 mesh is also solved using substructuring with the entire model treated as a single substructure.

Results and discussion

Hill's analytical solution for this problem is shown in *Figure 2* and *Figure 3* (radial displacements) and *Figure 4*, *Figure 5*, *Figure 6*, and *Figure 7* (tangential and radial stresses). The results obtained with finite element meshes F4FX and F4FR and with the finite/infinite element mesh I2 are compared to the analytical solution in *Figure 2*, *Figure 4*, and *Figure 6*. We see that models F4FX and F4FR provide upper and lower bound solutions, while the infinite element result agrees almost exactly with the theory. Even this very crude finite/infinite element mesh (with only a single layer of finite elements) provides accurate results for this simple case. *Figure 3*, *Figure 5*, and *Figure 7* show the results obtained with meshes F8FX, F8FR, and I4: again the infinite element results are in almost exact agreement with the analytical solution, while the better finite element representations also provide results that are close to the analytical solution. The results from the substructure analysis match the results that are obtained when substructures are not used.

Input files

sphericalcavinfmed_i2.inp	Model I2.

sphericalcavinfmed_i4.inp	Model I4.
sphericalcavinfmed_f4fx.inp	Model F4FX.
sphericalcavinfmed_f8fx.inp	Model F8FX.
sphericalcavinfmed_i2_sub.inp	Model I2 using substructuring.
sphericalcavinfmed_i2_sub_gen1.inp	Substructure generation referenced by the analysis sphericalcavinfmed_i2_sub.inp.
sphericalcavinfmed_i2_cin3d18r.inp	Model I2 using CIN3D18R elements.
sphericalcavinfmed_i4_cin3d18r.inp	Model I4 using CIN3D18R elements.

References

- Hill, R., The Mathematical Theory of Plasticity, Oxford University Press, 1950.
- Marques, J. M. M. C., and D. R. J. Owen, “Infinite Elements in Quasi–Static Materially Nonlinear Problems,” *University of Wales Report, Swansea*, 1983.

Figures

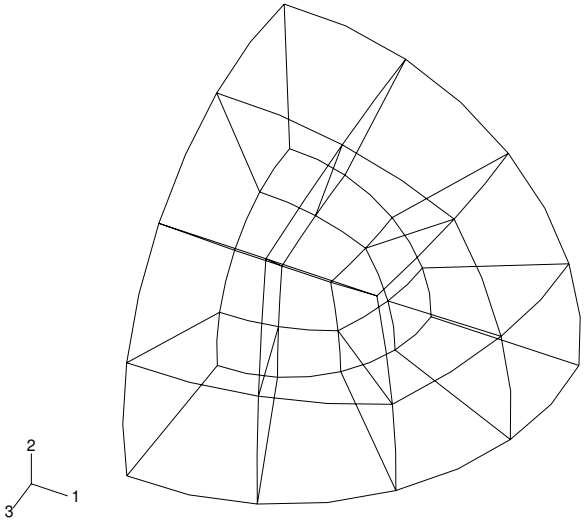


Figure 1: Typical element layer.

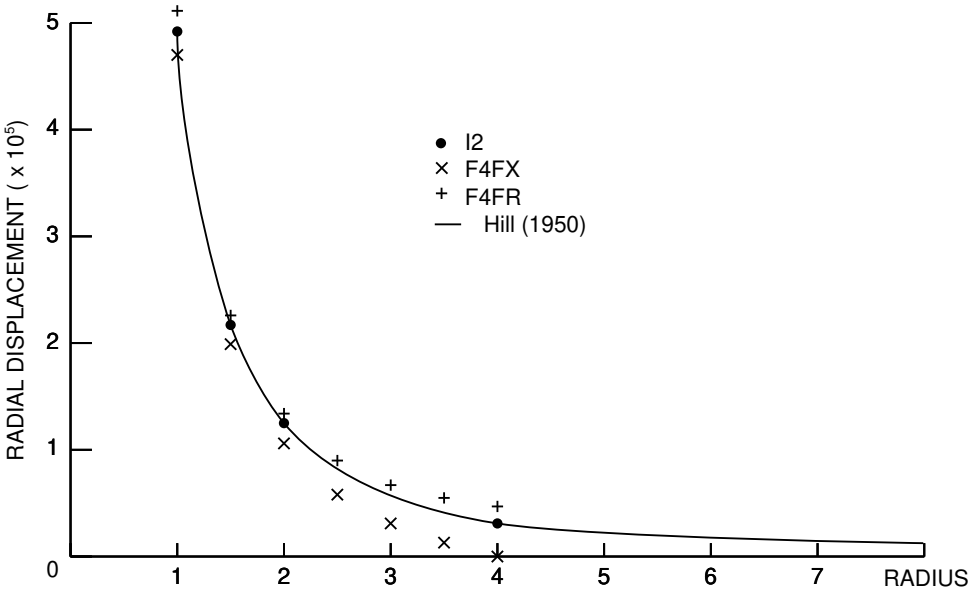


Figure 2: Radial displacement results—meshes F4FX, F4FR, and I2.

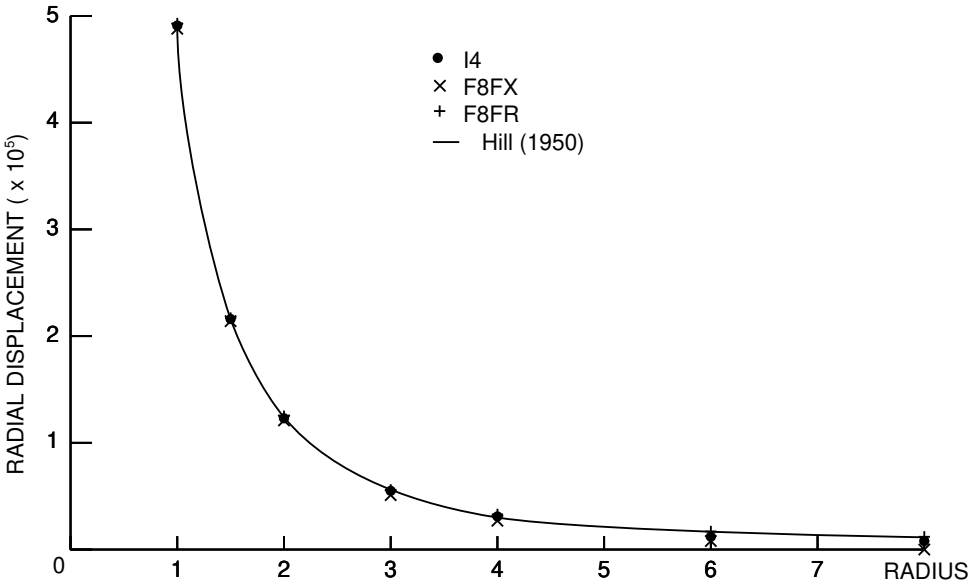


Figure 3: Radial displacement results—meshes F8FX, F8FR, and I4.

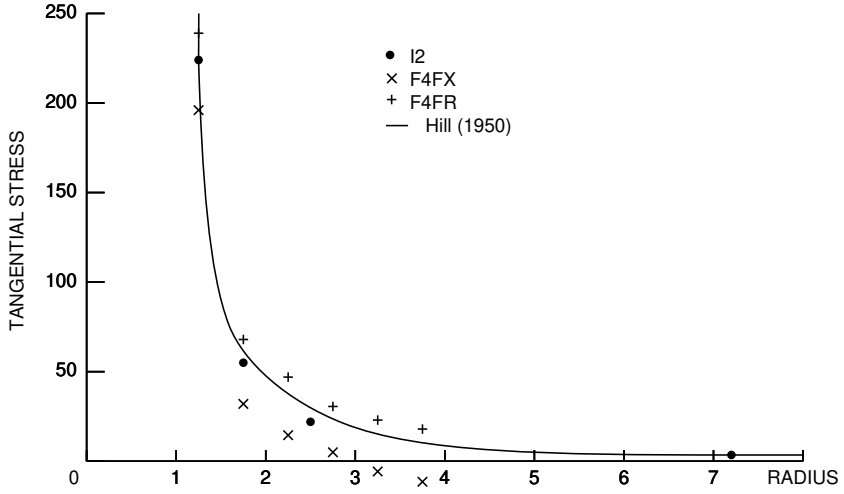


Figure 4: Tangential stress distribution—meshes F4FX, F4FR, and I2.

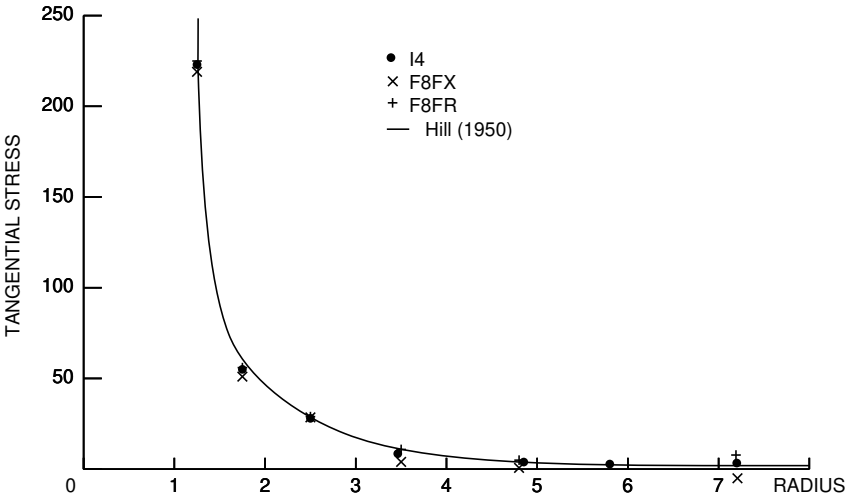


Figure 5: Tangential stress distribution—meshes F8FX, F8FR, and I4.

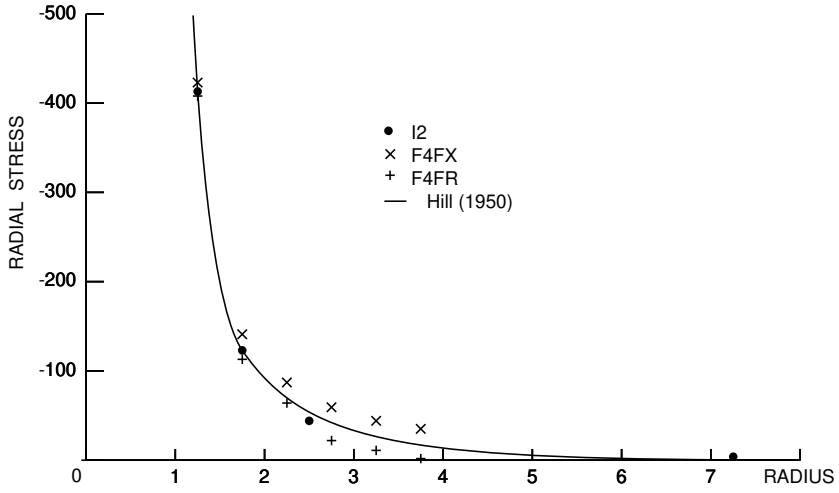


Figure 6: Radial stress distribution—meshes F4FX, F4FR, and I2.

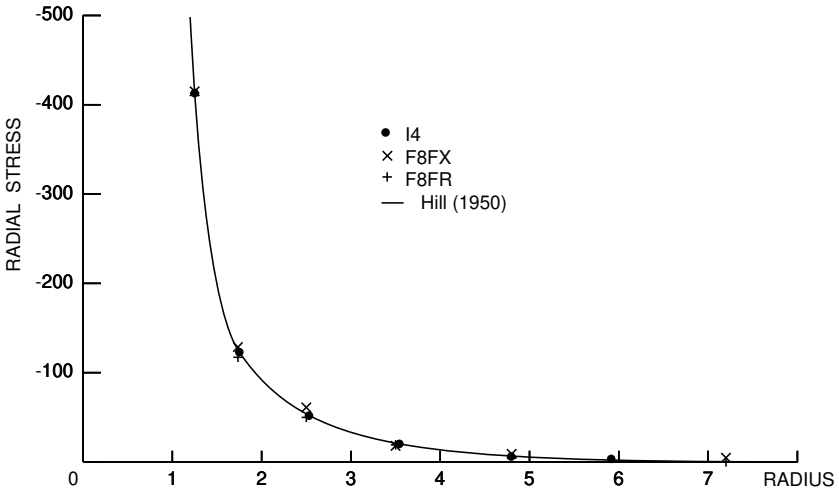


Figure 7: Radial stress distribution—meshes F8FX, F8FR, and I4.

Structural elements

In this section:

- *The barrel vault roof problem*
- *The pinched cylinder problem*
- *The pinched sphere problem*
- *Skew sensitivity of shell elements*
- *Performance of continuum and shell elements for linear analysis of bending problems*
- *Tip in-plane shear load on a cantilevered hook*
- *Analysis of a twisted beam*
- *Twisted ribbon test for shells*
- *Ribbon test for shells with applied moments*
- *Triangular plate-bending on three point supports*
- *Shell elements subjected to uniform thermal loading*
- *Shell bending under a tip load*
- *Variable thickness shells and membranes*
- *Transient response of a shallow spherical cap*
- *Simulation of propeller rotation*

The barrel vault roof problem

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example verifies the relative efficiency of using various shell elements available in Abaqus and different mesh densities to model a barrel vault roof deforming under its own weight.

Over the past several years a small set of linear test cases has emerged as a critical test set for shell elements (see, for example, the collection of papers on numerical modeling of shells—edited by Ashwell and Gallagher, 1976—and the survey paper by Belytschko, 1986). The set contains three cases: the barrel vault roof (this example), the cylinder with end diaphragm support subjected to pinching loads (*The pinched cylinder problem*), and the point loaded hemispherical shell (*LE3: Hemispherical shell with point loads*). It has been generally accepted that any elements that perform well on all three cases should provide accurate results for most general shell problems. These test cases are included in this guide so that the performance of the shell elements offered in Abaqus can be assessed.

Most modern shell elements, including those in Abaqus, do a good job on these problems. Although this is an indication that the elements usually provide good results, it should not be taken as a sufficient demonstration of the quality of an element's performance in all cases. For example, all three of these problems are completely regular geometries; the candidate element's usefulness in irregular geometries (and most practical cases involve a high degree of geometric irregularity) is not tested. In this example we make some attempt to address this issue by modeling not only with the regular mesh that would be the natural choice for the problem, but also with a mesh that might be the basis of analysis of a problem with the same underlying shape but with some type of local, irregular feature, such as a crack. Results for both types of mesh are reported below. As would be expected, the irregular mesh results are not as good as those provided by a regular mesh with the same number of variables.

Problem description

The problem is shown in *Figure 1*. The physical basis of the problem is a deeply arched roof supported only by diaphragms at its curved edges (an aircraft hanger), deforming under its own weight. It is interesting to observe that the geometry is such that the center point of the roof moves upward under the self-weight (downwardly directed) load. Perhaps this is one reason why the problem is not straightforward numerically.

Two discretizations are studied: a regular meshing and an irregular meshing of the type that might be used when a local refinement is desired (*Figure 2*). This method of mesh refinement is not being recommended: the irregular meshing is introduced here simply to record some results for the shell element used in this way.

The actual roof spans 15.24 m (600 in) between supports and has a thickness of 76.2 mm (3 in), so it would be considered to be a “thin” shell. Some of the shell elements in Abaqus/Standard (element types S4R5, S8R5, S9R5, STRI3, and STRI65) are intended to be used as thin shells. In these elements the Kirchhoff assumption, that lines initially normal to the shell's reference surface remain normal to that surface during the deformation, is imposed either algebraically (in element type STRI3) or numerically (in element types S4R5, S8R5, S9R5, and STRI65). Shell elements S4R, S4, and S3R and continuum shell elements SC6R and SC8R use an assumed strain treatment for the transverse shear that imposes the Kirchhoff constraint numerically for thin shells and provides accurate transverse shear predictions for thick shells. Element types S4R, S4, S3R, SC6R, and SC8R are, hence, valid for both thin and thick applications. Element type S8R is mainly intended to be used for thick shell modeling, where transverse shear flexibility may be an important part of the deformation. When this element is used to model thin shells, the transverse shear stiffness is treated as a penalty to impose the Kirchhoff assumption discretely, the penalty being chosen based on the technique described by Hughes et al. (1977).

In Abaqus/Explicit the problem is modeled using S4 elements, S4R elements with enhanced hourglass control, S3R elements, and S3RS elements.

Results and discussion

The results for both the regular and irregular meshes are described below.

Regular mesh

These results are summarized in *Table 1*, where the vertical motion of the center of the free edge is recorded. The generally accepted solution for this single displacement value, based on deep shell theory, is 91.2 mm (3.59 in) (see the chapter by Ashwell in Ashwell and Gallagher (1976) for a discussion of both semianalytical and purely numerical solutions to this example). *Table 1* records the error in this single displacement value compared to this "exact" solution.

From *Table 1* it is apparent that the second-order thin shell elements (S8R5, S9R5) are the most effective elements for this problem, with S8R, STRI65, and the first-order quadrilaterals (S4, S4R5, S4R) providing almost as good a solution except for the coarsest mesh used.

Irregular mesh

The results provided by the irregular mesh models are summarized in *Table 2*. These results are not as accurate as the results provided by regular mesh models having roughly the same number of elements and degrees of freedom. Nevertheless, when a relatively fine mesh is used, all the elements provide acceptable results even with the irregular mesh pattern. The poor accuracy of the results provided by element type S8R with the coarse irregular mesh is particularly noticeable and serves as a warning of how important it is to use a regular mesh with this element type in a thin shell in which there are high strain gradients.

Summary

In summary, based on this example:

1. For thin shell modeling the most effective elements provided in Abaqus/Standard are S8R5 and S9R5. Element STRI65 is fully compatible with S8R5 and S9R5 and is recommended for mesh refinement.
2. The first-order triangular elements are not as good as the corresponding mesh of first-order quadrilateral elements. It is generally recommended that the triangles be used only to complete meshes that cannot be generated easily with quadrilaterals, and then they should only be used where both the bending and membrane strain gradients are not large.
3. Whenever possible, ensure that the mesh follows lines of principal curvature of the shell.
4. Perform convergence studies on any problem for which the elements have not been used previously.
5. Expect deterioration of the elements' performance if they are distorted.

Input files

Abaqus/Standard input files:

S3R element models:

barrelvault_s3r_reg44.inp	Regular 4 × 4 mesh.
barrelvault_s3r_reg88.inp	Regular 8 × 8 mesh.
barrelvault_s3r_reg1818.inp	Regular 18 × 18 mesh.
barrelvault_s3r_irreg.inp	Coarse irregular mesh.
barrelvault_s3r_fineirreg.inp	Fine irregular mesh.

S4 element models:

barrelvault_s4_reg44.inp	Regular 4 × 4 mesh.
barrelvault_s4_reg88.inp	Regular 8 × 8 mesh.
barrelvault_s4_reg1818.inp	Regular 18 × 18 mesh.
barrelvault_s4_irreg.inp	Coarse irregular mesh.
barrelvault_s4_fineirreg.inp	Fine irregular mesh.

S4R element models:

barrelvault_s4r_reg44.inp	Regular 4 × 4 mesh.
barrelvault_s4r_reg88.inp	Regular 8 × 8 mesh.
barrelvault_s4r_reg1818.inp	Regular 18 × 18 mesh.
barrelvault_s4r_irreg.inp	Coarse irregular mesh.
barrelvault_s4r_fineirreg.inp	Fine irregular mesh.

S4R5 element models:

barrelvault_s4r5_reg44.inp	Regular 4 × 4 mesh.
barrelvault_s4r5_reg88.inp	Regular 8 × 8 mesh.
barrelvault_s4r5_reg1818.inp	Regular 18 × 18 mesh.
barrelvault_s4r5_irreg.inp	Coarse irregular mesh.
barrelvault_s4r5_fineirreg.inp	Fine irregular mesh.

S8R element models:

barrelvault_s8r_reg22.inp	Regular 2 × 2 mesh.
barrelvault_s8r_reg44.inp	Regular 4 × 4 mesh.
barrelvault_s8r_reg99.inp	Regular 9 × 9 mesh.
barrelvault_s8r_reg2020.inp	Regular 20 × 20 mesh.
barrelvault_s8r_reg3030.inp	Regular 30 × 30 mesh.
barrelvault_s8r_irreg.inp	Coarse irregular mesh.
barrelvault_s8r_fineirreg.inp	Fine irregular mesh.

S8R5 element models:

barrelvault_s8r5_reg22.inp	Regular 2 × 2 mesh.
barrelvault_s8r5_reg44.inp	Regular 4 × 4 mesh.
barrelvault_s8r5_reg99.inp	Regular 9 × 9 mesh.
barrelvault_s8r5_reg2020.inp	Regular 20 × 20 mesh.
barrelvault_s8r5_reg3030.inp	Regular 30 × 30 mesh.
barrelvault_s8r5_irreg.inp	Coarse irregular mesh.
barrelvault_s8r5_fineirreg.inp	Fine irregular mesh.
barrelvault_s8r5_post_reg44.inp	**POST OUTPUT* of barrelvault_s8r5_reg44.inp.

S9R5 element models:

barrelvault_s9r5_reg22.inp	Regular 2 × 2 mesh.
barrelvault_s9r5_reg44.inp	Regular 4 × 4 mesh.
barrelvault_s9r5_reg99.inp	Regular 9 × 9 mesh.
barrelvault_s9r5_reg2020.inp	Regular 20 × 20 mesh.
barrelvault_s9r5_reg3030.inp	Regular 30 × 30 mesh.
barrelvault_s9r5_irreg.inp	Coarse irregular mesh.

barrelvault_s9r5_fineirreg.inp	Fine irregular mesh.

SC6R element models:

barrelvault_sc6r_reg44.inp	Regular 4 × 4 mesh.
barrelvault_sc6r_reg88.inp	Regular 8 × 8 mesh.
barrelvault_sc6r_reg1818.inp	Regular 18 × 18 mesh.
barrelvault_sc6r_irreg.inp	Coarse irregular mesh.
barrelvault_sc6r_fineirreg.inp	Fine irregular mesh.

SC8R element models:

barrelvault_sc8r_reg44.inp	Regular 4 × 4 mesh.
barrelvault_sc8r_reg88.inp	Regular 8 × 8 mesh.
barrelvault_sc8r_reg1818.inp	Regular 18 × 18 mesh.
barrelvault_sc8r_irreg.inp	Coarse irregular mesh.
barrelvault_sc8r_fineirreg.inp	Fine irregular mesh.

STRI3 element models:

barrelvault_stri3_reg44.inp	Regular 4 × 4 mesh.
barrelvault_stri3_reg88.inp	Regular 8 × 8 mesh.
barrelvault_stri3_reg1818.inp	Regular 18 × 18 mesh.
barrelvault_stri3_irreg.inp	Coarse irregular mesh.
barrelvault_stri3_fineirreg.inp	Fine irregular mesh.

STRI65 element models:

barrelvault_stri65_reg22.inp	Regular 2 × 2 mesh.
barrelvault_stri65_reg44.inp	Regular 4 × 4 mesh.
barrelvault_stri65_reg99.inp	Regular 9 × 9 mesh.
barrelvault_stri65_reg2020.inp	Regular 20 × 20 mesh.
barrelvault_stri65_reg3030.inp	Regular 30 × 30 mesh.
barrelvault_stri65_irreg.inp	Coarse irregular mesh.
barrelvault_stri65_fineirreg.inp	Fine irregular mesh.

Abaqus/Explicit input files:

S3R element models:

barrelvault_s3r_reg44_exp.inp	Regular 4 × 4 mesh.
barrelvault_s3r_reg88_exp.inp	Regular 8 × 8 mesh.
barrelvault_s3r_reg1818_exp.inp	Regular 18 × 18 mesh.
barrelvault_s3r_irreg_exp.inp	Coarse irregular mesh.
barrelvault_s3r_fineirreg_exp.inp	Fine irregular mesh.

S3RS element models:

barrelvault_s3rs_reg44_exp.inp	Regular 4 × 4 mesh.
barrelvault_s3rs_reg88_exp.inp	Regular 8 × 8 mesh.
barrelvault_s3rs_reg1818_exp.inp	Regular 18 × 18 mesh.
barrelvault_s3rs_irreg_exp.inp	Coarse irregular mesh.
barrelvault_s3rs_fineirreg_exp.inp	Fine irregular mesh.

S4 element models:

barrelvault_s4_reg44_exp.inp	Regular 4 × 4 mesh.
barrelvault_s4_reg88_exp.inp	Regular 8 × 8 mesh.
barrelvault_s4_reg1818_exp.inp	Regular 18 × 18 mesh.
barrelvault_s4_irreg_exp.inp	Coarse irregular mesh.
barrelvault_s4_fineirreg_exp.inp	Fine irregular mesh.

S4R element models:

barrelvault_s4r_reg44_exp.inp	Regular 4 × 4 mesh.
barrelvault_s4r_reg88_exp.inp	Regular 8 × 8 mesh.
barrelvault_s4r_reg1818_exp.inp	Regular 18 × 18 mesh.
barrelvault_s4r_irreg_exp.inp	Coarse irregular mesh.
barrelvault_s4r_fineirreg_exp.inp	Fine irregular mesh.

References

- Ashwell, D. G., and R. H. Gallagher, Finite Elements for Thin Shells and Curved Members, John Wiley and Sons, London, 1976.
- Belytschko, T., "A Review of Recent Developments in Plate and Shell Elements," *Computational Mechanics—Advances and Trends*, AMD vol. 75, ASME, New York, 1986.
- Hughes, T. J. R., R. L. Taylor, and W. Kanoknukulchai, "A Simple and Efficient Finite Element for Plate Bending," *International Journal for Numerical Methods in Engineering*, vol. 11, no. 10, pp. 1529–1543, 1977.

Tables

Table 1: Shell roof: results for vertical displacement at the middle of the free edge, based on various regular meshes.

Element type	Mesh	Vertical displacement		Error compared to 91.2 mm (3.59 in)
		(mm)	(in)	
STRI3	4 × 4	67.44	2.665	−25.8%
	8 × 8	80.52	3.170	−11.7%
	18 × 18	88.93	3.501	−2.5%
S4R5	4 × 4	109.60	4.315	20.2%
	8 × 8	95.99	3.779	5.3%
	18 × 18	92.53	3.643	1.5%
S4R	4 × 4	109.2	4.298	19.7%
	8 × 8	95.91	3.776	5.2%
	18 × 18	92.61	3.646	1.6%
S4	4 × 4	95.48	3.759	4.7%
	8 × 8	92.37	3.637	1.3%
	18 × 18	91.89	3.618	0.77%

Element type	Mesh	Vertical displacement		Error compared to 91.2 mm (3.59 in)
		(mm)	(in)	
S4[1]	4 × 4	95.63	3.765	4.9%
	8 × 8	92.74	3.651	1.7%
	18 × 18	92.33	3.635	1.3%
S4R[2]	4 × 4	100.85	3.971	10.6%
	8 × 8	94.43	3.718	3.5%
	18 × 18	92.88	3.657	1.8%
S8R5	2 × 2	92.89	3.657	1.9%
	4 × 4	91.74	3.612	0.6%
	9 × 9	91.72	3.611	0.6%
S8R	2 × 2	89.17	3.511	–2.2%
	4 × 4	92.41	3.638	1.3%
	9 × 9	91.90	3.618	0.8%
S9R5	2 × 2	92.89	3.657	1.9%
	4 × 4	91.74	3.612	0.6%
	9 × 9	91.72	3.611	0.6%
SC6R	4 × 4	66.42	2.615	–27.1%
	8 × 8	80.77	3.180	–11.4%
	18 × 18	89.586	3.527	–1.75%
SC8R	4 × 4	110.9	4.367	21.6%
	8 × 8	96.80	3.811	6.15%
	18 × 18	93.27	3.672	2.28%
STRI65	2 × 2	74.67	2.940	–18.1%
	4 × 4	90.11	3.548	–1.2%
	9 × 9	91.67	3.609	0.5%
S3R	4 × 4	65.71	2.587	–27.9%
	8 × 8	80.11	3.154	–12.1%
	18 × 18	88.90	3.500	–2.5%
S3R[1]	4 × 4	65.43	2.576	–28.2%
	8 × 8	80.37	3.164	–11.9%
	18 × 18	90.68	3.57	–0.6%
S3RS[1]	4 × 4	67.87	2.672	–25.6%
	8 × 8	84.61	3.331	–7.2%
	18 × 18	91.14	3.588	–0.06%

[1]Abaqus/Explicit element

[2]Abaqus/Explicit element with enhanced hourglass control

Table 2: Shell roof: results for vertical displacement at the middle of the free edge, based on irregular meshes.

Element type	Mesh	Vertical displacement		Error compared to 91.2 mm (3.59 in)
		(mm)	(in)	
STRI3	coarse (258 d.o.f.)	72.57	2.857	−20.4%
	fine (894 d.o.f.)	83.34	3.281	−8.6%
S4R5	coarse (258 d.o.f.)	96.57	3.802	5.9%
	fine (894 d.o.f.)	93.98	3.700	3.1%
S4R	coarse (270 d.o.f.)	96.16	3.786	5.5%
	fine (918 d.o.f.)	93.93	3.698	3.0%
S4	coarse (270 d.o.f.)	88.37	3.479	−3.1%
	fine (918 d.o.f.)	91.94	3.620	0.83%
S4[1]	coarse (270 d.o.f.)	90.50	3.563	−0.75%
	fine (918 d.o.f.)	92.66	3.648	1.6%
S4R[2]	coarse (270 d.o.f.)	86.66	3.412	−4.95%
	fine (918 d.o.f.)	93.11	3.666	2.11%
S8R5	coarse (270 d.o.f.)	79.98	3.149	−12.3%
	fine (918 d.o.f.)	91.03	3.584	−0.2%
S8R	coarse (210 d.o.f.)	55.78	2.196	−38.8%
	fine (702 d.o.f.)	89.64	3.529	−1.7%
S9R5	coarse (270 d.o.f.)	82.75	3.258	−9.2%
	fine (918 d.o.f.)	93.80	3.693	2.9%
SC6R	coarse (270 d.o.f.)	71.0	2.796	−22.1%
	fine (918 d.o.f.)	83.7	3.294	−8.24%
SC8R	coarse (270 d.o.f.)	97.6	3.843	7.05%
	fine (918 d.o.f.)	94.7	3.730	3.90%
STRI65	coarse (270 d.o.f.)	81.53	3.209	−10.5%
	fine (918 d.o.f.)	90.80	3.575	−0.41%
S3R	coarse (258 d.o.f.)	70.56	2.778	−22.6%
	fine (894 d.o.f.)	82.96	3.266	−9.0%
S3R[1]	coarse (258 d.o.f.)	72.59	2.858	−20.4%
	fine (894 d.o.f.)	84.28	3.318	−7.6%
S3RS[1]	coarse (258 d.o.f.)	74.85	2.947	−17.9%
	fine (894 d.o.f.)	89.33	3.517	−2.0%

[1]Abaqus/Explicit element

[2]Abaqus/Explicit element with enhanced hourglass control

Figures

Material:

Young's modulus 20.68 GPa (3.0×10^6 lb/in^2)

Poisson's ratio 0.0

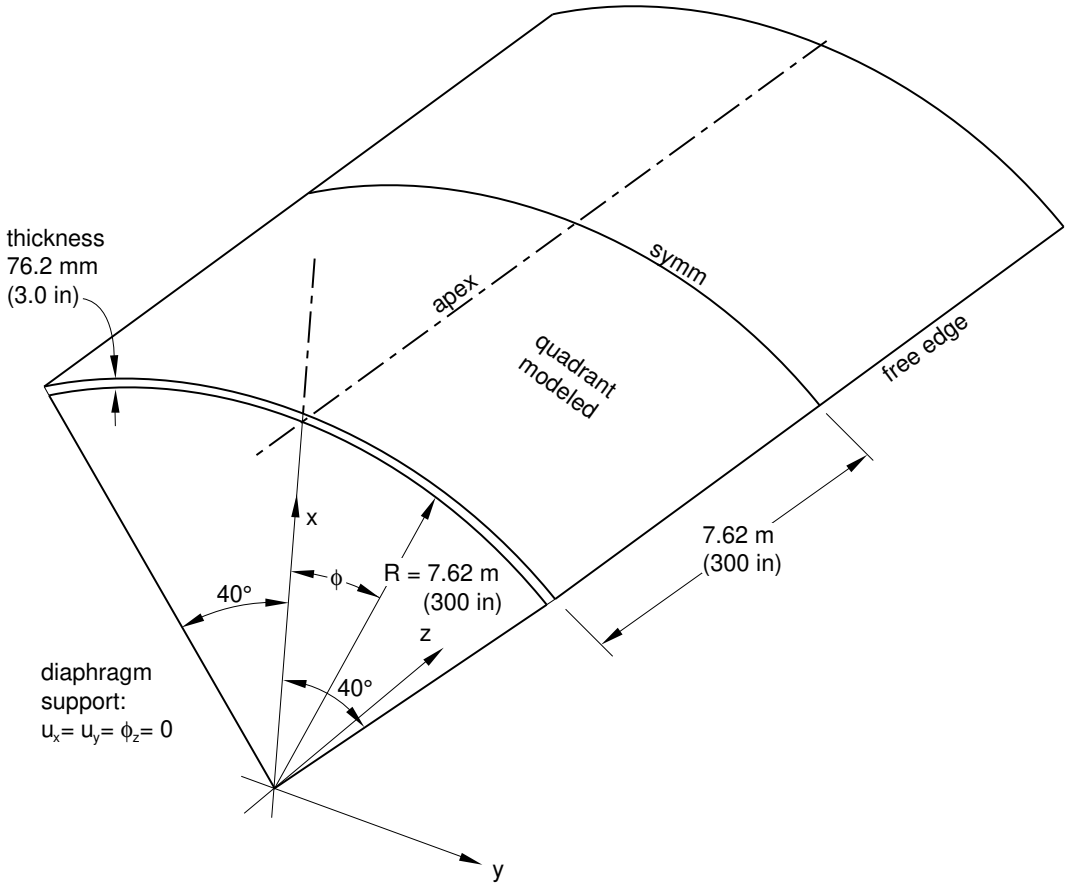


Figure 1: Barrel vault roof problem.

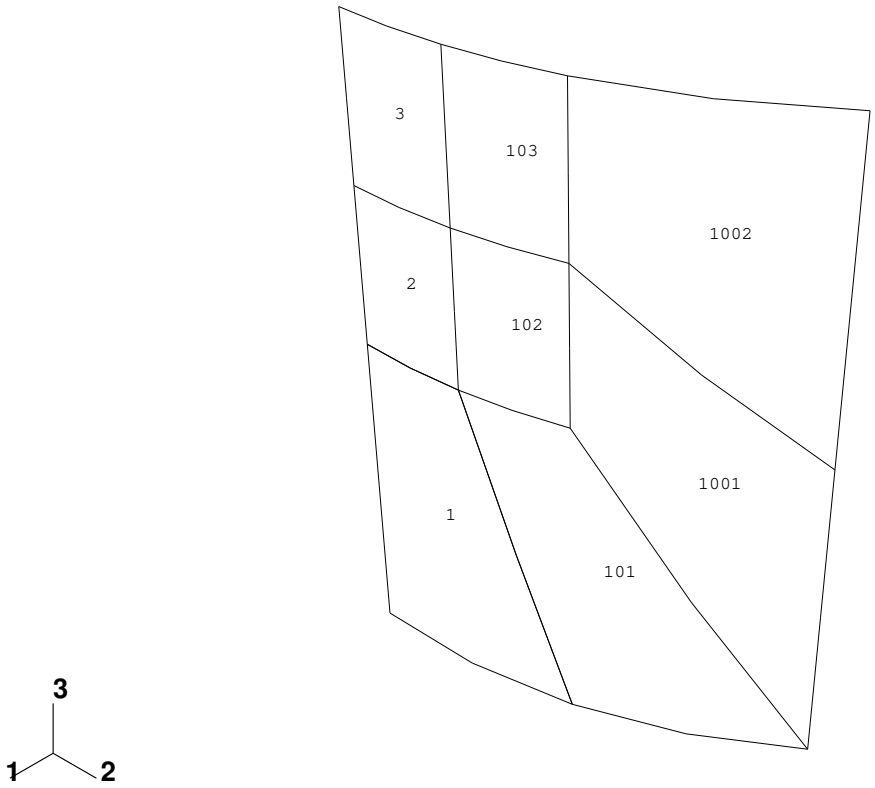


Figure 2: Coarse irregular mesh for barrel vault.

The pinched cylinder problem

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example demonstrates the performance of the shell element formulations in Abaqus, particularly with respect to the representation of inextensional bending modes and complex membrane states.
A finite length circular cylinder shell with rigid diaphragms in its ends, subjected to concentrated pinching loads, is analyzed; and the results are compared with known solutions (see Lindberg et al., 1969).

Problem description

The geometry and material properties used for the example are shown in *Figure 1*. No units are specified since the values given are in a self-consistent set of units. The thickness of the cylinder is 1/100 of its radius, so the structure can be considered a thin shell. The mesh covers a symmetric segment of the cylinder, as indicated in the figure, with symmetry boundary conditions imposed on three edges of the mesh, while the fourth edge (the end of the cylinder) is supported by a rigid diaphragm.

Two mesh patterns are used in this example: a regular mesh, shown in *Figure 2*, and two types of irregular meshes (coarse and fine), shown in *Figure 3* and *Figure 4*. When triangular elements are used, each quadrilateral is divided into two triangles. The irregular meshes are tested because such mesh patterns might be used in cases where local effects must be modeled, and they allow an assessment to be made of the distortion sensitivity of the elements. For comparison, the cylinder is analyzed with all the general shell elements available in Abaqus/Standard; the Abaqus/Explicit analyses test only the S3R and S4R elements.

The submodeling capability in Abaqus/Standard is also used in this example to analyze the region in the vicinity of the concentrated load. For shell-to-shell submodeling two regular mesh patterns of S8R elements, shown in *Figure 5*, are driven by various global analyses also using regular meshes. In each case symmetry boundary conditions are imposed on two edges of the mesh, while results from the global analyses are interpolated to the remaining two edges through the submodeling technique. A shell-to-solid submodel is also available for demonstration purposes.

The shell-to-solid coupling capability in Abaqus/Standard and Abaqus/Explicit is also used in this example. The region in the vicinity of the concentrated load is meshed with continuum elements, and the rest of the cylinder is meshed with shell elements (see *Figure 6*). S4R, S8R, C3D8I, C3D10, C3D10HS, and C3D20R elements are used in six different shell-to-solid combinations in Abaqus/Standard; S4R and C3D8R elements are used in Abaqus/Explicit.

The displacements are small, so it is appropriate to ignore geometric nonlinearities in the Abaqus/Explicit analyses. If the large-displacement theory is activated by considering geometric nonlinearities, the results are unchanged in all cases since the strains and rotations remain small. However, the analysis CPU times typically increase by about 30%.

Two input files are provided for the continuum shell element model to illustrate orienting the element thickness (stacking) direction independent of the nodal connectivity using a cylindrical system.

Results and discussion

The result used for comparison is the radial displacement at the point where the pinching load is applied. The solution given by Lindberg et al., based on Flügge's (1973) series solution, is 0.1825×10^{-4}.

Regular mesh

The results for the regular Abaqus/Standard mesh are shown in *Table 1*. The second-order elements (types S8R5 and S9R5) provide the most accurate solutions, whereas element type S8R (also a second-order element, but designed primarily for thick shell applications) provides a rather less accurate solution. Element type STRI3 provides the most accuracy among the first-order elements. None of the first-order elements provides acceptable solutions with the coarsest meshes used.

Element type STRI65 appears to converge rather slowly compared to the other element types. This result may appear counterintuitive, especially when compared to the STRI3 results, which demonstrate better convergence in this problem. Compared to STRI3, which is a flat facet element, element type STRI65 is preferable for modeling bending of thin shells and has complete quadratic representation of membrane strains; therefore, STRI65 is expected to perform better than STRI3 provided the number of elements in the two meshes is the same. In the present convergence study we have instead retained an equal number of nodes, which results in the relatively poor performance of the STRI65 element.

The results for the regular Abaqus/Explicit mesh are shown in *Table 2*. The results suggest that element types S3R and S4R are initially stiff but then converge to the correct solution. In addition, an energy plot is provided in *Figure 7*, which shows that by the end of the analysis a steady, static solution is obtained.

Irregular mesh

The second type of irregular mesh has more distorted element shapes than the first type of irregular mesh. The results for the two irregular Abaqus/Standard meshes are given in *Table 3*; and, as discussed in *The barrel vault roof problem*, they show less accurate results than the regular mesh problems.

Element types S8R5 and S9R5 again provide reasonably accurate results with fine meshes, although the coarse mesh results with these elements demonstrate poor accuracy. Interestingly, in this case all the first-order quadrilateral elements provide quite accurate values even with coarse meshing. This result may be fortuitous and should not be taken as a general indication of the quality of the elements in distorted meshes. For element type S4R both stiffness-based and enhanced hourglass controls are used to study the effect of mesh refinement and skew sensitivity. As expected, the coarse mesh results for enhanced hourglass control show poor accuracy compared with the fine mesh results.

The results for the irregular Abaqus/Explicit meshes are given in *Table 4*. These irregular meshes are more accurate despite the increased distortion because mesh refinement is concentrated in the area of highest solution gradients.

Submodeled analyses

Results from the submodeled Abaqus/Standard analyses for the shell-to-shell cases are given in *Table 5*. Clearly, the submodeling technique provides a more accurate solution in the vicinity of the point load than the coarser global analyses. When S4R elements are used on the global level, the radial displacement at the point of load application is within 40% of Lindberg's solution for the coarse mesh and 13% for the finer mesh. The submodeling technique significantly improves these results, giving radial displacements in the shell submodels within 11% and 2% for all four combinations of meshes.

When S8R elements are used to mesh the quarter cylinder, solution accuracy improves from within 6% on the global level to within 0.7% on the submodel level. Displacement contours for the shell submodels are shown in *Figure 8* for a representative analysis in which a 5×5 mesh of S8R elements is used on the global level and a 10×10 mesh of S8R elements is used on the submodel level.

Submodel analyses are tested with output from input files *pinchcyl_s4r_reg55.inp*, *pinchcyl_s4r_reg1010.inp*, and *pinchcyl_s8r_reg55.inp*. If five degree of freedom shells (S4R5, S8R5, etc.) are used at the global level, only the displacement degrees of freedom on the submodel boundary are driven since the rotations are not written to the results file for these elements.

A shell-to-solid submodel is also available for this problem, with a 10 × 10 C3D8I element mesh and four elements across the shell thickness. The submodel is driven from a 12 × 12 S4R element global model. The results are in good agreement with the shell-to-shell submodel results. Since the submodel in this case is made of solid elements, no comparison to the shell analytical solution is offered. The use of the shell-to-solid submodeling capability would be more justified in the case of concentrated loading applied on a finite area instead of the point load.

Shell-to-solid coupling analyses

Six shell-to-solid coupling cases are analyzed in Abaqus/Standard, as listed in *Table 6*. In all six cases a 12 × 12 shell element mesh is used. As is clearly seen, the shell-to-solid coupling analyses provide accurate solutions in the vicinity of the point load. The radial displacement at the point of load application is within 4.1% of Lindberg's solution for all six cases. As mentioned for submodeling, the use of the shell-to-solid coupling capability would be more justified in the case of concentrated loading applied on a finite area instead of the point load.

The results for the Abaqus/Explicit shell-to-solid coupling analysis are given in *Table 7*. The radial displacement at the point of load application is within 32% of Lindberg's solution.

Parametric study using the Abaqus parametric study capability

The performance of shell element formulations investigated in this example can be analyzed conveniently in a parametric study using the scripting capabilities offered in Abaqus. As an example we perform a parametric study in which eight analyses are automatically executed; these analyses correspond to combinations of three different (regular) mesh densities (5 × 5, 10 × 10, 20 × 20) for three different element types (S4, S8R, and S3R).

pinchcyl_parametric.inp shows the parametrized template input data used to generate the parametric variations of the parametric study. The script file (*pinchcyl_parametric.psf*) is used to perform the parametric study. The radial displacement at the point where the pinching load is applied is reported in the following table for each of the analyses of the parametric study:

```
_____________________________________________________

     Parametric study: pinchcyl_parametric
_____________________________________________________

        eltype,     m_density,     N2001_U.2,
_____________________________________________________

            S4,             5,  -9.51849e-06,
           S8R,             5,  -1.72138e-05,
            S4,            10,  -1.51895e-05,
           S8R,            10,  -1.80581e-05,
            S4,            20,   -1.7505e-05,
           S3R,             5,  -6.51879e-06,
           S3R,            10,   -1.3277e-05,
           S3R,            20,  -1.67431e-05,
_____________________________________________________
```

These results match the corresponding results found in *Table 1*.

Input files

Abaqus/Standard input files

pinchcyl_s8r_submodel_reg55.inp	Regular submodel mesh, S8R elements.
pinchcyl_s8r_sub_reg1010.inp	Submodel mesh, S8R elements.

pinchcyl_c3d8i_sub_reg10104.inp	Solid submodel mesh, C3D8I elements.
pinchcyl_s4r_c3d8i_shell2solid.inp	Shell-to-solid coupling model with S4R shell elements and C3D8I continuum elements.
pinchcyl_s4r_c3d10_shell2solid.inp	Shell-to-solid coupling model with S4R shell elements and C3D10 continuum elements.
pinchcyl_s4r_c3d10hs_shell2solid.inp	Shell-to-solid coupling model with S4R shell elements and C3D10HS continuum elements.
pinchcyl_s4r_c3d20r_shell2solid.inp	Shell-to-solid coupling model with S4R shell elements and C3D20R continuum elements.
pinchcyl_s8r_c3d8i_shell2solid.inp	Shell-to-solid coupling model with S8R shell elements and C3D8I continuum elements.
pinchcyl_s8r_c3d10_shell2solid.inp	Shell-to-solid coupling model with S8R shell elements and C3D10 continuum elements.
pinchcyl_s8r_c3d10hs_shell2solid.inp	Shell-to-solid coupling model with S8R shell elements and C3D10HS continuum elements.
pinchcyl_s8r_c3d20r_shell2solid.inp	Shell-to-solid coupling model with S8R shell elements and C3D20R continuum elements.
pinchcyl_parametric.inp	Parametrized template input data used to generate the parametric variations of the parametric study.

S3R elements:

pinchcyl_s3r_reg55.inp	5 × 5 mesh.
pinchcyl_s3r_reg1010.inp	10 × 10 mesh.
pinchcyl_s3r_reg2020.inp	20 × 20 mesh.
pinchcyl_s3r_irreg_typ1.inp	Coarse irregular mesh (Type 1).
pinchcyl_s3r_fineirreg_typ1.inp	Fine irregular mesh (Type 1).
pinchcyl_s3r_irreg_typ2.inp	Coarse irregular mesh (Type 2).
pinchcyl_s3r_fineirreg_typ2.inp	Fine irregular mesh (Type 2).

S4 elements:

pinchcyl_s4_reg55.inp	5 × 5 mesh.
pinchcyl_s4_reg1010.inp	10 × 10 mesh.
pinchcyl_s4_reg2020.inp	20 × 20 mesh.
pinchcyl_s4_irreg_typ1.inp	Coarse irregular mesh (Type 1).
pinchcyl_s4_fineirreg_typ1.inp	Fine irregular mesh (Type 1).
pinchcyl_s4_irreg_typ2.inp	Coarse irregular mesh (Type 2).
pinchcyl_s4_fineirreg_typ2.inp	Fine irregular mesh (Type 2).
pinchcyl_s4_reg22_typ1.inp	2 × 2 mesh (Type 1).
pinchcyl_s4_reg44_typ1.inp	4 × 4 mesh (Type 1).
pinchcyl_s4_reg66_typ1.inp	6 × 6 mesh (Type 1).
pinchcyl_s4_reg88_typ1.inp	8 × 8 mesh (Type 1).
pinchcyl_s4_reg1212_typ1.inp	12 × 12 mesh (Type 1).

S4R elements:

pinchcyl_s4r_reg55.inp	5 × 5 mesh.
pinchcyl_s4r_reg55_eh.inp	5 × 5 mesh with enhanced hourglass control.

pinchcyl_s4r_reg1010.inp	10 × 10 mesh.
pinchcyl_s4r_reg1010_eh.inp	10 × 10 mesh with enhanced hourglass control.
pinchcyl_s4r_reg2020.inp	20 × 20 mesh.
pinchcyl_s4r_reg2020_eh.inp	20 × 20 mesh with enhanced hourglass control.
pinchcyl_s4r_irreg_typ1.inp	Coarse irregular mesh (Type 1).
pinchcyl_s4r_irreg_typ1_eh.inp	Coarse irregular mesh with enhanced hourglass control (Type 1).
pinchcyl_s4r_fineirreg_typ1.inp	Fine irregular mesh (Type 1).
pinchcyl_s4r_fineirreg_typ1_eh.inp	Fine irregular mesh with enhanced hourglass control (Type 1).
pinchcyl_s4r_irreg_typ2.inp	Coarse irregular mesh (Type 2).
pinchcyl_s4r_irreg_typ2_eh.inp	Coarse irregular mesh with enhanced hourglass control (Type 2).
pinchcyl_s4r_fineirreg_typ2.inp	Fine irregular mesh (Type 2).
pinchcyl_s4r_fineirreg_typ2_eh.inp	Fine irregular mesh with enhanced hourglass control (Type 2).
pinchcyl_s4r_reg22_typ1.inp	2 × 2 mesh (Type 1).
pinchcyl_s4r_reg44_typ1.inp	4 × 4 mesh (Type 1).
pinchcyl_s4r_reg66_typ1.inp	6 × 6 mesh (Type 1).
pinchcyl_s4r_reg88_typ1.inp	8 × 8 mesh (Type 1).
pinchcyl_s4r_reg1212_typ1.inp	12 × 12 mesh (Type 1).

S4R5 elements:

pinchcyl_s4r5_reg55.inp	5 × 5 mesh.
pinchcyl_s4r5_reg1010.inp	10 × 10 mesh.
pinchcyl_s4r5_reg2020.inp	20 × 20 mesh.
pinchcyl_s4r5_irreg_typ1.inp	Coarse irregular mesh (Type 1).
pinchcyl_s4r5_fineirreg_typ1.inp	Fine irregular mesh (Type 1).
pinchcyl_s4r5_irreg_typ2.inp	Coarse irregular mesh (Type 2).
pinchcyl_s4r5_fineirreg_typ2.inp	Fine irregular mesh (Type 2).
pinchcyl_s4r5_reg22_typ1.inp	2 × 2 mesh (Type 1).
pinchcyl_s4r5_reg44_typ1.inp	4 × 4 mesh (Type 1).
pinchcyl_s4r5_reg66_typ1.inp	6 × 6 mesh (Type 1).
pinchcyl_s4r5_reg88_typ1.inp	8 × 8 mesh (Type 1).
pinchcyl_s4r5_reg1212_typ1.inp	12 × 12 mesh (Type 1).

S8R elements:

pinchcyl_s8r_reg55.inp	5 × 5 mesh.
pinchcyl_s8r_reg1010.inp	10 × 10 mesh.
pinchcyl_s8r_irreg_typ1.inp	Coarse irregular mesh (Type 1).
pinchcyl_s8r_fineirreg_typ1.inp	Fine irregular mesh (Type 1).
pinchcyl_s8r_irreg_typ2.inp	Coarse irregular mesh (Type 2).
pinchcyl_s8r_fineirreg_typ2.inp	Fine irregular mesh (Type 2).

S8R5 elements:

pinchcyl_s8r5_reg55.inp	5 × 5 mesh.
pinchcyl_s8r5_reg1010.inp	10 × 10 mesh.
pinchcyl_s8r5_irreg.inp	Coarse irregular mesh (Type 1).
pinchcyl_s8r5_fineirreg_typ1.inp	Fine irregular mesh (Type 1).
pinchcyl_s8r5_irreg_typ2.inp	Coarse irregular mesh (Type 2).
pinchcyl_s8r5_fineirreg_typ2.inp	Fine irregular mesh (Type 2).

S9R5 elements:

pinchcyl_s9r5_reg55.inp	5 × 5 mesh.
pinchcyl_s9r5_reg1010.inp	10 × 10 mesh.
pinchcyl_s9r5_irreg_typ1.inp	Coarse irregular mesh (Type 1).
pinchcyl_s9r5_fineirreg_typ1.inp	Fine irregular mesh (Type 1).
pinchcyl_s9r5_irreg_typ2.inp	Coarse irregular mesh (Type 2).
pinchcyl_s9r5_fineirreg_typ2.inp	Fine irregular mesh (Type 2).

STRI3 elements:

pinchcyl_stri3_reg55.inp	5 × 5 mesh.
pinchcyl_stri3_reg1010.inp	10 × 10 mesh.
pinchcyl_stri3_reg2020.inp	20 × 20 mesh.
pinchcyl_stri3_irreg_typ1.inp	Coarse irregular mesh (Type 1).
pinchcyl_stri3_fineirreg_typ1.inp	Fine irregular mesh (Type 1).
pinchcyl_stri3_irreg_typ2.inp	Coarse irregular mesh (Type 2).
pinchcyl_stri3_fineirreg_typ2.inp	Fine irregular mesh (Type 2).
pinchcyl_stri3_reg22_typ1.inp	2 × 2 mesh (Type 1).
pinchcyl_stri3_reg44_typ1.inp	4 × 4 mesh (Type 1).
pinchcyl_stri3_reg66_typ1.inp	6 × 6 mesh (Type 1).
pinchcyl_stri3_reg88_typ1.inp	8 × 8 mesh (Type 1).
pinchcyl_stri3_reg1212_typ1.inp	12 × 12 mesh (Type 1).
pinchcyl_stri3_reg22_typ2.inp	2 × 2 mesh (Type 2).
pinchcyl_stri3_reg44_typ2.inp	4 × 4 mesh (Type 2).
pinchcyl_stri3_reg66_typ2.inp	6 × 6 mesh (Type 2).
pinchcyl_stri3_reg88_typ2.inp	8 × 8 mesh (Type 2).
pinchcyl_stri3_reg1212_typ2.inp	12 × 12 mesh (Type 2).
pinchcyl_stri3_reg22_typ3.inp	2 × 2 mesh (Type 3).
pinchcyl_stri3_reg44_typ3.inp	4 × 4 mesh (Type 3).
pinchcyl_stri3_reg66_typ3.inp	6 × 6 mesh (Type 3).
pinchcyl_stri3_reg88_typ3.inp	8 × 8 mesh (Type 3).
pinchcyl_stri3_reg1212_typ3.inp	12 × 12 mesh (Type 3).

STRI65 elements:

pinchcyl_stri65_reg55.inp	5 × 5 mesh.
pinchcyl_stri65_reg1010.inp	10 × 10 mesh.
pinchcyl_stri65_irreg_typ1.inp	Coarse irregular mesh (Type 1).
pinchcyl_stri65_fineirreg_typ1.inp	Fine irregular mesh (Type 1).
pinchcyl_stri65_irreg_typ2.inp	Coarse irregular mesh (Type 2).

pinchcyl_stri65_fineirreg_typ2.inp	Fine irregular mesh (Type 2).
SC6R elements:	
pinchcyl_sc6r_reg55.inp	5 × 5 mesh.
pinchcyl_sc6r_reg1010.inp	10 × 10 mesh.
pinchcyl_sc6r_reg2020.inp	20 × 20 mesh.
pinchcyl_sc6r_stackdir_cylori.inp	20 × 20 mesh using the STACK DIRECTION=ORIENTATION parameter with a cylindrical orientation system to define the element thickness direction.
pinchcyl_sc6r_irreg_typ1.inp	Coarse irregular mesh (Type 1).
pinchcyl_sc6r_fineirreg_typ1.inp	Fine irregular mesh (Type 1).
pinchcyl_sc6r_irreg_typ2.inp	Coarse irregular mesh (Type 2).
pinchcyl_sc6r_fineirreg_typ2.inp	Fine irregular mesh (Type 2).
SC8R elements:	
pinchcyl_sc8r_reg55.inp	5 × 5 mesh.
pinchcyl_sc8r_reg1010.inp	10 × 10 mesh.
pinchcyl_sc8r_reg2020.inp	20 × 20 mesh.
pinchcyl_sc8r_stackdir_cylori.inp	20 × 20 mesh using the STACK DIRECTION=ORIENTATION parameter with a cylindrical orientation system to define the element thickness direction.
pinchcyl_sc8r_irreg_typ1.inp	Coarse irregular mesh (Type 1).
pinchcyl_sc8r_fineirreg_typ1.inp	Fine irregular mesh (Type 1).
pinchcyl_sc8r_irreg_typ2.inp	Coarse irregular mesh (Type 2).
pinchcyl_sc8r_fineirreg_typ2.inp	Fine irregular mesh (Type 2).

Abaqus/Explicit input files

pinchcyl_s4r_c3d8r_shell2solid.inp	Shell-to-solid coupling model with S4R shell elements and C3D8R continuum elements.
Large-displacement theory:	
pinch_cyl_coarse_irr1_s4r.inp	S4R element, coarse irregular mesh (Type 1).
pinch_cyl_fine_irr1_s4r.inp	S4R element, fine irregular mesh (Type 1).
pinch_cyl_coarse_irr1_s3r.inp	S3R element, coarse irregular mesh (Type 1).
pinch_cyl_fine_irr1_s3r.inp	S3R element, fine irregular mesh (Type 1).
pinch_cyl_coarse_reg_s4r.inp	S4R element, coarse regular mesh.
pinch_cyl_med_reg_s4r.inp	S4R element, medium regular mesh.
pinch_cyl_fine_reg_s4r.inp	S4R element, fine regular mesh.
pinch_cyl_coarse_reg_s3r.inp	S3R element, coarse regular mesh.
pinch_cyl_med_reg_s3r.inp	S3R element, medium regular mesh.
pinch_cyl_fine_reg_s3r.inp	S3R element, fine regular mesh.
Small-displacement theory:	
pinch_cyl_coarse_irr1_s4r_lk.inp	S4R element, coarse irregular mesh (Type 1).
pinch_cyl_fine_irr1_s4r_lk.inp	S4R element, fine irregular mesh (Type 1).
pinch_cyl_coarse_irr1_s3r_lk.inp	S3R element, coarse irregular mesh (Type 1).

pinch_cyl_fine_irr1_s3r_lk.inp	S3R element, fine irregular mesh (Type 1).
pinch_cyl_coarse_reg_s4r_lk.inp	S4R element, coarse regular mesh.
pinch_cyl_med_reg_s4r_lk.inp	S4R element, medium regular mesh.
pinch_cyl_fine_reg_s4r_lk.inp	S4R element, fine regular mesh.
pinch_cyl_coarse_reg_s3r_lk.inp	S3R element, coarse regular mesh.
pinch_cyl_med_reg_s3r_lk.inp	S3R element, medium regular mesh.
pinch_cyl_fine_reg_s3r_lk.inp	S3R element, fine regular mesh.

References

- Flügge, W., Stresses in Shells, Springer-Verlag, New York, Second, 1973.
- Lindberg, G. M. M., D. Olson, and G. R. Cowper, "New Developments in the Finite Element Analysis of Shells," *Quarterly Bulletin of the Division of Mechanical Engineering and the National Aeronautical Establishment, National Research Council of Canada*, vol. 4, 1969.

Tables

Table 1: Comparison of radial displacement results for pinched cylinder. Regular meshes. Abaqus/Standard analysis.

Element type	Number of dof	Displacement ($\times 10^{-5}$)	Error (compared to 1.825×10^{-5})
STRI3	216	1.134	–38%
	726	1.696	–7%
	2646	1.829	0.2%
S4R5	216	1.099	–40%
	726	1.597	–12%
	2646	1.778	–2.6%
S4	216	0.951	–47.8%
	726	1.519	–16.7%
	2646	1.750	–4.0%
S4R	216	1.089	–40.3%
	726	1.591	–12.8%
	2646	1.779	–2.5%
S4R*	216	0.954	–47.7%
	726	1.525	–16.4%
	2646	1.755	–3.8%
S8R5	726	1.804	–1.1%
	2646	1.833	0.4%
S8R	576	1.721	–5.7%
	2046	1.806	–1%
S9R5	726	1.804	–1.1%

Element type	Number of dof	Displacement ($\times 10^{-5}$)	Error (compared to 1.825×10^{-5})
	2646	1.833	0.4%
STRI65	726	1.358	–25.6%
	2646	1.765	–3.3%
S3R	216	0.653	–64%
	726	1.328	–27%
	2646	1.674	–8.3%
SC6R	216	0.652	–65.7%
	726	1.327	–27.3%
	2649	1.673	–8.3%
SC8R	216	1.123	–38.5%
	726	1.608	–11.9%
	2649	1.784	–2.25%
*Abaqus/Standard finite-strain element with enhanced hourglass control.			

Table 2: Comparison of radial displacement results for pinched cylinder. Regular meshes. Abaqus/Explicit analysis.

Element type	Number of elements	Displacement ($\times 10^{-5}$)	Error (compared to 1.825×10^{-5})
S3R	50	0.767	–58.%
	200	1.390	–24.%
	800	1.703	–6.7%
S4R	25	1.115	–39.%
	100	1.616	–11.%
	400	1.806	–1.0%

Table 3: Comparison of radial displacement results for pinched cylinder. Irregular meshes. Abaqus/Standard analysis.

Element type	Number of dof	Mesh type 1 Displacement ($\times 10^{-5}$)	Error	Mesh type 2 Displacement ($\times 10^{-5}$)	Error
STRI3	894	1.767	–3.2%	1.372	–25%
	3318	1.810	–0.8%	1.663	–9%
S4R5	894	1.815	–0.5%	1.790	–1.9%
	3318	1.835	0.5%	1.842	0.9%
S4R	894	1.814	–0.6%	1.781	–2.4%
	3318	1.849	1.3%	1.862	2.0%
S4R*	894	1.764	–3.3%	1.618	–10.7%

Element type	Number of dof	Mesh type 1 Displacement ($\times 10^{-5}$)	Error	Mesh type 2 Displacement ($\times 10^{-5}$)	Error
	3318	1.840	0.8%	1.845	1.1%
S4	894	1.687	–7.52%	1.454	–20.3%
	3318	1.814	–0.58%	1.777	–2.5%
S8R5	918	1.803	–1.2%	1.519	–17%
	3366	1.793	–1.8%	1.793	–1.8%
S8R	702	1.664	–9%	1.244	–32%
	2550	1.726	–5.4%	1.726	–5.4%
S9R5	918	1.793	–1.8%	1.504	–18%
	3366	1.831	0.3%	1.774	–2.8%
STRI65	918	1.723	–5.6%	1.551	–15.01%
	3366	1.850	1.4%	1.824	–0.05%
S3R	894	1.565	–14%	1.270	–30%
	3318	1.763	–3.4%	1.654	–9.4%
SC6R	894	1.563	–14.3%	1.273	–30.2%
	3318	1.762	–3.4%	1.655	–9.3%
SC8R	894	1.821	–0.22%	1.767	–3.18%
	3318	1.850	1.37%	1.865	2.19%

* Abaqus/Standard finite-strain element with enhanced hourglass control.

Table 4: Comparison of radial displacement results for pinched cylinder. Irregular meshes. Abaqus/Explicit analysis.

Element type	Number of elements	Displacement ($\times 10^{-5}$)	Error (compared to 1.825×10^{-5})
S3R	256	1.618	–11.3%
	1024	1.794	–1.69%
S4R	128	1.848	1.24%
	512	1.883	3.17%

Table 5: Comparison of radial displacement results for submodeled analyses in Abaqus/Standard. Reference solution: 1.825×10^{-5}

Global Element Type	Global Mesh Size	Submodel Mesh Size	Displacement ($\times 10^{-5}$)	Error
S4R	5 × 5	n/a	1.092	–40.2%
		5 × 5	1.6139	–11.6%
		10 × 10	1.6259	–10.9%
	10 × 10	n/a	1.592	–12.8%

Global Element Type	Global Mesh Size	Submodel Mesh Size	Displacement ($\times 10^{-5}$)	Error
		5 × 5	1.7775	–2.6%
		10 × 10	1.7881	–2.0%
S8R	5 × 5	n/a	1.721	–5.7%
		5 × 5	1.8004	–1.3%
		10 × 10	1.8123	–0.7%

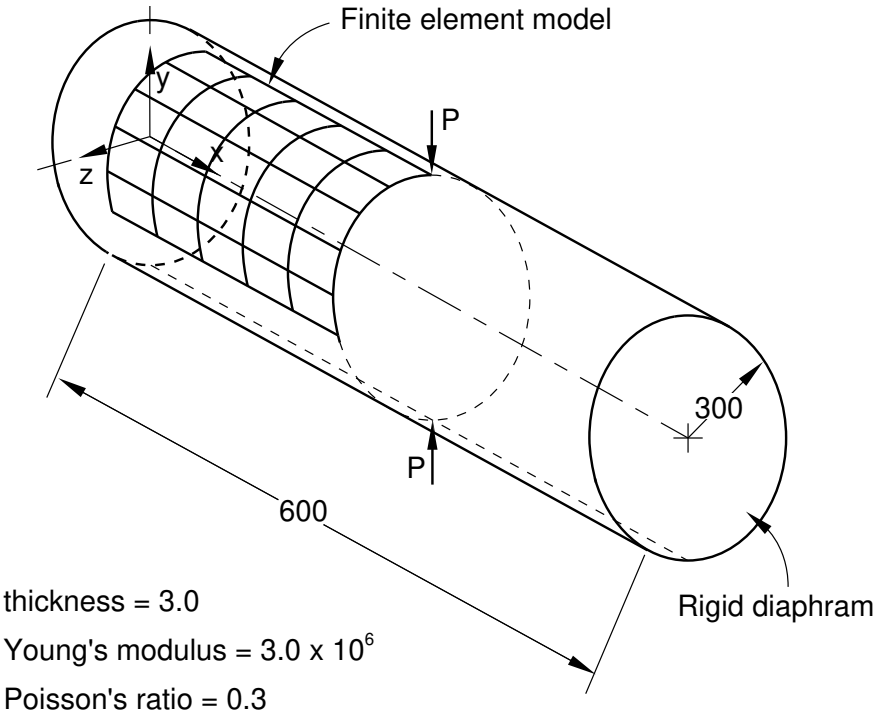
Table 6: Comparison of radial displacement results for Abaqus/Standard shell-to-solid coupling analyses.

Shell element	Continuum element	Displacement ($\times 10^{-5}$)	Error (compared to 1.825×10^{-5})
S4R	C3D8I	1.750	–4.11%
S4R	C3D10	1.775	–2.74%
S4R	C3D10HS	1.775	–2.74%
S4R	C3D20R	1.837	0.656%
S8R	C3D8I	1.766	–3.23%
S8R	C3D10	1.797	–1.53%
S8R	C3D10HS	1.797	–1.53%
S8R	C3D20R	1.854	1.59%

Table 7: Comparison of radial displacement results for Abaqus/Explicit shell-to-solid coupling analyses.

Shell element	Continuum element	Displacement ($\times 10^{-5}$)	Error (compared to 1.825×10^{-5})
S4R	C3D8R	1.24	–32%

Figures



thickness = 3.0

Young's modulus = 3.0×10^6

Poisson's ratio = 0.3

Figure 1: Pinched cylinder example.

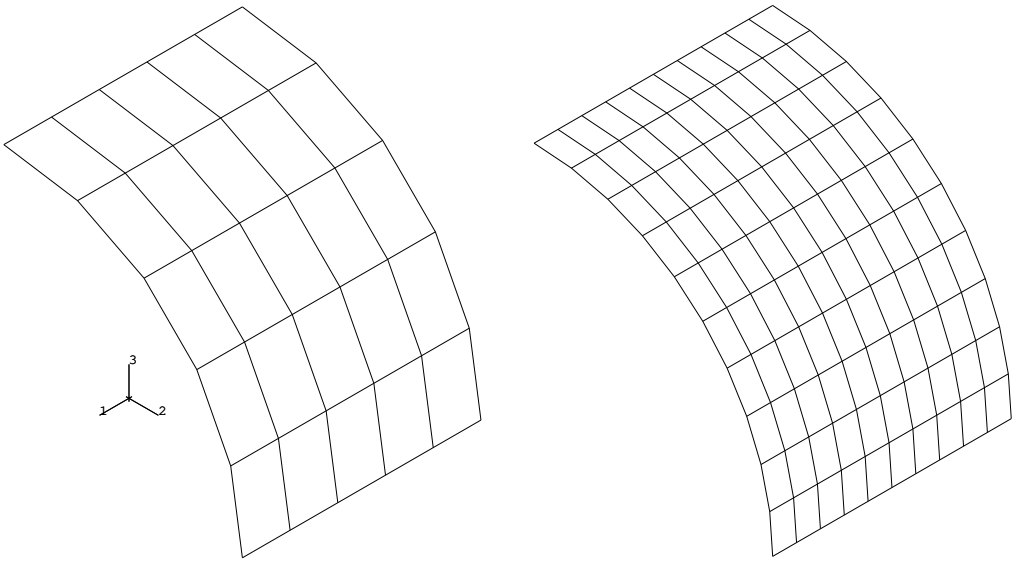


Figure 2: Regular meshes.

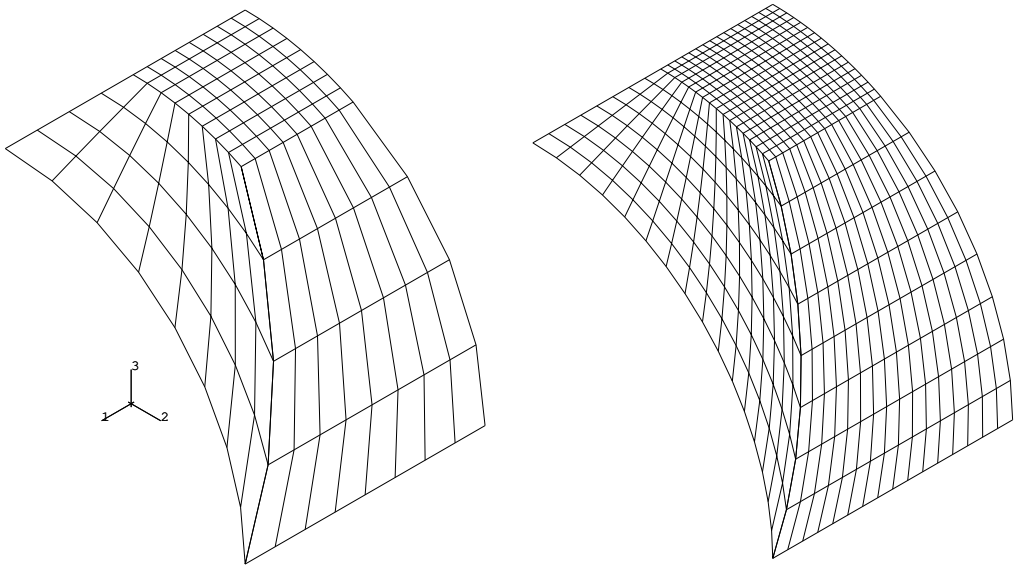


Figure 3: Irregular meshes of type 1.

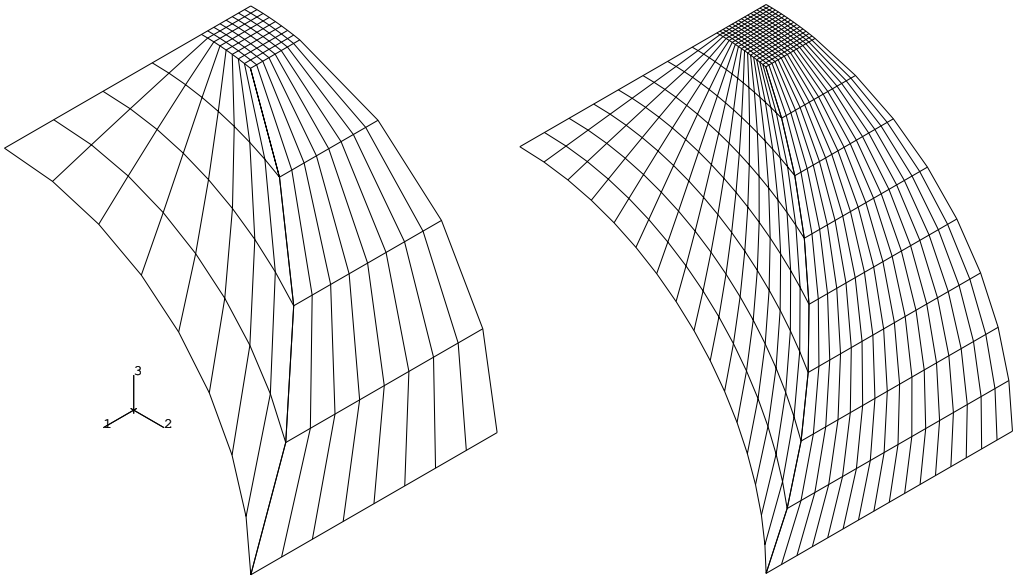


Figure 4: Irregular meshes of type 2.

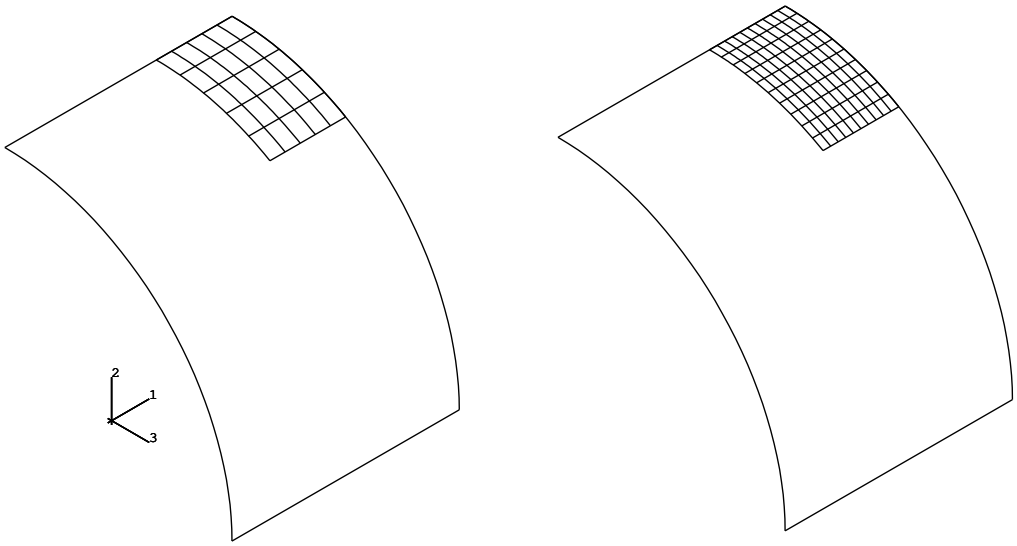


Figure 5: Superimposed submodel meshes.

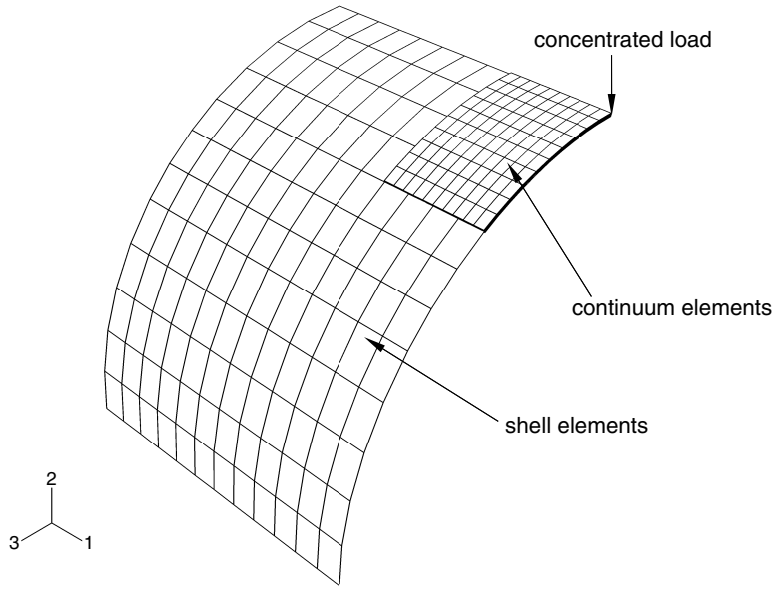


Figure 6: Shell-to-solid coupling mesh.

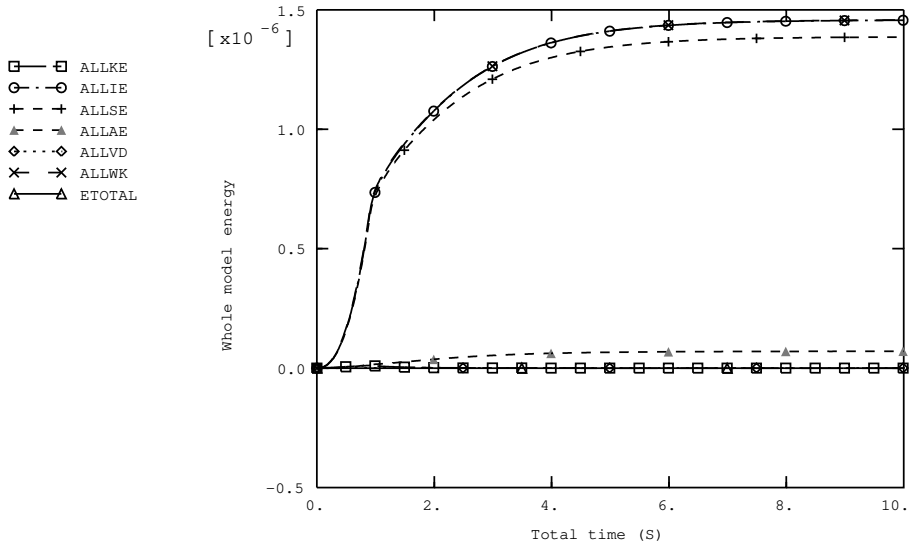
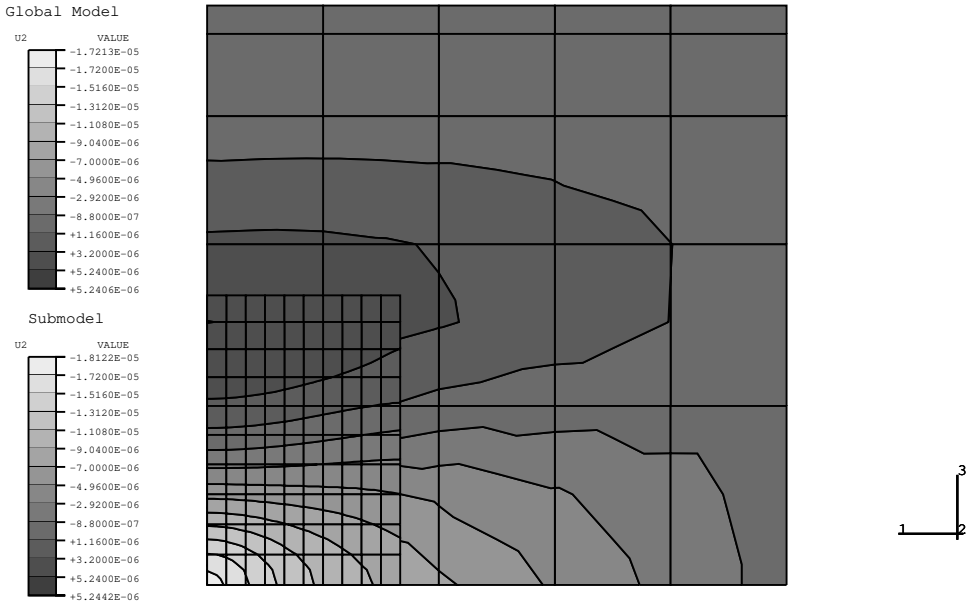


Figure 7: Energy plot for `pinch_cyl_coarse_irr1_s4r.inp` (Abaqus/Explicit).



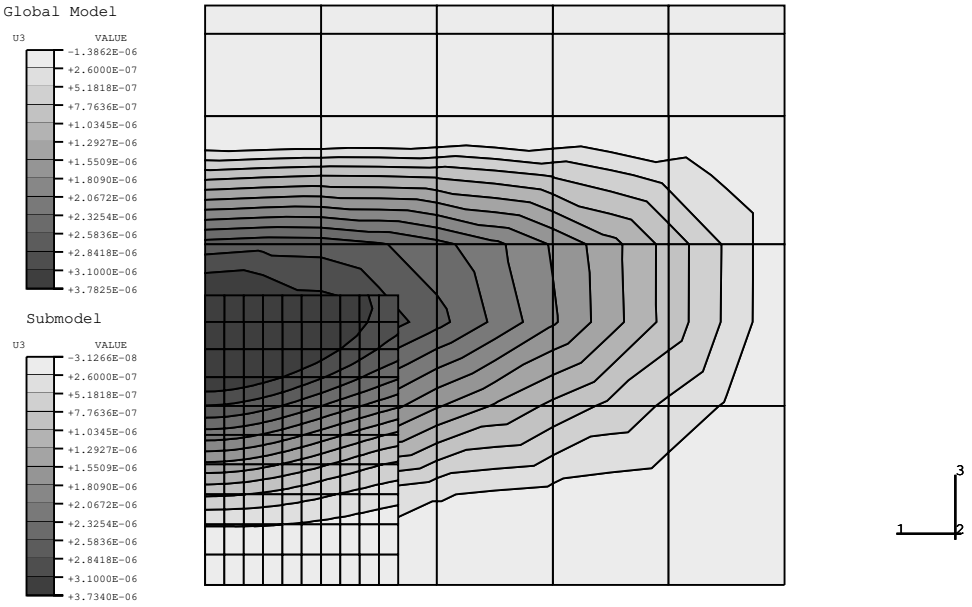


Figure 8: y- and z-displacement contours superimposed on the global analysis (Abaqus/Standard).

The pinched sphere problem

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example verifies the use of the axisymmetric shell elements to analyze a pinched sphere.

Most of the response is localized, so the case represents a more severe test than, for example, a sphere with internal pressure. Koiter (1963) has provided an analytical solution, which has been used as a standard test for several axisymmetric shell finite elements (see Ashwell and Gallagher, 1976).

Problem description

The physical problem consists of a hollow sphere with opposing point loads acting along a diameter of the sphere (see *Figure 1*). Taking advantage of symmetry in the model, only one-half of the sphere is modeled. The meshes used for this example are uniform, although in an actual problem that exhibits such localized response the mesh should be refined to concentrate the elements in the region where the strain gradients are most severe. Four axisymmetric meshes are used for the Abaqus/Standard analysis. For the linear 2-node element, SAX1, the meshes have 10 and 20 elements. For the quadratic 3-node element, SAX2, the meshes have 5 and 10 elements. Two three-dimensional models made up of a single strip of shell or continuum shell elements that subtend an arc of 6° in the circumferential direction are also tested in Abaqus/Standard. Two SAX1 meshes are used for the Abaqus/Explicit analysis, one with 10 elements and the other with 20 elements.

In the Abaqus/Explicit analysis the quasi-static pinching load is applied as a ramp function during the initial 10% of the time period for the step and is then held constant for the remainder of the step. Additionally, viscous pressure loading is applied to the structure to damp out dynamic effects. The time period for the step and the viscous pressure are chosen to obtain an optimal static solution.

Results and discussion

Figure 2 shows the numerical predictions of radial displacement compared with Koiter's (1963) exact solution. The radial displacement falls off very quickly and is essentially zero at angles more than 15° from the top of the sphere. In the coarser meshes of linear elements (SAX1) each element subtends a 9° arc; each quadratic element in the coarser mesh subtends 18°. Thus, the models are rather coarse compared to the variation of the exact solution.

Nonlinear geometric effects in this problem can be ignored because the displacements and rotations are small. By default, Abaqus/Explicit accounts for all geometric nonlinearities; the default can be overridden. The results for this problem are independent of whether or not the geometric nonlinearities are taken into account, which demonstrates that the small-displacement and the large-displacement deformation theories correctly converge to the same results when the strains and rotations are small. The advantage of using the small-displacement deformation theory in Abaqus/Explicit is a significant reduction in CPU time (often about 30%) for analyses that are dominated by element calculations.

The results for each of the meshes are summarized in *Table 1*, where the displacement at the pole of the shell is used as a representative measure of the solutions. The table indicates that the quadratic element meshes converge more rapidly than the linear element meshes with the same number of nodes. This is a common observation and reflects the higher theoretical convergence rate, with respect to element size, that is available with higher-order interpolation.

The explicit dynamic analysis is run until a steady, static solution is obtained. *Figure 3* shows an energy balance plot for the 20-element mesh. It can be seen that inertia effects are damped out very quickly.

Input files

Abaqus/Standard input files

pinchedsphere_s4r_ele20.inp	S4R, 20-element model.
pinchedsphere_sc8r_ele20.inp	SC8R, 20-element model.
pinchedsphere_sc8r_ele20_eh.inp	SC8R, 20-element model with enhanced hourglass control.
pinchedsphere_sax1_ele10.inp	SAX1, 10-element model.
pinchedsphere_sax1_ele20.inp	SAX1, 20-element model.
pinchedsphere_sax2_ele5.inp	SAX2, 5-element model.
pinchedsphere_sax2_ele10.inp	SAX2, 10-element model.

Abaqus/Explicit input files

pinch_sph_coarse.inp	10-element model, large-displacement analysis.
pinch_sph_fine.inp	20-element model, large-displacement analysis.
pinch_sph_coarse_lk.inp	10-element model, small-displacement analysis.
pinch_sph_fine_lk.inp	20-element model, small-displacement analysis.

References

- Ashwell, D. G., and R. H. Gallagher, Finite Elements for Thin Shells and Curved Members, John Wiley and Sons, London, 1976.
- Koiter, W. T., "A Spherical Shell Under Point Loads at Its Poles," *Progress in Applied Mechanics: The Prager Anniversary Volume, Macmillan, New York*, 1963.

Tables

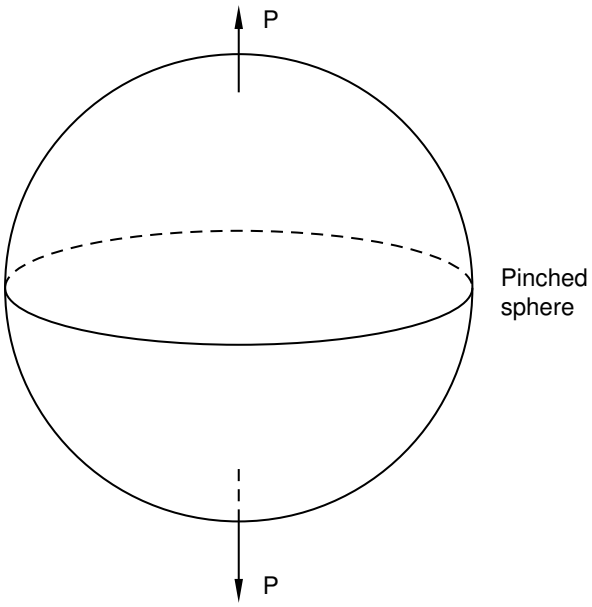
Table 1: Displacement at top of sphere.

Element type	Number of elements	Normalized displacement	Error
SAX1, Abaqus/Standard	10	15.52	–26.8%
SAX1, Abaqus/Standard	20	19.84	–6.4%
SAX1, Abaqus/Explicit	10	24.86	17.3%
SAX1, Abaqus/Explicit	20	24.88	17.3%
SAX2	5	15.62	–26.3%
SAX2	10	20.12	–5.1%
S4R	20	19.80	–6.6%
SC8R	20	19.80	–6.6%
SC8R*	20	13.24	–37.5%

The normalized displacement is wEt/P, where w is the actual displacement; E is Young's modulus; t is the shell thickness; and P is the applied load.

Element type	Number of elements	Normalized displacement	Error
Koiter's (1963) exact solution gives a normalized displacement of 21.2.			
*Abaqus/Standard results with enhanced hourglass control.			

Figures

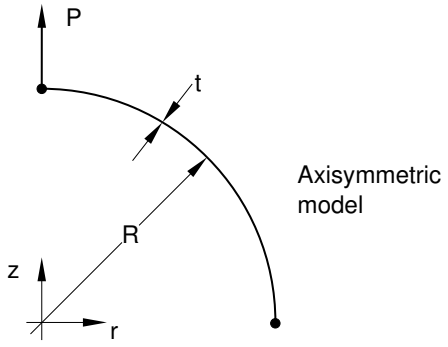


Model parameters:

P = 44.48 N (10.0 lb)

R = 508 mm (20.0 in)

t = 10.16 mm (0.4 in)



Material:

E = 69.0 GPa
(10.0 x 10^6 lb/in^2)

υ = 0.3

Figure 1: Pinched sphere example.

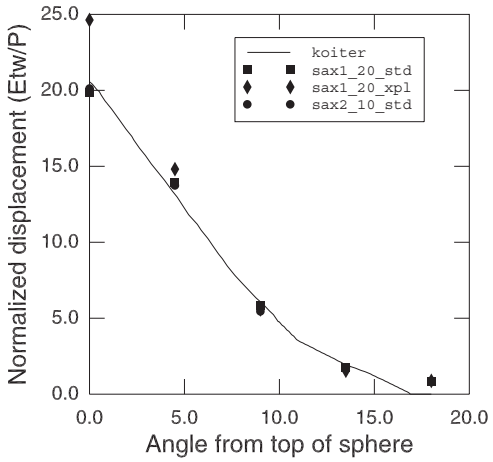


Figure 2: Radial displacement versus angle measured from the point of load application.

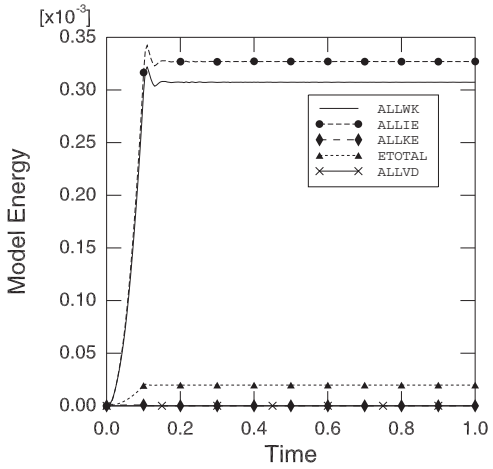


Figure 3: Energy balance for 20-element model, Abaqus/Explicit analysis.

Skew sensitivity of shell elements

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example illustrates the sensitivity of the shell elements in Abaqus to skew distortion when they are used as thin plates.

An analytical series solution to the boundary value problem is available in Morley (1963), and an identical evaluation of elements in numerous other commercial codes is presented by Robinson (1985).

Problem description

The geometry of the plate is shown in *Figure 1*, *Figure 2*, and *Figure 3*. The analysis is performed for five different values of the skew angle, δ: 90°, 80°, 60°, 40°, and 30°. Three meshes (4 × 4, 8 × 8, and 14 × 14) are used for each skew angle in the Abaqus/Standard analysis. In the Abaqus/Explicit analysis 4 × 4, 8 × 8, and 14 × 14 meshes are used for each skew angle with the quadrilateral elements and 2 × 2 × 4, 4 × 4 × 4, and 8 × 8 × 4 meshes are used for each skew angle with the triangular elements.

The plate is 10 mm thick. All sides are 1.0 m long. The length/thickness ratio is, thus, 100/1 so that the plate is thin in the sense that transverse shear deformation should not be significant. Young's modulus is 30 MPa, and Poisson's ratio is 0.3. The plate is loaded by a uniform pressure of 1.0×10^{-6} MPa applied over the entire surface. The edges of the plate are all simply supported.

The pressure is applied as a step function in the Abaqus/Explicit analysis. Viscous pressure loading is applied to the structure to damp out dynamic effects. The time period for the step and the viscous pressure are chosen to obtain an optimal static solution.

Results and discussion

Three response quantities are presented: the vertical displacement in the center of the plate, w_{center}, and the maximum and minimum bending moments per unit length at the center of the plate, defined as

$$\begin{aligned} M_{\text{max}} &= M_a + M_b, \\ M_{\text{min}} &= M_a - M_b, \end{aligned}$$

where

$$\begin{aligned} M_a &= \frac{1}{2}\left(M_x + M_y\right), \\ M_b &= \sqrt{\frac{1}{4}\left(M_x - M_y\right)^2 + {M_{xy}}^2}. \end{aligned}$$

The bending moment values M_x, M_y, and M_{xy} are obtained from the average nodal values obtained by requesting element output to the data file in the Abaqus/Standard analysis. These values are calculated by extrapolation from the integration point values in the elements, followed by averaging of these values over all elements attached to the node. They are, therefore, less accurate than the values at the integration points. In the Abaqus/Explicit analysis the bending moment values are obtained from an average of the integration point values for all elements that share the node at the center of the plate.

Abaqus/Standard results

The results for the 3-node triangular shells, S3R and STRI3, are given in *Table 1* and *Table 2*, respectively. These elements give reasonable results for all skew angles with all but the coarsest mesh used (4×4 elements).

The results for the 6-node triangular shell STRI65 are given in *Table 3*. This element gives reasonable results for all the skew angles with the various mesh discretizations, with the exception of the coarsest mesh used.

The results for the 4-node quadrilateral shells are presented in *Table 4* (S4R5), *Table 5* (S4R), and *Table 6* (S4). The performance of these elements in this case is rather similar to that of the triangular elements.

The results for element types S8R5 and S9R5, presented in *Table 7*, are essentially identical to each other. These second-order elements are more sensitive to the distortion in this problem than the first-order elements. For 80° and 90° angles they give slightly more accurate displacement values than S4R5; but at more severe angles their performance deteriorates noticeably, particularly in the prediction of the minimum moment at the center of the plate. It is possible that this is caused by the extrapolation and averaging technique used to obtain nodal values of bending moments rather than an intrinsic sensitivity of the elements to this type of distortion.

The results for element type S8R are given in *Table 8*. Except with the finest mesh used, this element generally shows greater loss of accuracy as the plate is skewed than any of the other elements.

The results for the continuum shell elements SC6R and SC8R are presented in *Table 9* and *Table 10*. The performance of these elements is similar to that of the S3R and S4R shell elements.

Abaqus/Explicit results

The explicit dynamic analysis is run until a steady, static solution is obtained. *Figure 4* shows an energy balance plot for the 14×14 mesh with a skew angle of 40°. It can be seen that inertia effects have died away.

The results for the 3-node triangular shell, S3R, are given in *Table 11*. These elements exhibit stiff response for the coarsest mesh used ($2 \times 2 \times 4$ elements) but converge to the correct answer as the mesh density is increased.

The results for the 4-node quadrilateral shells, S4R and S4RS, are presented in *Table 12* and *Table 13*, respectively. For all but the 40° and 30° skew angles, the S4R elements give reasonable answers for the coarsest mesh used. As the mesh density is increased, the elements converge to the analytical solutions for all skew angles.

The results for the continuum shell element SC8R are presented in *Table 14*. The performance of this element is similar to that of the S4R shell element.

General remarks

Abaqus gives a warning when quadrilateral elements are defined with skew distortions larger than 45°. The results in this case indicate that, with the possible exception of element type S8R, the elements can provide quite accurate results with reasonable meshes even with large skew distortions. Nevertheless it is also clear that the analyst should attempt to design meshes to avoid distortion of the elements in any region where there are large strain gradients.

Comparison of the results reported here with the evaluations given by Robinson (1985) indicate that the elements in Abaqus are among the most accurate and least sensitive to skew angle.

Parametric study using a parametric study script

The skew sensitivity investigation discussed in this example can be performed conveniently as a parametric study using the Python scripting capabilities offered in Abaqus. As an example we perform a parametric study in Abaqus/Standard in which 15 analyses are automatically executed; these analyses correspond to combinations of five different values of the skew angle (δ: 90°, 80°, 60°, 40°, and 30°) for three different element types (S8R, S4R, and S4). We also perform a parametric study in Abaqus/Explicit in which 12 analyses are executed

automatically; these analyses correspond to combinations of three different values of the skew angle (δ: 90°, 60°, and 30°), two different element types (S4R and S4RS), and two mesh discretizations (4 × 4 and 8 × 8 elements).

skewshell_parametric.inp shows the parametrized template input data used to generate the parametric variations of the Abaqus/Standard parametric study. The parametric study script file (*skewshell_parametric.psf*) is used to perform the parametric study. The vertical displacement in the center of the plate is reported in the following table for each of the analyses of the parametric study:

Parametric study: skewshell_parametric

elemType,	delta,	N405_U.3,
s8r,	90,	-0.00150858,
s4r,	90,	-0.00149891,
s4,	90,	-0.00144697,
s8r,	80,	-0.00141673,
s4r,	80,	-0.00143168,
s4,	80,	-0.00137446,
s8r,	60,	-0.000845317,
s4r,	60,	-0.000969093,
s4,	60,	-0.000885679,
s8r,	40,	-0.000258699,
s4r,	40,	-0.000371343,
s4,	40,	-0.000315966,
s8r,	30,	-9.5434e-05,
s4r,	30,	-0.000153366,
s4,	30,	-0.000130785,

These results match the corresponding results found in *Table 5* to *Table 8*.

skew_discr.inp shows the parametrized template input data used to generate the parametric variations for the Abaqus/Explicit parametric study. The parametric study script file (*skew_discr.psf*) is used to perform the parametric study. The vertical displacement at the center of the plate is reported in the following table for each analysis of the parametric study:

Parametric study: skewXpl

level,	elemType,	delta,	N405_U.3,
1,	s4r,	90,	-0.00144092,
2,	s4r,	90,	-0.00144511,
1,	s4rs,	90,	-0.00155302,
2,	s4rs,	90,	-0.00147813,
1,	s4r,	60,	-0.000954238,
2,	s4r,	60,	-0.000925741,
1,	s4rs,	60,	-0.00102325,
2,	s4rs,	60,	-0.000963277,
1,	s4r,	30,	-0.000151794,
2,	s4r,	30,	-0.000148982,
1,	s4rs,	30,	-0.000161744,
2,	s4rs,	30,	-0.000162303,

The results match the corresponding results found in *Table 11* to *Table 13*.

Input files

Abaqus/Standard input files

skewshell_typ_tri.inp	Typical input data for a triangular element.
skewshell_typ_quad.inp	Typical input data for a quadrilateral element.
skewshell_parametric.inp	Parametrized template input data used to generate the parametric variations of the parametric study.

The additional files included with the Abaqus release for this example have the following naming convention:

$$\text{skewshell_}\begin{Bmatrix}\text{s3r}\\ \text{s4}\\ \text{s4r}\\ \text{s4r5}\\ \text{s8r}\\ \text{s8r5}\\ \text{s9r5}\\ \text{stri3}\\ \text{stri65}\\ \text{sc6r}\\ \text{sc8r}\end{Bmatrix}\text{_}\begin{Bmatrix}\text{4x4}\\ \text{8x8}\\ \text{14x14}\end{Bmatrix}\text{_}\begin{Bmatrix}\text{ang30}\\ \text{ang40}\\ \text{ang60}\\ \text{ang80}\\ \text{ang90}\end{Bmatrix}\text{_}\{\text{eh}\}\text{.inp}$$

An “eh” in the input file name indicates that enhanced hourglass control was used in the analysis.

Abaqus/Explicit input files

skew_coarse_90_s4r.inp	Coarse (4×4) mesh using S4R elements for a skew angle of 90°.
skew_discr.inp	Parametrized template input data used to generate the parametric variations of the parametric study.

The additional files included with the Abaqus release for this example have the following naming convention:

$$\text{skew_}\begin{Bmatrix}\text{coarse}\\ \text{medium}\\ \text{fine}\end{Bmatrix}\text{_}\begin{Bmatrix}\text{90}\\ \text{80}\\ \text{60}\\ \text{40}\\ \text{30}\end{Bmatrix}\text{_}\begin{Bmatrix}\text{s3r}\\ \text{s4r}\\ \text{s4rs}\\ \text{sc8r}\end{Bmatrix}\text{.inp}$$

References

- Morley, L. S. D., Skew Plates and Structures, Pergamon Press, London, 1963.
- Robinson, J., “An Evaluation of Skew Sensitivity of Thirty-Three Plate Bending Elements in Nineteen FEM Systems,” *paper presented at the Finite Element Standards Forum at the AIAA/ASME/ASCE/AHS 26th Structures, Structural Dynamics, and Materials Conference*, April 1985.

Tables

Table 1: Skewed plate results: S3R, Abaqus/Standard analysis.

Skew angle	Mesh	w_{center}		$M_{\max}$		$M_{\min}$	
		(mm)	Error	(× 10^{-2} N-m/m)	Error	(× 10^{-2} N-m/m)	Error
90°	Series						
	solution	1.478		4.79		4.79	
	4 × 4	1.214	−17.9%	4.03	−15.9%	3.97	−17.1%
	8 × 8	1.425	−3.6%	4.86	1.5%	4.84	1.0%
	14 × 14	1.462	−1.1%	4.81	0.4%	4.80	0.2%
80°	Series						
	solution	1.409		4.86		4.48	
	4 × 4	1.148	−18.5%	4.09	−15.8%	3.60	−19.6%
	8 × 8	1.343	−4.7%	4.91	1.0%	4.44	−0.9%
	14 × 14	1.391	−1.3%	4.87	0.2%	4.51	0.7%
60°	Series						
	solution	0.932		4.25		3.33	
	4 × 4	0.615	−34.0%	2.98	−29.9%	1.93	−42.0%
	8 × 8	0.812	−12.9%	3.82	−10.1%	2.82	−15.3%
	14 × 14	0.913	−2.0%	4.19	−1.4%	3.31	−0.6%
40°	Series						
	solution	0.349		2.81		1.80	
	4 × 4	0.213	−39.0%	1.86	−33.8%	0.88	−51.1%
	8 × 8	0.292	−16.3%	2.42	−13.9%	1.39	−22.8%
	14 × 14	0.346	−0.8%	2.81	0.0%	1.82	1.1%
30°	Series						
	solution	0.148		1.91		1.08	
	4 × 4	0.080	−45.9%	1.14	−40.3%	0.46	−57.4%
	8 × 8	0.125	−15.5%	1.60	−16.2%	0.80	−25.9%
	14 × 14	0.148	0.0%	1.89	−1.0%	1.08	0.0%

Table 2: Skewed plate results: STRI3, Abaqus/Standard analysis.

Skew angle	Mesh	w_{center}		$M_{\max}$		$M_{\min}$	
		(mm)	Error	(× 10^{-2} N-m/m)	Error	(× 10^{-2} N-m/m)	Error
90°	Series						
	solution	1.478		4.79		4.79	
	4 × 4	1.488	0.7%	5.22	8.9%	5.22	8.9%

Skew angle	Mesh	w_{center}		M_{max}		M_{min}	
		(mm)	Error	(× 10^{-2} N-m/m)	Error	(× 10^{-2} N-m/m)	Error
	8 × 8	1.481	0.2%	4.89	2.0%	4.89	2.0%
	14 × 14	1.480	0.1%	4.82	0.6%	4.82	0.6%
80°	Series						
	solution	1.409		4.86		4.48	
	4 × 4	1.419	0.7%	5.37	10%	4.83	7.8%
	8 × 8	1.410	0.1%	4.98	2.4%	4.57	2.0%
	14 × 14	1.409	0.0%	4.89	0.7%	4.51	0.7%
60°	Series						
	solution	0.932		4.25		3.33	
	4 × 4	0.965	3.5%	4.86	14%	3.62	8.8%
	8 × 8	0.940	0.8%	4.43	4.2%	3.41	2.4%
	14 × 14	0.935	0.3%	4.31	1.4%	3.36	0.9%
40°	Series						
	solution	0.349		2.81		1.80	
	4 × 4	0.390	12%	3.40	21%	2.15	19%
	8 × 8	0.363	4.2%	3.05	8.5%	1.93	7.4%
	14 × 14	0.357	2.4%	2.91	3.4%	1.87	4.1%
30°	Series						
	solution	0.148		1.91		1.08	
	4 × 4	0.173	16%	2.35	23%	1.35	25%
	8 × 8	0.158	6.6%	2.12	11%	1.22	13%
	14 × 14	0.154	3.8%	2.01	5.3%	1.16	7.5%

Table 3: Skewed plate results: STRI65, Abaqus/Standard analysis.

Skew angle	Mesh	w_{center}		M_{max}		M_{min}	
		(mm)	Error	(× 10^{-2} N-m/m)	Error	(× 10^{-2} N-m/m)	Error
90°	Series						
	solution	1.478		4.79		4.79	
	4 × 4	1.481	–0.2%	5.11	6.7%	4.99	4.2%
	8 × 8	1.486	0.5%	4.91	2.5%	4.87	1.7%
	14 × 14	1.484	0.4%	4.83	0.8%	4.81	0.4%
80°	Series						
	solution	1.409		4.86		4.48	
	4 × 4	1.377	–2.3%	4.89	0.6%	4.65	3.8%
	8 × 8	1.413	0.3%	4.93	1.4%	4.61	2.9%
	14 × 14	1.414	0.3%	4.89	0.6%	4.54	1.3%

Skew angle	Mesh	w_{center}		M_{max}		M_{min}	
		(mm)	Error	(× 10^{-2} N-m/m)	Error	(× 10^{-2} N-m/m)	Error
60°	Series						
	solution	0.932		4.25		3.33	
	4 × 4	0.825	–3.5%	3.95	–7%	3.06	–8.2%
	8 × 8	0.919	–0.8%	4.27	0.5%	3.36	0.9%
	14 × 14	0.934	0.3%	4.27	0.5%	3.36	0.9%
40°	Series						
	solution	0.349		2.81		1.80	
	4 × 4	0.273	–22%	2.45	–13%	1.41	–21%
	8 × 8	0.333	–4.8%	2.76	–1.8%	1.73	–3.8%
	14 × 14	0.350	0.6%	2.81	0.0%	1.82	1.1%
30°	Series						
	solution	0.148		1.91		1.08	
	4 × 4	0.114	–23%	1.64	23%	0.80	–25%
	8 × 8	0.143	–3.4%	1.87	11%	1.03	–5%
	14 × 14	0.152	2.7%	1.92	5.3%	1.11	2.7%

Table 4: Skewed plate results: S4R5, Abaqus/Standard analysis.

Skew angle	Mesh	w_{center}		M_{max}		M_{min}	
		(mm)	Error	(× 10^{-2} N-m/m)	Error	(× 10^{-2} N-m/m)	Error
90°	Series						
	solution	1.478		4.79		4.79	
	4 × 4	1.502	1.6%	4.23	–12%	4.23	–12%
	8 × 8	1.485	0.5%	4.65	–2.8%	4.65	–2.8%
	14 × 14	1.482	0.3%	4.75	–0.9%	4.75	–0.9%
80°	Series						
	solution	1.409		4.86		4.48	
	4 × 4	1.436	1.9%	4.29	–11%	3.96	–12%
	8 × 8	1.415	0.5%	4.71	–3.0%	4.36	–2.6%
	14 × 14	1.412	0.2%	4.81	–1.0%	4.45	–0.7%
60°	Series						
	solution	0.932		4.25		3.33	
	4 × 4	0.981	5.3%	3.78	–11%	2.88	–14%
	8 × 8	0.943	1.2%	4.12	–3.1%	3.25	–2.3%
	14 × 14	0.937	0.5%	4.21	–0.9%	3.31	–0.7%
40°	Series						
	solution	0.349		2.81		1.80	

Skew angle	Mesh	w_{center}		M_{max}		M_{min}	
		(mm)	Error	(× 10^{-2} N-m/m)	Error	(× 10^{-2} N-m/m)	Error
	4 × 4	0.384	10%	2.58	−8.1%	1.45	−19%
	8 × 8	0.365	4.8%	2.74	−2.6%	1.80	−0.0%
	14 × 14	0.357	2.3%	2.79	−0.8%	1.83	−1.7%
30°	Series						
	solution	0.148		1.91		1.08	
	4 × 4	0.160	7.7%	1.74	−9.0%	0.80	−26%
	8 × 8	0.160	7.9%	1.89	−1.3%	1.08	0.0%
	14 × 14	0.155	4.6%	1.91	−0.2%	1.13	4.3%

Table 5: Skewed plate results: S4R, Abaqus/Standard analysis.

Skew angle	Mesh	w_{center}		M_{max}		M_{min}	
		(mm)	Error	(× 10^{-2} N-m/m)	Error	(× 10^{-2} N-m/m)	Error
90°	Series						
	solution	1.478		4.79		4.79	
	4 × 4	1.498	1.4%	4.22	−12%	4.22	−12%
	8 × 8	1.485	0.5%	4.65	−2.9%	4.65	−2.9%
	14 × 14	1.483	0.3%	4.75	−0.9%	4.75	−0.9%
80°	Series						
	solution	1.409		4.86		4.48	
	4 × 4	1.431	1.6%	4.28	−12%	3.94	−12%
	8 × 8	1.415	0.4%	4.71	−3.0%	4.36	−2.7%
	14 × 14	1.414	0.4%	4.81	−0.9%	4.45	−0.7%
60°	Series						
	solution	0.932		4.25		3.33	
	4 × 4	0.969	4.0%	3.76	−12%	2.84	−15%
	8 × 8	0.937	0.5%	4.11	−3.4%	3.23	−3.1%
	14 × 14	0.936	0.4%	4.21	−0.9%	3.30	−1.0%
40°	Series						
	solution	0.349		2.81		1.80	
	4 × 4	0.371	6.3%	2.52	−10%	1.41	−22%
	8 × 8	0.353	1.1%	2.68	−4.5%	1.72	−4.4%
	14 × 14	0.351	0.4%	2.76	−1.9%	1.77	−1.4%
30°	Series						
	solution	0.148		1.91		1.08	
	4 × 4	0.153	3.4%	1.70	−11%	0.78	−27%
	8 × 8	0.151	2.0%	1.82	−4.8%	1.00	−6.9%

Skew angle	Mesh	w_{center}		M_{max}		M_{min}	
		(mm)	Error	(× 10^{-2} N-m/m)	Error	(× 10^{-2} N-m/m)	Error
	14 × 14	0.149	0.7%	1.86	–2.7%	1.05	–2.5%
	4 × 4*	0.156	5.4%	1.72	11%	0.79	–27%
	8 × 8*	0.155	4.7%	1.86	2.6%	1.05	–2.7%
	14 × 14*	0.150	1.3%	1.88	–1.5%	1.10	–1.8%
*Abaqus/Standard finite-strain element with enhanced hourglass control.							

Table 6: Skewed plate results: S4, Abaqus/Standard analysis.

Skew angle	Mesh	w_{center}		M_{max}		M_{min}	
		(mm)	Error	(× 10^{-2} N-m/m)	Error	(× 10^{-2} N-m/m)	Error
90°	Series						
	solution	1.478		4.79		4.79	
	4 × 4	1.447	–2.1%	4.78	–0.2%	4.78	–0.2%
	8 × 8	1.474	–0.3%	4.80	0.2%	4.80	0.2%
	14 × 14	1.481	0.2%	4.80	0.2%	4.80	0.2%
80°	Series						
	solution	1.409		4.86		4.48	
	4 × 4	1.375	–2.4%	4.84	–0.4%	4.52	0.9%
	8 × 8	1.402	–0.5%	4.86	0.0%	4.50	0.4%
	14 × 14	1.410	0.1%	4.86	0.0%	4.50	0.4%
60°	Series						
	solution	0.932		4.86		3.33	
	4 × 4	0.886	–2.4%	4.84	–0.4%	3.38	1.5%
	8 × 8	0.910	–0.5%	4.86	0.0%	3.31	–0.6%
	14 × 14	0.925	0.1%	4.86	0.0%	3.32	–0.3%
40°	Series						
	solution	0.349		2.81		1.80	
	4 × 4	0.316	–4.9%	2.63	–6.4%	1.70	–5.6%
	8 × 8	0.323	–2.4%	2.71	–3.6%	1.74	–3.3%
	14 × 14	0.327	–0.8%	2.76	–1.9%	1.77	–1.4%
30°	Series						
	solution	0.148		1.91		1.08	
	4 × 4	0.131	–11.0%	1.67	–12.6%	0.92	–14.8%
	8 × 8	0.133	–10.1%	1.80	–5.8%	1.02	–5.6%
	14 × 14	0.141	–4.7%	1.85	–3.1%	1.06	–1.9%

Table 7: Skewed plate results: S8R5, S9R5; Abaqus/Standard analysis.

Skew angle	Mesh	w_{center}		M_{max}		M_{min}	
		(mm)	Error	(× 10^{-2} N-m/m)	Error	(× 10^{-2} N-m/m)	Error
90°	Series						
	solution	1.478		4.79		4.79	
	4 × 4	1.483	0.3%	5.16	7.8%	5.16	7.8%
	8 × 8	1.481	0.2%	4.88	1.9%	4.88	1.9%
	14 × 14	1.483	0.3%	4.82	0.7%	4.82	0.7%
80°	Series						
	solution	1.409		4.86		4.48	
	4 × 4	1.413	0.3%	5.18	6.5%	4.91	9.6%
	8 × 8	1.411	0.2%	4.94	1.6%	4.59	2.4%
	14 × 14	1.413	0.3%	4.89	0.6%	4.53	1.0%
60°	Series						
	solution	0.932		4.25		3.33	
	4 × 4	0.945	1.4%	4.43	4.3%	3.84	15%
	8 × 8	0.937	0.6%	4.31	1.4%	3.45	3.5%
	14 × 14	0.938	0.7%	4.28	0.8%	3.38	1.5%
40°	Series						
	solution	0.349		2.81		1.80	
	4 × 4	0.370	6.0%	2.92	4.0%	2.35	31%
	8 × 8	0.357	2.5%	2.85	1.3%	1.97	9.3%
	14 × 14	0.357	2.5%	2.85	1.4%	1.88	4.7%
30°	Series						
	solution	0.148		1.91		1.08	
	4 × 4	0.164	10%	2.05	7.4%	1.51	40%
	8 × 8	0.156	4.9%	1.94	1.7%	1.26	16%
	14 × 14	0.155	4.5%	1.95	2.2%	1.17	8.2%

Table 8: Skewed plate results: S8R, Abaqus/Standard analysis.

Skew angle	Mesh	w_{center}		M_{max}		M_{min}	
		(mm)	Error	(× 10^{-2} N-m/m)	Error	(× 10^{-2} N-m/m)	Error
90°	Series						
	solution	1.478		4.79		4.79	
	4 × 4	1.509	2.1%	5.22	9.1%	5.22	9.1%
	8 × 8	1.494	1.1%	4.91	2.5%	4.91	2.5%
	14 × 14	1.492	1.0%	4.85	1.2%	4.85	1.2%
80°	Series						

Skew angle	Mesh	w_{center}		M_{max}		M_{min}	
		(mm)	Error	(× 10^{-2} N-m/m)	Error	(× 10^{-2} N-m/m)	Error
	solution	1.409		4.86		4.48	
	4 × 4	1.417	0.6%	5.25	8.0%	4.89	9.1%
	8 × 8	1.421	0.9%	4.97	2.3%	4.60	2.8%
	14 × 14	1.421	0.9%	4.91	1.1%	4.55	1.5%
60°	Series						
	solution	0.932		4.25		3.33	
	4 × 4	0.845	–9.3%	4.46	4.8%	3.39	1.8%
	8 × 8	0.915	–1.8%	4.30	1.2%	3.32	–0.1%
	14 × 14	0.933	0.2%	4.29	0.8%	3.35	0.7%
40°	Series						
	solution	0.349		2.81		1.80	
	4 × 4	0.259	–26%	2.73	–3.0%	1.50	–17%
	8 × 8	0.308	–12%	2.68	–4.7%	1.57	–13%
	14 × 14	0.332	–4.7%	2.75	–2.2%	1.70	–5.5%
30°	Series						
	solution	0.148		1.91		1.08	
	4 × 4	0.095	–36%	1.72	–10%	0.78	–28%
	8 × 8	0.119	–20%	1.70	–11%	0.82	–24%
	14 × 14	0.149	0.5%	1.91	0.2%	1.09	0.8%

Table 9: Skewed plate results: SC6R, Abaqus/Standard analysis.

Skew angle	Mesh	w_{center}		M_{max}		M_{min}	
		(mm)	Error	(× 10^{-2} N-m/m)	Error	(× 10^{-2} N-m/m)	Error
90°	Series						
	solution	1.478		4.79		4.79	
	4 × 4	1.214	–17.8%	4.00	–6.5%	4.00	–16.5%
	8 × 8	1.425	–3.6%	4.85	1.3%	4.85	1.3%
	14 × 14	1.462	–1.1%	4.80	0.3%	4.80	0.3%
80°	Series						
	solution	1.409		4.86		4.48	
	4 × 4	1.147	–18.5%	4.07	–16.3%	3.54	–20.9%
	8 × 8	1.343	–4.7%	4.92	1.1%	4.43	–1.1%
	14 × 14	1.391	–1.3%	4.87	0.2%	4.50	0.5%
60°	Series						
	solution	0.932		4.25		3.33	
	4 × 4	0.615	–34%	3.02	–28.9%	1.94	–41.8%

Skew angle	Mesh	w_{center}		M_{max}		M_{min}	
		(mm)	Error	(× 10^{-2} N-m/m)	Error	(× 10^{-2} N-m/m)	Error
	8 × 8	0.812	–12.9%	3.83	–9.8%	2.81	–15.5%
	14 × 14	0.913	–2.1%	4.19	–1.5%	3.31	-0.7%
40°	Series						
	solution	0.349		2.81		1.80	
	4 × 4	0.213	–38.9%	1.89	–32.9%	0.89	–50.7%
	8 × 8	0.292	–16.3%	2.43	–13.6%	1.39	–22.8%
	14 × 14	0.346	–0.8%	2.81	0%	1.82	–1.1%
30°	Series						
	solution	0.148		1.91		1.08	
	4 × 4	0.080	–46.2%	1.16	–39.4%	0.46	–56.9%
	8 × 8	0.125	–15.8%	1.61	–15.9%	0.80	–26.1%
	14 × 14	0.148	0.1%	1.91	1.9%	1.07	0.5%

Table 10: Skewed plate results: SC8R, Abaqus/Standard analysis.

Skew angle	Mesh	w_{center}		M_{max}		M_{min}	
		(mm)	Error	(× 10^{-2} N-m/m)	Error	(× 10^{-2} N-m/m)	Error
90°	Series						
	solution	1.478		4.79		4.79	
	4 × 4	1.503	1.7%	4.23	–11.6%	4.23	–11.6%
	8 × 8	1.487	0.6%	4.66	–2.8%	4.66	–2.8%
	14 × 14	1.485	–0.5%	4.75	–0.8%	4.75	–0.8%
80°	Series						
	solution	1.409		4.86		4.48	
	4 × 4	1.436	1.9%	4.29	–11.8%	3.96	—11.6%
	8 × 8	1.417	0.6%	4.72	–2.9%	4.43	–2.5%
	14 × 14	1.414	–0.4%	4.82	–0.8%	4.46	–0.5%
60°	Series						
	solution	0.932		4.25		3.33	
	4 × 4	0.981	–5.2%	3.78	–11.0%	2.87	–13.7%
	8 × 8	0.943	1.2%	4.12	–3.1%	3.25	–2.3%
	14 × 14	0.938	0.6%	4.22	–0.7%	3.31	–0.5%
40°	Series						
	solution	0.349		2.81		1.80	
	4 × 4	0.383	9.6%	2.58	–8.2%	1.45	–19.5%
	8 × 8	0.363	4.1%	2.73	–2.8%	1.79	–0.5%
	14 × 14	0.355	1.7%	2.78	–0.9%	1.82	–1.3%

Skew angle	Mesh	w_{center}		M_{max}		M_{min}	
		(mm)	Error	(× 10^{-2} N-m/m)	Error	(× 10^{-2} N-m/m)	Error
30°	Series						
	solution	0.148		1.91		1.08	
	4 × 4	0.158	6.8%	1.74	–9.1%	0.80	–25.9%
	8 × 8	0.158	6.8%	1.88	–1.6%	1.07	–0.8%
	14 × 14	0.153	3.4%	1.90	–0.7%	1.11	3.1%

Table 11: Skewed plate results: S3R, Abaqus/Explicit analysis.

Skew Angle	Mesh	w_{center}		M_{max}		M_{min}	
		(mm)	Error	(×10^{-2} N-m/m)	Error	(×10^{-2} N-m/m)	Error
90°	Series						
	solution	1.478		4.79		4.79	
	2 × 2 × 4	0.949	–36.2%	2.69	–44.0%	2.69	–44.0%
	4 × 4 × 4	1.325	–10.4%	4.30	–10.2%	4.30	–10.2%
	8 × 8 × 4	1.413	–4.4%	4.56	–4.8%	4.56	–4.8%
80°	Series						
	solution	1.409		4.86		4.48	
	2 × 2 × 4	0.941	–33.0%	2.71	–44.1%	2.65	–40.8%
	4 × 4 × 4	1.257	–10.7%	4.26	–12.3%	4.18	–6.6%
	8 × 8 × 4	1.347	–4.4%	4.65	4.0%	4.26	–4.9%
60°	Series						
	solution	0.932		4.25		3.33	
	2 × 2 × 4	0.783	–15.9%	2.60	–38.1%	2.19	–34.3%
	4 × 4 × 4	0.822	–11.8%	3.99	–6.1%	2.98	–10.5%
	8 × 8 × 4	0.897	–3.7%	4.14	–2.5%	3.21	–3.6%
40°	Series						
	solution	0.349		2.81		1.80	
	2 × 2 × 4	0.332	–4.9%	1.78	–36.6%	1.10	–38.8%
	4 × 4 × 4	0.326	–6.6%	2.63	–6.4%	1.65	–8.3%
	8 × 8 × 4	0.348	0.2%	2.84	1.0%	1.80	0.0%
30°	Series						
	solution	0.148		1.91		1.08	
	2 × 2 × 4	0.145	–2.0%	1.16	–39.2%	0.60	–44.4%
	4 × 4 × 4	0.148	–0.0%	1.78	–6.8%	0.98	–9.3%
	8 × 8 × 4	0.152	2.7%	1.86	2.6%	1.10	1.9%

Table 12: Skewed plate results: S4R, Abaqus/Explicit analysis.

Skew Angle	Mesh	w_{center}		M_{max}		M_{min}	
		(mm)	Error	(×10^{-2} N-m/m)	Error	(×10^{-2} N-m/m)	Error
90°	Series						
	solution	1.478		4.79		4.79	
	4 × 4	1.444	–2.3%	3.95	–17%	3.95	–17%
	8 × 8	1.450	–1.9%	4.49	–6.0%	4.49	–6.0%
	14 × 14	1.445	–2.2%	4.60	–3.9%	4.60	–3.9%
80°	Series						
	solution	1.409		4.86		4.48	
	4 × 4	1.356	–4.0%	4.08	–16%	3.71	–17%
	8 × 8	1.372	–2.6%	4.57	–6.0%	4.20	–6.2%
	14 × 14	1.378	–2.2%	4.68	–2.5%	4.32	–3.5%
60°	Series						
	solution	0.932		4.25		3.33	
	4 × 4	0.957	2.7%	3.53	–16%	2.54	–23%
	8 × 8	0.930	–0.2%	4.02	–5.4%	3.09	–7.2%
	14 × 14	0.922	–1.0%	4.15	–2.3%	3.23	–3.0%
40°	Series						
	solution	0.349		2.81		1.80	
	4 × 4	0.305	–12%	2.21	–21%	1.17	–35%
	8 × 8	0.328	–6.0%	2.59	–7.9%	1.57	–13%
	14 × 14	0.344	–1.5%	2.74	–2.5%	1.73	–4.0%
30°	Series						
	solution	0.148		1.91		1.08	
	4 × 4	0.152	2.7%	1.45	–24%	0.63	–42%
	8 × 8	0.151	2.0%	1.71	–10%	0.88	–26%
	14 × 14	0.144	–2.7%	1.84	–3.7%	1.01	–6.4%

Table 13: Skewed plate results: S4RS, Abaqus/Explicit analysis.

Skew Angle	Mesh	w_{center}		M_{max}		M_{min}	
		(mm)	Error	(×10^{-2} N-m/m)	Error	(×10^{-2} N-m/m)	Error
90°	Series						
	solution	1.478		4.79		4.79	
	4 × 4	1.553	+5.1%	4.29	–10%	4.29	–10%
	8 × 8	1.477	–0.1%	4.60	–4.0%	4.60	–4.0%
	14 × 14	1.458	–1.4%	4.63	–3.3%	4.63	–3.3%
80°	Series						

Skew Angle	Mesh	w_{center}		M_{max}		M_{min}	
		(mm)	Error	($\times 10^{-2}$ N-m/m)	Error	($\times 10^{-2}$ N-m/m)	Error
	solution	1.409		4.86		4.48	
	4 × 4	1.516	+7.6%	4.46	−8.2%	4.08	−8.9%
	8 × 8	1.409	0.0%	4.72	−2.9%	4.26	−4.9%
	14 × 14	1.391	−1.3%	4.73	−2.7%	4.32	−3.6%
60°	Series						
	solution	0.932		4.25		3.33	
	4 × 4	1.026	+10%	3.91	−8.0%	2.96	−11%
	8 × 8	0.954	+2.1%	4.20	−1.2%	3.21	−3.6%
	14 × 14	0.942	+1.0%	4.19	−1.4%	3.33	0.0%
40°	Series						
	solution	0.349		2.81		1.80	
	4 × 4	0.397	+14%	2.65	−5.7%	1.49	−17%
	8 × 8	0.371	+6.3%	2.78	−1.1%	1.82	+1.1%
	14 × 14	0.361	+3.4%	2.80	−0.5%	1.87	+4.0%
30°	Series						
	solution	0.148		1.91		1.08	
	4 × 4	0.162	+9.5%	1.76	−7.9%	0.82	−24%
	8 × 8	0.162	+9.5%	1.91	0.0%	1.09	+1.0%
	14 × 14	0.156	+5.4%	1.92	+0.5%	1.14	+5.6%

Table 14: Skewed plate results: SC8R, Abaqus/Explicit analysis.

Skew Angle	Mesh	w_{center}		M_{max}		M_{min}	
		(mm)	Error	($\times 10^{-2}$ N-m/m)	Error	($\times 10^{-2}$ N-m/m)	Error
90°	Series						
	solution	1.478		4.79		4.79	
	4 × 4	1.466	−0.8%	4.15	−13%	4.15	−13%
	8 × 8	1.446	−2.1%	4.53	−5.4%	4.53	−5.4%
	14 × 14	1.443	−2.4%	4.61	−3.8%	4.61	−3.8%
80°	Series						
	solution	1.409		4.86		4.48	
	4 × 4	1.403	−0.4%	4.20	−14%	3.87	−14%
	8 × 8	1.381	−2.0%	4.59	−5.6%	4.25	−5.1%
	14 × 14	1.378	−2.2%	4.68	−3.7%	4.33	−3.3%
60°	Series						
	solution	0.932		4.25		3.33	
	4 × 4	0.957	1.6%	3.72	−13%	2.80	−16%

Skew Angle	Mesh	w_{center}		M_{max}		M_{min}	
		(mm)	Error	($\times 10^{-2}$ N-m/m)	Error	($\times 10^{-2}$ N-m/m)	Error
	8 × 8	0.926	−0.6%	4.06	−4.5%	3.18	−4.5%
	14 × 14	0.924	−0.9%	4.16	−2.1%	3.26	−2.1%
40°	Series						
	solution	0.349		2.81		1.80	
	4 × 4	0.367	5.2%	2.50	−11%	1.39	−23%
	8 × 8	0.350	0.3%	2.67	−5.0%	1.71	−5%
	14 × 14	0.349	0.0%	2.76	−1.8%	1.78	−1.1%
30°	Series						
	solution	0.148		1.91		1.08	
	4 × 4	0.152	2.0%	1.68	−12%	0.77	−29%
	8 × 8	0.149	0.7%	1.81	−5.2%	0.99	−8.3%
	14 × 14	0.149	0.7%	1.87	−2.1%	1.06	−1.9%

Figures

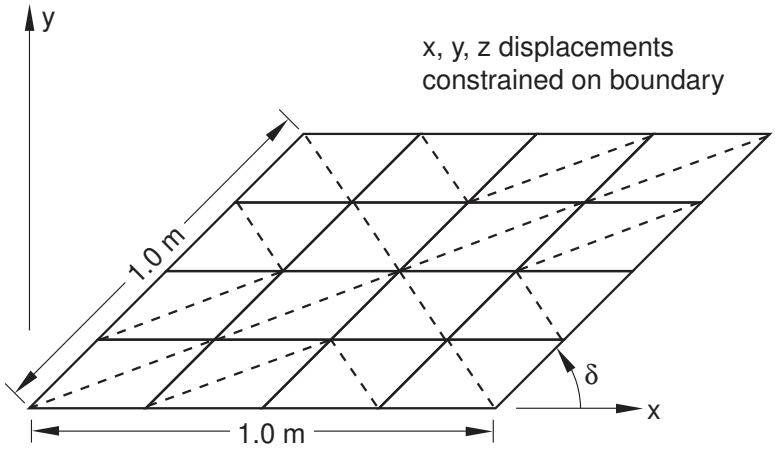


Figure 1: Simply supported skew plate with uniform distributed load. A 4 × 4 mesh for the complete plate of quadrilateral elements is shown. The corresponding mesh of triangular elements is shown by the dotted line.

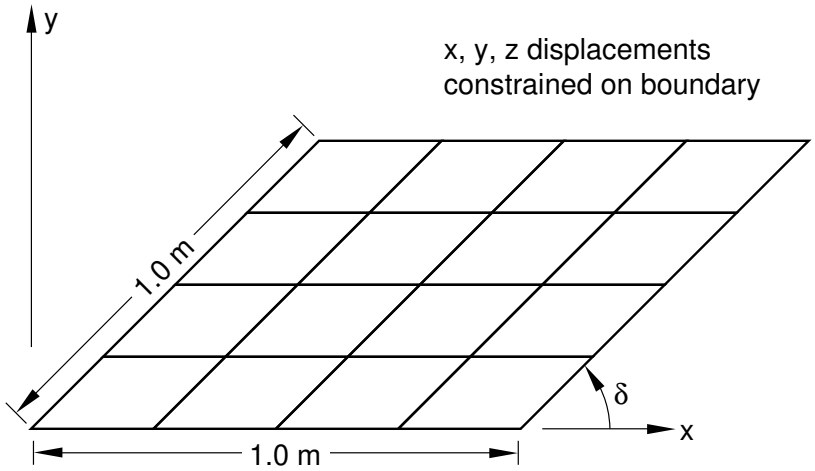


Figure 2: 4 × 4 mesh for the complete plate of quadrilateral elements.

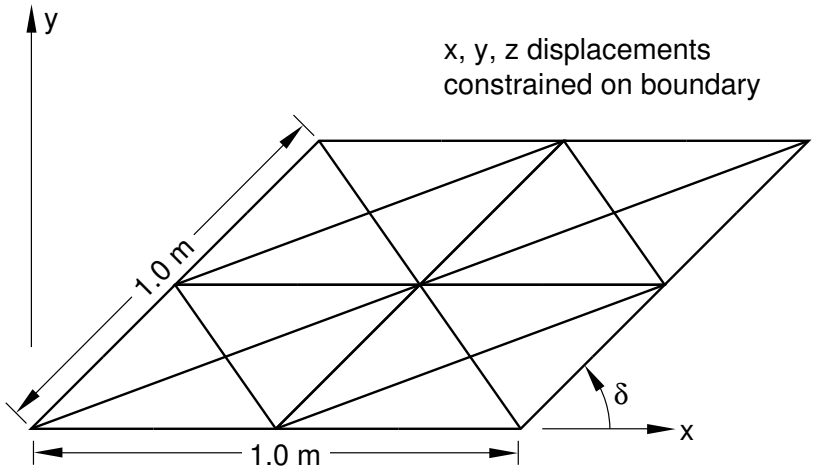


Figure 3: 2 × 2 × 4 mesh for the complete plate of triangular elements.

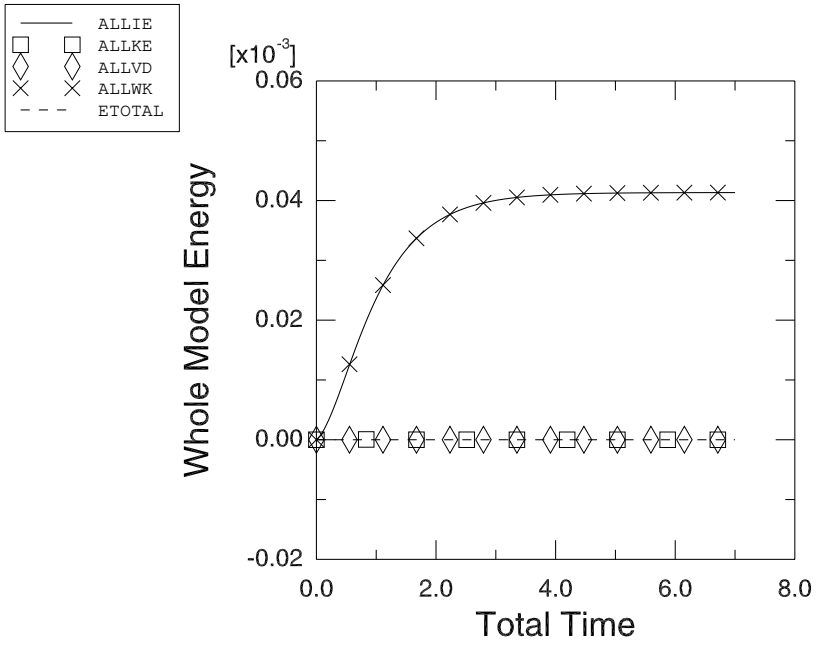


Figure 4: Energy balance for 14 × 14 mesh at 40° (Abaqus/Explicit analysis).

Performance of continuum and shell elements for linear analysis of bending problems

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example illustrates the rationale behind the use of various continuum and shell elements for bending problems, measures the performance of the elements using different meshes and loading conditions, and presents guidelines for the use of the elements.

Two problems are considered: a cantilever subjected to end loading and buckling of a ring in a plane under external pressure. The cantilever problem is studied with regular meshes and with two types of element distortion.

It is well known that fully integrated linear isoparametric continuum elements, in both two and three dimensions, are too stiff in modeling the simple flexural deformation of a beam. Similarly, fully integrated standard displacement formulations for 4-node shell elements are too stiff in modeling bending about an axis perpendicular to the plane of the shell; i.e., in-plane bending. Full integration refers to the Gauss integration order required for exact integration of the polynomial of the order being integrated when the element is rectangular. Although full-order integration elements can represent rigid-body and constant-strain displacement fields exactly, they tend to "lock" in bending problems because a disproportionately large shear-related strain energy arises, which greatly increases the flexural rigidity of the model.

Problem description

Continuum elements with good bending behavior

In Abaqus there are several alternative continuum elements that can be used to overcome this shear-locking deficiency:

- Second-order isoparametric elements—these elements can reproduce quadratic displacement fields, thus enabling them to model a pure bending response without any shear strains. They are available only in Abaqus/Standard.
- Incompatible mode elements—the addition of incompatible modes to the linear isoparametric elements eliminates shear locking and enables these elements to have excellent bending properties.
- Reduced-integration linear isoparametric elements—reduced integration in the evaluation of the element strain energy eliminates the shear locking phenomenon. These elements are available in both Abaqus/Standard and Abaqus/Explicit. Generally, multiple reduced-integration elements through the thickness are needed to model the bending response accurately. However, the enhanced hourglass control approach in Abaqus/Standard and Abaqus/Explicit can provide good bending behavior even with a coarse mesh. The displacement solutions for linear elastic materials obtained with reduced-integration elements using enhanced hourglass control closely match those obtained with incompatible mode elements since they are both based on the same assumed strain formulation.
- Continuum shell elements—these elements behave similar to shell elements and, therefore, can be used effectively for modeling slender structures dominated by bending behavior.

Solid isoparametric quadrilaterals and hexahedra and *Continuum elements with incompatible modes* provide detailed discussions of the element formulations.

Shell elements with good in-plane bending behavior

In many loading situations shell elements undergo substantial bending in the plane of the element. In Abaqus shell element S4 uses an assumed strain treatment for its membrane response that is designed to eliminate parasitic shear stresses that occur when the element is subjected to in-plane bending. In addition, the assumed strain field is designed to eliminate artificial stiffening during in-plane bending due to Poisson's ratio effects. See *Finite-strain shell element formulation* for a description of the assumed strain treatment in S4 elements. The reduced-integration element, S4R, exhibits good in-plane bending behavior with enhanced hourglass control in both Abaqus/Standard and Abaqus/Explicit.

Incompatible mode elements

The first-order quadrilateral continuum elements of type CPS4I, CPE4I, CAX4I, CPEG4I, and C3D8I, as well as the related hybrid elements, are enhanced by incompatible modes to improve the bending behavior. In addition to the displacement degrees of freedom, incompatible deformation modes are added internally to the elements. The primary effect of these degrees of freedom is to eliminate the parasitic shear stresses that are observed in regular linear continuum elements if they are loaded in bending.

In addition, these degrees of freedom eliminate artificial stiffening in bending due to Poisson's effect. In regular linear continuum elements, the linear variation of the axial stress due to bending is accompanied by a linear variation of the stress perpendicular to the bending direction, which leads to incorrect stresses and an overestimation of the stiffness. The incompatible modes prevent such a stress from occurring.

Continuum element integration schemes

The different numerical integration schemes used by the elements mentioned above in evaluating the stiffness matrices are discussed here. The integration scheme plays a vital role in determining the properties of an element.

- Linear isoparametric elements use selectively reduced integration. Selectively reduced integration is used in Abaqus for linear plane strain, generalized plane strain, axisymmetric, and three-dimensional isoparametric elements. In these elements second-order Gaussian integration is used for the deviatoric strains, with one point used to integrate volumetric strain terms, to avoid excessive constraint when the element's response is essentially incompressible.
- Second-order isoparametric elements use full or reduced integration. The full and reduced-integration schemes use third- and second-order Gaussian integration schemes, respectively.
- Incompatible mode elements use full integration. The use of full Gaussian quadrature (second-order) requires 2×2 integration points in two dimensions (e.g., CPS4I) and $2 \times 2 \times 2$ points in three dimensions (e.g., C3D8I).
- Reduced-integration, linear isoparametric elements use uniformly reduced integration. The integration scheme is based on the “uniform strain formulation,” where an average strain is calculated over the element volume. Uniformly reduced-integration rules are appealing computationally because a substantial reduction in the number of function evaluations is achieved.

Hourglass control for reduced-integration elements

Reduced integration has a serious drawback: it can result in a mesh instability, commonly referred to as “hourglassing.” Kinematic zero-energy modes are present in reduced-integration element formulations so that, if the mesh is geometrically consistent with a global pattern of such modes, a singular stiffness matrix is obtained and the element is rendered ineffective.

Flanagan and Belytschko (1981) and Belytschko et al. (1984) describe a control technique to deal with the hourglassing of first-order uniform-strain elements. The method involves the construction of generalized hourglass strains that are orthogonal to the rigid body modes. Typically, hourglass stresses are related to the hourglass

strains through artificial stiffness parameters. These stiffness coefficients are relatively small when compared to the actual stiffness of the material. The *Introduction: general* describes the artificial stiffness values used by default if they are not specified. The enhanced hourglass controls available in Abaqus uses stiffness coefficients based on the enhanced assumed strain method. It gives good displacement solutions for linear elastic materials and provides increased resistance to hourglassing for nonlinear materials. Element types CPS4R, CPE4R, CPEG4R, CAX4R, C3D8R, and the corresponding hybrid elements are based on this approach. Shell element types S4R and S4R5 are based on similar concepts. For modified tetrahedral and triangular elements the total stiffness approach is specified.

Geometry and models

Three examples are considered here to illustrate the behavior of the various elements in modeling bending behavior.

The same number of first- and second-order elements is used for each mesh layout. Obviously, for an equal number of elements, the mesh for the second-order elements will contain more nodes and, thus, have a larger number of degrees of freedom. However, the objective is to show that, even with fewer degrees of freedom, the incompatible mode elements and—to a certain extent—the reduced-integration, linear elements give comparably accurate results with respect to the second-order elements. Hence, the meshes included here are coarse and should not be taken as good modeling practice.

Cantilever beam with end shear load

The first example is the linear static analysis of a cantilever with an end shear load. The geometry is shown in *Figure 1*. The Young's modulus is 689.5 GPa (10×10^7 lb/in^2), and Poisson's ratio is 0. In Abaqus/Standard and Abaqus/Explicit the following four meshes are used: 1, 2, 4, or 8 elements through the depth of the beam, combined with 4 or 16 elements along the beam. The 8×16 mesh is included to examine the “converged” solution. The resulting end deflections are normalized with respect to the Euler-Bernoulli beam theory prediction of 2.74 mm (0.108 in) and tabulated in *Table 1*. Although this example is a beam problem, the thickness is chosen to be small so that the problem can be reasonably modeled with both continuum and shell elements. Due to the small thickness, warning messages may be issued for poor aspect ratios for analyses involving coarser meshes of three-dimensional continuum elements. The analytical solution depends only on the thickness through the combination Young's modulus times thickness (Et). Hence, any thickness can be used so long as the product Et remains fixed; the solution remains the same with the exception of the tetrahedral elements, where the thickness influences the tetrahedral quality measure and, hence, the displacement solution.

In the Abaqus/Explicit analyses the load is applied to the structure using a SMOOTH STEP amplitude curve to minimize the dynamic effects.

Skew sensitivity analysis of cantilever beam

The second example examines the skew sensitivity of the elements with respect to two shapes: parallelogram and trapezoidal. The cantilever beam and loading are the same as in the first example. A 1×8 mesh is used for both shapes when the elements are quadrilateral or hexahedral, as shown in *Figure 2* and *Figure 3*. When the problem is modeled using triangular or tetrahedral elements, the basic parallelogram or trapezoidal shape meshed with one quadrilateral or hexahedral element is filled with either two triangular elements in the two-dimensional case or five tetrahedral elements in the three-dimensional case. Therefore, the triangular and tetrahedral elements do not have a true parallelogram or trapezoidal shape. The following skew angles are tested for the Abaqus/Standard and Abaqus/Explicit runs: 0°, 15°, 30°, and 45°. The resulting end deflections are again normalized with respect to the Euler-Bernoulli beam theory prediction of 2.74 mm (0.108 in) and tabulated in *Table 2* and *Table 3*. The use of a single layer of elements precludes the testing of the reduced-integration linear isoparametric elements (except when using enhanced hourglass control) since a minimum of four layers is required for acceptable results.

Buckling analysis of ring under external pressure

The final example is the buckling analysis of a ring in a plane under external pressure. The geometry and material are the same as those used in *Buckling of a ring in a plane under external pressure* and are shown in *Figure 4*. The ring buckling problem requires a rather fine mesh in the circumferential direction, presumably to model the strain gradients accurately in the buckling mode. Using symmetry boundary conditions, only a 45° sector of the ring is modeled, which is enough to reproduce the primary buckling mode. *Table 5* gives the solutions obtained with various models for this case. The exact solution, based on Euler-Bernoulli beam theory, is a critical buckling pressure of 51.71 KPa (7.5 lb/in^2).

Results and discussion

The results for each analysis are described below.

Cantilever beam with end shear load

As expected, the second-order elements in Abaqus/Standard give excellent results even with the coarse 1 × 4 mesh, with the reduced-integration CPS8R and C3D20R elements giving the most accurate results. The regular second-order and the modified triangular and tetrahedral elements also exhibit excellent behavior in bending. The incompatible mode elements in Abaqus/Standard (CPS4I and C3D8I) and the reduced-integration elements in Abaqus/Standard and Abaqus/Explicit (C3D8R, CPS4R, and S4R) with enhanced hourglass control perform just as well as the second-order elements, indicating their excellent bending properties when they are used as rectangles and lined up with the principal axes of bending. The results for the one-layer 1 × 4 mesh are already very good. Increasing the number of layers does not improve the results. However, more layers are required for accurate analysis when material nonlinearities are present. The assumed strain shell element S4 performs relatively well given the coarseness of the mesh and the fact that S4 does not have internal degrees of freedom like CPS4I and C3D8I. With mesh refinement the solution improves.

On the other hand, the linear isoparametric elements exhibit extremely stiff bending behavior, far too stiff for practical applications. Even with the very fine 8 × 16 mesh, the displacement is still less than half of the correct value. In a critical analysis of an exactly integrated plane stress element, Prathap (1985) points out that "improvement of the idealization by increasing the number of elements through the depth does not relieve the parasitic shear locking [i.e., excessively stiff behavior in bending situations]." This observation is confirmed by the results with element types CPS3, CPS4, C3D4, and C3D8 in this example.

The reduced-integration linear elements CPS4R, C3D8R, S4R, and SC8R converge rapidly as the mesh is refined. However, the convergence to the correct result is no longer monotonic. This is an effect of the under-integration of the element stiffness: there is no guarantee of an upper bound to the stiffness of the solution, and the response may fall on the soft side. With reduced integration the number of elements through the depth plays a critical role. Two elements through the depth fail to provide engineering accuracy. Four elements through the depth, with four elements along the length, provide acceptable results. If the idealization involved only one element through the depth, the material integration points would all lie on the neutral axis, and the bending behavior would depend entirely on the (artificial) hourglass stiffness. Combined with enhanced hourglass control, the CPS4R, C3D8R, and S4R elements provide excellent results even with the coarse 1 × 4 mesh. The results closely match those obtained with incompatible mode elements in Abaqus/Standard since both analyses are based on the same assumed strain formulation.

Skew sensitivity analysis of cantilever beam

Figure 5 and *Figure 6* show the effect of skewing the elements into parallelograms and trapezoids. We see that the second-order elements do not show strong sensitivity to such distortion, while the incompatible mode elements and element S4 show more sensitivity, especially to trapezoidal distortion, which quickly causes them to be impractically stiff. Both the regular second-order and modified triangular/tetrahedral elements show less sensitivity to the trapezoidal skew than the second-order quadrilateral/hexahedral elements, but they show more sensitivity

than the rectangular elements in the parallelogram skew test. The Abaqus/Explicit results and the linear reduced-integration elements in Abaqus/Standard with enhanced hourglass control closely match the results obtained with incompatible mode elements in Abaqus/Standard.

Table 2 and *Table 3* summarize the effect of skewing continuum, shell, and continuum shell elements into parallelograms and trapezoids. The skewing is done in the element in-plane direction for the shell and continuum shell models. *Table 4* summarizes the effect of skewing the continuum shell element SC8R in the thickness direction of the element. The results indicate that the behavior is insensitive to mesh distortion that occurs in the thickness direction.

Buckling analysis of ring under external pressure

Both the incompatible mode and second-order elements give excellent results, even with the coarse mesh. The linear reduced-integration elements with hourglass control based on total stiffness give acceptable results that again hover about the correct value: higher for the coarse mesh and lower for the fine mesh. The linear reduced-integration elements with enhanced hourglass control give results that closely match the results obtained with incompatible mode elements. As expected, the linear full-integration elements once again give extremely stiff results: between 8 to 93 times the actual buckling pressure.

Additional modeling considerations

These analyses illustrate the use of continuum elements to simulate bending behavior of thin structures. Similarly, these analyses illustrate the use of shell element S4 for bending in the plane of the element. In general, the use of beam or shell elements (with out-of-plane bending) is recommended for thin structures; the difficulties discussed here would be encountered only if the analyst could not use the more appropriate bending elements for some reason. The results presented here do not reflect the usefulness or importance of continuum elements in other types of problems.

We have seen that the incompatible mode elements perform almost as well as the second-order elements in many situations if the elements have an approximately rectangular shape. The performance is considerably poorer if the elements have a parallelogram shape and quickly becomes unacceptable with trapezoidal element shapes. In addition, these elements offer no advantages when a side is degenerated or collapsed into a node. The degenerated elements can only represent a constant strain field, and the incompatible modes cannot improve on such fields.

Due to the internal degrees of freedom (4 for CPS4I; 5 for CPE4I, CAX4I, and CPEG4I; and 13 for C3D8I) the incompatible mode elements are somewhat more expensive than regular displacement elements but are more economical than the second-order elements. The additional degrees of freedom do not increase the wavefront size substantially, since they can be eliminated immediately. In addition, it is not necessary to use selectively reduced integration, which partially offsets the cost of the additional degrees of freedom.

The reduced-integration, linear elements also give satisfactory solutions for the set of problems attempted here when a minimum of four layers is used. However, there are cases where the elements may yield a nearly singular stiffness and, hence, physically unreasonable solutions. This is especially true when large-strain problems are analyzed. Thus, as with all modeling decisions, analysts must develop their discretization with careful testing of the effectiveness of the elements for a particular application.

Further examples

The use of the elements discussed above is illustrated further in the following example problems:

- *Geometrically nonlinear analysis of a cantilever beam*
- *Cylinder under internal pressure*
- *Nonlinear dynamic analysis of a structure with local inelastic collapse*

Input files

linbending_typ_cant.inp	Typical input data for the cantilever analysis.
linbending_typ_paral.inp	Typical input data for the skew sensitivity analysis of parallelogram-shaped elements.
linbending_typ_trap.inp	Typical input data for the skew sensitivity analysis of trapezoidal-shaped elements.
linbending_typ_ring.inp	Typical input data for the ring buckling problem.

Abaqus/Standard input files:

Cantilever analysisC3D4 elements:

linbending_c3d4_cant_1x4.inp	1 × 4 mesh.
linbending_c3d4_cant_2x4.inp	2 × 4 mesh.
linbending_c3d4_cant_4x4.inp	4 × 4 mesh.
linbending_c3d4_cant_8x16.inp	8 × 16 mesh.

C3D8 elements:

linbending_c3d8_cant_1x4.inp	1 × 4 mesh.
linbending_c3d8_cant_2x4.inp	2 × 4 mesh.
linbending_c3d8_cant_4x4.inp	4 × 4 mesh.
linbending_c3d8_cant_8x16.inp	8 × 16 mesh.

C3D8I elements:

linbending_c3d8i_cant_1x4.inp	1 × 4 mesh.
linbending_c3d8i_cant_2x4.inp	2 × 4 mesh.
linbending_c3d8i_cant_4x4.inp	4 × 4 mesh.
linbending_c3d8i_cant_8x16.inp	8 × 16 mesh.

C3D8R elements:

linbending_c3d8r_cant_2x4.inp	2 × 4 mesh.
linbending_c3d8r_cant_4x4.inp	4 × 4 mesh.
linbending_c3d8r_cant_8x16.inp	8 × 16 mesh.
linbending_c3d8r_cant_1x4_eh.inp	1 × 4 mesh.
linbending_c3d8r_cant_2x4_eh.inp	2 × 4 mesh.
linbending_c3d8r_cant_4x4_eh.inp	4 × 4 mesh.
linbending_c3d8r_cant_8x16_eh.inp	8 × 16 mesh.

C3D10 elements:

linbending_c3d10_cant_1x4.inp	1 × 4 mesh.
linbending_c3d10_cant_2x4.inp	2 × 4 mesh.
linbending_c3d10_cant_4x4.inp	4 × 4 mesh.
linbending_c3d10_cant_8x16.inp	8 × 16 mesh.

C3D10HS elements:

linbending_c3d10hs_cant_1x4.inp	1 × 4 mesh.
linbending_c3d10hs_cant_2x4.inp	2 × 4 mesh.

linbending_c3d10hs_cant_4x4.inp	4 × 4 mesh.
linbending_c3d10hs_cant_8x16.inp	8 × 16 mesh.
C3D10M elements:	
linbending_c3d10_mcant_1x4.inp	1 × 4 mesh.
linbending_c3d10_mcant_2x4.inp	2 × 4 mesh.
linbending_c3d10_mcant_4x4.inp	4 × 4 mesh.
linbending_c3d10_mcant_8x16.inp	8 × 16 mesh.
C3D20 elements:	
linbending_c3d20_cant_1x4.inp	1 × 4 mesh.
linbending_c3d20_cant_2x4.inp	2 × 4 mesh.
linbending_c3d20_cant_4x4.inp	4 × 4 mesh.
linbending_c3d20_cant_8x16.inp	8 × 16 mesh.
C3D20R elements:	
linbending_c3d20r_cant_1x4.inp	1 × 4 mesh.
linbending_c3d20r_cant_2x4.inp	2 × 4 mesh.
linbending_c3d20r_cant_4x4.inp	4 × 4 mesh.
linbending_c3d20r_cant_8x16.inp	8 × 16 mesh.
CPS3 elements:	
linbending_cps3_cant_1x4.inp	1 × 4 mesh.
linbending_cps3_cant_2x2.inp	2 × 2 mesh.
linbending_cps3_cant_4x4.inp	4 × 4 mesh.
linbending_cps3_cant_8x16.inp	8 × 16 mesh.
CPS4 elements:	
linbending_cps4_cant_1x4.inp	1 × 4 mesh.
linbending_cps4_cant_2x4.inp	2 × 4 mesh.
linbending_cps4_cant_4x4.inp	4 × 4 mesh.
linbending_cps4_cant_8x16.inp	8 × 16 mesh.
CPS4I elements:	
linbending_cps4i_cant_1x4.inp	1 × 4 mesh.
linbending_cps4i_cant_2x4.inp	2 × 4 mesh.
linbending_cps4i_cant_4x4.inp	4 × 4 mesh.
linbending_cps4i_cant_8x16.inp	8 × 16 mesh.
CPS4R elements:	
linbending_cps4r_cant_2x4.inp	2 × 4 mesh.
linbending_cps4r_cant_4x4.inp	4 × 4 mesh.
linbending_cps4r_cant_8x16.inp	8 × 16 mesh.
linbending_cps4r_cant_1x4_eh.inp	1 × 4 mesh.
linbending_cps4r_cant_2x4_eh.inp	2 × 4 mesh.
linbending_cps4r_cant_4x4_eh.inp	4 × 4 mesh.
linbending_cps4r_cant_8x16_eh.inp	8 × 16 mesh.

CPS6 elements:

linbending_cps6_cant_1x4.inp	1 × 4 mesh.
linbending_cps6_cant_2x4.inp	2 × 4 mesh.
linbending_cps6_cant_4x4.inp	4 × 4 mesh.
linbending_cps6_cant_8x16.inp	8 × 16 mesh.

CPS6M elements:

linbending_cps6m_cant_1x4.inp	1 × 4 mesh.
linbending_cps6m_cant_2x4.inp	2 × 4 mesh.
linbending_cps6m_cant_4x4.inp	4 × 4 mesh.
linbending_cps6m_cant_8x16.inp	8 × 16 mesh.

CPS8 elements:

linbending_cps8_cant_1x4.inp	1 × 4 mesh.
linbending_cps8_cant_2x4.inp	2 × 4 mesh.
linbending_cps8_cant_4x4.inp	4 × 4 mesh.
linbending_cps8_cant_8x16.inp	8 × 16 mesh.

CPS8R elements:

linbending_cps8r_cant_1x4.inp	1 × 4 mesh.
linbending_cps8r_cant_2x4.inp	2 × 4 mesh.
linbending_cps8r_cant_4x4.inp	4 × 4 mesh.
linbending_cps8r_cant_8x16.inp	8 × 16 mesh.

CSS8 elements:

linbending_css8_cant_1x4.inp	1 × 4 mesh.
linbending_css8_cant_2x4.inp	2 × 4 mesh.
linbending_css8_cant_4x4.inp	4 × 4 mesh.
linbending_css8_cant_8x16.inp	8 × 16 mesh.

S4 elements:

linbending_s4_cant_1x4.inp	1 × 4 mesh.
linbending_s4_cant_2x4.inp	2 × 4 mesh.
linbending_s4_cant_4x4.inp	4 × 4 mesh.
linbending_s4_cant_8x16.inp	8 × 16 mesh.

S4R elements:

linbending_s4r_cant_1x4_eh.inp	1 × 4 mesh.
linbending_s4r_cant_2x4_eh.inp	2 × 4 mesh.
linbending_s4r_cant_4x4_eh.inp	4 × 4 mesh.
linbending_s4r_cant_8x16_eh.inp	8 × 16 mesh.

Skew sensitivity analysis of parallelogram and trapezoidal-shaped elementsC3D4 elements:

linbending_c3d4_1x8_ang0.inp	1 × 8 mesh; skew angle of 0°.
linbending_c3d4_paral_ang15.inp	Parallelogram shape; skew angle of 15°.
linbending_c3d4_paral_ang30.inp	Parallelogram shape; skew angle of 30°.
linbending_c3d4_paral_ang45.inp	Parallelogram shape; skew angle of 45°.

linbending_c3d4_trap_ang15.inp	Trapezoidal shape; skew angle of 15°.
linbending_c3d4_trap_ang30.inp	Trapezoidal shape; skew angle of 30°.
linbending_c3d4_trap_ang45.inp	Trapezoidal shape; skew angle of 45°.

C3D8 elements:

linbending_c3d8_1x8_ang0.inp	1 × 8 mesh; skew angle of 0°.
linbending_c3d8_paral_ang15.inp	Parallelogram shape; skew angle of 15°.
linbending_c3d8_paral_ang30.inp	Parallelogram shape; skew angle of 30°.
linbending_c3d8_paral_ang45.inp	Parallelogram shape; skew angle of 45°.
linbending_c3d8_trap_ang15.inp	Trapezoidal shape; skew angle of 15°.
linbending_c3d8_trap_ang30.inp	Trapezoidal shape; skew angle of 30°.
linbending_c3d8_trap_ang45.inp	Trapezoidal shape; skew angle of 45°.

C3D8I elements:

linbending_c3d8i_1x8_ang0.inp	1 × 8 mesh; skew angle of 0°.
linbending_c3d8i_paral_ang15.inp	Parallelogram shape; skew angle of 15°.
linbending_c3d8i_paral_ang30.inp	Parallelogram shape; skew angle of 30°.
linbending_c3d8i_paral_ang45.inp	Parallelogram shape; skew angle of 45°.
linbending_c3d8i_trap_ang15.inp	Trapezoidal shape; skew angle of 15°.
linbending_c3d8i_trap_ang30.inp	Trapezoidal shape; skew angle of 30°.
linbending_c3d8i_trap_ang45.inp	Trapezoidal shape; skew angle of 45°.

C3D8R elements:

linbending_c3d8r_1x8_ang0_eh.inp	1 × 8 mesh; skew angle of 0°.
linbending_c3d8r_paral_ang15_eh.inp	Parallelogram shape; skew angle of 15°.
linbending_c3d8r_paral_ang30_eh.inp	Parallelogram shape; skew angle of 30°.
linbending_c3d8r_paral_ang45_eh.inp	Parallelogram shape; skew angle of 45°.
linbending_c3d8r_trap_ang15_eh.inp	Trapezoidal shape; skew angle of 15°.
linbending_c3d8r_trap_ang30_eh.inp	Trapezoidal shape; skew angle of 30°.
linbending_c3d8r_trap_ang45_eh.inp	Trapezoidal shape; skew angle of 45°.

C3D10 elements:

linbending_c3d10_1x8_ang0.inp	1 × 8 mesh; skew angle of 0°.
linbending_c3d10_paral_ang15.inp	Parallelogram shape; skew angle of 15°.
linbending_c3d10_paral_ang30.inp	Parallelogram shape; skew angle of 30°.
linbending_c3d10_paral_ang45.inp	Parallelogram shape; skew angle of 45°.
linbending_c3d10_trap_ang15.inp	Trapezoidal shape; skew angle of 15°.
linbending_c3d10_trap_ang30.inp	Trapezoidal shape; skew angle of 30°.
linbending_c3d10_trap_ang45.inp	Trapezoidal shape; skew angle of 45°.

C3D10HS elements:

linbending_c3d10hs_1x8_ang0.inp	1 × 8 mesh; skew angle of 0°.
linbending_c3d10hs_paral_ang15.inp	Parallelogram shape; skew angle of 15°.
linbending_c3d10hs_paral_ang30.inp	Parallelogram shape; skew angle of 30°.
linbending_c3d10hs_paral_ang45.inp	Parallelogram shape; skew angle of 45°.
linbending_c3d10hs_trap_ang15.inp	Trapezoidal shape; skew angle of 15°.

linbending_c3d10hs_trap_ang30.inp	Trapezoidal shape; skew angle of 30°.
linbending_c3d10hs_trap_ang45.inp	Trapezoidal shape; skew angle of 45°.

C3D10M elements:

linbending_c3d10m_1x8_ang0.inp	1 × 8 mesh; skew angle of 0°.
linbending_c3d10m_paral_ang15.inp	Parallelogram shape; skew angle of 15°.
linbending_c3d10m_paral_ang30.inp	Parallelogram shape; skew angle of 30°.
linbending_c3d10m_paral_ang45.inp	Parallelogram shape; skew angle of 45°.
linbending_c3d10m_trap_ang15.inp	Trapezoidal shape; skew angle of 15°.
linbending_c3d10m_trap_ang30.inp	Trapezoidal shape; skew angle of 30°.
linbending_c3d10m_trap_ang45.inp	Trapezoidal shape; skew angle of 45°.

C3D20 elements:

linbending_c3d20_1x8_ang0.inp	1 × 8 mesh; skew angle of 0°.
linbending_c3d20_paral_ang15.inp	Parallelogram shape; skew angle of 15°.
linbending_c3d20_paral_ang30.inp	Parallelogram shape; skew angle of 30°.
linbending_c3d20_paral_ang45.inp	Parallelogram shape; skew angle of 45°.
linbending_c3d20_trap_ang15.inp	Trapezoidal shape; skew angle of 15°.
linbending_c3d20_trap_ang30.inp	Trapezoidal shape; skew angle of 30°.
linbending_c3d20_trap_ang45.inp	Trapezoidal shape; skew angle of 45°.

C3D20R elements:

linbending_c3d20r_1x8_ang0.inp	1 × 8 mesh; skew angle of 0°.
linbending_c3d20r_paral_ang15.inp	Parallelogram shape; skew angle of 15°.
linbending_c3d20r_paral_ang30.inp	Parallelogram shape; skew angle of 30°.
linbending_c3d20r_paral_ang45.inp	Parallelogram shape; skew angle of 45°.
linbending_c3d20r_trap_ang15.inp	Trapezoidal shape; skew angle of 15°.
linbending_c3d20r_trap_ang30.inp	Trapezoidal shape; skew angle of 30°.
linbending_c3d20r_trap_ang45.inp	Trapezoidal shape; skew angle of 45°.

CPS3 elements:

linbending_cps3_1x8_ang0.inp	1 × 8 mesh; skew angle of 0°.
linbending_cps3_paral_ang15.inp	Parallelogram shape; skew angle of 15°.
linbending_cps3_paral_ang30.inp	Parallelogram shape; skew angle of 30°.
linbending_cps3_paral_ang45.inp	Parallelogram shape; skew angle of 45°.
linbending_cps3_trap_ang15.inp	Trapezoidal shape; skew angle of 15°.
linbending_cps3_trap_ang30.inp	Trapezoidal shape; skew angle of 30°.
linbending_cps3_trap_ang45.inp	Trapezoidal shape; skew angle of 45°.

CPS4 elements:

linbending_cps4_1x8_ang0.inp	1 × 8 mesh; skew angle of 0°.
linbending_cps4_paral_ang15.inp	Parallelogram shape; skew angle of 15°.
linbending_cps4_paral_ang30.inp	Parallelogram shape; skew angle of 30°.
linbending_cps4_paral_ang45.inp	Parallelogram shape; skew angle of 45°.
linbending_cps4_trap_ang15.inp	Trapezoidal shape; skew angle of 15°.
linbending_cps4_trap_ang30.inp	Trapezoidal shape; skew angle of 30°.

linbending_cps4_trap_ang45.inp	Trapezoidal shape; skew angle of 45°.
CPS4I elements:	
linbending_cps4i_mesh18_ang0.inp	1 × 8 mesh; skew angle of 0°.
linbending_cps4i_paral_ang15.inp	Parallelogram shape; skew angle of 15°.
linbending_cps4i_paral_ang30.inp	Parallelogram shape; skew angle of 30°.
linbending_cps4i_paral_ang45.inp	Parallelogram shape; skew angle of 45°.
linbending_cps4i_trap_ang15.inp	Trapezoidal shape; skew angle of 15°.
linbending_cps4i_trap_ang30.inp	Trapezoidal shape; skew angle of 30°.
linbending_cps4i_trap_ang45.inp	Trapezoidal shape; skew angle of 45°.
CPS4R elements:	
linbending_cps4r_1x8_ang0_eh.inp	1 × 8 mesh; skew angle of 0°.
linbending_cps4r_paral_ang15_eh.inp	Parallelogram shape; skew angle of 15°.
linbending_cps4r_paral_ang30_eh.inp	Parallelogram shape; skew angle of 30°.
linbending_cps4r_paral_ang45_eh.inp	Parallelogram shape; skew angle of 45°.
linbending_cps4r_trap_ang15_eh.inp	Trapezoidal shape; skew angle of 15°.
linbending_cps4r_trap_ang30_eh.inp	Trapezoidal shape; skew angle of 30°.
linbending_cps4r_trap_ang45_eh.inp	Trapezoidal shape; skew angle of 45°.
CPS6 elements:	
linbending_cps6_1x8_ang0.inp	1 × 8 mesh; skew angle of 0°.
linbending_cps6_paral_ang15.inp	Parallelogram shape; skew angle of 15°.
linbending_cps6_paral_ang30.inp	Parallelogram shape; skew angle of 30°.
linbending_cps6_paral_ang45.inp	Parallelogram shape; skew angle of 45°.
linbending_cps6_trap_ang15.inp	Trapezoidal shape; skew angle of 15°.
linbending_cps6_trap_ang30.inp	Trapezoidal shape; skew angle of 30°.
linbending_cps6_trap_ang45.inp	Trapezoidal shape; skew angle of 45°.
CPS6M elements:	
linbending_cps6m_1x8_ang0.inp	1 × 8 mesh; skew angle of 0°.
linbending_cps6m_paral_ang15.inp	Parallelogram shape; skew angle of 15°.
linbending_cps6m_paral_ang30.inp	Parallelogram shape; skew angle of 30°.
linbending_cps6m_paral_ang45.inp	Parallelogram shape; skew angle of 45°.
linbending_cps6m_trap_ang15.inp	Trapezoidal shape; skew angle of 15°.
linbending_cps6m_trap_ang30.inp	Trapezoidal shape; skew angle of 30°.
linbending_cps6m_trap_ang45.inp	Trapezoidal shape; skew angle of 45°.
CPS8 elements:	
linbending_cps8_1x8_ang0.inp	1 × 8 mesh; skew angle of 0°.
linbending_cps8_paral_ang15.inp	Parallelogram shape; skew angle of 15°.
linbending_cps8_paral_ang30.inp	Parallelogram shape; skew angle of 30°.
linbending_cps8_paral_ang45.inp	Parallelogram shape; skew angle of 45°.
linbending_cps8_trap_ang15.inp	Trapezoidal shape; skew angle of 15°.
linbending_cps8_trap_ang30.inp	Trapezoidal shape; skew angle of 30°.
linbending_cps8_trap_ang45.inp	Trapezoidal shape; skew angle of 45°.

CPS8R elements:

linbending_cps8r_1x8_ang0.inp	1 × 8 mesh; skew angle of 0°.
linbending_cps8r_paral_ang15.inp	Parallelogram shape; skew angle of 15°.
linbending_cps8r_paral_ang30.inp	Parallelogram shape; skew angle of 30°.
linbending_cps8r_paral_ang45.inp	Parallelogram shape; skew angle of 45°.
linbending_cps8r_trap_ang15.inp	Trapezoidal shape; skew angle of 15°.
linbending_cps8r_trap_ang30.inp	Trapezoidal shape; skew angle of 30°.
linbending_cps8r_trap_ang45.inp	Trapezoidal shape; skew angle of 45°.

CSS8 elements:

linbending_css8_1x8_ang0.inp	1 × 8 mesh; skew angle of 0°.
linbending_css8_paral_ang15.inp	Parallelogram shape; skew angle of 15°.
linbending_css8_paral_ang30.inp	Parallelogram shape; skew angle of 30°.
linbending_css8_paral_ang45.inp	Parallelogram shape; skew angle of 45°.
linbending_css8_trap_ang15.inp	Trapezoidal shape; skew angle of 15°.
linbending_css8_trap_ang30.inp	Trapezoidal shape; skew angle of 30°.
linbending_css8_trap_ang45.inp	Trapezoidal shape; skew angle of 45°.

S4 elements:

linbending_s4_1x8_ang0.inp	1 × 8 mesh; skew angle of 0°.
linbending_s4_paral_ang15.inp	Parallelogram shape; skew angle of 15°.
linbending_s4_paral_ang30.inp	Parallelogram shape; skew angle of 30°.
linbending_s4_paral_ang45.inp	Parallelogram shape; skew angle of 45°.
linbending_s4_trap_ang15.inp	Trapezoidal shape; skew angle of 15°.
linbending_s4_trap_ang30.inp	Trapezoidal shape; skew angle of 30°.
linbending_s4_trap_ang45.inp	Trapezoidal shape; skew angle of 45°.

S4R elements:

linbending_s4r_1x8_ang0_eh.inp	1 × 8 mesh; skew angle of 0°.
linbending_s4r_paral_ang15_eh.inp	Parallelogram shape; skew angle of 15°.
linbending_s4r_paral_ang30_eh.inp	Parallelogram shape; skew angle of 30°.
linbending_s4r_paral_ang45_eh.inp	Parallelogram shape; skew angle of 45°.
linbending_s4r_trap_ang15_eh.inp	Trapezoidal shape; skew angle of 15°.
linbending_s4r_trap_ang30_eh.inp	Trapezoidal shape; skew angle of 30°.
linbending_s4r_trap_ang45_eh.inp	Trapezoidal shape; skew angle of 45°.

SC8R elements:

linbending_sc8r_1x8_ang0_eh.inp	1 × 8 mesh; skew angle of 0°.
linbending_sc8r_paral_ang15_eh.inp	Parallelogram shape; skew angle of 15°.
linbending_sc8r_paral_ang30_eh.inp	Parallelogram shape; skew angle of 30°.
linbending_sc8r_paral_ang45_eh.inp	Parallelogram shape; skew angle of 45°.
linbending_sc8r_trap_ang15_eh.inp	Trapezoidal shape; skew angle of 15°.
linbending_sc8r_trap_ang30_eh.inp	Trapezoidal shape; skew angle of 30°.
linbending_sc8r_trap_ang45_eh.inp	Trapezoidal shape; skew angle of 45°.
linbending_sc8r_1x8_ang0_thk.inp	1 × 8 mesh; skew angle of 0° in element thickness direction.

linbending_sc8r_paral_ang15_thk.inp	Parallelogram shape; skew angle of 15° in element thickness direction.
linbending_sc8r_paral_ang30_thk.inp	Parallelogram shape; skew angle of 30° in element thickness direction.
linbending_sc8r_paral_ang45_thk.inp	Parallelogram shape; skew angle of 45° in element thickness direction.
linbending_sc8r_trap_ang15_thk.inp	Trapezoidal shape; skew angle of 15° in element thickness direction.
linbending_sc8r_trap_ang30_thk.inp	Trapezoidal shape; skew angle of 30° in element thickness direction.
linbending_sc8r_trap_ang45_thk.inp	Trapezoidal shape; skew angle of 45° in element thickness direction.

Ring buckling problemC3D4 elements:

linbending_ring_c3d4_4x10.inp	4 × 10 mesh.
linbending_ring_c3d4_4x20.inp	4 × 20 mesh.

C3D8 elements:

linbending_ring_c3d8_4x10.inp	4 × 10 mesh.
linbending_ring_c3d8_4x20.inp	4 × 20 mesh.

C3D8I elements:

linbending_ring_c3d8i_4x10.inp	4 × 10 mesh.
linbending_ring_c3d8i_4x20.inp	4 × 20 mesh.

C3D8R elements:

linbending_ring_c3d8r_4x10.inp	4 × 10 mesh.
linbending_ring_c3d8r_4x20.inp	4 × 20 mesh.
linbending_ring_c3d8r_4x10_eh.inp	4 × 10 mesh.
linbending_ring_c3d8r_4x20_eh.inp	4 × 20 mesh.

C3D10 elements:

linbending_ring_c3d10_4x10.inp	4 × 10 mesh.
linbending_ring_c3d10_4x20.inp	4 × 20 mesh.

C3D10HS elements:

linbending_ring_c3d10hs_4x10.inp	4 × 10 mesh.
linbending_ring_c3d10hs_4x20.inp	4 × 20 mesh.

C3D10M elements:

linbending_ring_c3d10m_4x10.inp	4 × 10 mesh.
linbending_ring_c3d10m_4x20.inp	4 × 20 mesh.

C3D20 elements:

linbending_ring_c3d20_4x10.inp	4 × 10 mesh.
linbending_ring_c3d20_4x20.inp	4 × 20 mesh.

C3D20R elements:

linbending_ring_c3d20r_4x10.inp	4 × 10 mesh.

linbending_ring_c3d20r_4x20.inp	4 × 20 mesh.

CPS3 elements:

linbending_ring_cps3_eq4x10.inp	4 × 10 mesh.
linbending_ring_cps3_eq4x20.inp	4 × 20 mesh.

CPS4 elements:

linbending_ring_cps4_4x10.inp	4 × 10 mesh.
linbending_ring_cps4_4x20.inp	4 × 20 mesh.

CPS4I elements:

linbending_ring_cps4i_4x10.inp	4 × 10 mesh.
linbending_ring_cps4i_4x20.inp	4 × 20 mesh.

CPS4R elements:

linbending_ring_cps4r_4x10.inp	4 × 10 mesh.
linbending_ring_cps4r_4x20.inp	4 × 20 mesh.
linbending_ring_cps4r_4x10_eh.inp	4 × 10 mesh.
linbending_ring_cps4r_4x20_eh.inp	4 × 20 mesh.

CPS6 elements:

linbending_ring_cps6_eq4x10.inp	4 × 10 mesh.
linbending_ring_cps6_eq4x20.inp	4 × 20 mesh.

CPS6M elements:

linbending_ring_cps6m_eq4x10.inp	4 × 10 mesh.
linbending_ring_cps6m_eq4x20.inp	4 × 20 mesh.

CPS8 elements:

linbending_ring_cps8_4x10.inp	4 × 10 mesh.
linbending_ring_cps8_4x20.inp	4 × 20 mesh.

CPS8R elements:

linbending_ring_cps8r_4x10.inp	4 × 10 mesh.
linbending_ring_cps8r_4x20.inp	4 × 20 mesh.

CSS8 elements:

linbending_ring_css8_4x10.inp	4 × 10 mesh.
linbending_ring_css8_4x20.inp	4 × 20 mesh.

Abaqus/Explicit input files:

Cantilever analysisC3D8 elements:

expl_linbending_c3d8_cant1x4.inp	1 × 4 mesh.
expl_linbending_c3d8_cant2x4.inp	2 × 4 mesh.
expl_linbending_c3d8_cant4x4.inp	4 × 4 mesh.
expl_linbending_c3d8_cant8x16.inp	8 × 16 mesh.

C3D8I elements:

expl_linbending_c3d8i_cant1x4.inp	1 × 4 mesh.
expl_linbending_c3d8i_cant2x4.inp	2 × 4 mesh.
expl_linbending_c3d8i_cant4x4.inp	4 × 4 mesh.
expl_linbending_c3d8i_cant8x16.inp	8 × 16 mesh.

C3D8R elements:

expl_linbending_c3d8r_cant1x4enh.inp	1 × 4 mesh.
expl_linbending_c3d8r_cant2x4enh.inp	2 × 4 mesh.
expl_linbending_c3d8r_cant4x4enh.inp	4 × 4 mesh.
expl_linbending_c3d8r_cant8x16enh.inp	8 × 16 mesh.

CPS4R elements:

expl_linbending_cps4r_cant1x4enh.inp	1 × 4 mesh.
expl_linbending_cps4r_cant2x4enh.inp	2 × 4 mesh.
expl_linbending_cps4r_cant4x4enh.inp	4 × 4 mesh.
expl_linbending_cps4r_cant8x16enh.inp	8 × 16 mesh.

S4 elements:

expl_linbending_s4_cant1x4.inp	1 × 4 mesh.
expl_linbending_s4_cant2x4.inp	2 × 4 mesh.
expl_linbending_s4_cant4x4.inp	4 × 4 mesh.
expl_linbending_s4_cant8x16.inp	8 × 16 mesh.

S4R elements:

expl_linbending_s4r_cant1x4enh.inp	1 × 4 mesh.
expl_linbending_s4r_cant2x4enh.inp	2 × 4 mesh.
expl_linbending_s4r_cant4x4enh.inp	4 × 4 mesh.
expl_linbending_s4r_cant8x16enh.inp	8 × 16 mesh.

Skew sensitivity analysis of parallelogram and trapezoidal-shaped elementsC3D8 elements:

expl_linbending_c3d8_cant1x8_ang0.inp	1 × 8 mesh; skew angle of 0°.
expl_linbending_c3d8_cant1x8_ang15.inp	Parallelogram shape; skew angle of 15°.
expl_linbending_c3d8_cant1x8_ang30.inp	Parallelogram shape; skew angle of 30°.
expl_linbending_c3d8_cant1x8_ang45.inp	Parallelogram shape; skew angle of 45°.
expl_linbending_c3d8_trap_ang15.inp	Trapezoidal shape; skew angle of 15°.
expl_linbending_c3d8_trap_ang30.inp	Trapezoidal shape; skew angle of 30°.
expl_linbending_c3d8_trap_ang45.inp	Trapezoidal shape; skew angle of 45°.

C3D8I elements:

expl_linbending_c3d8i_cant1x8_ang0.inp	1 × 8 mesh; skew angle of 0°.
expl_linbending_c3d8i_cant1x8_ang15.inp	Parallelogram shape; skew angle of 15°.
expl_linbending_c3d8i_cant1x8_ang30.inp	Parallelogram shape; skew angle of 30°.
expl_linbending_c3d8i_cant1x8_ang45.inp	Parallelogram shape; skew angle of 45°.
expl_linbending_c3d8i_trap_ang15.inp	Trapezoidal shape; skew angle of 15°.
expl_linbending_c3d8i_trap_ang30.inp	Trapezoidal shape; skew angle of 30°.
expl_linbending_c3d8i_trap_ang45.inp	Trapezoidal shape; skew angle of 45°.

C3D8R elements:

expl_linbending_c3d8r_cant1x8_ang0enh.inp	1 × 8 mesh; skew angle of 0°.
expl_linbending_c3d8r_cant1x8_ang15enh.inp	Parallelogram shape; skew angle of 15°.
expl_linbending_c3d8r_cant1x8_ang30enh.inp	Parallelogram shape; skew angle of 30°.
expl_linbending_c3d8r_cant1x8_ang45enh.inp	Parallelogram shape; skew angle of 45°.
expl_linbending_c3d8r_trap_ang15enh.inp	Trapezoidal shape; skew angle of 15°.
expl_linbending_c3d8r_trap_ang30enh.inp	Trapezoidal shape; skew angle of 30°.
expl_linbending_c3d8r_trap_ang45enh.inp	Trapezoidal shape; skew angle of 45°.

CPS4R elements:

expl_linbending_cps4r_cant1x8_ang0enh.inp	1 × 8 mesh; skew angle of 0°.
expl_linbending_cps4r_cant1x8_ang15enh.inp	Parallelogram shape; skew angle of 15°.
expl_linbending_cps4r_cant1x8_ang30enh.inp	Parallelogram shape; skew angle of 30°.
expl_linbending_cps4r_cant1x8_ang45enh.inp	Parallelogram shape; skew angle of 45°.
expl_linbending_cps4r_trap_ang15enh.inp	Trapezoidal shape; skew angle of 15°.
expl_linbending_cps4r_trap_ang30enh.inp	Trapezoidal shape; skew angle of 30°.
expl_linbending_cps4r_trap_ang45enh.inp	Trapezoidal shape; skew angle of 45°.

S4 elements:

expl_linbending_s4_1x8_ang0.inp	1 × 8 mesh; skew angle of 0°.
expl_linbending_s4_1x8_ang15.inp	Parallelogram shape; skew angle of 15°.
expl_linbending_s4_1x8_ang30.inp	Parallelogram shape; skew angle of 30°.
expl_linbending_s4_1x8_ang45.inp	Parallelogram shape; skew angle of 45°.
expl_linbending_s4_trap_ang15.inp	Trapezoidal shape; skew angle of 15°.
expl_linbending_s4_trap_ang30.inp	Trapezoidal shape; skew angle of 30°.
expl_linbending_s4_trap_ang45.inp	Trapezoidal shape; skew angle of 45°.

S4R elements:

expl_linbending_s4r_1x8_ang0enh.inp	1 × 8 mesh; skew angle of 0°.
expl_linbending_s4r_1x8_ang15enh.inp	Parallelogram shape; skew angle of 15°.
expl_linbending_s4r_1x8_ang30enh.inp	Parallelogram shape; skew angle of 30°.
expl_linbending_s4r_1x8_ang45enh.inp	Parallelogram shape; skew angle of 45°.
expl_linbending_s4r_trap_ang15enh.inp	Trapezoidal shape; skew angle of 15°.
expl_linbending_s4r_trap_ang30enh.inp	Trapezoidal shape; skew angle of 30°.
expl_linbending_s4r_trap_ang45enh.inp	Trapezoidal shape; skew angle of 45°.

References

- Belytschko, T., W. K. Liu, and J. M. Kennedy, “Hourglass Control in Linear and Nonlinear Problems,” *Computer Methods in Applied Mechanics and Engineering*, vol. 43, pp. 251–276, 1984.
- Flanagan, D. P., and T. Belytschko, “A Uniform Strain Hexahedron and Quadrilateral with Hourglass Control,” *International Journal For Numerical Methods in Engineering*, vol. 17, pp. 679–706, 1981.
- Prathap, G., “The Poor Bending Response of the Four-Node Plane Stress Quadrilateral,” *International Journal for Numerical Methods in Engineering*, vol. 21, pp. 825–835, 1985.

Tables

Table 1: Normalized tip deflection of a cantilever beam ($w_{\text{fem}}/w_{\text{exact}}$). Bernoulli-Euler theory prediction: $w_{\text{exact}} = 2.743$ mm (0.108 in).

Element Type	Mesh Size (Depth × Length)			
	1 × 4	2 × 4	4 × 4	8 × 16
CPS3	0.012	0.012	0.012	0.159
CPS4I	0.985	0.985	0.985	1.000
S4	0.899	0.943	0.937	0.966
S4[1]	0.873	0.887	0.834	0.923
S4R[2]	0.985	0.985	0.985	1.000
S4R[3]	0.985	0.985	0.985	1.000
CPS4	0.034	0.034	0.034	0.363
CPS4R	*	1.151	0.944	1.008
CPS4R[2]	0.985	0.985	0.985	1.000
CPS4R[3]	0.985	0.985	0.985	1.000
C3D4	0.001	0.001	0.002	0.065
C3D8I	0.985	0.985	0.985	1.000
C3D8I[1]	0.984	0.985	0.985	1.000
C3D8	0.035	0.034	0.034	0.364
C3D8[1]	0.034	0.034	0.034	0.364
C3D8R	*	1.306	1.050	1.016
C3D8R[2]	0.984	0.985	0.985	1.000
C3D8R[3]	0.985	0.985	0.985	1.000
CPS6	0.986	0.986	0.986	1.000
CPS6M	0.940	0.946	0.947	0.999
CPS8	0.987	0.987	0.987	1.000
CPS8R	1.001	1.001	1.001	1.001
C3D10	0.985	0.985	0.985	1.000
C3D10HS	0.985	0.985	0.985	1.000
C3D10M	1.021	0.985	0.969	1.000
C3D20	0.987	0.987	0.988	1.000
C3D20R	1.001	1.001	1.001	1.001
CSS8	0.985	0.985	0.985	1.000
* yields singular stiffness matrix				
[1]Abaqus/Explicit				
[2]Abaqus/Explicit with enhanced hourglass control				
[3]Abaqus/Standard with enhanced hourglass control				

Table 2: Normalized tip deflection of a cantilever beam with parallelogram-shaped elements ($w_{\text{fem}}/w_{\text{exact}}$). Bernoulli-Euler theory prediction: $w_{\text{exact}} = 2.743$ mm (0.108 in).

Element Type	Skew Angle			
	0°	15°	30°	45°
CPS3	0.042	0.032	0.022	0.017
CPS4I	0.996	0.898	0.791	0.742
S4	0.903	0.833	0.470	0.226
S4[1]	0.901	0.826	0.471	0.239
S4R[2]	0.996	0.898	0.791	0.742
S4R[3]	0.996	0.898	0.791	0.742
CPS4	0.125	0.110	0.079	0.049
CPS4R[2]	0.996	0.898	0.791	0.742
CPS4R[3]	0.996	0.898	0.791	0.742
C3D4	0.001	0.001	0.002	0.002
C3D8I	0.997	0.898	0.791	0.742
C3D8I[1]	0.996	0.896	0.791	0.743
C3D8	0.132	0.121	0.093	0.061
C3D8[1]	0.132	0.121	0.093	0.061
C3D8R[2]	0.996	0.897	0.790	0.742
C3D8R[3]	0.996	0.897	0.791	0.742
CPS6	0.997	0.982	0.931	0.821
CPS6M	0.991	0.985	0.965	0.926
CPS8	0.998	0.998	0.996	0.988
CPS8R	1.001	1.001	1.000	0.997
C3D10	0.997	0.897	0.711	0.484
C3D10HS	0.999	0.880	0.674	0.455
C3D10M	1.040	1.004	0.920	0.814
C3D20	0.998	0.998	0.983	0.961
C3D20R	1.001	0.988	0.980	0.896
CSS8	0.996	0.523	0.226	0.152
SC8R[3]	0.996	0.898	0.791	0.742

[1]Abaqus/Explicit

[2]Abaqus/Explicit with enhanced hourglass control

[3]Abaqus/Standard with enhanced hourglass control

Table 3: Normalized tip deflection of a cantilever beam with trapezoidal-shaped elements (w_{fem}/w_{exact}). Bernoulli-Euler theory prediction: $w_{exact} = 2.743$ mm (0.108 in).

Element Type	Skew Angle			
	0°	15°	30°	45°
CPS3	0.042	0.041	0.034	0.025
CPS4I	0.997	0.411	0.140	0.067
S4	0.903	0.469	0.169	0.102
S4[1]	0.901	0.479	0.216	0.142
S4R[2]	0.996	0.411	0.140	0.067
S4R[3]	0.996	0.411	0.140	0.067
CPS4	0.125	0.102	0.060	0.035
CPS4R[2]	0.997	0.411	0.140	0.067
CPS4R[3]	0.996	0.411	0.140	0.067
C3D4	0.001	0.002	0.006	0.010
C3D8I	0.997	0.411	0.140	0.067
C3D8I[1]	0.996	0.410	0.140	0.067
C3D8	0.132	0.108	0.063	0.037
C3D8[1]	0.132	0.108	0.063	0.037
C3D8R[2]	0.996	0.410	0.140	0.067
C3D8R[3]	0.996	0.411	0.140	0.067
CPS6	0.997	0.997	0.994	0.986
CPS6M	0.991	0.990	0.990	0.990
CPS8	0.998	0.959	0.985	0.915
CPS8R	1.001	0.971	0.996	0.981
C3D10	0.997	0.995	0.986	0.963
C3D10HS	0.999	0.995	0.986	0.963
C3D10M	1.040	1.038	1.035	1.024
C3D20	0.998	0.959	0.985	0.914
C3D20R	1.001	0.956	0.984	0.974
CSS8	0.996	0.390	0.142	0.081
SC8R[3]	0.996	0.411	0.140	0.067
[1]Abaqus/Explicit				
[2]Abaqus/Explicit with enhanced hourglass control				
[3]Abaqus/Standard with enhanced hourglass control				

Table 4: Normalized tip deflection of a cantilever beam using continuum shell elements with skewing in the thickness direction of the element ($w_{\text{fem}}/w_{\text{exact}}$). Bernoulli-Euler theory prediction: $w_{\text{exact}} = 2.743$ mm (0.108 in).
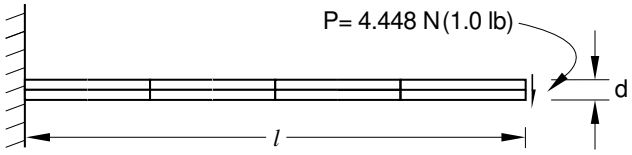
Mesh	Skew Angle			
	0°	15°	30°	45°
Parallelogram-shaped elements	1.00	1.00	1.00	1.00
Trapezoidal-shaped elements	1.00	1.00	1.00	1.00

Table 5: Normalized eigenvalue pressure estimates ($p_{\text{fem}}/p_{\text{exact}}$). Bernoulli-Euler theory prediction: $p_{\text{exact}} = 51.71$ KPa (7.5 lb/in^2).

Element Type	Mesh Size (Depth × Length)	
	4 × 10	4 × 20
CPS3	93.83	24.27
CPS4I	1.005	1.001
CPS4	32.00	8.72
CPS4R	1.059	0.968
CPS4R[1]	1.006	1.001
C3D4	64.27	16.82
C3D8I	1.005	1.001
C3D8	31.98	8.70
C3D8R	1.043	0.966
C3D8R[1]	1.006	1.001
CPS6	1.010	1.001
CPS6M	1.002	1.000
CPS8	1.027	1.002
CPS8R	1.000	1.000
C3D10	1.013	1.001
C3D10HS	1.076	1.001
C3D10M	1.014	0.998
C3D20	1.027	1.002
C3D20R	1.000	1.000
CSS8	1.006	1.001

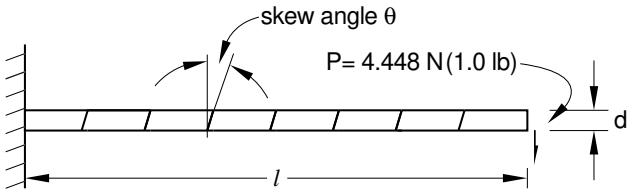
[1]Abaqus/Standard with enhanced hourglass control

Figures



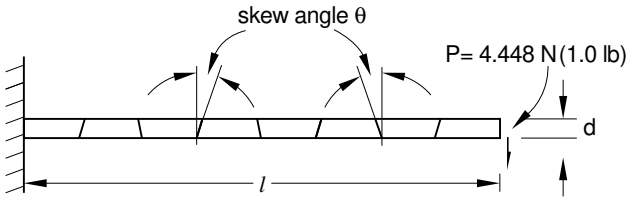
l = 152.4 mm (6 in), d = 5.08 mm (0.2 in), t = 0.254 mm (0.01 in)

Figure 1: Typical mesh and loading.



l = 152.4 mm (6 in), d = 5.08 mm (0.2 in), t = 0.254 mm (0.01 in)

Figure 2: Typical parallelogram-shaped elements.



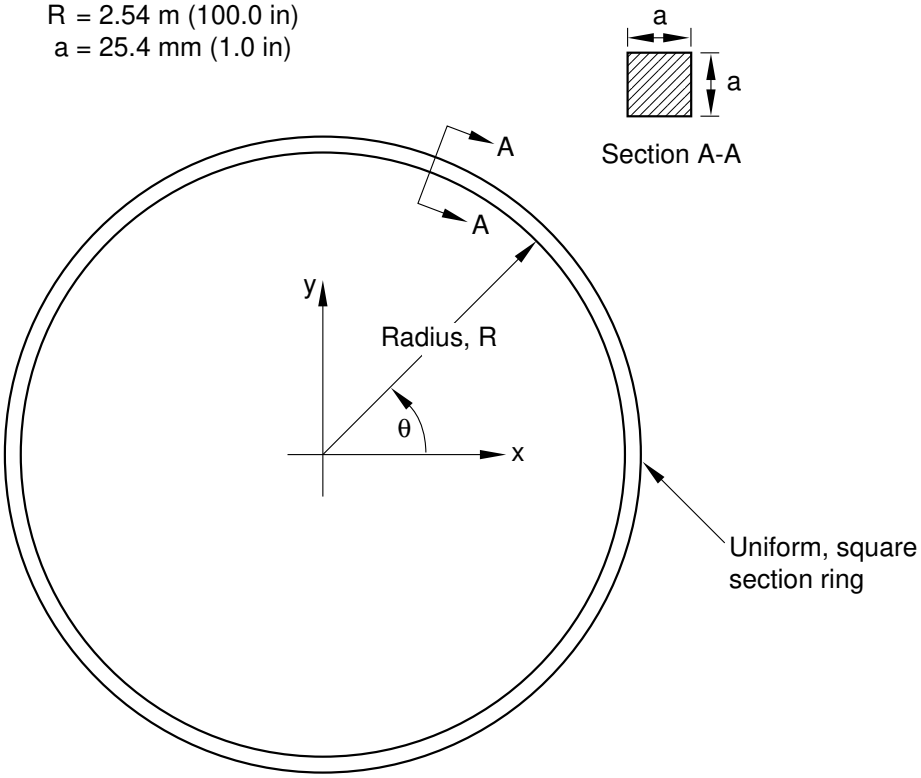
l = 152.4 mm (6 in), d = 5.08 mm (0.2 in), t = 0.254 mm (0.01 in)

Figure 3: Typical trapezoidal-shaped elements.

Geometry values used:

R = 2.54 m (100.0 in)

a = 25.4 mm (1.0 in)



Material: linear elastic

Young's modulus = 206.8 GPa
(30.0 x 10^6 lb/in^2)

Poisson's ratio = 0.0

Loading: uniform external pressure

Figure 4: Ring buckling problem.

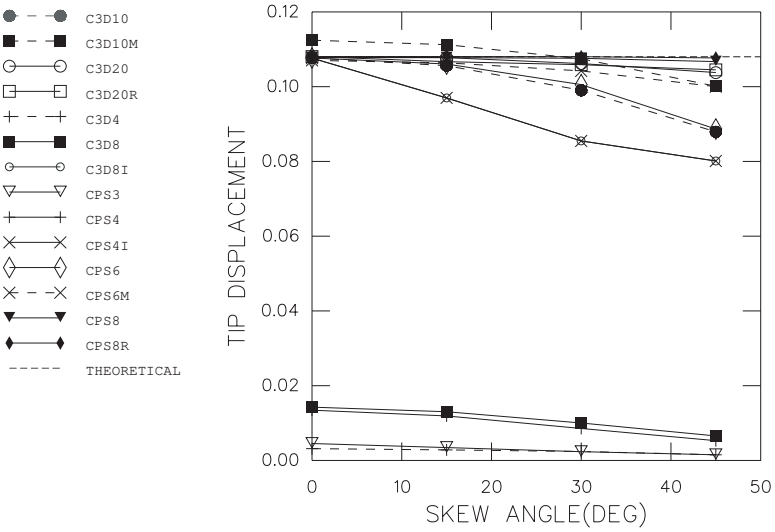


Figure 5: Parallelogram-shaped continuum elements: tip displacement versus skew angle.

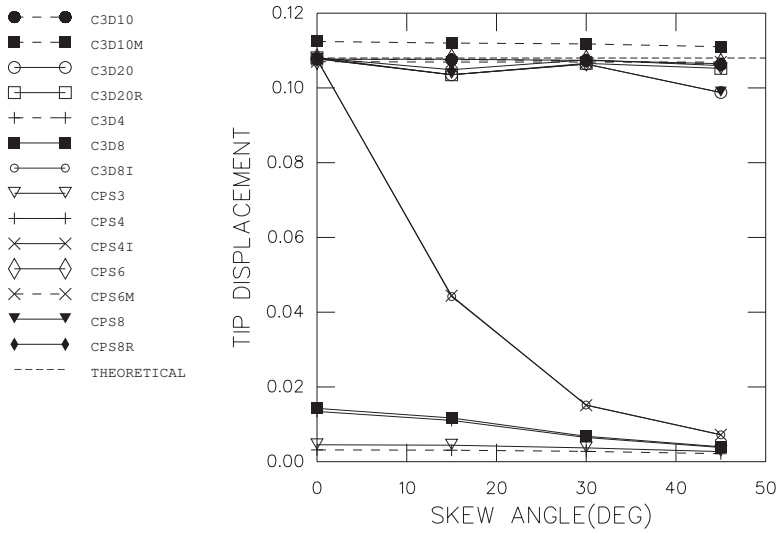


Figure 6: Trapezoidal-shaped continuum elements: tip displacement versus skew angle.

Tip in-plane shear load on a cantilevered hook

Products: Abaqus/Standard

Overview

The Raasch Challenge problem is used to demonstrate how Abaqus models in-plane shear loading in a curved strip using shell elements.
Transverse shear flexibility and proper treatment of twisting deformations of the shell elements are important factors in determining the bending behavior. The results are compared to the reference solution by Knight (1997).

Problem description

The geometry consists of a “hook” in the form of a curved strip rigidly clamped at one end and loaded with a unit in-plane shear along the width at the other end. It has two circular segments that are connected at the tangent point. The smaller segment has a mean radius of 0.3556 m (14 inches) and spans 60° from the clamped end to the tangent point. The larger segment spans 150° from the tangent point to the free end and has a mean radius of 1.1684 m (46 inches). The hook is 0.0508 m (2 inches) thick and 0.508 m (20 inches) wide, modeled as linear elastic with an elastic modulus of 22.77 MPa (3300 psi) and a Poisson's ratio of 0.35. In most tests the shear force is applied through the use of a distributing coupling constraint. The coupling constraint provides coupling between a reference node on which the load is prescribed and the nodes located on the free end. The distributed nodal loads on the free end are equivalent to a uniformly distributed load of 8.7563 N/m (0.05 lb/in). In two of the tests an equivalent shear force is applied as a distributed shear traction instead.

The problem is modeled using fully integrated S4 shell elements with five different meshes: 1 × 9, 3 × 18, 5 × 36, 10 × 72, and 20 × 144. For comparison the problem is also analyzed with S4R shell elements and SC8R continuum shell elements that use reduced integration. The reference solution is obtained with a refined mesh using C3D20R continuum elements with reduced integration.

Results and discussion

The solution reported is the in-plane displacement along the centerline of the loaded end. A comparison of the tip displacement normalized with the reference solution is shown in *Table 1*. The reduced-integration elements, S4R and SC8R, show excessively large displacement for the coarse mesh (1 × 9) because of the elements' poor treatment of in-plane bending deformations with coarse meshes. The coarse mesh for S4 elements gives a solution that is only about 3.5% stiffer than the reference solution. The refined meshes for both types of shell elements give comparable solutions, which are 0.2% more flexible than the reference solution.

The solution for the continuum shell mesh with a single element in the thickness direction converges to an excessively large displacement. This may be due to the element's poor treatment of drill stiffness. Stacking two or more elements yields the exact solution even for a coarse mesh (3 × 18 × 2).

The solutions using a distributed shear traction to apply the load agree exactly with the solutions using a coupling constraint.

Input files

C3D20R elements:

raasch_c3d20r_20x144x2.inp — 20 × 144 × 2 mesh.

S4 elements:

raasch_s4_1x9.inp	1 × 9 mesh.
raasch_s4_1x9_edld.inp	1 × 9 mesh loaded with a distributed edge traction.
raasch_s4_3x18.inp	3 × 18 mesh.
raasch_s4_5x36.inp	5 × 36 mesh.
raasch_s4_10x72.inp	10 × 72 mesh.
raasch_s4_20x144.inp	20 × 144 mesh.

S4R elements:

raasch_s4r_1x9.inp	1 × 9 mesh.
raasch_s4r_3x18.inp	3 × 18 mesh.
raasch_s4r_5x36.inp	5 × 36 mesh.
raasch_s4r_10x72.inp	10 × 72 mesh.
raasch_s4r_20x144.inp	20 × 144 mesh.

SC8R elements:

raasch_sc8r_1x9x1.inp	1 × 9 × 1 mesh.
raasch_sc8r_3x18x1.inp	3 × 18 × 1 mesh.
raasch_sc8r_5x36x1.inp	5 × 36 × 1 mesh.
raasch_sc8r_10x72x1.inp	10 × 72 × 1 mesh.
raasch_sc8r_20x144x1.inp	20 × 144 × 1 mesh.
raasch_sc8r_1x9x2.inp	1 × 9 × 2 mesh.
raasch_sc8r_3x18x2.inp	3 × 18 × 2 mesh.
raasch_sc8r_3x18x2_trvec.inp	3 × 18 × 2 mesh loaded with a distributed general traction.
raasch_sc8r_5x36x2.inp	5 × 36 × 2 mesh.
raasch_sc8r_10x72x2.inp	10 × 72 × 2 mesh.
raasch_sc8r_20x144x2.inp	20 × 144 × 2 mesh.

References

- Knight, N. F., Jr., “The Raasch Challenge for Shell Elements,” *AIAA Journal*, vol. 35, no. 2, pp. 375–381, 1997.

Tables

Table 1: Comparison of tip deflections (normalized with continuum solution) in the direction of load. (Continuum solution is 5.020 for 20 × 144 × 2 mesh of C3D20R elements.)

Mesh	S4	S4R	SC8R	
			single element	two elements stacked
1 × 9	0.967	2.951	2.622	1.693
3 × 18	0.972	0.979	1.534	1.019
5 × 36	0.987	0.989	1.522	1.007
10 × 72	0.998	0.999	1.530	1.011
20 × 144	1.003	1.003	1.535	1.015

Figures

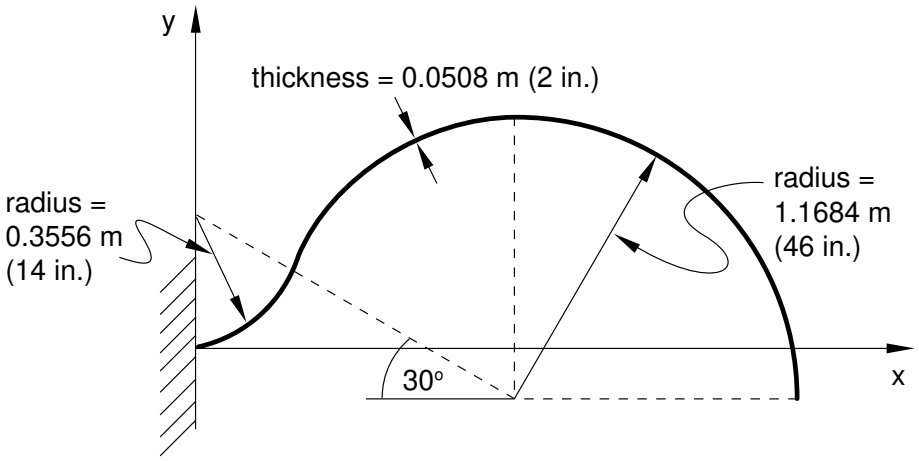


Figure 1: The Raasch Challenge problem.

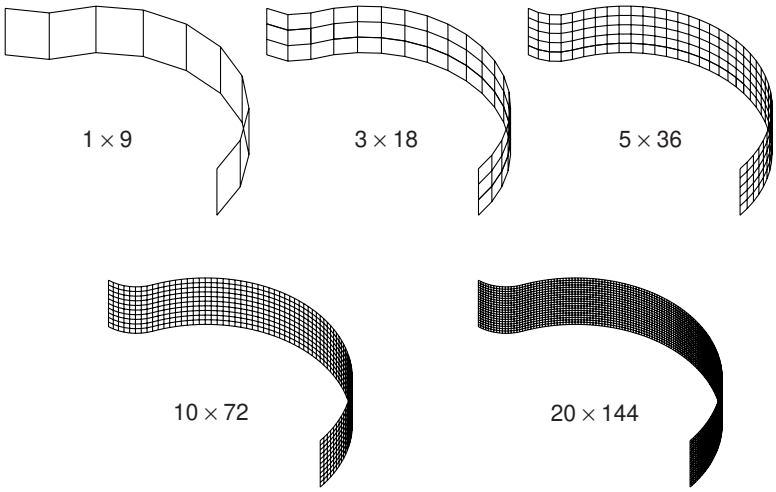


Figure 2: Meshes used for shell elements.

Analysis of a twisted beam

Products: Abaqus/Standard Abaqus/Explicit

Overview

This problem illustrates the accuracy of shell and beam finite element solutions for bending of warped structures. The responses of both a thick and thin twisted cantilever beam subjected to either an in-plane or out-of-plane shear load are obtained. The test was proposed by MacNeal and Harder (1985), who provided the analytical solution for the thick twisted beam. The reference solution for the thin twisted beam was provided by Simo et al. (1989).

Problem description

The structure is a cantilever beam, 12.0 in long and 1.1 in wide, that twists 90° from end to end, as shown in *Figure 1*. The beam is aligned with the x-axis. Its thickness, b, is 0.32 in for the thick case and 0.05 in for the thin case.

The beam is modeled in Abaqus/Standard with 4-node shell elements (S4, S4R, and S4R5), 3-node shell elements (S3R and STRI3), quadratic shell elements (STRI65, S8R, S8R5, and S9R5), continuum shell elements (SC8R), and beam elements (B31, B32, and B33). Three mesh densities are considered for each element type. The coarsest mesh of 4-node shell elements (2×12 with a warp angle of 7.5° per element length) is illustrated in *Figure 1*. The 3-node shell mesh has the same number of elements as the equivalent 4-node shell mesh. The quadratic shell mesh has half as many elements in each direction (in general, the same number of degrees of freedom) as the corresponding linear shell mesh. The coarsest mesh of beam elements uses 12 linear elements.

The beam is modeled in Abaqus/Explicit with a 2×12 mesh of S4R, S4RS, or S4RSW elements.

The material is steel with a Young's modulus of 29.0 Msi and a Poisson's ratio of 0.22. A point load of 1.0 lb is applied at the center of the free end in the y- and z-directions, respectively.

Results and discussion

The results are listed in *Table 1* to *Table 12*. The tip displacements in the load directions are compared with the analytical solution.

Abaqus/Standard results

The shell element models all converge to the analytical solution for both load cases and thicknesses. Even for the coarsest meshes, where for the 4-node shells the warp angle is 7.5° per element, the results are in good agreement.

The 4-node quadrilateral results are listed in *Table 1* and *Table 2*, the 3-node triangular results are listed in *Table 5* and *Table 6*, and the second-order shell results are listed in *Table 7* and *Table 8*. For the coarsest meshes the first-order shell results are somewhat better for the in-plane than for the out-of-plane loading case. The out-of-plane loading case causes in-plane bending deformation at the built-in end, where the maximum bending moments occur (refer to *Figure 1*). First-order triangular and reduced-integration quadrilateral elements require mesh refinement to model this in-plane bending accurately. The first-order fully integrated shell, S4, and the second-order reduced-integration elements capture the correct in-plane bending behavior.

The continuum shell results are listed in *Table 3* and *Table 4* for both load cases and thicknesses. Results are compared for cases in which 1, 2, 4, and 8 elements are stacked in the thickness direction. For the case with a single element stacked in the thickness direction, the results show excessively large displacements. This may be

due to the element's poor treatment of drill stiffness. The results show good agreement for cases with multiple elements (even two elements) stacked in the thickness direction.

The beam element models accurately reproduce the analytical result for both load cases and thicknesses; see *Table 9* and *Table 10*. Since the coarsest mesh is sufficiently refined to capture the analytical solution, the results do not improve with mesh refinement.

Abaqus/Explicit results

Figure 2 shows the time history of the tip displacement and various energies for the in-plane shear load case when the beam has a thickness of 0.32 in. The tip displacement values indicated in the tabulated results are the displacement values at node 132 in the direction of the applied tip load. *Table 11* and *Table 12* compare the solutions obtained with elements S4R, S4RS, and S4RSW. The response of S4RS is quite similar to that of S4R.

Input files

B31 elements:

twistedbeam_b31_12_thick.inp	12-element, thick beam.
twistedbeam_b31_12_thin.inp	12-element, thin beam.
twistedbeam_b31_24_thick.inp	24-element, thick beam.
twistedbeam_b31_24_thin.inp	24-element, thin beam.
twistedbeam_b31_48_thick.inp	48-element, thick beam.
twistedbeam_b31_48_thin.inp	48-element, thin beam.

B32 elements:

twistedbeam_b32_6_thick.inp	6-element, thick beam.
twistedbeam_b32_6_thin.inp	6-element, thin beam.
twistedbeam_b32_12_thick.inp	12-element, thick beam.
twistedbeam_b32_12_thin.inp	12-element, thin beam.
twistedbeam_b32_24_thick.inp	24-element, thick beam.
twistedbeam_b32_24_thin.inp	24-element, thin beam.

B33 elements:

twistedbeam_b33_12_thick.inp	12-element, thick beam.
twistedbeam_b33_12_thin.inp	12-element, thin beam.
twistedbeam_b33_24_thick.inp	24-element, thick beam.
twistedbeam_b33_24_thin.inp	24-element, thin beam.
twistedbeam_b33_48_thick.inp	48-element, thick beam.
twistedbeam_b33_48_thin.inp	48-element, thin beam.

S3R elements:

twistedbeam_s3r_2x12_thick.inp	2 × 12 mesh, thick beam.
twistedbeam_s3r_2x12_thin.inp	2 × 12 mesh, thin beam.
twistedbeam_s3r_4x24_thick.inp	4 × 24 mesh, thick beam.
twistedbeam_s3r_4x24_thin.inp	4 × 24 mesh, thin beam.
twistedbeam_s3r_8x48_thick.inp	8 × 48 mesh, thick beam.
twistedbeam_s3r_8x48_thin.inp	8 × 48 mesh, thin beam.

S4 elements:

twistedbeam_s4_2x12_thick.inp	2 × 12 mesh, thick beam.
twistedbeam_s4_2x12_thin.inp	2 × 12 mesh, thin beam.
twistedbeam_s4_4x24_thick.inp	4 × 24 mesh, thick beam.
twistedbeam_s4_4x24_thin.inp	4 × 24 mesh, thin beam.
twistedbeam_s4_8x48_thick.inp	8 × 48 mesh, thick beam.
twistedbeam_s4_8x48_thin.inp	8 × 48 mesh, thin beam.

S4R elements:

twistedbeam_s4r_2x12_thick.inp	2 × 12 mesh, thick beam.
twistedbeam_s4r_2x12_thin.inp	2 × 12 mesh, thin beam.
twistedbeam_s4r_4x24_thick.inp	4 × 24 mesh, thick beam.
twistedbeam_s4r_4x24_thin.inp	4 × 24 mesh, thin beam.
twistedbeam_s4r_8x48_thick.inp	8 × 48 mesh, thick beam.
twistedbeam_s4r_8x48_thin.inp	8 × 48 mesh, thin beam.

S4R5 elements:

twistedbeam_s4r5_2x12_thick.inp	2 × 12 mesh, thick beam.
twistedbeam_s4r5_2x12_thin.inp	2 × 12 mesh, thin beam.
twistedbeam_s4r5_4x24_thick.inp	4 × 24 mesh, thick beam.
twistedbeam_s4r5_4x24_thin.inp	4 × 24 mesh, thin beam.
twistedbeam_s4r5_8x48_thick.inp	8 × 48 mesh, thick beam.
twistedbeam_s4r5_8x48_thin.inp	8 × 48 mesh, thin beam.

2 × 12 mesh of S4RSW elements in Abaqus/Explicit:

twistedbeam_thick_fy.inp	Thick beam, tip load in *y*-direction.
twistedbeam_thick_fz.inp	Thick beam, tip load in *z*-direction.
twistedbeam_thin_fy.inp	Thin beam, tip load in *y*-direction.
twistedbeam_thin_fz.inp	Thin beam, tip load in *z*-direction.

2 × 12 mesh of S4RS elements in Abaqus/Explicit:

twistedbeam_s4rs_thick_fy.inp	Thick beam, tip load in *y*-direction.
twistedbeam_s4rs_thick_fz.inp	Thick beam, tip load in *z*-direction.
twistedbeam_s4rs_thin_fy.inp	Thin beam, tip load in *y*-direction.
twistedbeam_s4rs_thin_fz.inp	Thin beam, tip load in *z*-direction.

2 × 12 mesh of S4R elements in Abaqus/Explicit:

twistedbeam_s4r_thick_fy.inp	Thick beam, tip load in *y*-direction.
twistedbeam_s4r_thick_fz.inp	Thick beam, tip load in *z*-direction.
twistedbeam_s4r_thin_fy.inp	Thin beam, tip load in *y*-direction.
twistedbeam_s4r_thin_fz.inp	Thin beam, tip load in *z*-direction.

S8R elements:

twistedbeam_s8r_2x12_thick.inp	2 × 12 mesh, thick beam.
twistedbeam_s8r_2x12_thin_s8r.inp	2 × 12 mesh, thin beam.
twistedbeam_s8r_4x24_thick.inp	4 × 24 mesh, thick beam.
twistedbeam_s8r_4x24_thin_s8r.inp	4 × 24 mesh, thin beam.

twistedbeam_s8r_8x48_thick.inp	8 × 48 mesh, thick beam.
twistedbeam_s8r_8x48_thin_s8r.inp	8 × 48 mesh, thin beam.

S8R5 elements:

twistedbeam_s8r5_2x12_thick.inp	2 × 12 mesh, thick beam.
twistedbeam_s8r5_2x12_thin.inp	2 × 12 mesh, thin beam.
twistedbeam_s8r5_4x24_thick.inp	4 × 24 mesh, thick beam.
twistedbeam_s8r5_4x24_thin.inp	4 × 24 mesh, thin beam.
twistedbeam_s8r5_8x48_thick.inp	8 × 48 mesh, thick beam.
twistedbeam_s8r5_8x48_thin.inp	8 × 48 mesh, thin beam.

S9R5 elements:

twistedbeam_s9r5_2x12_thick.inp	2 × 12 mesh, thick beam.
twistedbeam_s9r5_2x12_thin.inp	2 × 12 mesh, thin beam.
twistedbeam_s9r5_4x24_thick.inp	4 × 24 mesh, thick beam.
twistedbeam_s9r5_4x24_thick.inp	4 × 24 mesh, thin beam.
twistedbeam_s9r5_8x48_thick.inp	8 × 48 mesh, thick beam.
twistedbeam_s9r5_8x48_thin.inp	8 × 48 mesh, thin beam.

STRI3 elements:

twistedbeam_stri3_2x12_thick.inp	2 × 12 mesh, thick beam.
twistedbeam_stri3_2x12_thin.inp	2 × 12 mesh, thin beam.
twistedbeam_stri3_4x24_thick.inp	4 × 24 mesh, thick beam.
twistedbeam_stri3_4x24_thin.inp	4 × 24 mesh, thin beam.
twistedbeam_stri3_8x48_thick.inp	8 × 48 mesh, thick beam.
twistedbeam_stri3_8x48_thin.inp	8 × 48 mesh, thin beam.

STRI65 elements:

twistedbeam_stri65_2x12_thick.inp	2 × 12 mesh, thick beam.
twistedbeam_stri65_2x12_thin.inp	2 × 12 mesh, thin beam.
twistedbeam_stri65_4x24_thick.inp	4 × 24 mesh, thick beam.
twistedbeam_stri65_4x24_thin.inp	4 × 24 mesh, thin beam.
twistedbeam_stri65_8x48_thick.inp	8 × 48 mesh, thick beam.
twistedbeam_stri65_8x48_thin.inp	8 × 48 mesh, thin beam.

SC8R elements:

twistedbeam_sc8r_2x12x1_thick.inp	2 × 12 × 1 mesh, thick beam.
twistedbeam_sc8r_2x12x2_thick.inp	2 × 12 × 2 mesh, thick beam.
twistedbeam_sc8r_2x12x4_thick.inp	2 × 12 × 4 mesh, thick beam.
twistedbeam_sc8r_2x12x8_thick.inp	2 × 12 × 8 mesh, thick beam.
twistedbeam_sc8r_4x24x1_thick.inp	4 × 24 × 1 mesh, thick beam.
twistedbeam_sc8r_4x24x2_thick.inp	4 × 24 × 2 mesh, thick beam.
twistedbeam_sc8r_4x24x4_thick.inp	4 × 24 × 4 mesh, thick beam.
twistedbeam_sc8r_4x24x8_thick.inp	4 × 24 × 8 mesh, thick beam.
twistedbeam_sc8r_8x48x1_thick.inp	8 × 48 × 1 mesh, thick beam.
twistedbeam_sc8r_8x48x2_thick.inp	8 × 48 × 2 mesh, thick beam.

twistedbeam_sc8r_8x48x4_thick.inp	8 × 48 × 4 mesh, thick beam.
twistedbeam_sc8r_8x48x8_thick.inp	8 × 48 × 8 mesh, thick beam.
twistedbeam_sc8r_2x12x1_thin.inp	2 × 12 × 1 mesh, thin beam.
twistedbeam_sc8r_2x12x2_thin.inp	2 × 12 × 2 mesh, thin beam.
twistedbeam_sc8r_2x12x4_thin.inp	2 × 12 × 4 mesh, thin beam.
twistedbeam_sc8r_2x12x8_thin.inp	2 × 12 × 8 mesh, thin beam.
twistedbeam_sc8r_4x24x1_thin.inp	4 × 24 × 1 mesh, thin beam.
twistedbeam_sc8r_4x24x2_thin.inp	4 × 24 × 2 mesh, thin beam.
twistedbeam_sc8r_4x24x4_thin.inp	4 × 24 × 4 mesh, thin beam.
twistedbeam_sc8r_4x24x8_thin.inp	4 × 24 × 8 mesh, thin beam.
twistedbeam_sc8r_8x48x1_thin.inp	8 × 48 × 1 mesh, thin beam.
twistedbeam_sc8r_8x48x2_thin.inp	8 × 48 × 2 mesh, thin beam.
twistedbeam_sc8r_8x48x4_thin.inp	8 × 48 × 4 mesh, thin beam.
twistedbeam_sc8r_8x48x8_thin.inp	8 × 48 × 8 mesh, thin beam.

References

- MacNeal, R. H., and R. L. Harder, “A Proposed Standard Set of Problems to Test Finite Element Accuracy,” *Finite Elements in Analysis Design*, vol. 11, pp. 3–20, 1985.
- Simo, J. C., D. D. Fox, and M. S. Rifai, “On a Stress Resultant Geometrically Exact Shell Model. Part II: The Linear Theory; Computational Aspects,” *Computational Methods in Applied Mechanical Engineering*, vol. 73, pp. 53–92, 1989.

Tables

Table 1: Tip displacements for 4-node shell meshes, thick case (b = 0.32 in).

Loading		In-plane (F_y = 1.0 lb)		Out-of-plane (F_z = 1.0 lb)	
Reference solution		5.424×10^{-3} (in)		1.754×10^{-3} (in)	
Element	Mesh	FE solution	% error	FE solution	% error
S4	2 × 12	5.440×10^{-3}	0.29	1.730×10^{-3}	−1.37
	4 × 24	5.428×10^{-3}	0.07	1.747×10^{-3}	−0.40
	8 × 48	5.427×10^{-3}	0.05	1.753×10^{-3}	−0.06
S4R	2 × 12	5.479×10^{-3}	1.01	1.868×10^{-3}	6.50
	4 × 24	5.437×10^{-3}	0.24	1.777×10^{-3}	1.31
	8 × 48	5.430×10^{-3}	0.11	1.761×10^{-3}	0.40
S4R5	2 × 12	5.443×10^{-3}	0.35	1.879×10^{-3}	7.10
	4 × 24	5.418×10^{-3}	−0.10	1.768×10^{-3}	0.78
	8 × 48	5.416×10^{-3}	−0.15	1.755×10^{-3}	0.05

Table 2: Tip displacements for 4-node shell meshes, thin case (b = 0.05 in).

Loading		In-plane (F_y = 1.0 lb)		Out-of-plane (F_z = 1.0 lb)	
Reference solution		1.390 (in)		0.3431 (in)	
Element	Mesh	FE solution	% error	FE solution	% error
S4	2 × 12	1.391	0.07	0.3397	–0.99
	4 × 24	1.388	–0.14	0.3421	–0.29
	8 × 48	1.388	–0.14	0.3427	–0.12
S4R	2 × 12	1.394	0.28	0.3403	–0.81
	4 × 24	1.389	–0.07	0.3422	–0.26
	8 × 48	1.388	–0.14	0.3428	–0.09
S4R5	2 × 12	1.389	–0.07	0.3388	–1.25
	4 × 24	1.387	–0.22	0.3418	–0.38
	8 × 48	1.387	–0.22	0.3426	–0.15

Table 3: Tip displacements for continuum shell meshes, thick case (b = 0.32 in).

Loading		In-plane (F_y = 1.0 lb)		Out-of-plane (F_z = 1.0 lb)	
Reference solution		5.424×10^{-3} (in)		1.754×10^{-3} (in)	
Element	Mesh	FE solution	% error	FE solution	% error
SC8R	2 × 12 × 1	7.819×10^{-3}	44.2	2.428×10^{-3}	38.4
	2 × 12 × 2	5.254×10^{-3}	–3.13	1.887×10^{-3}	7.59
	2 × 12 × 4	5.352×10^{-3}	–1.33	1.874×10^{-3}	6.82
	2 × 12 × 8	5.410×10^{-3}	–0.27	1.873×10^{-3}	6.79
	4 × 24 × 1	7.696×10^{-3}	41.9	2.388×10^{-3}	36.2
	4 × 24 × 2	5.229×10^{-3}	–3.59	1.798×10^{-3}	2.53
	4 × 24 × 4	5.349×10^{-3}	–1.38	1.777×10^{-3}	1.28
	4 × 24 × 4	5.395×10^{-3}	–0.53	1.775×10^{-3}	1.17
	8 × 48 × 1	7.635×10^{-3}	40.8	2.380×10^{-3}	35.7
	8 × 48 × 2	5.220×10^{-3}	–3.76	1.781×10^{-3}	1.54
	8 × 48 × 4	5.331×10^{-3}	–1.72	1.781×10^{-3}	0.3
	8 × 48 × 8	5.393×10^{-3}	–0.57	1.757×10^{-3}	0.19

Table 4: Tip displacements for continuum shell meshes, thin case (b = 0.05 in).

Loading		In-plane (F_y = 1.0 lb)		Out-of-plane (F_z = 1.0 lb)	
Reference solution		1.390 (in)		0.3431 (in)	
Element	Mesh	FE solution	% error	FE solution	% error
SC8R	2 × 12 × 1	1.927	38.6	0.5826	69.8
	2 × 12 × 2	1.347	–3.09	0.3574	4.17

Loading		In-plane (F_y = 1.0 lb)		Out-of-plane (F_z = 1.0 lb)	
Reference solution		1.390 (in)		0.3431 (in)	
Element	Mesh	FE solution	% error	FE solution	% error
	2 × 12 × 4	1.366	–1.73	0.3412	–0.55
	2 × 12 × 8	1.378	–0.86	0.3384	–1.37
	4 × 24 × 1	1.908	37.3	0.5828	69.9
	4 × 24 × 2	1.346	–3.17	0.3608	5.16
	4 × 24 × 4	1.368	–1.58	0.3451	0.58
	4 × 24 × 4	1.381	–0.65	0.3423	–0.23
	8 × 48 × 1	1.903	36.9	0.5829	69.9
	8 × 48 × 2	1.346	–3.17	0.3617	5.42
	8 × 48 × 4	1.368	–1.58	0.3461	0.87
	8 × 48 × 8	1.382	–0.59	0.3433	0.06

Table 5: Tip displacements for 3-node shell meshes, thick case (b = 0.32 in).

Loading		In-plane (F_y = 1.0 lb)		Out-of-plane (F_z = 1.0 lb)	
Reference solution		5.424×10^{-3} (in)		1.754×10^{-3} (in)	
Element	Mesh	FE solution	% error	FE solution	% error
S3R	4 × 6	5.262×10^{-3}	–2.99	1.400×10^{-3}	–20.18
	8 × 12	5.361×10^{-3}	–1.16	1.581×10^{-3}	–9.86
	16 × 24	5.405×10^{-3}	–0.35	1.696×10^{-3}	–3.31
STRI3	4 × 6	5.323×10^{-3}	–1.86	1.438×10^{-3}	–18.01
	8 × 12	5.359×10^{-3}	–1.20	1.594×10^{-3}	–9.18
	16 × 24	5.386×10^{-3}	–0.70	1.698×10^{-3}	–3.19

Table 6: Tip displacements for 3-node shell meshes, thin case (b = 0.05 in).

Loading		In-plane (F_y = 1.0 lb)		Out-of-plane (F_z = 1.0 lb)	
Reference solution		1.390 (in)		0.3431 (in)	
Element	Mesh	FE solution	% error	FE solution	% error
S3R	4 × 6	1.352	–2.73	0.3251	–5.25
	8 × 12	1.372	–1.29	0.3381	–1.46
	16 × 24	1.383	–0.50	0.3417	–0.41
STRI3	4 × 6	1.383	–0.50	0.3382	–1.43
	8 × 12	1.384	–0.43	0.3413	–0.52
	16 × 24	1.386	–0.29	0.3424	–0.20

Table 7: Tip displacements for quadratic shell meshes, thick case (b = 0.32 in).

Loading		In-plane (F_y = 1.0 lb)		Out-of-plane (F_z = 1.0 lb)	
Reference solution		5.424×10^{-3} (in)		1.754×10^{-3} (in)	
Element	Mesh	FE solution	% error	FE solution	% error
STRI65	2 × 6	5.408×10^{-3}	–0.29	1.751×10^{-3}	–0.17
	4 × 12	5.412×10^{-3}	–0.22	1.752×10^{-3}	–0.11
	8 × 24	5.414×10^{-3}	–0.18	1.752×10^{-3}	–0.11
S8R	1 × 6	5.376×10^{-3}	–0.88	1.745×10^{-3}	–0.51
	2 × 12	5.411×10^{-3}	–0.24	1.752×10^{-3}	–0.11
	4 × 24	5.415×10^{-3}	–0.17	1.752×10^{-3}	–0.11
S8R5 & S9R5	1 × 6	5.405×10^{-3}	–0.35	1.746×10^{-3}	–0.46
	2 × 12	5.413×10^{-3}	–0.20	1.752×10^{-3}	–0.11
	4 × 24	5.416×10^{-3}	–0.15	1.753×10^{-3}	–0.06

Table 8: Tip displacements for quadratic shell meshes, thin case (b = 0.05 in).

Loading		In-plane (F_y = 1.0 lb)		Out-of-plane (F_z = 1.0 lb)	
Reference solution		1.390 (in)		0.3431 (in)	
Element	Mesh	FE solution	% error	FE solution	% error
STRI65	2 × 6	1.384	–0.43	0.3420	–0.32
	4 × 12	1.384	–0.43	0.3429	–0.06
	8 × 24	1.386	–0.29	0.3429	–0.06
S8R	1 × 6	1.214	–12.66	0.3311	–3.50
	2 × 12	1.379	–0.79	0.3427	–0.11
	4 × 24	1.387	–0.22	0.3429	–0.05
S8R5 & S9R5	1 × 6	1.386	–0.29	0.3423	–0.23
	2 × 12	1.387	–0.22	0.3429	–0.05
	4 × 24	1.387	–0.21	0.3429	–0.05

Table 9: Tip displacements for beam meshes, thick case (b = 0.32 in).

Loading		In-plane (F_y = 1.0 lb)		Out-of-plane (F_z = 1.0 lb)	
Reference solution		5.424×10^{-3} (in)		1.754×10^{-3} (in)	
Element	Mesh	FE solution	% error	FE solution	% error
B31	12	5.422×10^{-3}	–0.04	1.753×10^{-3}	–0.06
	24	5.428×10^{-3}	0.07	1.750×10^{-3}	–0.23
	48	5.429×10^{-3}	0.09	1.750×10^{-3}	–0.23
B32	6	5.429×10^{-3}	0.09	1.750×10^{-3}	–0.23
	12	5.429×10^{-3}	0.09	1.750×10^{-3}	–0.23

Loading		In-plane (F_y = 1.0 lb)		Out-of-plane (F_z = 1.0 lb)	
Reference solution		5.424×10^{-3} (in)		1.754×10^{-3} (in)	
Element	Mesh	FE solution	% error	FE solution	% error
	24	5.429×10^{-3}	0.09	1.750×10^{-3}	–0.23
B33	12	5.430×10^{-3}	0.11	1.743×10^{-3}	–0.63
	24	5.429×10^{-3}	0.09	1.743×10^{-3}	–0.63
	48	5.428×10^{-3}	0.07	1.743×10^{-3}	–0.63

Table 10: Tip displacements for beam meshes, thin case (b = 0.05 in).

Loading		In-plane (F_y = 1.0 lb)		Out-of-plane (F_z = 1.0 lb)	
Reference solution		1.390 (in)		0.3431 (in)	
Element	Mesh	FE solution	% error	FE solution	% error
B31	12	1.392	0.15	0.3438	0.26
	24	1.394	0.29	0.3430	–0.03
	48	1.394	0.29	0.3428	–0.03
B32	6	1.394	0.29	0.3427	–0.03
	12	1.394	0.29	0.3427	–0.03
	24	1.394	0.29	0.3427	–0.03
B33	12	1.395	0.36	0.3417	–0.32
	24	1.395	0.36	0.3418	–0.32
	48	1.395	0.36	0.3421	–0.32

Table 11: Tip displacements for 4-node shell 2 x 12 mesh in Abaqus/Explicit, thick case (b = 0.32 in).

Loading		In-plane (F_y = 1.0 lb)		Out-of-plane (F_z = 1.0 lb)	
Reference solution		5.424×10^{-3} (in)		1.754×10^{-3} (in)	
Element	Mesh	FE solution	% error	FE solution	% error
S4R	2 × 12	5.542 x 10^{-3}	2.18	1.800×10^{-3}	2.62
S4RS	2 × 12	5.438×10^{-3}	2.57	1.802×10^{-3}	2.74
S4RSW	2 × 12	5.435×10^{-3}	0.20	1.869×10^{-3}	6.56

Table 12: Tip displacements for 4-node shell 2 x 12 mesh in Abaqus/Explicit, thin case (b = 0.05 in).

Loading		In-plane (F_y = 1.0 lb)		Out-of-plane (F_z = 1.0 lb)	
Reference solution		1.390 (in)		0.3431 (in)	
Element	Mesh	FE solution	% error	FE solution	% error
S4R	2 × 12	1.366	-1.73	0.3443	0.35
S4RS	2 × 12	1.376	-1.01	0.3390	–1.19

Loading		In-plane (F_y = 1.0 lb)		Out-of-plane (F_z = 1.0 lb)	
Reference solution		1.390 (in)		0.3431 (in)	
Element	Mesh	FE solution	% error	FE solution	% error
S4RSW	2 × 12	1.424	2.45	0.3821	11.37

Figures

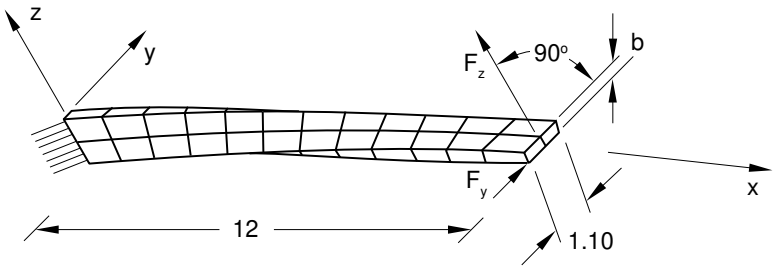


Figure 1: Twisted beam.

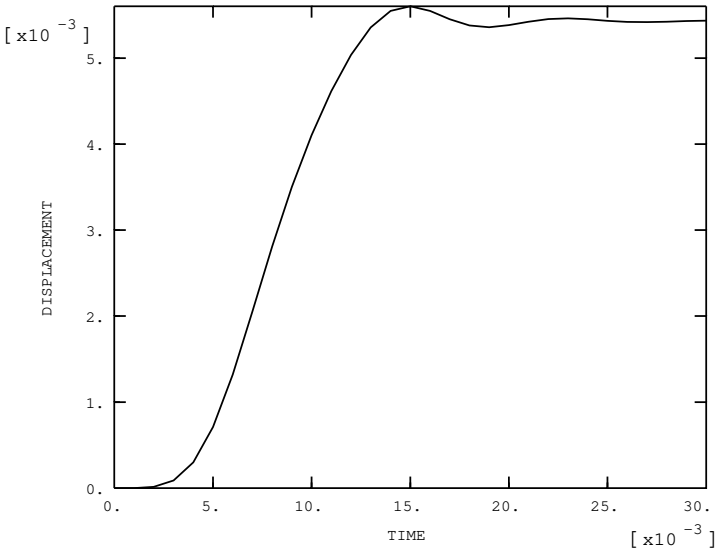


Figure 2: Variation of u_2 at node 132 with time, Abaqus/Explicit analysis.

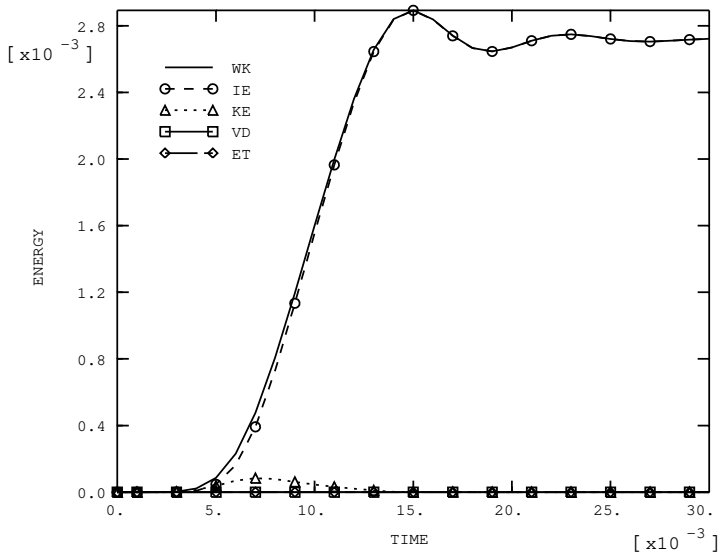


Figure 3: Energy variation with time, Abaqus/Explicit analysis.

Twisted ribbon test for shells

Products: Abaqus/Standard

Overview

This problem illustrates the use of shell elements for bending of a twisted plate. The results are compared with the reference solution by Batoz (1982).

Problem description

The original problem description and the benchmark solutions can be found in Batoz (1982).

The length of the plate is varied from 2.0 to 10.0 to study aspect ratio effects. The material is linear elastic with a Young's modulus of 1×10^7 and a Poisson's ratio of 0.25. Nodes along edge AD are clamped. The plate is twisted by applying a force of -1.0 at node B, and an equal and opposite force at node C. Gauss integration is used for the shell cross-section for the S9R5 element.

Results and discussion

The vertical displacement at node C is found as a function of the plate's aspect ratio. Since the plate's width is 1.0, the aspect ratio is equal to the length, L. *Figure 2* plots vertical displacement at C versus the length of the plate. All elements give reasonable numerical predictions compared with the reference solution; see Batoz (1982).

Input files

ese4sxsa.inp	S4 elements.
esf4sxsa.inp	S4R elements.
es54sxsa.inp	S4R5 elements.
es68sxsa.inp	S8R elements.
es58sxsa.inp	S8R5 elements.
es59sxsa.inp	S9R5 elements.
es63sxsa.inp	STRI3 elements.
es56sxsa.inp	STRI65 elements.

References

- Batoz, J. L., “An Explicit Formulation for an Efficient Triangular Plate Bending Element,” *International Journal for Numerical Methods in Engineering*, vol. 18, pp. 1077–1089, 1982.

Figures

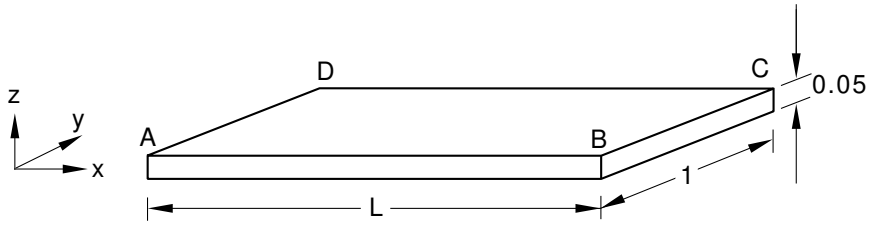


Figure 1: Model of plate.

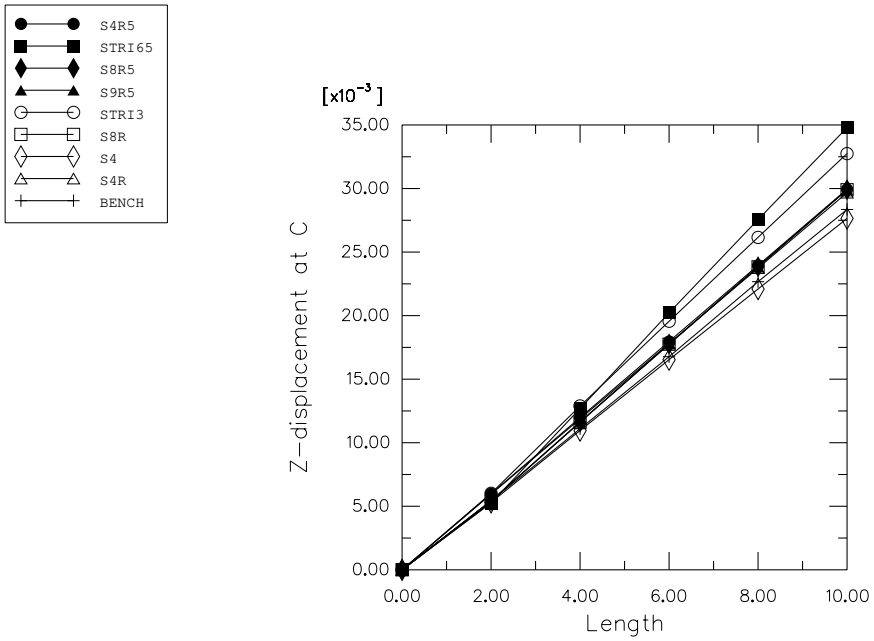


Figure 2: Comparison of results.

Ribbon test for shells with applied moments

Products: Abaqus/Standard

Overview

The problem illustrates the use of shell elements with applied moments in a plate. The results are compared with the reference solution by Batoz (1982).

Problem description

The original problem description and the benchmark solutions can be found in Batoz (1982).

The length of the plate is varied from 2.0 to 10.0 to study aspect ratio effects. The material is linear elastic with a Young's modulus of 1×10^7 and a Poisson's ratio of 0.25. Nodes along edge ***AD*** are clamped. Moments of 1.0 are applied in the *y*-direction at nodes *B* and ***C***.

Results and discussion

The vertical displacement at node *C* is found as a function of the plate's aspect ratio. Since the plate's width is 1.0, the aspect ratio is equal to the length, *L*. *Figure 2* plots vertical displacement at ***C*** versus the length of the plate. All elements give reasonable numerical predictions compared with the reference solution; see Batoz (1982).

Input files

ese4sxsb.inp	S4 elements.
esf4sxsb.inp	S4R elements.
es54sxsb.inp	S4R5 elements.
es68sxsb.inp	S8R elements.
es58sxsb.inp	S8R5 elements.
es59sxsb.inp	S9R5 elements.
es63sxsb.inp	STRI3 elements.
es56sxsb.inp	STRI65 elements.

References

- Batoz, J. L., “An Explicit Formulation for an Efficient Triangular Plate Bending Element,” *International Journal for Numerical Methods in Engineering*, vol. 18, pp. 1077–1089, 1982.

Figures

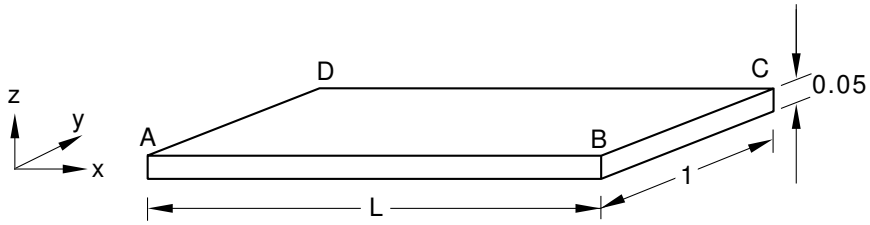


Figure 1: Model of plate.

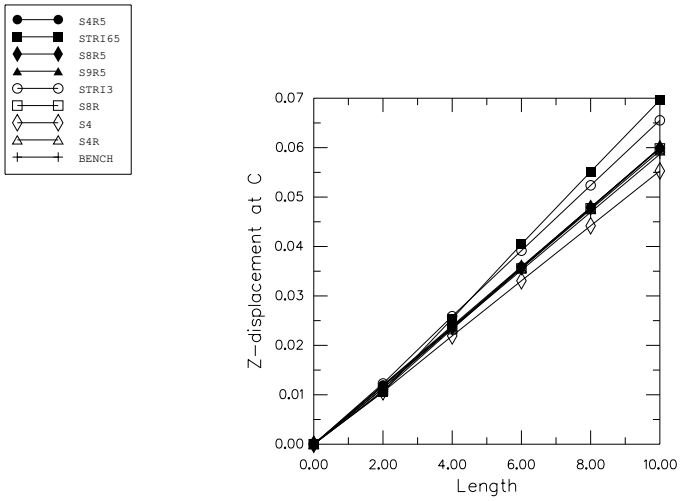


Figure 2: Comparison of results.

Triangular plate-bending on three point supports

Products: Abaqus/Standard

Overview

This problem illustrates triangular plate-bending on three point supports. The results are compared to the theoretical solution.

Problem description

Two meshes, coarse and fine, are considered. The coarse mesh is discretized with seven nodes and either six 3-node elements or three 4-node elements. The fine mesh is discretized with nineteen nodes and either twenty-four 3-node elements or twelve 4-node elements. The material is linear elastic with a Young's modulus of 207×10^9 and a Poisson's ratio of 0.25. The plate has a thickness, t, of 0.00254. $c = 0.138564$ and $l = 0.24$. There are three corner-point supports in the z-direction. A uniform distributed tangential moment of 300/length and a linear distributed twisting moment of 194.85/length are applied on each boundary.

Results and discussion

The equivalent nodal moments are calculated at nodes 1, 2, 3, 6, and 7. The vertical displacement of the centroid, node 7, is also calculated. The theoretical solution is given in *Table 1*, where the equivalent nodal moments have been calculated by applying the principle of virtual displacements with a linear function for the rotation corresponding to the tangential moment and a quadratic function for the rotation corresponding to the twisting moment. Results for the coarse meshes are given in *Table 2* to *Table 5*, and results for the fine meshes are given in *Table 6* to *Table 9*. For the mesh densities used and due to the extrapolation of integration point quantities, the nodal moments show sizable errors compared to the theoretical solution. The predicted centroidal displacements are larger than the theoretical value, approaching the theoretical values as the mesh density increases.

Input files

ese4sfsh.inp	S4 elements, fine mesh.
ese4smsh.inp	S4 elements, coarse mesh.
esf4sfsh.inp	S4R elements, fine mesh.
esf4smsh.inp	S4R elements, coarse mesh.
es54sfsh.inp	S4R5 elements, fine mesh.
es54smsh.inp	S4R5 elements, coarse mesh.
es63sfsh.inp	STRI3 elements, fine mesh.
es63smsh.inp	STRI3 elements, coarse mesh.

References

- Robinson, J., “Triangular Plate-Bending on Three Point Supports,” *Finite Element News*, no. 1, 1992.

Tables

Table 1: Theoretical solution.

NODE	M_x	M_y	M_{xy}
1	300.0	75.0	194.86
2	–37.7	412.5	0.0
3	300.0	75.0	–194.86
6	300.0	75.0	0.0
7	187.5	187.5	0.0
Centroidal displacement = 2.1226×10^{-3}			

Table 2: S4 elements, coarse mesh.

NODE	M_x	M_y	M_{xy}
1	194.7	64.38	–112.9
2	–0.7924	259.9	0.0
3	194.7	64.38	112.9
6	273.0	73.07	0.0
7	303.4	303.4	0.0
Centroidal displacement = 3.6602×10^{-3}			

Table 3: S4R elements, coarse mesh.

NODE	M_x	M_y	M_{xy}
1	243.0	132.0	–96.16
2	76.47	298.5	0.0
3	243.0	132.0	96.16
6	243.0	132.0	0.0
7	187.5	187.5	0.0
Centroidal displacement = 3.2232×10^{-3}			

Table 4: S4R5 elements, coarse mesh.

NODE	M_x	M_y	M_{xy}
1	243.6	131.4	–97.11
2	75.38	299.6	0.0
3	243.6	131.4	97.11
6	243.6	131.4	0.0
7	187.5	187.5	0.0
Centroidal displacement = 3.1924×10^{-3}			

Table 5: STRI3 elements, coarse mesh.

NODE	M_x	M_y	M_{xy}
1	101.5	251.5	−129.9
2	326.4	26.56	0.0
3	101.5	251.5	129.9
6	50.33	355.5	0.0
7	183.1	183.1	0.0
Centroidal displacement = 2.7551×10^{-3}			

Table 6: S4 elements, fine mesh.

NODE	M_x	M_y	M_{xy}
1	233.3	59.27	−151.5
2	−9.773	332.3	0.0
3	233.3	59.27	151.5
6	275.7	71.22	0.0
7	240.1	247.5	0.0
Centroidal displacement = 2.5038×10^{-3}			

Table 7: S4R elements, fine mesh.

NODE	M_x	M_y	M_{xy}
1	260.9	102.0	−139.4
2	19.36	352.0	0.0
3	260.9	102.0	139.4
6	273.7	108.2	0.0
7	184.7	191.2	0.0
Centroidal displacement = 2.4042×10^{-3}			

Table 8: S4R5 elements, fine mesh.

NODE	M_x	M_y	M_{xy}
1	261.3	101.1	−140.6
2	18.91	353.9	0.0
3	261.3	101.1	140.6
6	273.6	108.7	0.0
7	184.6	191.3	0.0
Centroidal displacement = 2.4022×10^{-3}			

Table 9: STRI3 elements, fine mesh.

NODE	M_x	M_y	M_{xy}
1	83.59	272.1	−160.3
2	370.3	−8.347	0.0
3	83.59	272.1	160.3
6	62.72	333.9	0.0
7	183.6	187.4	0.0
Centroidal displacement = 2.3259×10^{-3}			

Figures

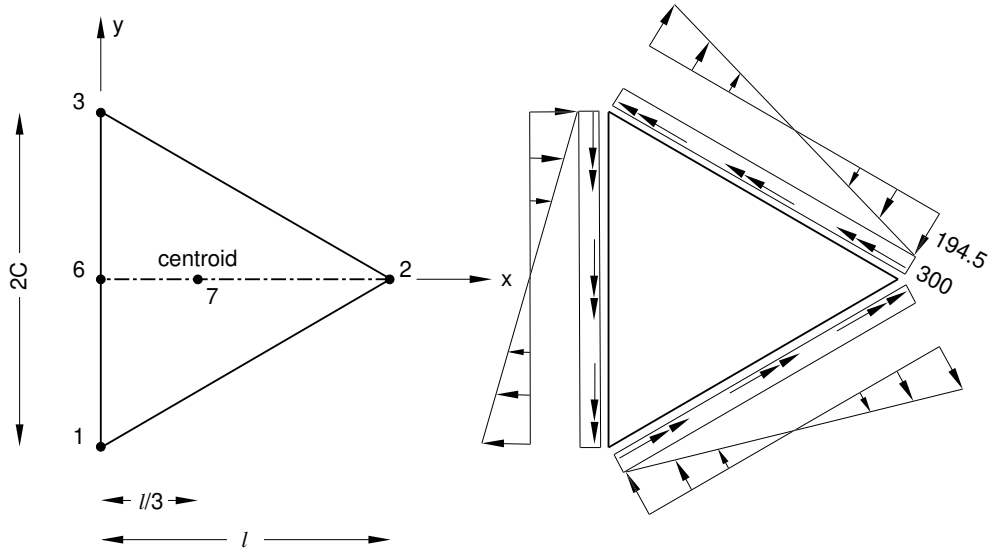


Figure 1: Model of triangular plate with applied moments.

Shell elements subjected to uniform thermal loading

Products: Abaqus/Standard

Overview

This examples illustrates the free thermal expansion of shell elements. The results are compared to the analytical solution.

Problem description

A one-eighth symmetrical segment of a spherical surface with a cutout is modeled as a shell section. Symmetry boundary conditions are applied appropriately to the section's edges. The structure has a thickness of 0.4 inch and a radius of 100.0 inches and is subjected to a uniform temperature change. The discretization of this surface yields the following meshes: one consisting of 64 quad elements for use with S4, S4R, S4R5, S8R, and SC8R elements and the other consisting of 128 triangular elements for use with STRI3, STRI65, S3/S3R, and SC6R elements. The shell section is heated uniformly from 0° F to 430° F.

The first step of each input file is a static linear perturbation step. In the second step the loading is repeated as a general step that accounts for geometric nonlinearities. Since the deformations are small, the displacements obtained in the second step are virtually the same as the displacements obtained in the linear perturbation step. In some of the input files, solution controls are used to relax the equilibrium tolerances somewhat. This is necessary because the nodal forces in the final solution are zero; hence, the maximum force and moment residuals are (almost) of the same order of magnitude as the force and moment norms.

Material properties

The material is isotropic with the following constants:

Young's modulus, $E = 68.25 \times 10^6$ psi

Poisson's ratio, $\nu = 0.3$

Thermal expansion coefficient, $\alpha = 1.0 \times 10^{-6}$ in/in°F

Analytical solution

The analytical solution to this problem is uniform radial expansion with zero stress.

Results and discussion

In establishing a reasonable numerical result for the shell section, acceptable stress output may be values that are at least five or six orders of magnitude smaller than those of a completely constrained shell section with identical geometry. If the modeled section were completely constrained, the temperature change would provide a uniform compressive stress calculated as $\sigma_c = E' \alpha \Delta\theta$, where $E' = E/(1-\nu)$ and $\Delta\theta$ is the temperature change. Using the material properties above and these relations, a fully constrained shell model should produce compressive stresses, σ_c, equal to 42,000 psi. Thus, an acceptable numerical solution would be less than approximately 0.1 psi.

All elements provide principal stresses that are well below this value for both the linear static analysis step (Step 1) and the geometrically nonlinear step (Step 2). In the linear static step all elements tested, with the exception

of S4R5, provide maximum principal stress magnitudes that are $O(10^{-8})$ psi or smaller. The S4R5 element's maximum principal stress magnitude is $O(10^{-3})$ psi. The problem becomes more challenging when nonlinear geometry (NLGEOM) is included. In these tests this effect is reflected by the higher stress magnitudes of some of the geometrically nonlinear results of Step 2. Elements S3/S3R, STRI3, and S4R produce maximum principal stress magnitudes that are $O(10^{-7})$ psi or smaller. Element S4 produces maximum principal stress magnitudes of $O(10^{-6})$ psi. The principal stress magnitudes of STRI65, S4R5, and S8R elements—all $O(10^{-2})$ psi—are considerable higher. However, even these relatively high stresses are considered to be very reasonable.

Input files

esf3sxf1.inp	S3/S3R elements.
ese4sxf1.inp	S4 elements.
esf4sxf1.inp	S4R elements.
es54sxf1.inp	S4R5 elements.
esf8sxf1.inp	S8R elements.
es63sxf1.inp	STRI3 elements.
es56sxf1.inp	STRI65 elements.
esc6sxf1.inp	SC6R elements.
esc8sxf1.inp	SC8R elements.

Files that include the OFFSET parameter:

esf3sxf2.inp	S3/S3R elements.
ese4sxf2.inp	S4 elements.
esf4sxf2.inp	S4R elements.
es54sxf2.inp	S4R5 elements.
es63sxf2.inp	STRI3 elements.
es56sxf2.inp	STRI65 elements.

Figures

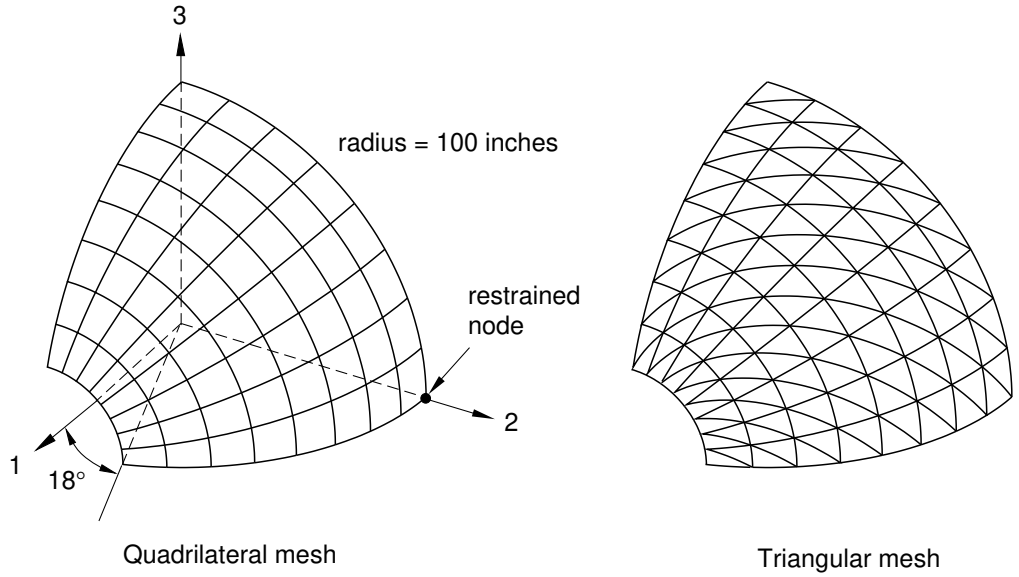


Figure 1: Meshes used in this analysis.

Shell bending under a tip load

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example illustrates the use of shell elements to model a beam bending as a result of a tip load.

Problem description

Two beams are analyzed in the same run, the first (BEAM1) with a follower load and the second (BEAM2) with a constant-direction load (a nonfollower load).

Each beam is 400 mm long in the *x*-direction, with a 20-mm-square cross-section. There are 40 elements along the length. Thus, each beam has 40 S4R elements and 82 nodes. All nodes are constrained in degrees of freedom 3–5, and the ends of both beams are constrained in degrees of freedom 1–6.

The material for this problem is elastic, with a constant Young's modulus of 1000 MPa and a Poisson's ratio of 0. The density is 10000 kg/m^3.

Both beams are loaded with a general edge traction of 12500 N/m in the negative *y*-direction at each end node. The load on the first beam is a follower force, while the load on the second beam does not change direction as the beam deforms.

Results and discussion

Figure 1 displays the Abaqus/Explicit contours of the axial membrane section force, SF1, for both beams loaded by the distributed force. In Beam 1 the section force is zero at the tip, increasing to maximum compression at the base. In Beam 2 the section force is zero at the base, increasing to maximum tension at the tip. *Figure 2* shows a time history of the tip deflection of both beams.

The predicted vertical steady-state displacement at the tip of the first beam (with a follower load) compares well with the Abaqus/Standard result of 291 mm.

The predicted vertical steady-state displacement at the tip of the second beam (with a constant-direction load) compares well with the analytical result of 240 mm given by Bisshopp and Drucker (1945) and the Abaqus/Standard result of 242 mm.

Input files

shellfoll_xpl_edld.inp	Abaqus/Explicit analysis of two beams, one with a follower edge traction and the other with a nonfollower edge traction.
shellfoll_std_edld.inp	Abaqus/Standard analysis of two beams, one with a follower edge traction and the other with a nonfollower edge traction.

References

- Bisshopp, K. E., and D. C. Drucker, “Large Deflections of Cantilever Beams,” *Quarterly of Applied Mathematics*, vol. 3 272, 1945.

Figures

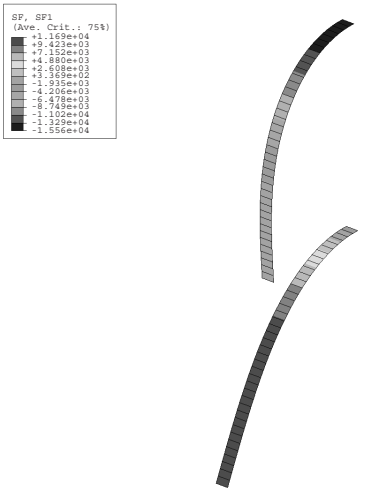


Figure 1: Contours of axial section force.

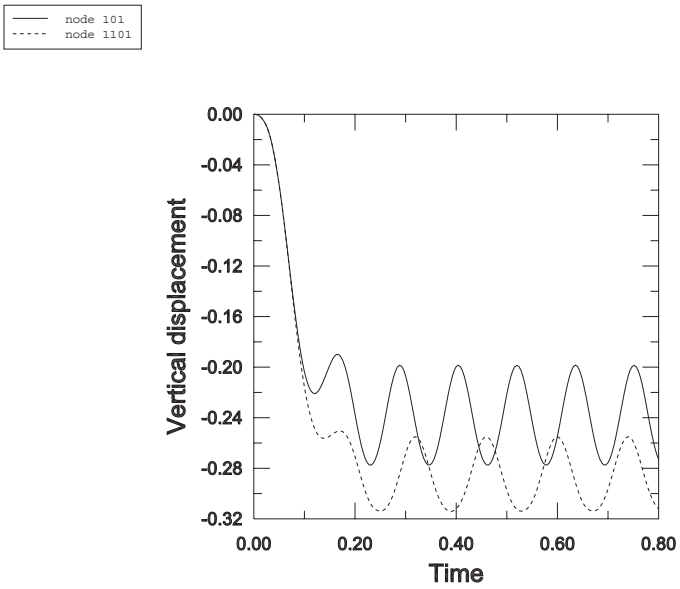


Figure 2: Tip deflection history.

Variable thickness shells and membranes

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example illustrates the use of shell and membrane elements in models defined with variable thickness.

Problem description

For the general shell, membrane, and continuum shell elements the model consists of a tapered plate of length 100 and width 20, as shown in *Figure 1*. The plate is clamped at one end, and the thickness varies linearly across the plate from 3 at the clamped end to 1 at the free end. The model consists of 10 elements along the length and 2 across the width. For the model using continuum shell elements the thickness is defined by the nodal geometry.

For the axisymmetric elements the model consists of a tapered cylinder, with a radius of 1×10^6 and a length of 100, as shown in *Figure 2*. The cylinder is clamped at one end, and the thickness varies linearly along the length of the cylinder from 3 at the clamped end to 1 at the free end. The radius is chosen to be very large to ensure that the effects of circumferential stresses are negligible. The cylinder is meshed with 10 elements.

A linear elastic material with a Young's modulus of 10×10^9, a Poisson's ratio of 0, and a density of 8000 is used for all the tests. The thicker ends of the plate and cylinder are fully clamped. The shell elements are loaded with a bending moment of 3 per unit length at the thin end of the shell, and the membrane elements are loaded with an in-plane force of 50 per unit length at the thin end of the membrane.

Results and discussion

The Abaqus/Standard results for shell and continuum shell elements are shown in *Table 1*. The Abaqus/Explicit results for shell and continuum shell elements are shown in *Table 2* and for membrane elements in *Table 3*. The differences in the Abaqus/Explicit results are due to dynamic effects and mesh discretization. For the shell elements in Abaqus/Explicit the results using a shell offset with a value of 0.5 are also shown in *Table 2*.

Input files

Abaqus/Standard input files

varthick_std_s3r.inp	S3R element using **SHELL SECTION* to define section properties.
varthick_std_s4r.inp	S4R element using **SHELL SECTION* to define section properties.
varthick_std_s4r_edmom.inp	S4R element using **SHELL SECTION* to define section properties. Bending moment applied with a distributed edge moment.
varthick_std_sc6r.inp	SC6R element using **SHELL SECTION* to define section properties.
varthick_std_sc6r_sgs.inp	SC6R element using **SHELL GENERAL SECTION* to define section properties.
varthick_std_sc8r.inp	SC8R element using **SHELL SECTION* to define section properties.
varthick_std_sc8r_sgs.inp	SC8R element using **SHELL GENERAL SECTION* to define section properties.
varthick_std_sc8r_eh.inp	SC8R element with enhanced hourglass control using **SHELL SECTION* to define section properties.

varthick_std_sc8r_sgs_eh.inp	SC8R element with enhanced hourglass control using **SHELL GENERAL SECTION* to define section properties.
varthick_std_css8.inp	CSS8 element.

Abaqus/Explicit input files

varthick_s3r.inp	S3R element.
varthick_s3r_offset.inp	S3R element and a shell offset.
varthick_s4r.inp	S4R element.
varthick_s4r_offset.inp	S4R element and a shell offset.
varthick_sc6r.inp	SC6R element.
varthick_sc8r.inp	SC8R element.
varthick_sc6r_sgs.inp	SC6R element using **SHELL GENERAL SECTION* to define section properties.
varthick_sc8r_sgs.inp	SC8R element using **SHELL GENERAL SECTION* to define section properties.
varthick_sax1.inp	SAX1 element.
varthick_sax1_offset.inp	SAX1 element and a shell offset.
varthick_m3d3.inp	M3D3 element.
varthick_m3d4r.inp	M3D4R element.
varthick_m3d4r_hgc.inp	COMBINED hourglass option for membrane elements included for the sole purpose of testing the performance of the code.
varthick_m3d4r_hgv.inp	VISCOUS hourglass option for membrane elements included for the sole purpose of testing the performance of the code.

Tables

Table 1: Abaqus/Standard shell bending results.

Element Type	Tip Displacement (u_z)	Tip Rotation
EXACT	2.00×10^{-6}	8.00×10^{-8}
S3R	2.06×10^{-6}	8.09×10^{-8}
S4R	2.02×10^{-6}	7.91×10^{-8}
SC6R	1.93×10^{-6}	
SC6R (sgs)	1.93×10^{-6}	
SC8R	2.02×10^{-6}	
SC8R (sgs)	2.02×10^{-6}	
SC8R*	2.02×10^{-6}	
SC8R* (sgs)	2.02×10^{-6}	
CSS8	2.00×10^{-6}	
*Abaqus/Standard results with enhanced hourglass control.		

Table 2: Abaqus/Explicit shell bending results.

Element Type	Offset Value	Tip Displacement (u_z)	Tip Rotation
EXACT		2.00×10^{-6}	8.00×10^{-8}
SAX1	0	2.26×10^{-6}	8.33×10^{-8}
SAX1	0.5	2.23×10^{-6}	8.26×10^{-8}
S3R	0	2.26×10^{-6}	8.50×10^{-8}
S3R	0.5	2.23×10^{-6}	8.38×10^{-8}
S4R	0	2.26×10^{-6}	8.33×10^{-8}
S4R	0.5	2.24×10^{-6}	8.23×10^{-8}
SC6R		1.93×10^{-6}	
SC8R		2.01×10^{-6}	
SC6R (sgs)		1.93×10^{-6}	
SC8R (sgs)		2.01×10^{-6}	

Table 3: Abaqus/Explicit membrane element results.

Element Type	Tip Displacement (u_y)
EXACT	2.746×10^{-7}
M3D3	2.751×10^{-7}
M3D4R	2.742×10^{-7}

Figures

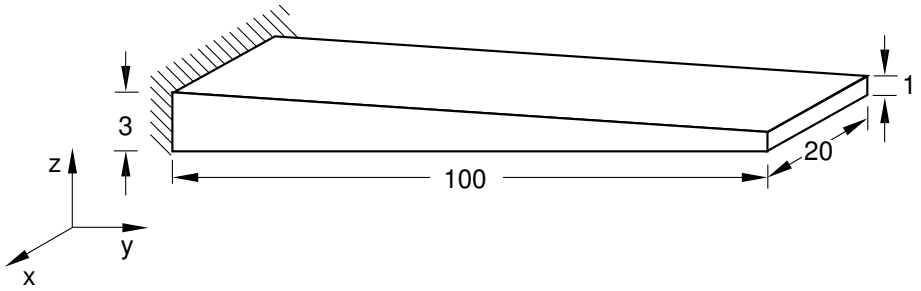


Figure 1: Tapered plate for general elements.

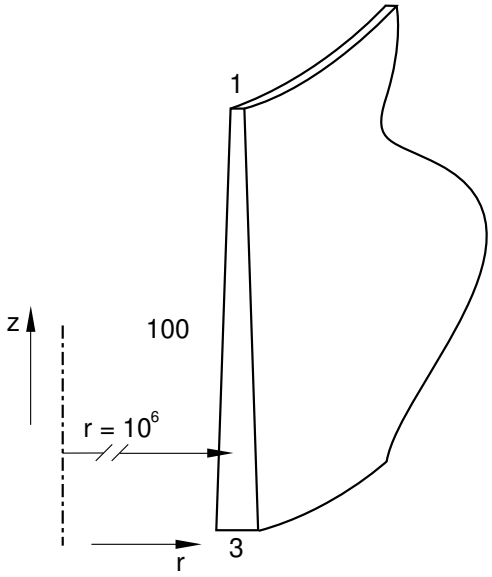


Figure 2: Tapered cylinder.

Transient response of a shallow spherical cap

Products: Abaqus/Explicit

Overview

This example illustrates the transient response of a shallow spherical cap subjected to uniform pressure.

Problem description

The spherical cap has a radius of 22.27 in. and a thickness of 0.41 in. The response of the spherical cap (shown in *Figure 1*) is dominated by bending. Both an axisymmetric and a three-dimensional analysis are performed. The three-dimensional model consists of a quadrant modeled with S4R or S4RS elements with appropriate symmetry boundary conditions (see *Figure 2*).

The material is modeled as an elastic-plastic material with the following properties:

Young's modulus = 10.5×10^6 psi

Poisson's ratio = 0.3

Density = 2.45×10^{-4} lb-sec^2/in^4

Initial yield stress = 240000 psi

Hardening modulus = 0.21×10^6 psi

A uniform pressure of 600 psi is applied over the shell as a step function in time. Three meshes are used for each geometry. The three-dimensional analysis is performed using 75, 147, and 243 S4R and S4RS elements; and the axisymmetric analysis is performed using 20, 40, and 60 SAX1 elements.

Results and discussion

Figure 3 and *Figure 4* show the time histories of the center displacement of the spherical cap predicted by the three-dimensional S4R and S4RS models, respectively. *Figure 5* shows some predictions from the axisymmetric models. *Figure 6* shows a comparison of the time histories obtained with the finest axisymmetric mesh and the finest three-dimensional mesh. *Figure 7* shows a comparison of the time history of the kinetic energy obtained with the finest axisymmetric and three-dimensional meshes.

The results indicate that the SAX1 element, the S4R element, and the S4RS element converge for this problem. They compare well with the existing solutions in the literature (see Bathe et al., 1975, and Belytschko et al., 1984).

Input files

sphr_axa_fine.inp	Axisymmetric analysis with the fine mesh.
sphr_axa_med.inp	Axisymmetric analysis with the medium mesh.
sphr_axa_coarse.inp	Axisymmetric analysis with the coarse mesh.
sphr_coarse.inp	Three-dimensional analysis with the coarse mesh using S4R elements.
sphr_med.inp	Three-dimensional analysis with the medium mesh using S4R elements.
sphr_fine.inp	Three-dimensional analysis with the fine mesh using S4R elements.
sphr_coarse_s4rs.inp	Three-dimensional analysis with the coarse mesh using S4RS elements.
sphr_med_s4rs.inp	Three-dimensional analysis with the medium mesh using S4RS elements.

sphr_fine_s4rs.inp Three-dimensional analysis with the fine mesh using S4RS elements.

References

- Bathe, K. J., et al.., "Finite Element Formulations for Large Deformation Dynamic Analysis," *International Journal for Numerical Methods in Engineering*, vol. 9, pp. 353–386, 1975.
- Belytschko, T. B., et al.., "Explicit Algorithms for the Nonlinear Dynamics of Shells," *Computational Methods in Applied Mechanics and Engineering*, vol. 42, pp. 225–251, 1984.

Figures

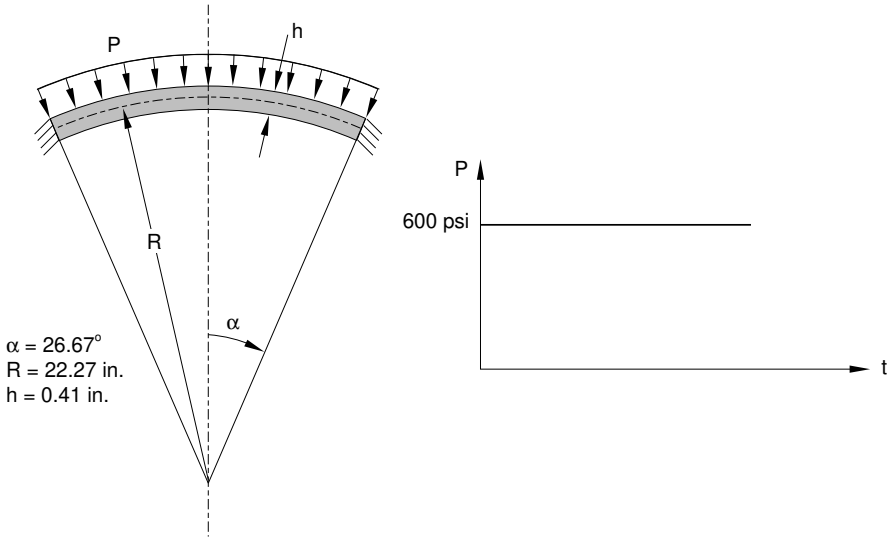


Figure 1: Geometric characteristics of the spherical cap.

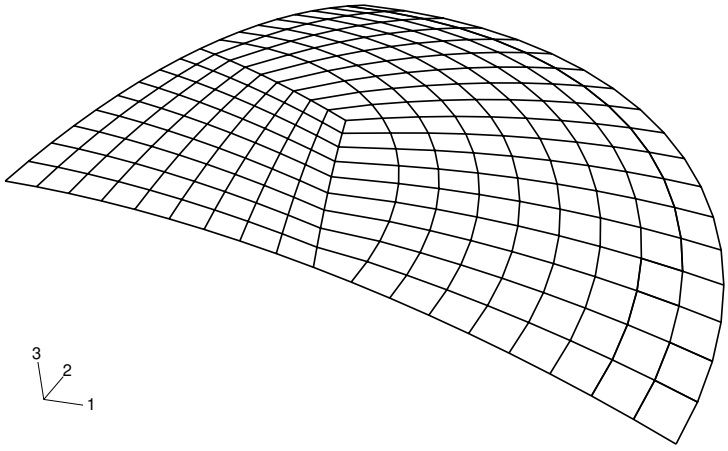


Figure 2: Finest mesh used in the three-dimensional analysis.

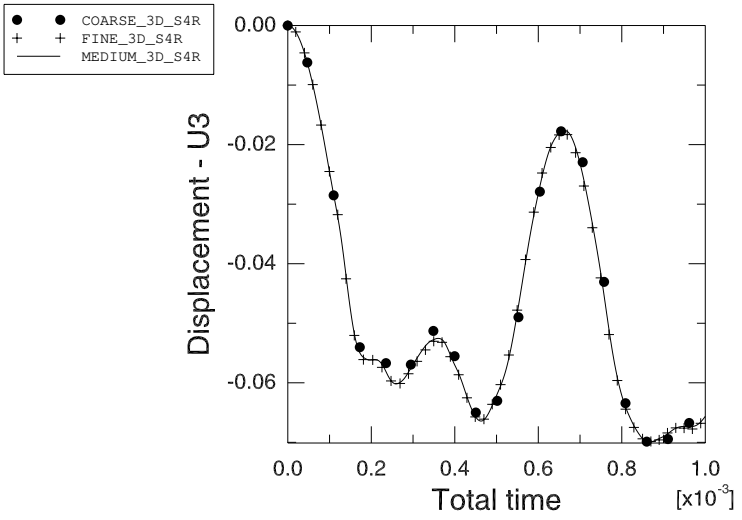


Figure 3: Convergence of the center displacement of the spherical cap using S4R elements.

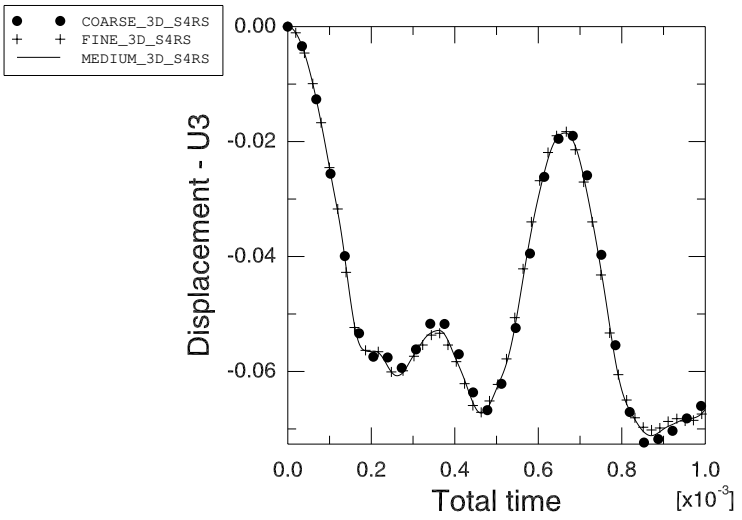


Figure 4: Convergence of the center displacement of the spherical cap using S4RS elements.

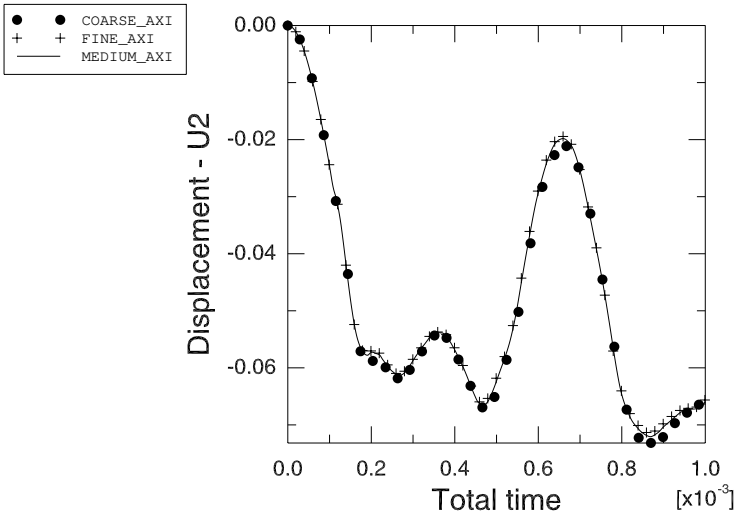


Figure 5: Convergence of the center displacement of the spherical cap using SAX1 elements.

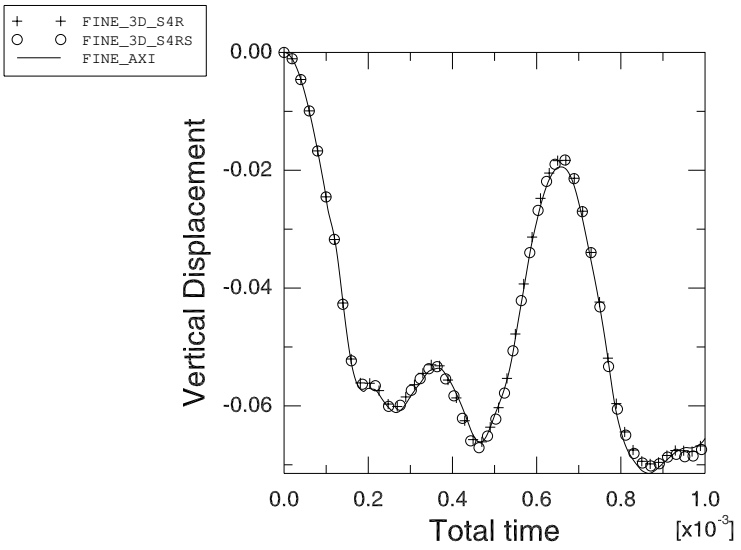


Figure 6: Comparison of the time history of the center displacement of the spherical cap for the fine S4R, the fine S4RS, and the fine SAX1 meshes.

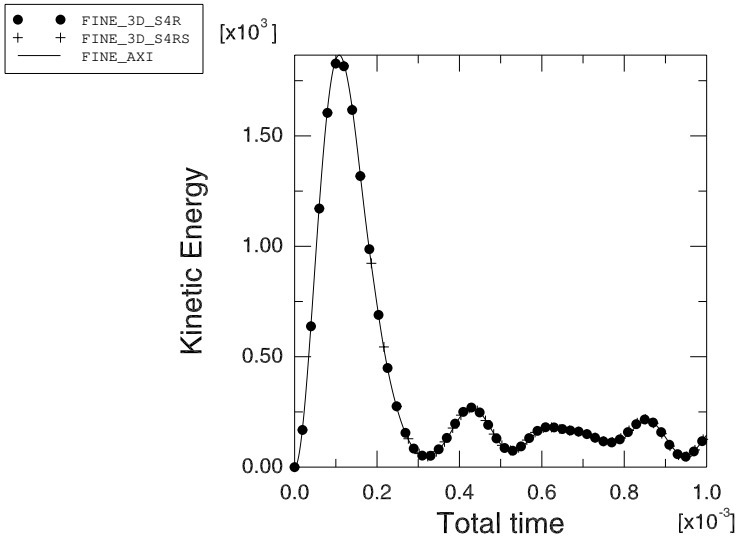


Figure 7: Comparison of the time history of the kinetic energy of the spherical cap for the fine S4R, the fine S4RS, and the fine SAX1 meshes.

Simulation of propeller rotation

Products: Abaqus/Explicit

Overview

This example illustrates the performance of shell and solid elements in Abaqus/Explicit.
The example includes a large number of rigid body rotations.

Problem description

The rotation of a propeller that has an initial spin rate of 314.16 rad/s (3000 rpm or 50 revolutions per second) is simulated for a duration of 2 seconds. The propeller consists of a rigid central hub and two deformable blades, as shown in *Figure 1*. The hollow cylindrical hub is 5 inches (0.1270 m) long and has inner and outer radii of 4 inches (0.1016 m) and 5 inches (0.1270 m), respectively. The blades are 0.5 inches (0.0127 m) thick and span 50 inches (1.27 m). The blades make a 20° angle with the central axis of the propeller. The entire propeller assembly is made of steel with the following properties:

Young's modulus = 30.0×10^6 psi (207 GPa)

Poisson's ratio = 0.3

Density = 7.3×10^{-4} lbf-sec^2/in^4 (7800 kg/m^3)

The propeller is assumed to be spinning in a vacuum with zero initial stresses. Once the stress state due to the centrifugal loading is established, the solution (displacements, stresses, and strains) is expected to oscillate about this state with time. The mass moment of inertia of the propeller about its central axis increases slightly due to the stretching of the blades in the mean stress state under the centrifugal loading. Since the angular momentum is conserved, the increase in the mass moment of inertia is expected to result in a mean spin rate that is slightly less than the initial spin rate.

Meshing of the blades and the hub

In this study the hub region of the propeller is modeled as a rigid body that is discretized with C3D8R elements. The propeller blades are discretized with either S4R, S4RS, or C3D8R elements. Each propeller blade is discretized with a single element type, although two different element types can be used for the two blades. This leads to six different model permutations, all of which are studied in this example.

Order of accuracy in element formulation

Since the finite elements undergo a large number (> 5) of rigid body revolutions, the second-order accurate element formulation is specified in the section controls. The six models of the propeller problem with different element combinations are analyzed using second-order accurate elements. In addition, the model with both blades discretized using S4R elements is also analyzed using the default first-order accurate element formulation option.

Results and discussion

All the analyses in this problem are performed using double precision floating point accuracy since the number of time increments to complete the simulation for a duration of 2 seconds is quite large (about 2.3 million increments with C3D8R elements in any of the two blades and about 1.3 million increments with shell elements in the both blades).

A representative deformed configuration plot at the end of the 2 second time period is shown in *Figure 2* for the model where S4R elements are used to discretize one blade and C3D8R elements are used for the other blade. The deformed configuration in all seven cases considered is free from element distortions, and in each case the propeller has undergone about 99.5 revolutions by the end of the 2 second time period. Correspondingly, a mean rotational velocity of about 312 rad/s is established with time as the mean stress state is reached from the initial stress-free state (see *Figure 3*).

Figure 4 shows the total model energy (ETOTAL) and the total model kinetic energy (ALLKE) for the seven cases considered here. As ETOTAL remains constant, the energy balance is clearly maintained throughout the analysis. The kinetic energy is also fairly constant with time and is only about 0.4% lower than the initial kinetic energy of the model. This difference of 0.4% is explained by the internal energy of the elements. In general, the element elastic (ALLSE), artificial (ALLAE), and viscous (ALLVD) energies are found to be insignificant compared to the model kinetic energy (ALLKE).

Summary

In summary, based upon this example:

1. For problems involving a large number of rigid body rotations, the second-order accurate element formulation captures the behavior without any element distortions and is highly recommended for problems with elements undergoing more than 5 revolutions.
2. For the propeller model with both blades discretized using S4R elements, the first-order accurate formulation option yields results that compare well with the results from the second-order accurate formulation option.

Input files

Second-order accurate element formulation:

propeller_c3d8r_c3d8r.inp	Both blades made of C3D8R elements.
propeller_s4r_s4r.inp	Both blades made of S4R elements.
propeller_s4rs_s4rs.inp	Both blades made of S4RS elements.
propeller_s4r_c3d8r.inp	One blade made of S4R elements and the other of C3D8R elements.
propeller_s4r_s4rs.inp	One blade made of S4R elements and the other of S4RS elements.
propeller_s4rs_c3d8r.inp	One blade made of S4RS elements and the other of C3D8R elements.

First-order accurate element formulation:

propeller_s4r_s4r_1storder.inp	Both blades made of S4R elements.

Figures

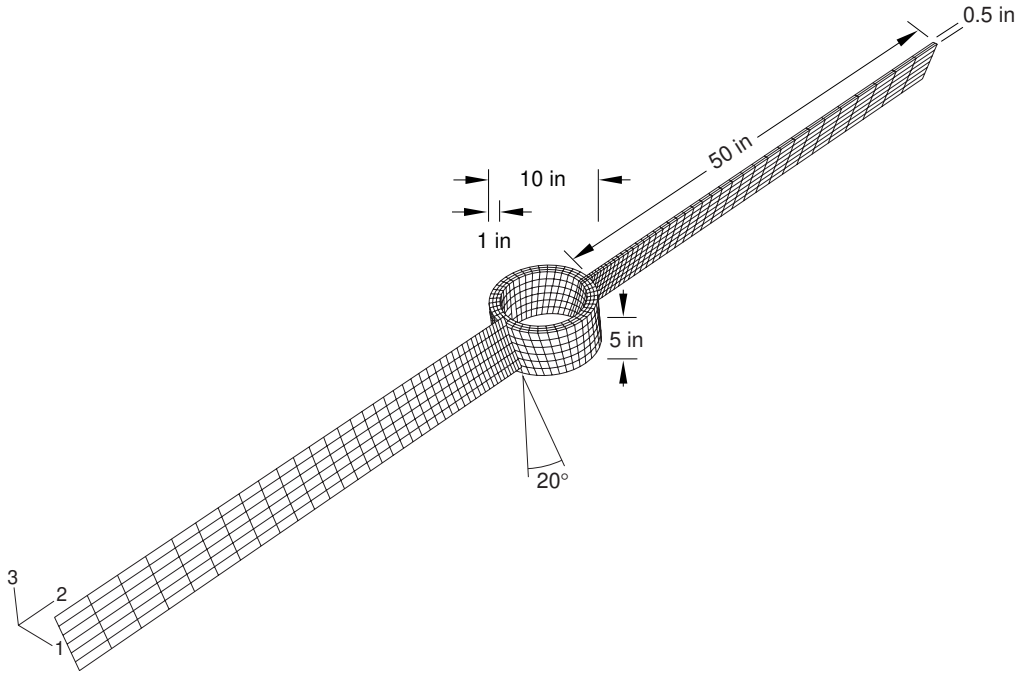


Figure 1: Propeller problem.

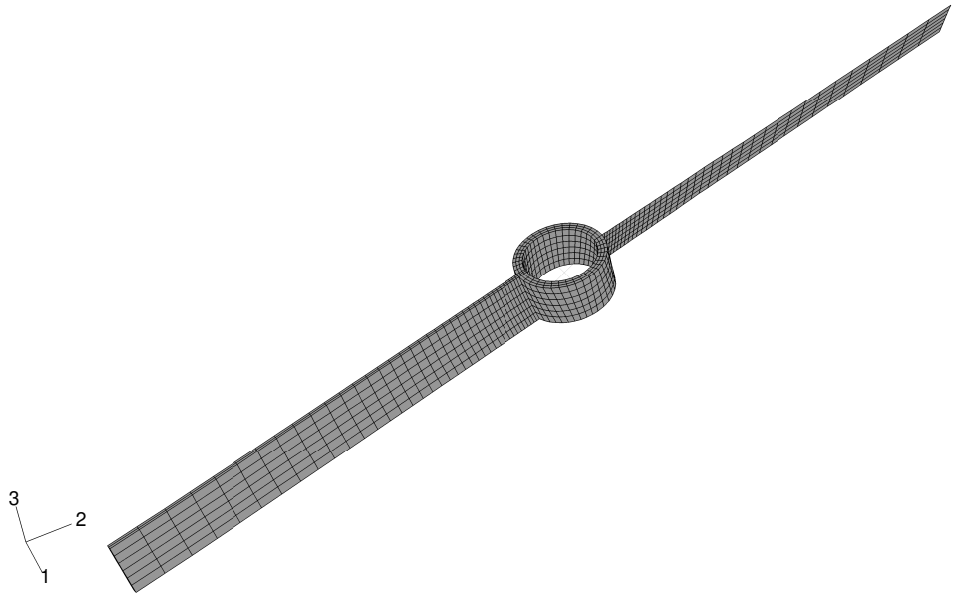


Figure 2: Deformed plot of propeller discretized with C3D8R elements in one blade and S4R elements in the other after about 99.5 revolutions.

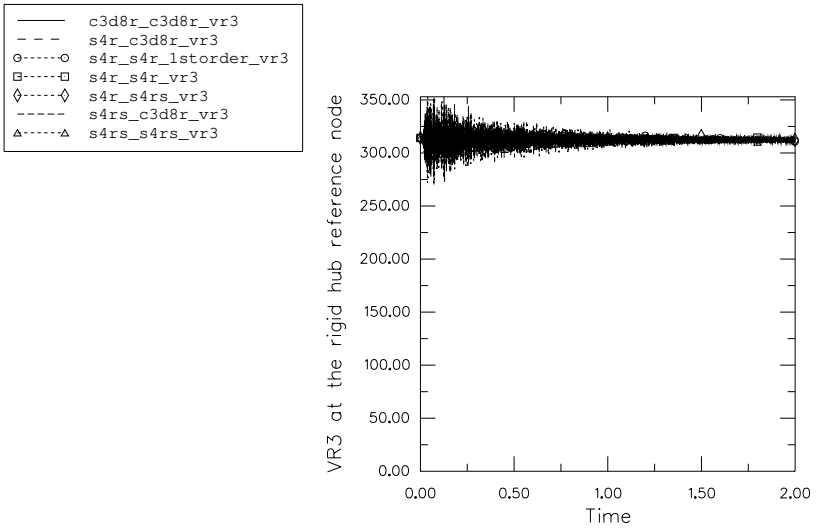


Figure 3: Rotational velocity of the rigid hub about its axis.

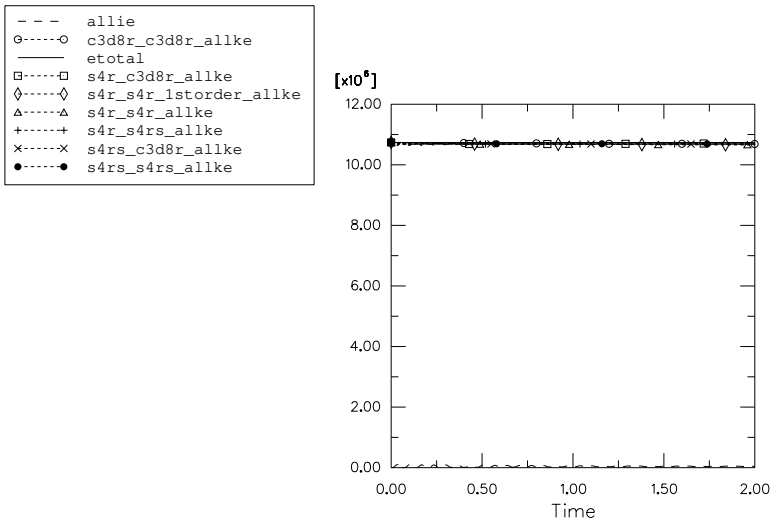


Figure 4: Whole model energy history in different models of the propeller.

Connector elements

In this section:

- *Dynamic response of a two degree of freedom system*
- *Linear behavior of spring and dashpot elements*

Dynamic response of a two degree of freedom system

Products: Abaqus/Standard

Overview

This example verifies the use of nonlinear springs and direct, implicit, dynamic integration in a simple example for which an independent solution is available (Underwood and Park, 1981).

Problem description

The system consists of two nonlinear springs, each connecting a mass to a fixed point, with a linear spring and a dashpot between the masses. The system is shown in *Figure 1*. The spring characteristics, the initial conditions, and the forcing functions are also shown in the figure. All values are assumed to be in consistent units.

For direct comparison with the solution of Underwood and Park (1981), the analysis is run with fixed time increments. In Underwood and Park (1981) a time increment of 0.0005 is shown to be very accurate with the central difference (explicit) integration operator, while a time increment of 0.03 is less accurate. In this study the time increment chosen is 0.01. This gives results that agree closely with those reported by Underwood and Park (1981).

Results and discussion

The displacement and velocity histories of the two masses are shown in *Figure 2* and *Figure 3*. The results obtained by Underwood and Park (1981) are shown in the same figures. The agreement is quite close.

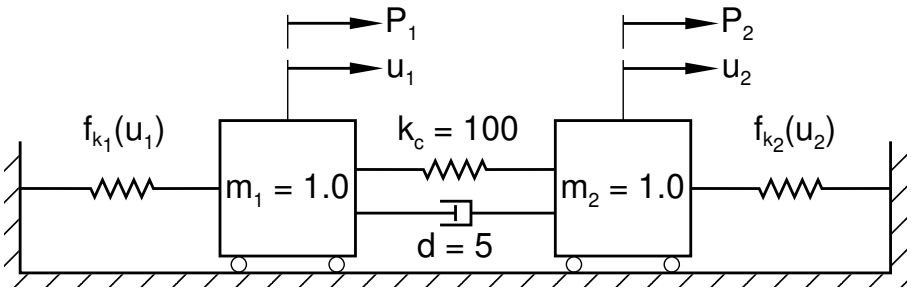
Input files

2dofdynamics.inp	Dynamic analysis.
2dofdynamics_depend.inp	Identical to the input data shown in 2dofdynamics.inp, except that temperature- and field-variable-dependent spring and dashpot properties are used.

References

- Underwood, P., and K. C. Park, "STIND/CD: A Stand-Alone Explicit Time Integration Package for Structural Dynamic Analysis," *International Journal for Numerical Methods in Engineering*, vol. 17, pp. 1285–1312, 1981.

Figures



$\dot{u}_1(0) = 100.0$ $\quad u_1(0) = u_2(0) = 0.0$

$\dot{u}_2(0) = 0.0$

$P_1(t) = 0.0$ $\quad f_{k_1}(u_1) = 1000 \tanh u_1$

$P_2(t) = 0.0 \quad t < 0.5;\ t > 0.55 \quad f_{k_2}(u_2) = 1000 \sinh u_2$

$P_2(t) = 3000.0 \quad 0.5 \leq t \geq 0.55$

Figure 1: Two degree of freedom nonlinear spring-mass system.

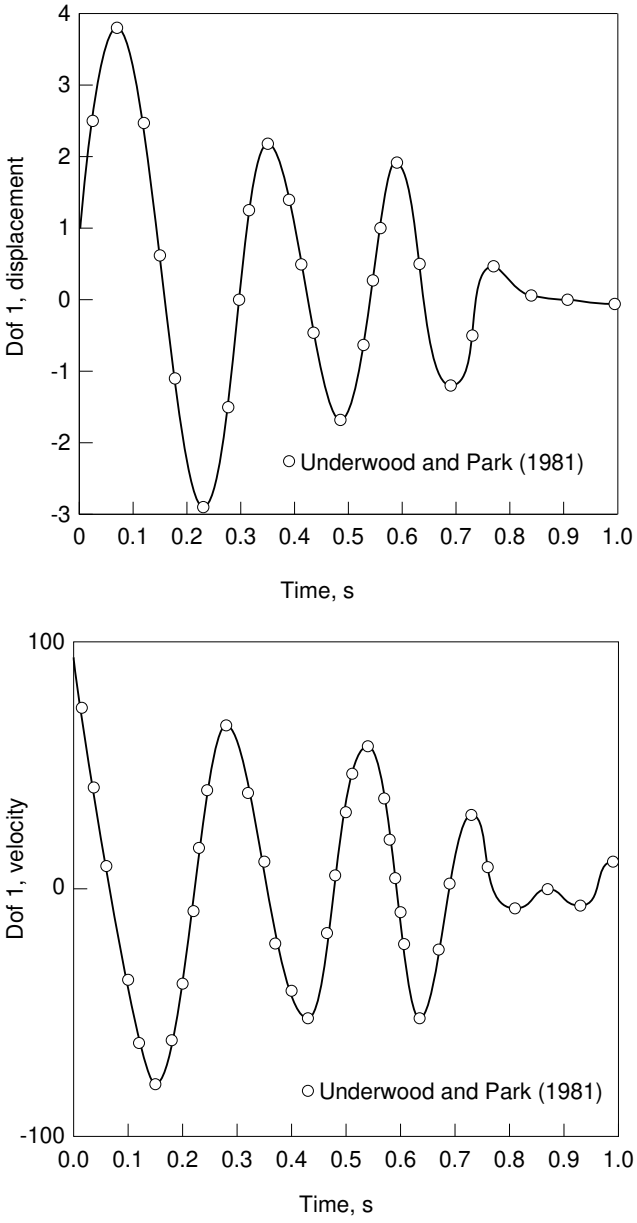


Figure 2: Displacement and velocity histories of left-hand mass.

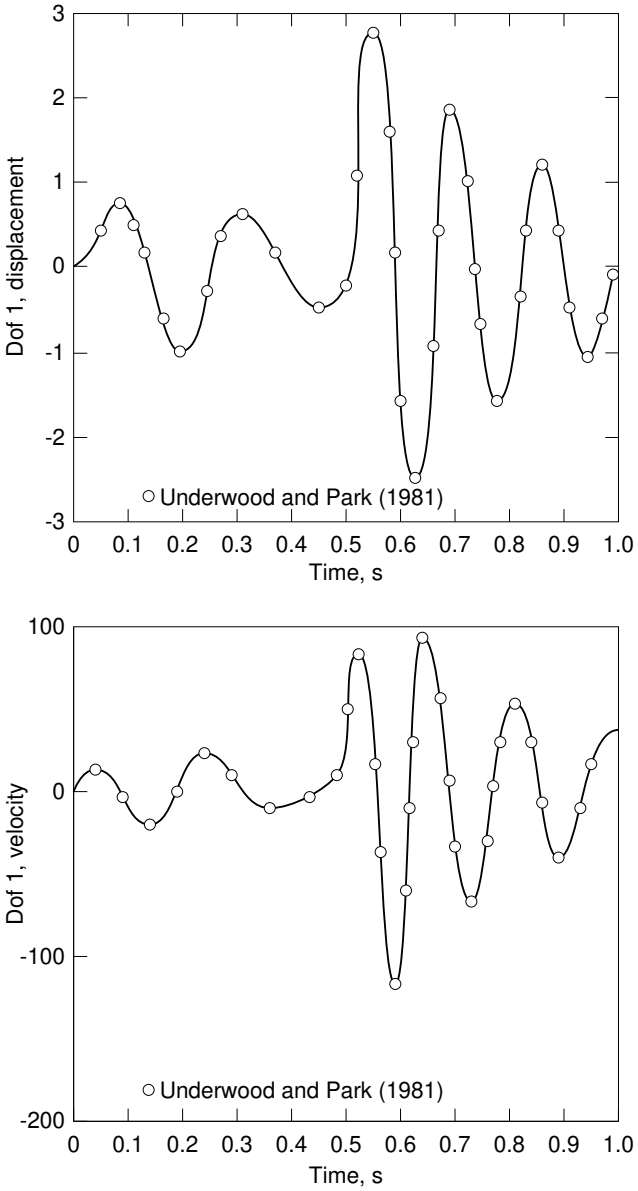


Figure 3: Displacement and velocity histories of right-hand mass.

Linear behavior of spring and dashpot elements

Products: Abaqus/Standard

Overview

This example illustrates the linear behavior of three independent spring/dashpot/mass systems for which analytical solutions are available.

Problem description

Each of the independent spring/dashpot/mass systems (A, B, and C, as shown in *Figure 1*) has a common component: a mass attached to a parallel spring/dashpot system. In system A the MASS element is attached to SPRING1 and DASHPOT1 elements, so the system is grounded. Instead of directly grounding the parallel spring/dashpot systems, systems B and C add another MASS element attached to ground by means of a second spring. Element types SPRING2 and DASHPOT2 are tested in system B, while system C tests element types SPRINGA and DASHPOTA. For SPRINGA and DASHPOTA elements, the direction of action is the line joining the two nodes. This behavior is tested by orienting system C such that the spring/dashpot line of action lies 45° counterclockwise from the horizontal, thereby undergoing motion along both the global 1- and 2-directions. All the results for system C are reported in the local coordinate system.

The mass, spring, and dashpot constants are the same for all three systems: mass = 0.02588, spring constant = 30.0, dashpot coefficient = 0.12.

Grounded tests

A two-step analysis is performed on each of the three systems. In the first step the mass at the free end is displaced one unit, while in systems B and C the other mass is constrained to simulate the grounded condition of system A. The end mass is then free to displace in the second step. The results of this test should be exactly the same for all three systems.

Boundary conditions:

Step 1: $u_y = 0$ at all nodes. $u_x = 0$ at nodes 11, 12, 101, and 102.

$u_x = 1$ at nodes 3, 13, and 103.

Step 2: $u_x = u_y = 0$ at nodes 11, 12, 101, and 102.

Node-to-node tests

A two-step analysis is performed on systems B and C. As in test *exsd3glx.inp*, the mass at the free end is displaced one unit during the first step, while the other mass is fixed. After stretching, however, both masses are free to move. The movement is constrained to be along the 1-direction so that systems B and C can be compared. The results of this test should be exactly the same for these two systems.

Boundary conditions:

Step 1: $u_y = 0$ at all nodes. $u_x = 0$ at nodes 11, 12, 101, and 102.

$u_x = 1$ at nodes 3, 13, and 103.

Step 2: $u_x = u_y = 0$ at nodes 11 and 101.

Analytical solutions

The analytical solutions for each of the analyses are presented here.

Grounded tests

Force balance on the system yields the second-order linear differential equation for a single degree of freedom damped oscillator. Let x be the position of the mass node along the x-axis. We obtain the equation

$$m\ddot{x} + c\dot{x} + kx = 0,$$

which is solved subject to $x(0) = 1$ and $\dot{x}(0) = 0$ to yield

$$x(t) = e^{-\xi\omega_n t}\left(A\sin\omega_d t + B\cos\omega_d t\right),$$

where the constants have the following values:

$$\xi = 0.0681, \omega_n = 34.047, \omega_d = 33.9678$$
$$A = 0.0, B = 1.0.$$

Node-to-node tests

The analytical solution to this problem is obtained by writing out the force balance equation for both masses. This yields a coupled system of two ordinary differential equations. Let x_1 and x_2 refer to the position of the middle and end mass, respectively. The following equations are obtained:

$$m\ddot{x}_1 + k(2x_1 - x_2) + c(\dot{x}_1 - \dot{x}_2) = 0$$
$$m\ddot{x}_2 + k(x_2 - x_1) + c(\dot{x}_2 - \dot{x}_1) = 0,$$

subject to $x_1(0) = 0$, $\dot{x}_1(0) = 0$, $x_2(0) = 1$, and $\dot{x}_2(0) = 0$.

Taking the Laplace transform and solving for $X_1(s)$ and $X_2(s)$ yields

$$X_1(s) = \frac{kms}{m^2s^4+2cms^3+3kms^2+cks+k^2},$$
$$X_2(s) = \frac{m^2s^3+2mcs^2+2mks+kc}{m^2s^4+2cms^3+3kms^2+cks+k^2}.$$

By using partial fraction expansion and taking the inverse transform of each of the terms, the following closed-form solution is obtained:

$$x_1(t) = e^{-\alpha_1 t}\left(A_1 cos\beta_1 t + B_1 sin\beta_1 t\right) + e^{-\alpha_2 t}\left(C_1 cos\beta_2 t + D_1 sin\beta_2 t\right)$$
$$x_2(t) = e^{-\alpha_1 t}\left(A_2 cos\beta_1 t + B_2 sin\beta_1 t\right) + e^{-\alpha_2 t}\left(C_2 cos\beta_2 t + D_2 sin\beta_2 t\right)$$

where the constants have the following values:

$$\alpha_1 = 4.3928, \alpha_2 = 0.243988, \beta_1 = 54.868, \beta_2 = 21.0582$$
$$A_1 = -C_1 = -0.44687, B_1 = -0.043391, D_1 = 0.025017$$
$$A_2 = 0.28072, B_2 = -2.00798 \times 10^{-4}, C_2 = 0.71928, D_2 = 0.067415$$

Results and discussion

The Abaqus results for each analysis are presented here.

Grounded tests

This test verifies spring and dashpot elements in two ways. First, it is shown that the results compare favorably with the analytical solution. Second, the output for similar spring/dashpot systems that use different element types is shown to compare well with respect to one another (e.g., a grounded SPRING1/DASHPOT1 system should yield the same results as a SPRINGA/DASHPOTA system that has one of the nodes constrained).

Figure 2 and *Table 1* to *Table 4* show that all three systems yield exactly the same results and that they all match the analytical solution very closely.

Node-to-node tests

Figure 3 shows the time history of the middle and end masses. Notice that, for either mass, system B yields the same result as system C, as expected. Both systems' time histories are also very close to the analytical solution.

Input files

exsd3glx.inp	Linear test of grounded springs and dashpots.
exsdbnlx.inp	Linear test of node-to-node springs and dashpots.
exsd3gla_po.inp	**POST OUTPUT* analysis.

Input files *exsd3gla.inp* and *exsdbnla.inp* are modified versions of files *exsd3glx.inp* and *exsdbnlx.inp*, respectively. They include temperature- and/or field variable-dependent behavior for spring constants and dashpot coefficients where applicable. These modified files are designed to provide the exact same results as those files from which they are derived.

Tables

Table 1: Nodal results at end of dynamic step: grounded tests.

Node	u_1	v_1	a_1
3	-1.2019×10^{-3}	6.905	−29.89
13	-1.2019×10^{-3}	6.905	−29.89
103	-1.2019×10^{-3}	6.905	−29.89

Table 2: Element results at end of dynamic step: grounded tests.

Element	S11	E11
1	-3.6056×10^{-2}	-1.2019×10^{-3}
3	-3.6056×10^{-2}	-1.2019×10^{-3}
5	-3.6056×10^{-2}	-1.2019×10^{-3}
2	0.0	0.0
4	0.0	0.0
6	0.8286	-1.2019×10^{-3}
7	0.8286	-1.2019×10^{-3}
8	0.8286	-1.2019×10^{-3}

Table 3: Nodal results at end of dynamic step: node-to-node tests.

Node	u_1	v_1	a_1
12	−0.2105	−6.531	151.8
102	−0.2105	−6.531	151.8
13	−0.2698	−11.55	91.23
103	−0.2698	−11.55	91.23

Table 4: Element results at end of dynamic step: node-to-node tests.

Element	S11	E11
2	−6.315	−0.2105
4	−6.315	−0.2105
3	−1.778	-5.9276×10^{-2}
5	−1.778	-5.9276×10^{-2}
7	−0.6022	-5.9276×10^{-2}
8	−0.6022	-5.9276×10^{-2}

Figures

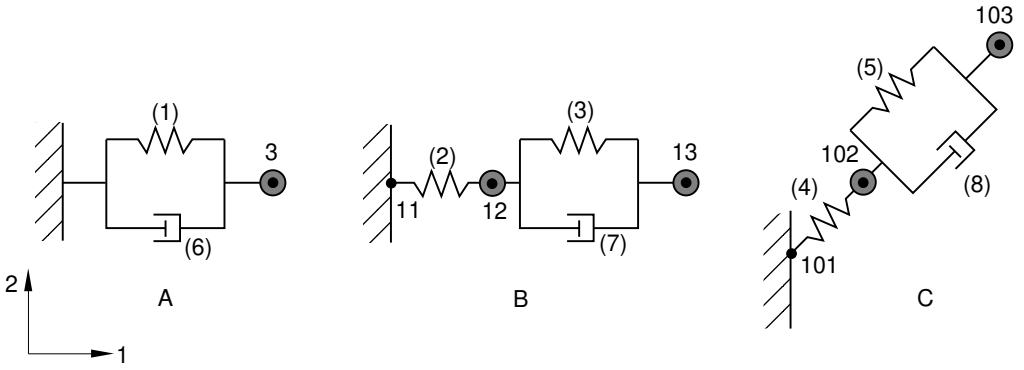


Figure 1: Three independent spring/dashpot/mass systems.

LINE	VARIABLE	SCALE FACTOR
1	u1 at node 3	+1.00E+00
2	u1 at node 13	+1.00E+00
3	u1 at node 103	+1.00E+00
4	analytical	+1.00E+00

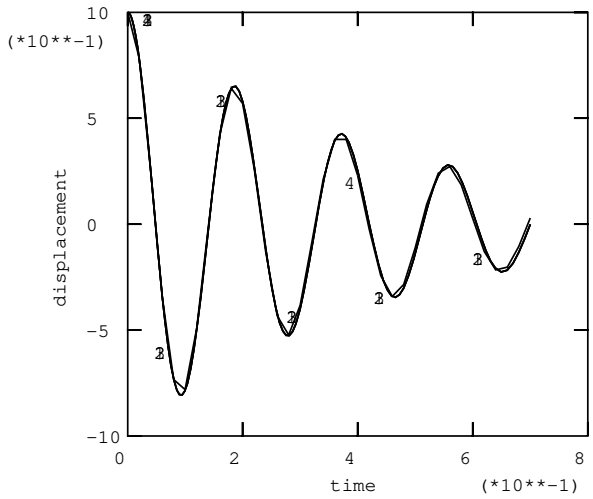


Figure 2: Free node displacement history: grounded tests.

LINE	VARIABLE	SCALE FACTOR
1	u1 at node 13	+1.00E+00
2	u1 at node 12	+1.00E+00
3	u1 at node 103	+1.00E+00
4	u1 at node 102	+1.00E+00
5	x1(t)	+1.00E+00
6	x2(t)	+1.00E+00

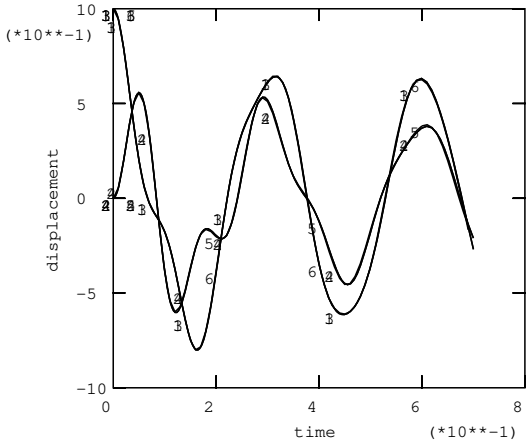


Figure 3: Free node displacement history: node-to-node tests.

Special-purpose elements

In this section:

- *Acoustic modes of an enclosed cavity*
- *Fluid filled rubber bladders*
- *Delamination analysis of laminated composites*

Acoustic modes of an enclosed cavity

Products: Abaqus/Standard

Overview

This example illustrates how to calculate the vibration frequencies of an enclosed two-dimensional acoustic cavity. The results provided by the acoustic elements are compared with published results for the same problem.

Problem description

The cavity is shown in *Figure 1*. Its walls are rigid, and it is fully enclosed. It is a rectangular cavity of length 236 mm and height 113 mm and contains a rigid wall located halfway along the longer side of the cavity. The wall is 10 mm thick and extends from one side of the cavity halfway across to the other wall. The cavity is filled with an acoustic fluid whose density is 1.0 kg/m^3 and whose bulk modulus is 0.1183 MPa.

Two models are used, one with first-order elements (element type AC2D4) and one with second-order elements (element type AC2D8). The mesh of first-order elements is shown in *Figure 1*. The mesh for the second-order elements uses the same pattern, with each block of four first-order elements replaced with a single element. No mesh convergence studies have been performed, but the close agreement between the frequencies computed and those given by Petyt et al. (1977) suggests that the meshes are adequate.

Since the acoustic fluid is fully enclosed by rigid walls, the acoustic pressure is not prescribed anywhere in the fluid. This means that an arbitrary acoustic pressure value is present in the solution—the equivalent of a rigid body mode in a structural problem, resulting in a zero frequency mode. During the frequency procedure (*Natural Frequency Extraction*) we, therefore, introduce a shift of −10 cycles/sec^2. This eliminates the difficulty of having a singularity in the matrix that must be solved during the eigenvalue extraction. The negative shift ensures that the frequencies are still extracted in ascending order, starting with the zero frequency.

Since the cavity is geometrically symmetric and we are only interested in obtaining the natural modes, the results are also available by modeling only half of the cavity, using symmetry and antisymmetry boundary conditions on the plane of geometric symmetry. We illustrate this by using half of the first-order model. This analysis is done in two steps. In the first step we impose the symmetry (natural) acoustic boundary condition on the plane of symmetry. This boundary condition is that the gradient of pressure normal to the plane, $\partial p/\partial n$, is zero. Since $\partial p/\partial n$ corresponds to surface "loading" in the acoustic problem, this boundary condition requires no data—it is an unloaded surface. The second step includes the antisymmetry boundary condition, $p = 0$, on the plane of symmetry.

Results and discussion

The first six nonzero frequencies are shown in *Table 1*, where they are compared with the calculated and experimentally measured values given by Petyt et al. (1977). There is fairly close agreement between all of the results, with the second-order model mostly providing higher frequencies (about 2% higher than those obtained with the first-order model). The pressure distributions predicted for these first six modes are shown in *Figure 2*.

The half-model that takes advantage of symmetry provides identical results, as shown in *Figure 3*. Modes 2, 3, and 6 are obtained in the first step with the symmetry condition and modes 1, 4, and 5 are obtained in the second step with the antisymmetry condition.

Input files

acousticmodes_ac2d4.inp	AC2D4 elements.
acousticmodes_ac2d8.inp	AC2D8 elements.
acousticmodes_ac2d4_half.inp	The same problem as acousticmodes_ac2d4.inp with appropriate boundary conditions so that only half the cavity need be modeled.

References

- Petyt, M., G. H. Koopman, and R. J. Pinnington, "Acoustic Modes of a Rectangular Cavity with a Rigid, Incomplete Partition," *Journal of Sound and Vibration*, vol. 53, pp. 71–82, 1977.

Tables

Table 1: Natural frequencies of an acoustic cavity.

Frequencies, Hz.				
Mode	Abaqus		Petyt et al.	
	AC2D4	AC2D8	Calculated	Measured
1	590	586	577	570
2	1443	1484	1450	1470
3	1502	1548	1550	1534
4	527	1573	1610	1555
5	1786	1870	1860	1840
6	1982	2149	2160	2120

Figures

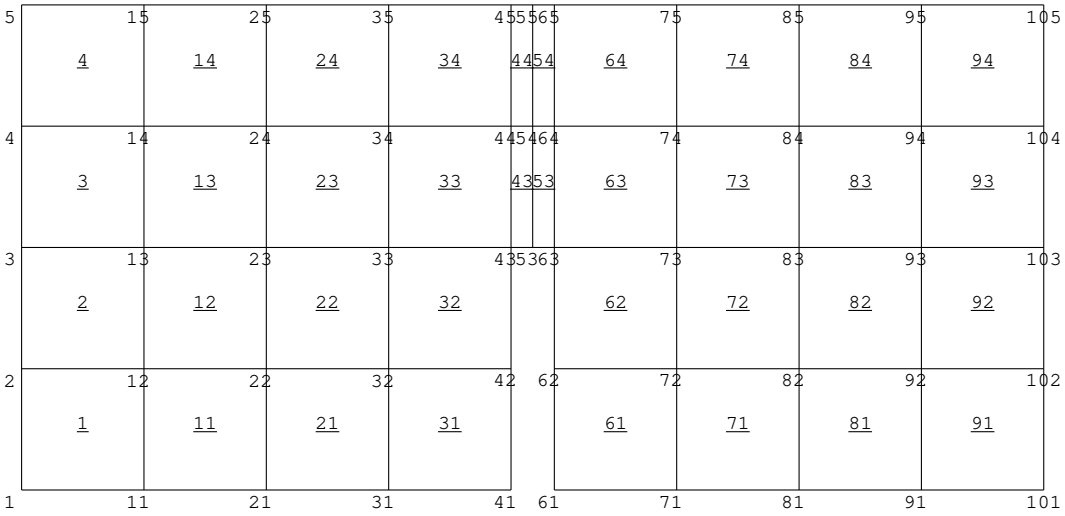


Figure 1: Acoustic cavity, showing the finite element mesh.

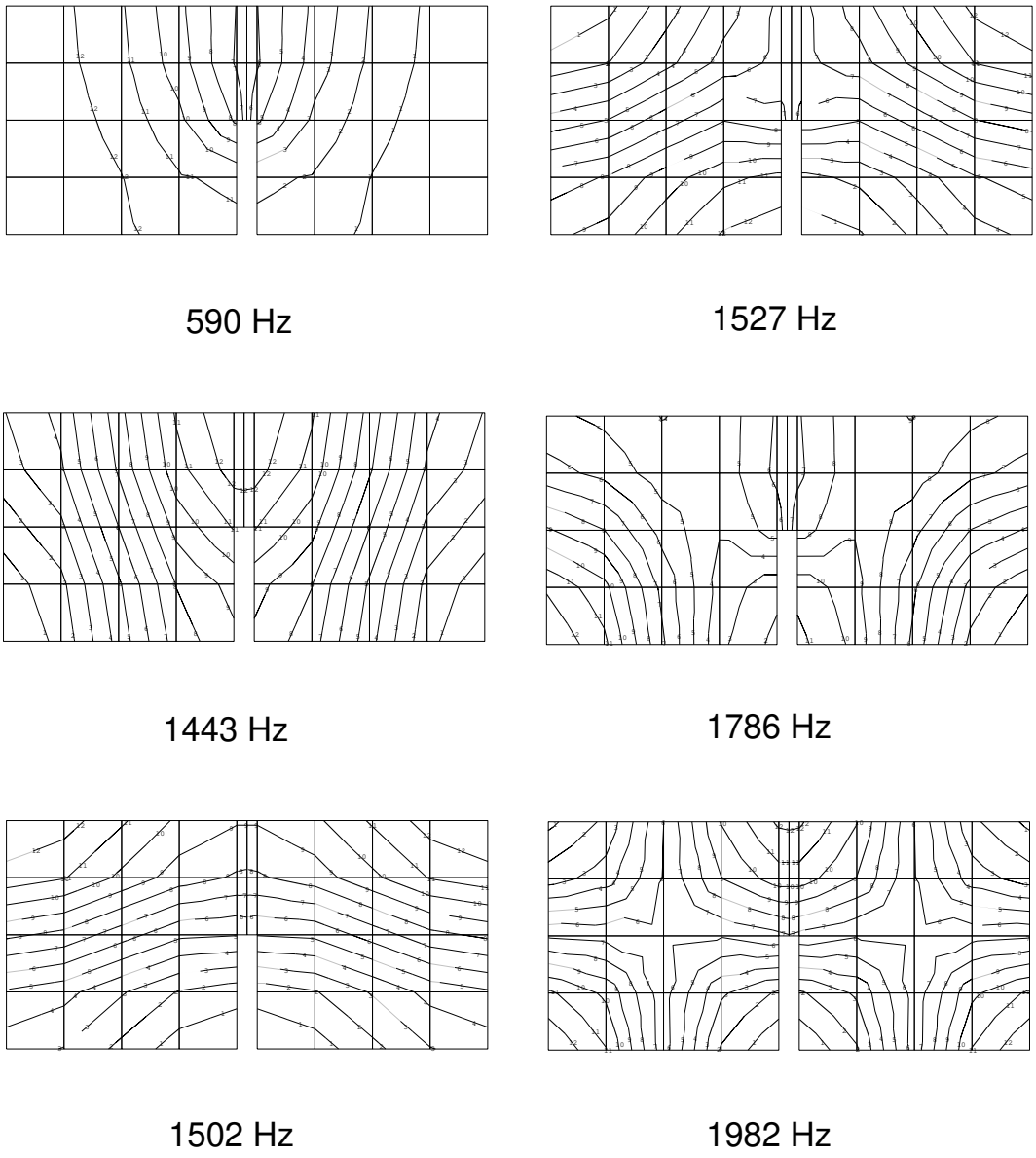


Figure 2: Pressure distributions in the first six modes.

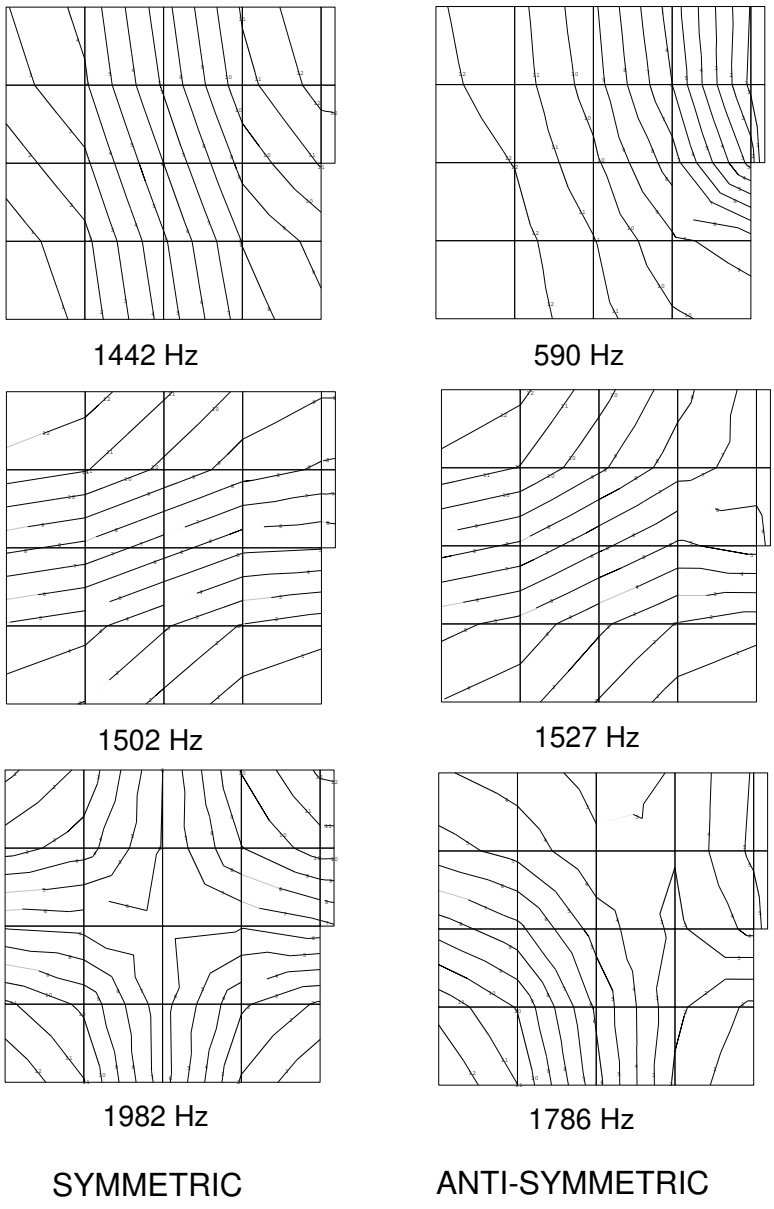


Figure 3: Pressure distributions in the first three modes of the half-model for the symmetric and antisymmetric cases.

Fluid filled rubber bladders

Products: Abaqus/Explicit

Overview

This example illustrates the transfer of fluid between two rubber bladders subjected to external axial compression. There is a fluid exchange between the two bladders so that fluid can move between them. The amount of mass transfer depends on the pressure differentials between the bladders.

Problem description

The bladders are modeled as short cylinders (with M3D4R elements) or spheres (with SAX1 elements) with a radius of 1.0 m and a wall thickness of 0.05 m. Ogden hyperelasticity (N=3) with material coefficients calibrated by Abaqus from experimental stress-strain data is used for the rubber constitutive equation. The fluid in the bladders is modeled as fluid cavities. The fluid cavity's reference node must lie on the symmetry plane or the symmetry axis, respectively. The normals to the element-based surface must point into the fluid cavity to obtain the correct cavity volume. The ambient pressure is assumed to be 50.0 kPa, and the fluid is prepressurized to a gauge (additional) pressure of 8.2736 kPa. Static equilibrium requires that the rubber bladders also be subjected to a uniform initial stress of 165.972 kPa along the circumferential direction in the M3D4R elements and a uniform in-plane initial stress of 82.736 kPa in the spheres.

The transfer of fluid is modeled by using a fluid exchange definition and specifying the bulk viscosity for the fluid exchange property. The viscous coefficient, C_V, and the hydrodynamic resistance coefficient, C_H, are chosen to be 10000.0 and 100.0, respectively. These resistance coefficients determine the mass flow rate at any time instant as a function of the pressure differential between the two bladders.

Results and discussion

The bladder systems are impacted by a rigid body at a constant downward velocity of 4.0 m/s. The total time of the event is 0.64 sec. *Figure 1* gives the initial configuration of the three-dimensional model. *Figure 2* shows the final deformed shape of the cylindrical rubber bladders. *Figure 3* gives the initial configuration of the axisymmetric model. *Figure 4* shows the final deformed shape of the spherical rubber bladders. The fluid pressures inside the two containers are plotted in *Figure 5* and *Figure 6*. The pressures in the two bladders are almost the same, owing to the fluid link, which drives the flow of fluid if there is any pressure differential between the two bladders. The pressure in the cylinders rises more than the pressure in the spheres as the relative volume change in the spheres is less than in the cylinders. The pressure in the cylinders and the spheres shows oscillations that are caused by the stiffness associated with flattening of the spheres.

Input files

fluidfilled_3d_surfcav.inp	Three-dimensional membrane elements.
fluidfilled_ax_surfcav.inp	Axisymmetric case where two rubber spheres instead of barrels are modeled.

Figures

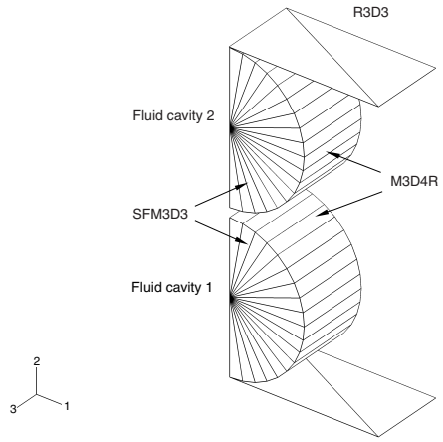


Figure 1: Initial (undeformed) configuration for the 3D case.

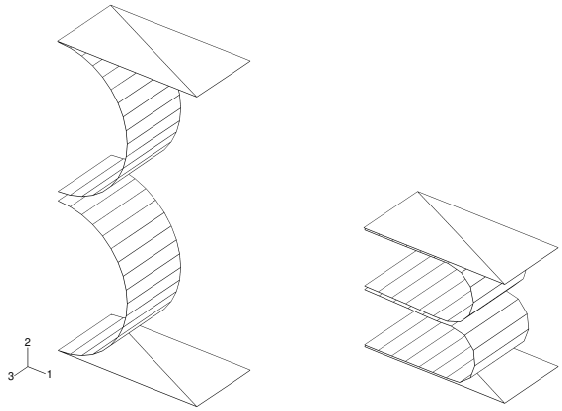


Figure 2: Undeformed rubber bladders on left and deformed rubber bladders on right. The bladders are modeled with M3D4R elements.

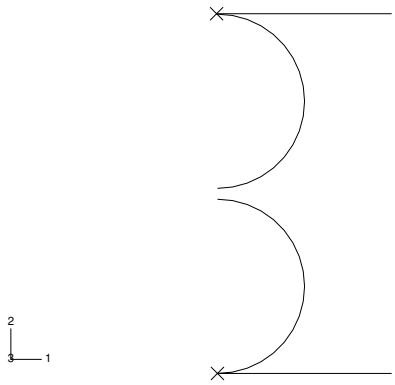


Figure 3: Initial (undeformed) configuration for the axisymmetric case.

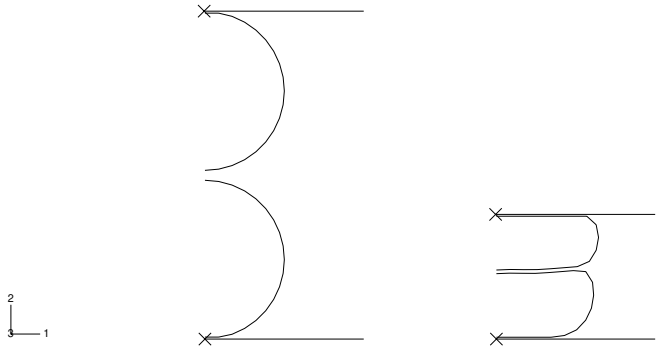


Figure 4: Undeformed rubber bladders on left and deformed rubber bladders on right. The bladders are modeled with SAX1 elements.

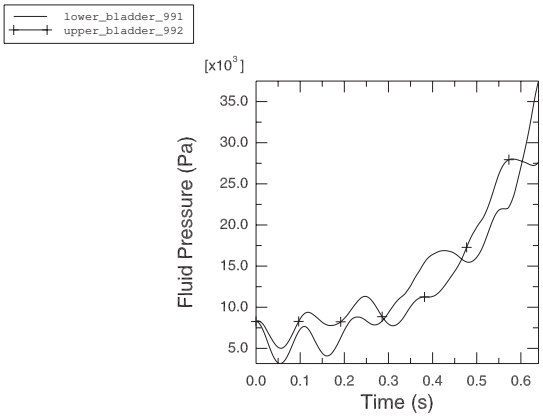


Figure 5: Fluid pressures in the two cylindrical rubber bladders.

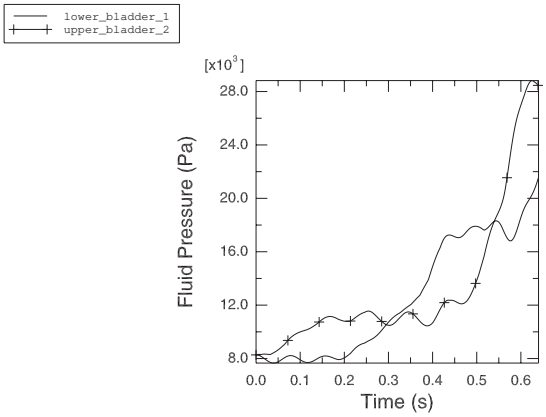


Figure 6: Fluid pressures in the two spherical rubber bladders.

Delamination analysis of laminated composites

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example verifies the use of Abaqus to predict mixed-mode multidelamination in a layered composite specimen. Cohesive elements, connector elements, traction-separation in contact, and a crack propagation analysis with VCCT criterion are used for this purpose.

The example studied is the one that appears in Alfano (2001). The results presented are compared against the experimental results included in that reference, taken from Robinson (1999).

Problem description

The model with cohesive elements is analyzed in Abaqus/Standard as well as Abaqus/Explicit and uses a damaged, linear elastic constitutive model. The model with VCCT criterion is also analyzed in both Abaqus/Standard and Abaqus/Explicit to predict debond growth. In addition, the model with VCCT criterion in Abaqus/Standard is analyzed using the Paris law to assess the fatigue life when it is subjected to subcritical cyclic displacement loading.

Geometry and model

The problem geometry and loading are depicted in *Figure 1*: a layered composite specimen, 200 mm long, with a total thickness of 3.18 mm and a width of 20 mm, loaded by equal and opposite displacements in the thickness direction at one end. The maximum displacement value is set equal to 20 mm in the monotonic loading case. In the low-cycle fatigue analysis, cyclic displacement loading with a peak value of 1 mm is specified. The thickness direction is composed of 24 layers. The model has two initial cracks: the first (of length 40 mm) is positioned at the midplane of the specimen at the left end, and the second (of length 20 mm) is located to the right of the first and two layers below.

When cohesive elements are used, the problem is modeled in both two and three dimensions, using solid elements to represent the bulk behavior and cohesive elements to capture the potential delamination at the interfaces between the 10th and 11th layers and between the 12th and 13th layers, counting from the bottom. In the two-dimensional finite element model, the top part of the specimen (consisting of 12 layers), the middle section (2 layers), and the bottom part (10 layers) are each modeled with a mesh of 1×200 CPE4I elements in Abaqus/Standard and CPE4R elements in Abaqus/Explicit. In both Abaqus/Standard and Abaqus/Explicit the initially uncracked portions of the two interfaces are modeled by one layer each of COH2D4 elements that share nodes with the adjacent solid elements. A similar, matching mesh is adopted for the equivalent three-dimensional model, where the corresponding element types used areC3D8I and COH3D8 in Abaqus/Standard and C3D8R and COH3D8 in Abaqus/Explicit, with one element in the width direction. The nodes where the equal and opposite displacements are prescribed are constrained in the length direction of the specimen; these are the only boundary conditions in the two-dimensional case. For the equivalent three-dimensional model all the nodes are also constrained in the width direction to simulate the plane strain effect. In addition, contact is defined between the open faces of the second, pre-existing crack to avoid penetrations if the faces are compressed against each other during the analysis.

In Abaqus/Standard, when the surface-based traction-separation capability available with the contact pair algorithm is used, the problem is modeled in both two and three dimensions. In Abaqus/Explicit the problem is modeled in three dimensions since surface-based traction-separation is available with the general contact algorithm. The models are very similar to those created for use with cohesive elements, as described in the previous paragraph, except that the cohesive elements are replaced with cohesive surfaces.

When the VCCT debond method is used, the problem is modeled in two dimensions in Abaqus/Standard but in three dimensions in Abaqus/Explicit. The models created above can also be adopted. Instead of using cohesive elements or traction-separation in contact in Abaqus/Standard, you can activate the crack propagation capability with the VCCT criterion. The same model is also used in a low-cycle fatigue analysis. When the same model is analyzed using Abaqus/Explicit, the VCCT criterion is obtained by assigning contact clearances, specifying cohesive behavior properties, and specifying crack propagation criteria with general contact.

When connector elements are used, the problem is modeled only in two dimensions in Abaqus/Standard. Two node-based surfaces are generated: one along the top surface of the 10th layer and the other along the bottom surface of the 11th layer. Both surfaces are tied to adjacent layers using surface-based tie constraints. CARTESIAN connector elements are used to bond the two node-based surfaces together to represent the interface. For the interface between the 12th and 13th layers, matched solid element nodes along the interface are connected directly using connector elements.

Material

The material data given in Alfano (2001) for the bulk material composite properties are $E_1 = 115.0$ GPa, $E_2 = 8.5$ GPa, $E_3 = 8.5$ GPa, $\nu_{12} = 0.29$, $\nu_{13} = 0.29$, $\nu_{23} = 0.3$, $G_{12} = 4.5$ GPa, $G_{13} = 3.3$ GPa, and $G_{23} = 4.5$ GPa.

The response of the cohesive elements in the model is specified through the cohesive section definition as a "traction-separation" response type. The elastic properties of the cohesive layer material are specified in terms of the traction-separation response with stiffness values $E = 850$ MPa, $G_1 = 850$ MPa, and $G_2 = 850$ MPa. The quadratic traction-interaction failure criterion is selected for damage initiation in the cohesive elements; and a mixed-mode, energy-based damage evolution law based on a power law criterion is selected for damage propagation. The relevant material data are as follows: $N_0 = 3.3$ MPa, $T_0 = 7.0$ MPa, $S_0 = 7.0$ MPa, $G_{1C} = 0.33 \times 10^3$ N/m, $G_{2C} = 0.80 \times 10^3$ N/m, $G_{3C} = 0.80 \times 10^3$ N/m, and $\alpha = 1.0$.

The same damage initiation criterion and damage evolution law with the same damage data are used for the surface-based traction-separation approach. However, in the absence of cohesive elements, their thickness is accounted for by scaling the elastic properties by a factor of 0.0132×10^{-3} (since the cohesive elements have a thickness of 0.0132 mm), and the properties are specified as $E =$ 64,200 GPa, $G_1 =$ 64,200 GPa, and $G_2 =$ 64,200 GPa.

For the VCCT debond approach, the BK mixed-mode failure law is used with the same critical energy release rates as those used for cohesive elements; that is, $G_{1C} = 0.33 \times 10^3$ N/m, $G_{2C} = 0.80 \times 10^3$ N/m, and $G_{3C} = 0.80 \times 10^3$ N/m. The exponent of the BK law is specified as $\eta = 2.284$. When the low-cycle fatigue analysis using the Paris law is performed, the additional relevant data are as follows: $c_1 = 0.5$, $c_2 = -1.44$, $c_3 = 4.88 \times 10^{-6}$, $c_4 = 1.15$, $\frac{G_{thresh}}{G_C} = 0.01$, and $\frac{G_{pl}}{G_C} = 0.85$.

Force-based damage initiation and a tabular form of motion-based damage evolution are used to define the connector damage mechanisms. Initiation forces are calculated based on the value of N_0 given above for cohesive elements. For example, the initiation force for the lower interface is calculated as 66 N, which is equal to $N_0 \times A$. The interface area over one cohesive element, A, is 20×10^{-6}. The stiffnesses of the connector elements are calculated as $k = \frac{EA}{L} = 1.283 \times 10^9$ N/m, where L is the thickness of the cohesive element. To improve the convergence behavior of this model, viscous regularization has been applied.

Results and discussion

The plots of the prescribed displacement versus the corresponding reaction force for the delamination problem are presented in *Figure 2* and compared with the experimental results included in Alfano (2001). Both the Abaqus/Standard and Abaqus/Explicit results displayed in the graph are from the two-dimensional analyses. The results from the equivalent three-dimensional models are almost identical to their two-dimensional counterparts and are not included in *Figure 2*. It can be seen from *Figure 2* that the curve produced using the surface-based traction-separation approach is nearly the same as that obtained using cohesive elements. Both curves have good agreement with the experimental results up to an applied displacement of approximately 20 mm; then, a sharp drop in the reaction force is observed at this point by the Abaqus analysis, after which the reaction force values appear to be underpredicted by approximately 30% when compared to the experimental data. The reason for this deviation, which appears to coincide with the simultaneous propagation of both of the cracks, is related to the sudden failure of a relatively large number of cohesive elements in a very short period of time. On the other hand, the data predicted using the VCCT debond method agree well with the experimental results, without the sharp drop previously noted. While the Abaqus/Explicit results, both with cohesive elements as well as from the three-dimensional model with surface-based traction-separation, follow the same pattern as the Abaqus/Standard results, they are not as smooth due to inertia effects. A second-order Butterworth-type filter was applied to the nodal reaction force history output from the Abaqus/Explicit analysis to eliminate high-frequency oscillations from the response curve.

Figure 3 shows the results using cohesive elements from a series of Abaqus/Standard analyses incorporating a viscous regularization scheme to improve convergence and demonstrates the effect on the predicted results of the choice of the viscosity parameter, μ. Larger values of μ, while providing better convergence, affect the results more than smaller viscosity values. The appropriate value of the viscosity parameter that results in the right balance between improved convergence behavior of the nonlinear system and accuracy of the results is problem dependent and requires judgment on the part of the user. In the cohesive zone approach to modeling delamination, the complex fracture process at the micro-scale is modeled using only a few macroscopic parameters (such as peak strength and fracture energy). While viscous regularization is not intended to be used to model rate effects, it does provide an additional parameter that can be "fitted" to the material model at hand. For the particular delamination problem analyzed, as can be seen from *Figure 3*, a larger value of μ causes the first peak of the reaction force curve to be higher than the experimental value and predicts a milder and smoother drop in the reaction force following the peak compared to the experimental data. However, the results with viscous regularization (for example, the curve for $\mu = 1.0 \times 10^{-4}$ in *Figure 3*) appear to match the experiments better for prescribed displacement values greater than 20 mm.

Figure 2 also shows the results from the analysis using connector elements to model the bonded interfaces. The same trend of delamination is observed as seen in the experimental data. *Figure 4* illustrates the effect of viscous regularization. In one case, a viscous regularization factor of 0.0008 and maximum degradation factor of 0.99 are used. In the other case, the values are 0.0005 and 0.9, respectively. As can be seen in *Figure 4*, a larger value of viscous regularization causes the peak of the reaction force to be higher.

Figure 5 illustrates how the ratio of the peak reaction force over the corresponding peak prescribed displacement (stiffness) degrades as a function of the cycle number after using the direct cyclic approach. Similar results are obtained when using the general and simplified fatigue crack growth approaches.

A comparison of the deformed configurations between the three-dimensional Abaqus/Explicit model and the two-dimensional Abaqus/Standard model is shown in *Figure 6*. *Figure 7* depicts the delamination of both the top and bottom layers obtained from the Abaqus/Explicit and Abaqus/Standard analyses. A comparison of reaction forces versus displacement, illustrated in *Figure 8*, verifies the consistency in predicting the debond growth of both analyses. Inertia effects were observed in Abaqus/Explicit later in the analysis when both bonded layers started to debond. Although the forces are not as smooth, they follow the same pattern as the Abaqus/Standard results.

Input files

alfano_2d_std.inp	Abaqus/Standard two-dimensional model.
alfano_2d_reg_std.inp	Abaqus/Standard two-dimensional model using viscous regularization.
alfano_vcct_2d_1.inp	Abaqus/Standard two-dimensional model using the VCCT debond method.
alfano_vcct_fatigue_2d.inp	Abaqus/Standard two-dimensional model using the Paris law to analyze the fatigue delamination growth using the direct cyclic approach.
alfano_vcct_fatigue_2d_2.inp	Abaqus/Standard two-dimensional model using the Paris law to analyze the fatigue delamination growth using the general fatigue crack growth approach.
alfano_vcct_fatigue_2d_2_simp.inp	Abaqus/Standard two-dimensional model using the Paris law to analyze the fatigue delamination growth using the simplified fatigue crack growth approach.
alfano_2d_std_surf.inp	Abaqus/Standard two-dimensional model using contact surface-based traction-separation behavior with a viscous regularization factor of 1×10^{-7}.
alfano_3d_std.inp	Abaqus/Standard three-dimensional model.
alfano_3d_std_surf.inp	Abaqus/Standard three-dimensional model using contact surface-based traction-separation behavior with a viscous regularization factor of 1×10^{-7}.
alfano_2d_xpl.inp	Abaqus/Explicit two-dimensional model with cohesive elements.
alfano_3d_xpl.inp	Abaqus/Explicit three-dimensional model with cohesive elements.
alfano_3d_xpl_surf.inp	Abaqus/Explicit three-dimensional model with surface-based traction-separation behavior.
alfano_3d_xpl_vcct.inp	Abaqus/Explicit three-dimensional model with VCCT criterion.
alfano_std_conn2d_reg1.inp	Abaqus/Standard two-dimensional model using connector elements and a viscous regularization factor of 0.0005.
alfano_std_conn2d_reg2.inp	Abaqus/Standard two-dimensional model using connector elements and a viscous regularization factor of 0.0008.

Python scripts

alfano_2d_model.py	Script for creating the two-dimensional version of this model using Abaqus/CAE.

References

- Alfano, G., and M. A. Crisfield, "Finite Element Interface Models for the Delamination Analysis of Laminated Composites: Mechanical and Computational Issues," *International Journal for Numerical Methods in Engineering*, vol. 50, pp. 1701–1736, 2001.
- Robinson, P., T. Besant, and D. Hitchings, "Delamination Growth Prediction Using a Finite Element Approach," *2nd ESIS TC4 Conference on Polymers and Composites, Les Diablerets, Switzerland*, 1999.

Figures

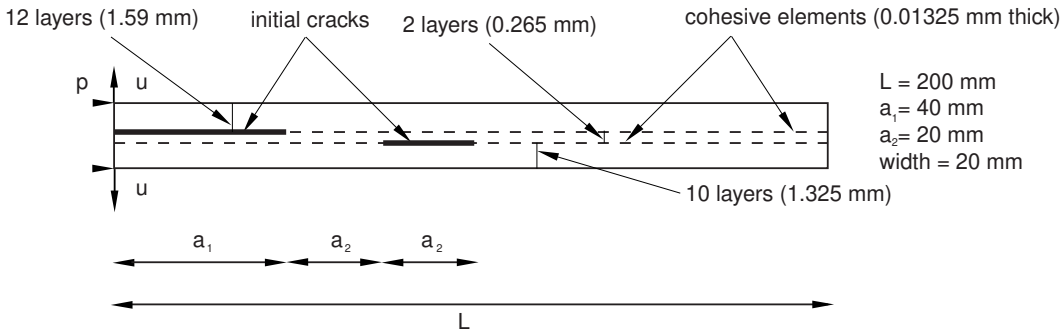


Figure 1: Model geometry for the Alfano delamination problem.

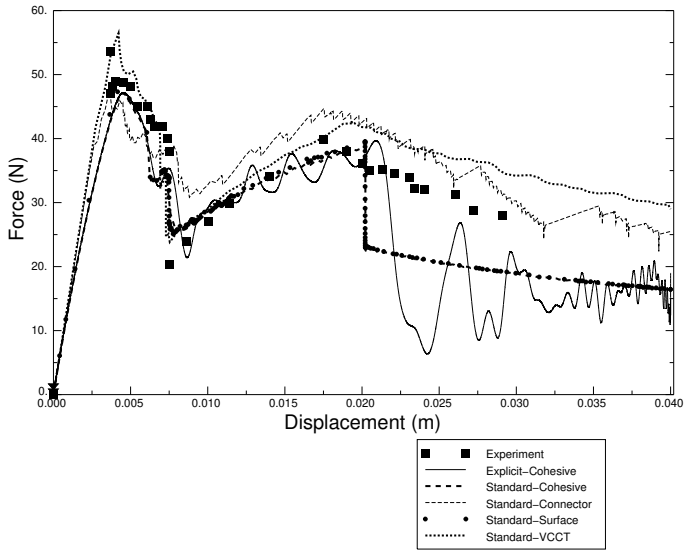


Figure 2: Reaction force vs. prescribed displacement: experimental and numerical results.

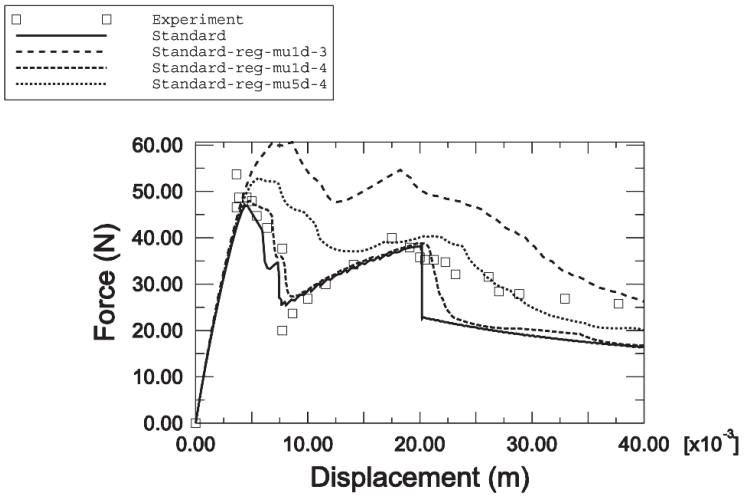


Figure 3: Effect of viscous regularization on the predicted force-displacement response using cohesive elements.

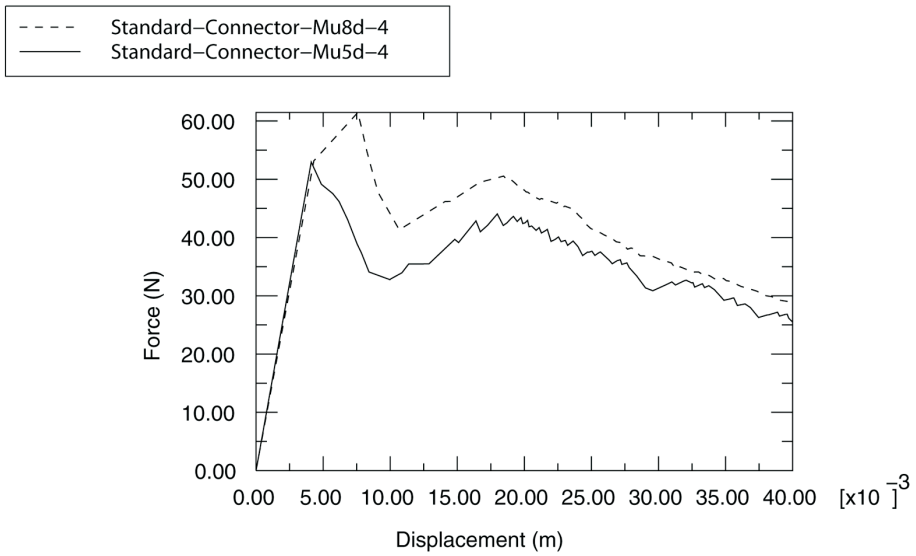


Figure 4: Effect of viscous regularization on the predicted force-displacement response using connector elements.

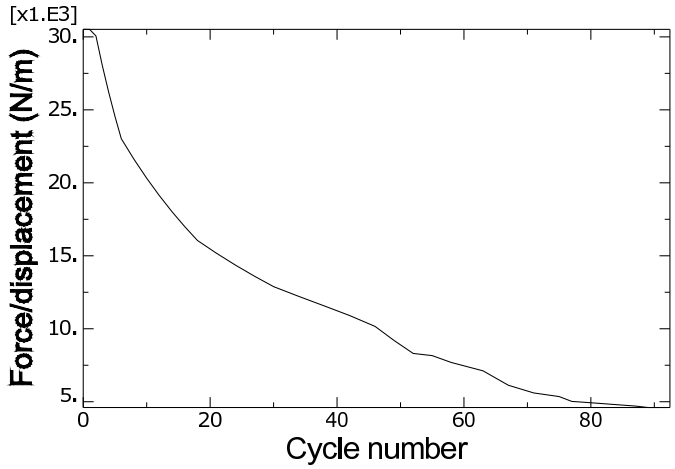


Figure 5: Stiffness degradation as a function of cycle number.

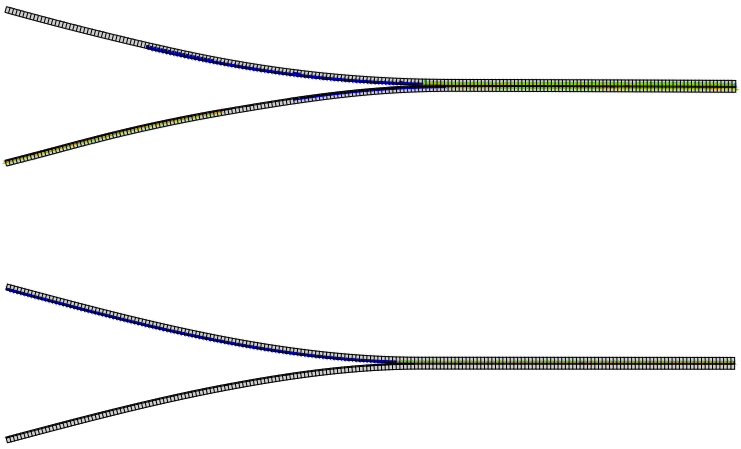


Figure 6: Deformed configuration comparison between Abaqus/Explicit (top) and Abaqus/Standard (bottom).

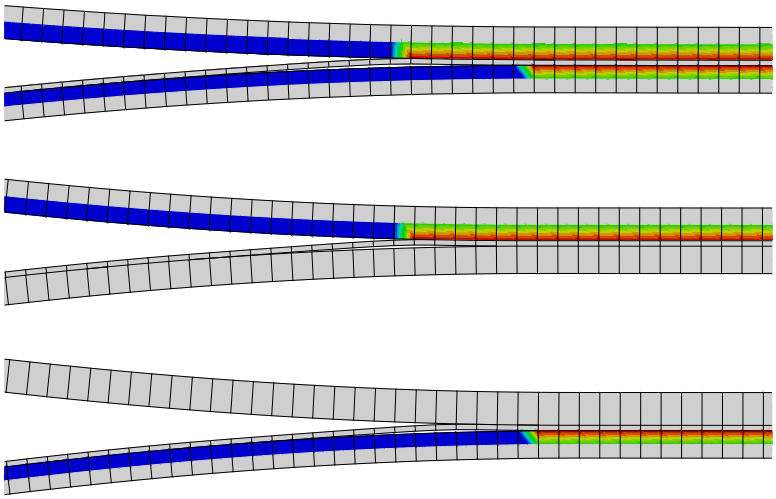


Figure 7: Delamination comparison between Abaqus/Explicit (top) and Abaqus/Standard (middle and bottom).

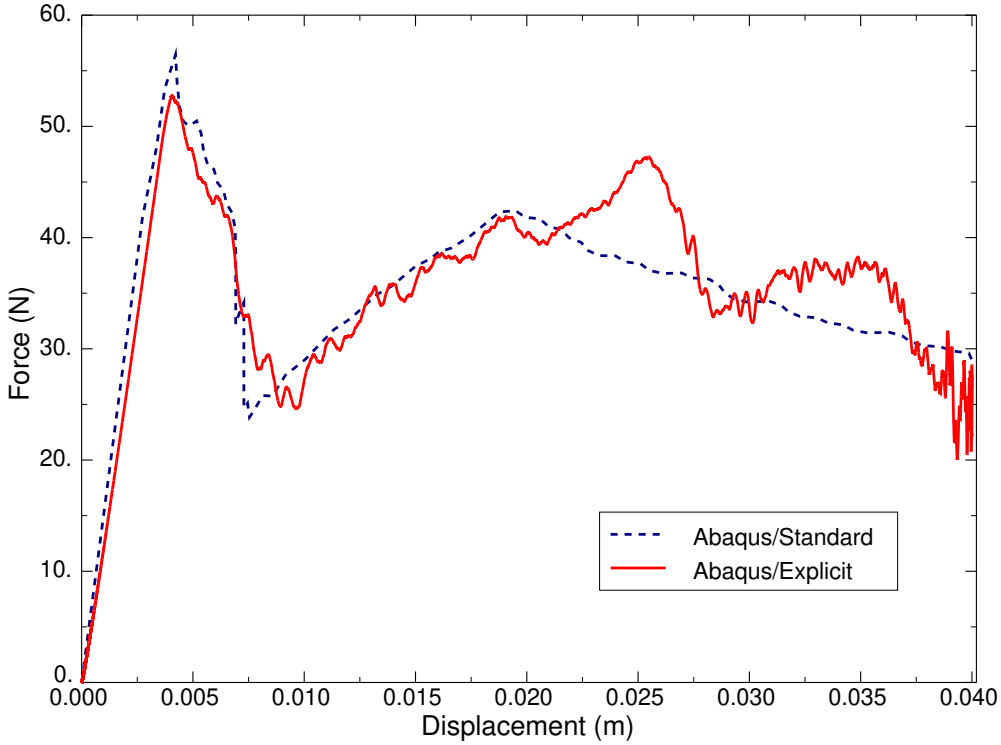


Figure 8: Comparison of the force-displacement response between Abaqus/Explicit and Abaqus/Standard.

Material Tests

In this section:

- *Elasticity*
- *Plasticity and creep*

Elasticity

In this section:

- *Viscoelastic rod subjected to constant axial load*
- *Transient thermal loading of a viscoelastic slab*
- *Uniform strain, viscoplastic truss*
- *Fitting of rubber test data*
- *Fitting of elastomeric foam test data*
- *Rubber under uniaxial tension*
- *Anisotropic hyperelastic modeling of arterial layers*

Viscoelastic rod subjected to constant axial load

Products: Abaqus/Standard

Overview

This example verifies the use of the time domain linear viscoelastic material model in Abaqus.

Problem description

The example, taken from *Collingwood et al. (1985)*, is a rod of length of 254 mm (10 in) and diameter of 25.4 mm (1 in). The rod is fixed in the axial direction on one end and a constant axial load of 0.689 MPa (100 psi) is applied suddenly to the other end. The rod is modeled using one quadratic, axisymmetric, hybrid continuum element (CAX8H).

Material

The linear viscoelastic material model used in this example can be represented by a combination of linear springs and a dashpot, as shown in *Figure 1*. The extensional relaxation function is

$$E_R(t) = k_1 + k_2 e^{-t/\tau_R},$$

where $\tau_R = \eta/k_2$, η is the damping coefficient and k_1 and k_2 are constants. In this case k_1 is 6.89 MPa (1000 psi); k_2 is 62.01 MPa (9000 psi); τ_R is 1.0 sec; and the bulk modulus, $\mathbf{K}$, is 689 MPa (100,000 psi) and is independent of time.

Short-term material properties are specified using linear elastic behavior (*Linear Elastic Behavior*), which requires the instantaneous Young's modulus, E_0, and Poisson's ratio, ν_0. The time-dependent behavior is specified using viscoelastic behavior, in which the shear relaxation modulus and the bulk modulus are defined by a Prony series (see *Time Domain Viscoelasticity*). For the Abaqus analysis of this problem, it is assumed that no volumetric relaxation occurs.

E_0 is immediately available as $E_0 = k_1 + k_2 =$ 68.9 MPa (10000 psi), and ν_0 is

$$\nu_0 = \frac{3\mathbf{K} - k_1 - k_2}{6\mathbf{K}} = 0.4833.$$

The time-dependent material behavior is approximated with a single-term Prony series for the shear relaxation modulus:

$$G_R(t) = G_0\left(1 - \bar{g}_1^P\left(1 - e^{-t/\tau_1^G}\right)\right).$$

We need to compute G_0, $\bar{g}_1^P$, and τ_1^G from the extensional relaxation function. The limiting cases of both the shear and extensional relaxation functions are used for this purpose. The long-term ($t \to \infty$) properties of the material approach that of a linear elastic solid, with $E = k_1$. The long-term shear modulus, G, can be calculated using the relationship between the bulk, shear, and extensional moduli:

$$G = \frac{3\mathbf{K}k_1}{9\mathbf{K} - k_1}.$$

Likewise, the “instantaneous” or “glassy” shear modulus, G_0, is

$$G_0 = \frac{3\mathbf{K}(k_1 + k_2)}{9\mathbf{K} - (k_1 + k_2)}.$$

Then $\overline{g}_1^P = 1 - G/G_0 =$ 0.901001.

The shear relaxation time, τ_1^G, is obtained by writing the rate of change of the shear modulus in terms of the rate of change of the extensional modulus at time $t =$ 0:

$$\left.\frac{dG_R}{dt}\right|_{t=0} = \left.\left(\frac{dG_R}{dE_R}\frac{dE_R}{dt}\right)\right|_{t=0}.$$

After some algebra we obtain

$$\tau_1^G = \frac{9K-k_1-k_2}{9K-k_1}\tau_R = 0.9899\text{sec}.$$

The same problem is also treated as a large-strain example. The relaxation behavior is defined in the same way, but the short-term elastic properties are given using hyperelastic behavior. The polynomial formulation with $N = 1$ is used, and the constants are $C_{10} =$ 6.89 MPa (1000 psi), $C_{01} =$ 4.59 MPa (666.67 psi), and $D_1 =$ 0.0029MPa^{-1} (0.00002 psi^{-1}). These constants are such that the initial Young's modulus and initial Poisson's ratio are equal to E_0 and ν_0, respectively, and produce a close fit to a linear material. (See *Hyperelastic Behavior of Rubberlike Materials* for further discussion of the choice of constants when $N = 1$.)

Loading

A distributed load of 0.689 MPa (100 psi) is applied instantaneously and held constant throughout the analysis. To model this, we use the quasi-static procedure (*Quasi-Static Analysis*) in two steps. The load is applied in the first step, which has a time period of 0.001 seconds, so that the instantaneous (glassy) behavior dominates. Since this step uses only one increment, a tolerance to control the accuracy of the creep integration is not specified. The second step has a time period of 50 seconds, during which the load is held constant and the rod is allowed to relax toward its long-term behavior. Automatic time incrementation is chosen by giving a tolerance value for the maximum difference in the creep strain increment over a time increment. This tolerance is selected so that its value is of the same order of magnitude as the maximum elastic strain. Therefore, for this example the tolerance is set to 5×10^{-3}. The total force and the total moment on the loaded face of the model are output to the results file.

Results and discussion

The instantaneous and long-term behaviors provide a check on the Abaqus results. The instantaneous and long-term axial displacements of the rod tip can be calculated as follows:

$$\frac{u_2}{L}(t=0) = \frac{\sigma_{22}}{k_1+k_2} = 0.01$$

$$\frac{u_2}{L}(t\to\infty) = \frac{\sigma_{22}}{k_1} = 0.1.$$

These values agree well with the Abaqus results. Similarly, the instantaneous and long-term values of the Poisson's ratio can also be calculated exactly:

$$\nu(t=0) = \frac{3K-k_1-k_2}{6K} = 0.4833$$

$$\nu(t\to\infty) = \frac{3K-k_1}{6K} = 0.4983.$$

The Poisson's ratio can be extracted from the Abaqus results by taking the ratio of the lateral strain to the axial strain at $t =$ 0.001 and $t =$ 50:

$$\nu\left(t=0.001\right) = -\frac{\varepsilon_{11}}{\varepsilon_{22}} = -\frac{-4.8353\times10^{-3}}{1.0005\times10^{-2}} = 0.4833$$

$$\nu\left(t=50\right) = -\frac{\varepsilon_{11}}{\varepsilon_{22}} = -\frac{-4.9478\times10^{-2}}{9.9291\times10^{-2}} = 0.4983.$$

Since this is an applied stress problem, obtaining the exact solution for the entire time period of the analysis requires inverting the original constitutive integral equation defining uniaxial stress in terms of uniaxial strain. To perform this inversion, we use the following relation (Pipkin, 1972) between the time-dependent relaxation modulus, $E_R\left(t\right)$, and the time-dependent creep compliance, $J_C\left(t\right)$:

$$\int_0^t J_C\left(\tau\right) E_R\left(t-\tau\right) d\tau = t.$$

Differentiation of this relation with respect to t yields

$$J_C\left(t\right) \quad E_R\left(0\right) + \int_0^t J_C\left(\tau\right) \dot{E}_R\left(t-\tau\right) d\tau = 1.$$

With the previously used expression for $E_R\left(t\right)$ this takes the form

$$\left(k_1+k_2\right) J_C\left(t\right) - \frac{k_2}{\tau_R}\int_0^t J_C\left(\tau\right) e^{(\tau-t)/\tau_R} d\tau = 1.$$

Differentiating this expression once more provides

$$\left(k_1+k_2\right) \dot{J}_C\left(t\right) - \frac{k_2}{\tau_R} J_C\left(t\right) + \frac{k_2}{\tau_R^2}\int_0^t J_C\left(\tau\right) e^{(\tau-t)/\tau_R} d\tau = 0.$$

Multiplying this equation by τ_R and adding it to the previous equation yields the differential equation

$$\left(k_1+k_2\right)\tau_R \dot{J}_C\left(t\right) + k_1 J_C\left(t\right) = 1.$$

With the introduction of the creep time constant $\tau_C = \left(1+k_2/k_1\right)\tau_R$ this can be written as:

$$\tau_C \dot{J}_C\left(t\right) + J_C\left(t\right) = 1/k_1.$$

The general solution to this differential equation is

$$J_C\left(t\right) = 1/k_1 + Ce^{-t/\tau_C},$$

where the coefficient C is defined by the initial condition $J_C\left(0\right) = 1/\left(k_1+k_2\right)$, which yields

$$J_C\left(t\right) = \left(1 - \frac{k_2}{k_1+k_2} e^{-t/\tau_C}\right)\frac{1}{k_1}.$$

For this problem the stress σ_{22} is a constant, so that

$$\varepsilon_{22} = J_C\left(t\right)\sigma_{22} = \left(1 - \frac{k_2}{k_1+k_2} e^{-t/\tau_C}\right)\frac{\sigma_{22}}{k_1}.$$

From the values given above for k_1, k_2, and τ_R, as well as the fact that $\sigma_{22} = 0.689$ MPa (100 psi), ε_{22} becomes

$$\varepsilon_{22} = 0.1\left(1 - 0.9e^{-t/10}\right).$$

From this equation, it is evident that the effective time constant for the problem is dramatically different (by a factor of 10 in this case), depending upon whether the loading is applied force or applied displacement. *Figure 2* is a time history plot of ε_{22} as predicted by the above equation and as calculated by Abaqus. The plot shows acceptable agreement between the Abaqus results and the exact solution. Closer agreement can be obtained by using a smaller tolerance value for the maximum difference in the creep strain increment over a time increment.

The solution obtained with the large-strain formulation differs negligibly from the small-strain solution. Abaqus automatically converts frequency domain data into a time domain Prony series representation. The analysis results using Prony parameters calibrated from tabulated frequency-dependent moduli data are in good agreement with the analyses using time domain data directly.

Input files

viscorod_smallstrain.inp	Small-strain input data for this problem.
viscorod_small_frq2tim.inp	Small-strain input data for time domain analysis using Prony parameters calibrated from tabulated frequency-dependent moduli data.
viscorod_largestrain.inp	Large-strain input data for this problem.
viscorod_large_frq2tim.inp	Large-strain input data for this problem using Prony parameters calibrated from tabulated frequency-dependent moduli data.
viscorod_c3d8.inp	Model using the three-dimensional 8-node brick element, C3D8.
viscorod_cps4.inp	Model using the 4-node plane stress element, CPS4.
viscorod_t3d2.inp	Model using the 2-node truss element, T3D2.
viscorod_postoutput.inp	**POST OUTPUT* job for the restart file generated in viscorod_largestrain.inp.

References

- Collingwood, G. A., E. B. Becker, and T. Miller, User's Manual for the TEXVISC Computer Program, Morton Thiokol, Inc., Document Numbers U-85-4550A and U-85-4550B , 1985.
- Pipkin, A. C., Lectures on Viscoelasticity Theory, Springer Verlag, New York , 1972.

Figures

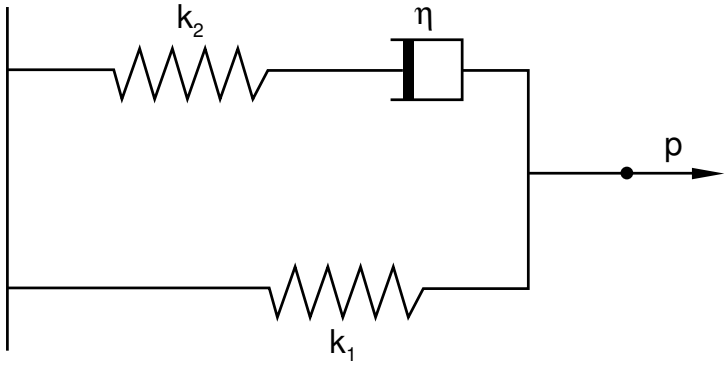


Figure 1: Spring and dashpot model of viscoelastic rod.

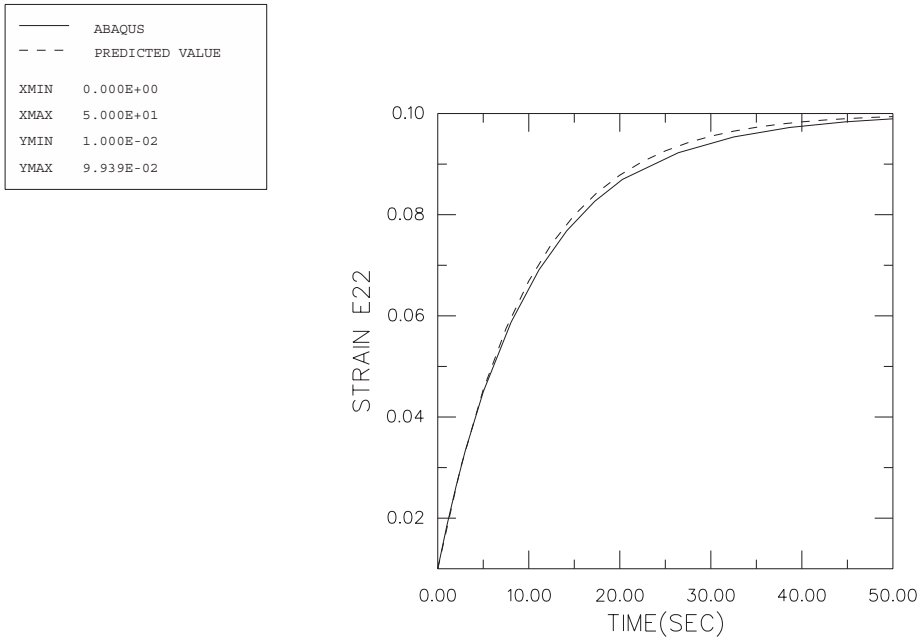


Figure 2: Time history of strain in the direction of load for viscoelastic rod.

Transient thermal loading of a viscoelastic slab

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example verifies the use of the time domain linear viscoelastic material model in conjunction with a temperature-time shift function.

Problem description

The example, taken from Collingwood et al. (1985), models a viscoelastic slab under plane strain restrained in all directions in its plane and examines the response of the slab after the temperature of its faces is raised suddenly to 100°.

The slab has unit half-thickness. Since the problem is one-dimensional, the slab is modeled with a single row of plane strain continuum elements. In Abaqus/Standard a sequential thermal-stress analysis is performed with two-dimensional, 8-node heat transfer elements, DC2D8, used for the heat transfer analysis and the corresponding 8-node plane strain continuum elements, CPE8R, used for the stress analysis. The mesh is shown in *Figure 1*. In Abaqus/Explicit a coupled thermal-stress analysis is performed using first-order plane strain elements (CPE3T and CPE4RT) to model the slab. Twenty elements are used along the length of the slab in the Abaqus/Explicit simulation.

The initial temperature throughout the slab is 0°. The outside face of the slab, at $X = 1$, is instantaneously raised to 100°. The mesh (*Figure 1*) is finer toward $X = 1$, where the temperature gradient is expected to be highest. The resulting transient temperature distribution is written to the results file and used as input to the subsequent stress analysis. Plane strain is imposed in the Y-direction by setting $u_y = 0$ on the two faces of the mesh at $Y = 0$ and $Y = 1$. Symmetry about $X = 0$ is also imposed.

Material

The thermal material properties are arbitrarily defined (in consistent units) as thermal conductivity (k) of 1.0, specific heat (c) of 1.0, and density (ρ) of 1.0.

The viscoelastic material models (small-strain and large-strain) are the same as the ones used in *Viscoelastic rod subjected to constant axial load*, with the addition of a temperature-time shift. The temperature-time shift uses the Williams-Landel-Ferry approximation,

$$\log(A) = -\frac{C_1(\theta-\theta_0)}{C_2+(\theta-\theta_0)},$$

where θ_0 is the reference temperature at which the relaxation data are given and C_1, C_2 are calibration constants obtained at this temperature (for additional information on the WLF equation, see *Viscoelasticity*). When $\theta \le \theta_0 - C_2$, the material behavior is elastic. In this example $C_1 = 4.92$, $C_2 = 215.0°$, and $\theta_0 = 70°$. The coefficient of thermal expansion is 1.0×10^{-5} per degree.

Analysis

The transient heat transfer problem is analyzed in Abaqus/Standard using a heat transfer procedure for a time period of 6 seconds, so the structure is allowed to come to thermal equilibrium. The integration procedure used in Abaqus/Standard for transient heat transfer analysis introduces a relationship between the minimum usable

time increment and the element size and material properties. The guideline given in *Uncoupled Heat Transfer Analysis* is

$$\Delta t > \frac{\rho c}{6k}\Delta l^2,$$

where Δl is the size of the smallest element in the mesh. If time increments smaller than this value are used, spurious oscillations may appear in the solution. Since the mesh is rather coarse, the minimum usable time increment predicted by the above formula is 4.17×10^{-4} seconds. A suggested initial time increment of 5×10^{-4} seconds is, therefore, used.

Automatic time incrementation is chosen by setting a value for the maximum temperature change allowed in an increment during a heat transfer analysis. Smaller values for the maximum temperature change cannot be used in this problem because they result in time increments that are smaller than the minimum usable time increment described above. As a consequence, the thermal analysis is rather approximate. A finer mesh would be necessary to obtain more accurate results.

The stress analysis uses the temperature distribution obtained in the heat transfer analysis to define the thermal loading. The quasi-static procedure (*Quasi-Static Analysis*) is used with automatic incrementation, chosen by specifying a value for the maximum difference in the creep strain increment over a time increment. This value is set to 2.0×10^{-3}, which is of the same order of magnitude as the maximum elastic strain. The time period is 6 seconds, and the initial suggested time increment is 5.0×10^{-4} seconds to capture the high temperature gradients that occur very early in the analysis.

In Abaqus/Explicit the thermal and mechanical responses of the slab are determined simultaneously. The automatic time incrementation scheme available in Abaqus/Explicit is used to ensure numerical stability and to advance the solution in time.

Results and discussion

The temperature distribution in the first part of the problem is given in Carslaw and Yeager (1959). *Table 1* compares that exact solution with the Abaqus results after an elapsed time of one second.

Table 1: Exact solution compared to Abaqus results.

X/L	Temperature		
	Carslaw and Yeager (1959)	Abaqus/Standard	Abaqus/Explicit
0.9	98.2	97.8	98.3
0.7	95.0	93.5	95.1
0.5	92.2	89.9	92.3
0.2	89.5	86.4	89.7
0.0	89.2	85.7	89.1

The Abaqus/Standard results are of limited accuracy because of the relatively large time increments used.

The stress and strain distributions at various times during the solution are shown in *Figure 2* and *Figure 3*. The final stress in the slab is calculated as −0.0138 MPa (−2 psi), while the final strain is 2.99×10^{-3}. *Figure 4* shows the time history of the stress at the leftmost and rightmost integration points in the structure. Time histories for the same problem, solved without the temperature-time shift, are also shown in this figure. As expected, the shift considerably shortens the time required for the structure to reach equilibrium.

The equilibrium stress and strain distributions obtained from the stress analyses of the viscoelastic slab can be compared with those of an elastic slab whose properties correspond to the long-term properties of the viscoelastic material. The extensional relaxation function for the viscoelastic material is

$$E_R(t) = 69\left(1 - 0.901001\left(1 - e^{-t/0.99}\right)\right) \text{ MPa}$$
$$\text{or} \quad E_R(t) = 10000\left(1 - 0.901001\left(1 - e^{-t/0.99}\right)\right) \text{ psi.}$$

The long-term Young's modulus, E_L, is $E_R(t \to \infty)$ and is 6.83 MPa (989.99 psi). The long-term Poisson's ratio can be calculated from the long-term Young's modulus and the bulk modulus,

$$\nu_L = \frac{3K - E_L}{6K} = 0.4984.$$

The equilibrium values of ε_{xx} and σ_{yy} are obtained using linear elasticity

$$\varepsilon_{xx} = \frac{1}{E_L}\left(\sigma_{xx} - \nu_L\sigma_{yy} - \nu_L\sigma_{zz}\right) + \alpha\theta$$
$$\varepsilon_{yy} = \frac{1}{E_L}\left(\sigma_{yy} - \nu_L\sigma_{zz} - \nu_L\sigma_{xx}\right) + \alpha\theta.$$

By symmetry $\sigma_{yy} = \sigma_{zz}$, and by the assumptions of plane strain $\varepsilon_{yy} = \varepsilon_{zz} = 0$. The slab is unrestrained in the X-direction, so $\sigma_{xx} = 0$. These conditions result in stress and strain distributions that follow the temperature distribution,

$$\sigma_{yy} = -\frac{E_L}{1-\nu_L}\alpha\theta \qquad \varepsilon_{xx} = \frac{1+\nu_L}{1-\nu_L}\alpha\theta.$$

At steady state the temperature is 100° throughout the slab, so the final stress and strain are

$$\sigma_{yy} = -0.0136\,\text{MPa} = -1.97\text{ psi} \qquad \varepsilon_{xx} = 2.99 \times 10^{-3},$$

which agree with the steady-state values obtained in the analyses.

Input files

Abaqus/Standard input files

viscoslabthermload_heat.inp	Heat transfer analysis.
viscoslabthermload_smallstrain.inp	Small-strain analysis of the viscoelastic slab with the temperature-time shift included.
viscoslabthermload_largestrain.inp	Equivalent large-strain analysis.
viscoslabthermload_usr_utrs.inp	Stress analysis making use of user subroutine *UTRS* to define the `WLF` shift function. The solution using user subroutine *UTRS* is identical to that obtained from the job in viscoslabthermload_smallstrain.inp.
viscoslabthermload_usr_utrs.f	User subroutine *UTRS* used in conjunction with viscoslabthermload_usr_utrs.inp.
viscoslabthermload_postoutput.inp	**POST OUTPUT* analysis.

Abaqus/Explicit input files

viscoslabthermload_x_cpe3t.inp	Small-strain analysis of the viscoelastic slab with the temperature-time shift included; CPE3T elements.
viscoslabthermload_x_cpe4rt.inp	Small-strain analysis of the viscoelastic slab with the temperature-time shift included; CPE4RT elements.

viscoslabthermload_usr_cpe4rt.inp	Stress analysis making use of user subroutine *VUTRS* to define the `WLF` shift function. The solution using user subroutine *VUTRS* is identical to that obtained from the job in viscoslabthermload_x_cpe4rt.inp.
viscoslabthermload_usr_cpe4rt.f	User subroutine *VUTRS* used in conjunction with viscoslabthermload_usr_cpe4rt.inp.
viscoslabthermload_xh_cpe3t.inp	Large-strain analysis of the viscoelastic slab with the temperature-time shift included; CPE3T elements.
viscoslabthermload_xh_cpe4rt.inp	Large-strain analysis of the viscoelastic slab with the temperature-time shift included; CPE4RT elements.

To run the stress analyses without the shift, simply remove the **TRS* option and the one data line that follows it from *viscoslabthermload_smallstrain.inp*, *viscoslabthermload_largestrain.inp*, *viscoslabthermload_x_cpe3t.inp*, *viscoslabthermload_x_cpe4rt.inp*, *viscoslabthermload_xh_cpe3t.inp*, and *viscoslabthermload_xh_cpe4rt.inp*.

References

- Carslaw, H. S., and J. C. Yeager, Conduction of Heat in Solids, Clarendon Press, Oxford, 1959.
- Collingwood, G. A., E. B. Becker, and T. Miller, User's Manual for the TEXVISC Computer Program, Morton Thiokol, Inc., Document Numbers U-85-4550A and U-85-4550B, 1985.

Figures

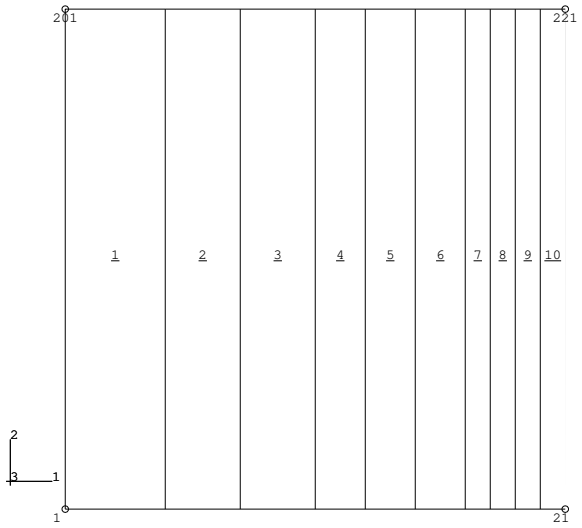


Figure 1: Finite element model of viscoelastic slab (Abaqus/Standard).

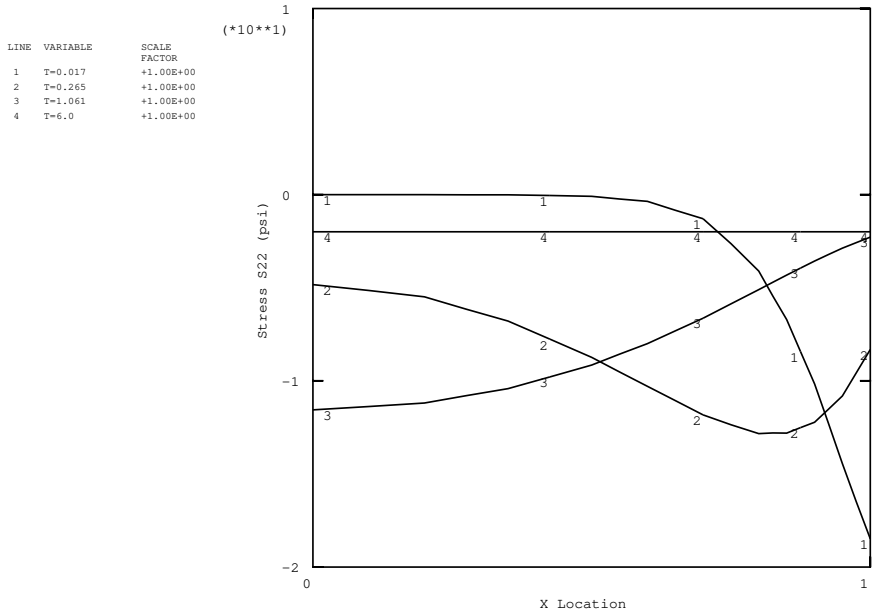


Figure 2: Stresses through the thickness of the slab at various times (Abaqus/Standard).

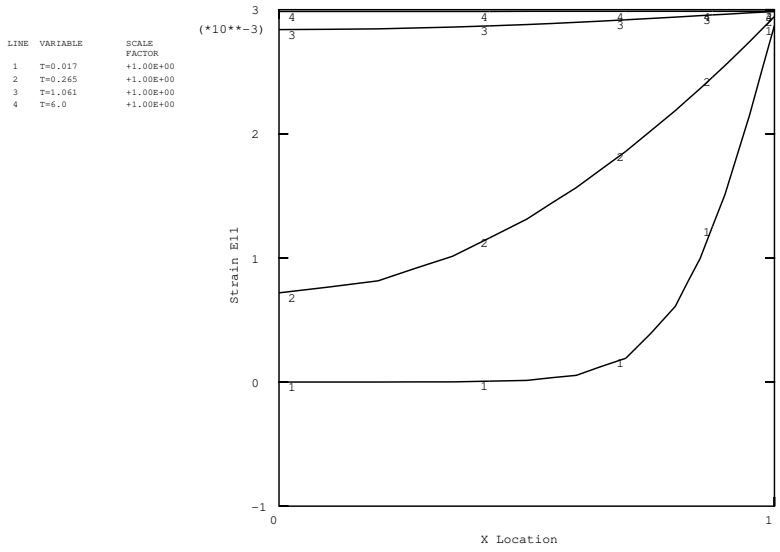


Figure 3: Strains through the thickness of the slab at various times (Abaqus/Standard).

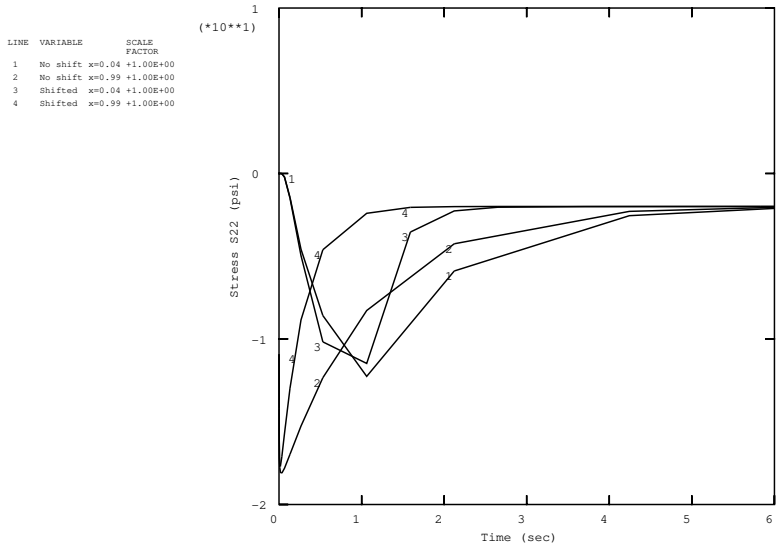


Figure 4: Comparison of stress time histories with and without temperature-time shift (Abaqus/Standard).

Uniform strain, viscoplastic truss

Products: Abaqus/Standard

Overview

This example verifies the viscoplastic constitutive model used in Abaqus for the response of elastic-plastic materials at high strain rates.

The problem is a one degree of freedom system, consisting of a uniformly strained truss and a point mass (see *Figure 1*). The truss is assumed to be made of an elastic, viscoplastic material (see *Metal plasticity*). The system is excited by an initial velocity imposed while the truss is unstrained, the velocity being sufficient to cause quite extensive yield. Due to the simplicity of the system, the exact response is easily developed, thus providing a basic check case for the implementation of this constitutive model in Abaqus.

The problem has been analyzed for a rigid viscoplastic material model by Symonds and Ting (1964). In that paper the authors provide the criteria that ensure that the rigid, viscoplastic analysis will give an accurate prediction of the final strain. Essentially, the requirement is that the initial velocity be large enough such that the initial kinetic energy exceeds the maximum strain energy that can be stored in the rod. The parameters chosen for the case analyzed here satisfy this criterion comfortably.

Problem description

The one degree of freedom model is shown in *Figure 1*. The dimensions are:

Truss length	25.4 mm (1 in)
Truss cross-sectional area	64.52 mm^2 (0.1 in^2)

The truss is assumed to have the following material properties:

Young's modulus	207 GPa (30×10^6 lb/in^2)
Static yield stress (σ^0)	276 MPa (40×10^3 lb/in^2) (no strain hardening)
Mass density	7827 kg/m^3 (7.324×10^{-4} lb-s^2/in^4)
Viscoplastic parameters	$\dot{\bar{\varepsilon}}^{pl} = D\left(\bar{\sigma}/\sigma^0 - 1\right)^p$; $D =$ 40 per s, $p = 5$

The mass on the end of the truss is 5.254 kg (0.030 lb-s^2/in), and the initial velocity is 5.08 m/s (200 in/s).

The truss is modeled with a single truss element. The same case is also modeled using a plane stress element as an additional verification exercise. The lumped mass is modeled with a mass element. Plane stress is usually the most difficult case for implementation of plasticity models of this type, because the constitutive calculations must be done in a multidimensional stress space with the constraint that one direct stress component is zero.

Results and discussion

The exact equation of motion for this system in terms of the stress, σ, in the rod, is

$$\ddot{\sigma} + \gamma \frac{EDp}{\sigma^o}\left(\frac{|\sigma|}{\sigma^o} - 1\right)^{p-1}\dot{\sigma} + \frac{AE}{mL}\sigma = 0,$$

where E is Young's modulus, D and p are the viscoplastic parameters, σ° is the static yield stress in uniaxial tension, A is the cross-sectional area of the rod, L is the length of the rod, m is the attached mass, and γ is 0 if $|\sigma| \le \sigma^\circ$; γ is 1 if $|\sigma| > \sigma^\circ$.

The initial conditions on this equation are $\sigma(0) = \dot{\sigma}(0) = EV^\circ/L$, where V° is the initial velocity given to the mass.

An accurate solution to this equation can be developed by standard numerical methods. In this case the fourth-order Runge-Kutta algorithm has been used, giving the stress-time plot shown in *Figure 2*. The rigid, viscoplastic analysis of Symonds and Ting (1964) and the solution obtained by Abaqus are also shown in the figure. The Abaqus plane stress results are identical to those obtained with the truss element. Fixed time stepping is used so that the numerical solution can be compared continuously with the exact solution. Three different time increment sizes are used to obtain the response details. These are 2.5 μs, 10 μs, and 25 μs for the time periods 0–50 μs, 50 μs–0.1 ms, and 0.1–5 ms, respectively.

Figure 2 shows that the numerical solution accurately predicts the exact solution until the very end of the plot, when a small phase error begins to appear in the numerical solution.

It is interesting to see the form of the solution: in the first half-cycle the stress increases very rapidly to about 2.38 times the static yield, then unloads, slowly at first, then more rapidly. During this first half-cycle about 99% of the initial kinetic energy is dissipated in plastic work (this result is available by printing a summary of the total energy content to the data file). The remaining response analyzed (up to 5 ms) continues to have some dissipation at peak stress in each half-cycle. Since there is so little energy left in the system compared to the rod's ability to store strain energy, the resulting damping of the response amplitude in each cycle is not large. The rigid, viscoplastic solution of Symonds and Ting (1964) follows the elastic, viscoplastic curves accurately because the initial energy is so large compared to the strain energy that can be stored. The rigid, viscoplastic solution estimates a final strain of 6.75% in the rod. The numerical analysis shows a total plastic strain of 6.59% at 0.7 ms, with still some slight increase of this value during each half-cycle, as the peak stress continues to exceed static yield.

Input files

viscoplastictruss_t3d2.inp	Truss element problem.
viscoplastictruss_cps4.inp	Plane stress model with the overstress power law entered as a piecewise linear function.

References

- Symonds, P. S., and T. C. T. Ting, “Longitudinal Impact on Visco-Plastic Rods—Approximate Methods and Comparisons,” *Journal of Applied Mechanics*, vol. 31, pp. 611–620, 1964.

Figures

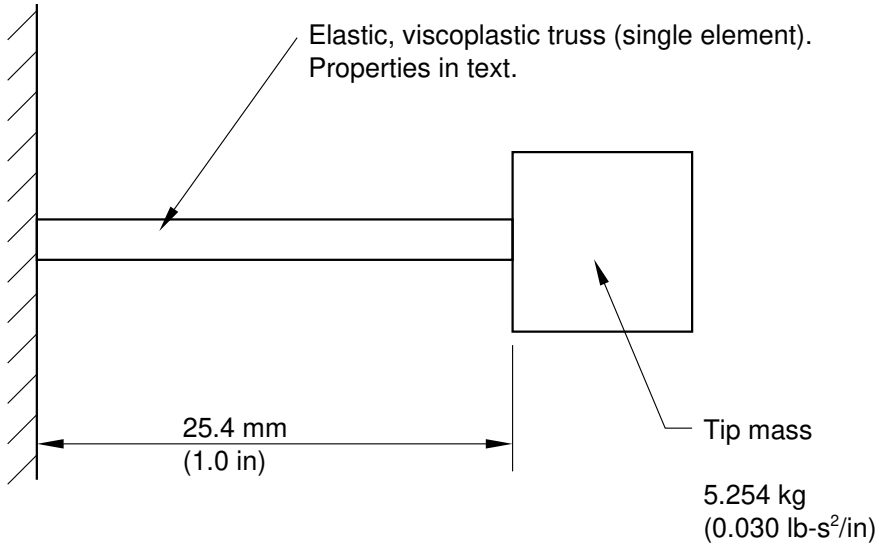


Figure 1: One degree of freedom elastic, viscoplastic verification problem.

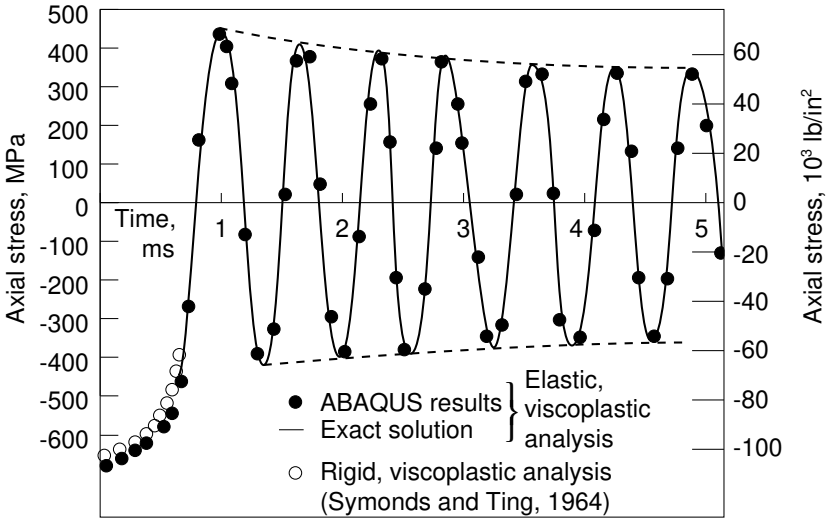


Figure 2: Stress-time history of the uniform-strain viscoplastic truss.

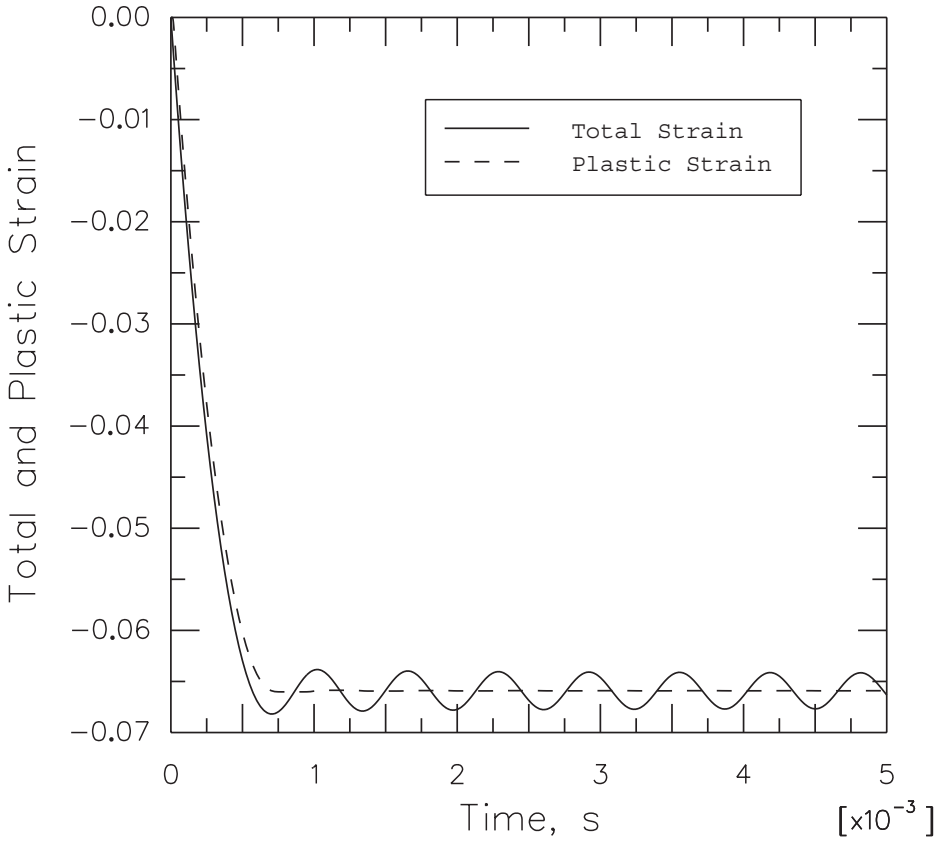


Figure 3: Total strain and plastic strain versus time.

Fitting of rubber test data

Products: Abaqus/Standard

Overview

This example illustrates how elastomeric (rubber) materials are modeled in Abaqus using the hyperelasticity material model.

Several hyperelastic strain energy potentials are available—the polynomial model (including its particular cases, such as the reduced polynomial, neo-Hookean, Mooney-Rivlin, and Yeoh forms), the Ogden form, the Arruda-Boyce form, the Van der Waals form (which is also known as the Kilian model), and the Marlow form.

Problem description

The form of the polynomial strain energy potential is

$$U = \sum_{i+j=1}^{N} C_{ij}\left(\overline{I}_1 - 3\right)^i \left(\overline{I}_2 - 3\right)^j + \sum_{i=1}^{N} \frac{1}{D_i}(J_{el} - 1)^{2i},$$

where U is the strain energy potential; J_{el} is the elastic volume ratio; $\overline{I}_1$ and $\overline{I}_2$ are the first and second invariants of the deviatoric strain; and N, C_{ij}, and D_i are material constants. C_{ij} describes the shear behavior of the material, and D_i introduces compressibility.

Particular forms of the polynomial model can be obtained by setting specific coefficients to zero. If all C_{ij} with $j \neq 0$ are set to zero, the reduced polynomial form is obtained:

$$U = \sum_{i=1}^{N} C_{i0}\left(\overline{I}_1 - 3\right)^i + \sum_{i=1}^{N} \frac{1}{D_i}(J_{el} - 1)^{2i}.$$

If in addition N is set to 3, the Yeoh model is obtained. For $N = 1$, the reduced polynomial model reduces to the neo-Hookean model. If in the (general) polynomial model N is set to 1, the Mooney-Rivlin form is obtained.

The form of the Ogden strain energy potential is

$$U = \sum_{i=1}^{N} \frac{2\mu_i}{\alpha_i^2}\left(\overline{\lambda}_1^{\alpha_i} + \overline{\lambda}_2^{\alpha_i} + \overline{\lambda}_3^{\alpha_i} - 3\right) + \sum_{i=1}^{N} \frac{1}{D_i}(J_{el} - 1)^{2i},$$

where $\overline{\lambda}_i = J^{-\frac{1}{3}}\lambda_i$, λ_i are the principal stretches and J is the volume ratio. The constants μ_i and α_i describe the shear behavior of the material, and D_i, the compressibility.

The Arruda-Boyce model—also known as the eight-chain model—has the form

$$U = \mu \sum_{i=1}^{5} \frac{C_i}{\lambda_m^{2i-2}}\left(\overline{I}_1^i - 3^i\right) + \frac{1}{D}\left(\frac{J_{el}^2 - 1}{2} - \ln J_{el}\right),$$

where

$$C_1 = \frac{1}{2}, \qquad C_2 = \frac{1}{20}, \qquad C_3 = \frac{11}{1050}, \qquad C_4 = \frac{19}{7050}, \quad \text{and} \quad C_5 = \frac{519}{673750}.$$

The shear behavior is described by the parameters μ and λ_m, while D governs the compressibility.

The Van der Waals strain energy potential has the form

$$U = \mu \left\{ -\left(\lambda_m^2 - 3\right) \left[\ln(1-\eta) + \eta\right] - \frac{2}{3} a \left(\frac{\tilde{I}-3}{2}\right)^{\frac{3}{2}} \right\} + \frac{1}{D} \left(\frac{J_{el}^2 - 1}{2} - \ln J_{el} \right),$$

where

$$\tilde{I} = (1-\beta)\,\bar{I}_1 + \beta \bar{I}_2 \quad \text{and} \quad \eta = \sqrt{\frac{\tilde{I}-3}{\lambda_m^2 - 3}}.$$

The parameters μ, λ_m, a, and β describe the deviatoric behavior, while the coefficient D controls the compressibility.

The Marlow strain energy potential has the form

$$U = U_{dev}\left(\bar{I}_1\right) + U_{vol}\left(J_{el}\right),$$

where U is the strain energy per unit of reference volume, with U_{dev} as its deviatoric part and U_{vol} as its volumetric part. The deviatoric part of the potential is defined by providing uniaxial, equibiaxial, or planar test data; while the volumetric part is defined by providing volumetric test data, defining the Poisson's ratio, or specifying the lateral strains on the uniaxial, equibiaxial, or planar test data.

Details of the formulation are given in *Hyperelastic Behavior of Rubberlike Materials*, *Hyperelastic material behavior*, and *Fitting of hyperelastic and hyperfoam constants*.

The hyperelastic constants C_{ij}, D_i (polynomial form); μ_i, α_i, D_i (Ogden form); μ, λ_m, D (Arruda-Boyce form); and $\mu, \lambda_m, a, \beta, D$ (Van der Waals form) are determined from the material test data. This example illustrates the steps in doing so.

Specification of material data

The following steps are needed to specify the material data in an analysis:

- Perform different types of tests to measure stress-strain data.
- Fit hyperelastic constants to the test data.
- Check correlation between the numerical results from hyperelastic model and test data.
- If satisfactory, proceed with finite element analysis; otherwise, perform corrective measures, and try the fitting procedure again.

When evaluating the curve fits, the following criteria should be used:

- If uniaxial, biaxial, and planar data are available, how well do the calculated curves approximate measured data?
- If only limited test data are available, how realistic is the prediction of deformation modes other than those measured? In the absence of material data this would require some engineering judgement. In this example we simulate this situation by restricting the curve fit to uniaxial tension data even though all data are available.
- Is the Drucker stability criterion satisfied?

Experimental data of Treloar

For this example experimental test data measured by Treloar (1944) are used. The stress-strain data were measured for 8% sulfur rubber, which exhibits highly reversible behavior. Nevertheless, specimens were conditioned by prestraining to induce any permanent deformation before actual measurements were performed. Some slight

hysteresis was observed at higher strains. The hyperelasticity model assumes ideal elasticity. Separate viscoelastic material data can be defined with viscoelastic behavior to model the hysteresis effects.

With the assumption of full incompressibility, $J = \lambda_1\lambda_2\lambda_3 = 1$. The deformation modes for the tests described in terms of the principal stretches λ_i are:

- Uniaxial tension: $\lambda_1 = \lambda_U\ , \quad \lambda_2 = \lambda_3 = \lambda_U^{-\frac{1}{2}}$
- Equibiaxial tension: $\lambda_1 = \lambda_2 = \lambda_B\ , \quad \lambda_3 = \lambda_B^{-2}$
- Planar tension (pure shear): $\lambda_1 = \lambda_S\ , \quad \lambda_2 = 1\ , \quad \lambda_3 = \lambda_S^{-1}$

The principal stretch λ_i is related to the principal nominal strain ϵ_i through $\lambda_i = 1 + \epsilon_i$. The nominal stress–nominal strain curves are shown in *Figure 1*. The curves are quite nonlinear and extend into fairly large strains: the maximum uniaxial tensile strain is 6.64, the maximum equibiaxial tensile strain is 3.45, and the maximum planar tensile strain is 4.06. The stress has units of kgf/cm^2 (1 kgf/cm^2=0.0981 MPa). These units are consistent with the units Treloar used in presenting his experimental results.

Fitting procedures

In Abaqus the test data are specified as nominal stress–nominal strain data pairs using uniaxial test data, biaxial test data, and planar test data for hyperelastic behavior with material constants computed by Abaqus from the test data: shear constants C_{ij} (polynomial forms); μ_i, α_i (Ogden form); μ, λ_m (Arruda-Boyce form); or μ, λ_m, a, β (Van der Waals form). If required, pressure-volume ratio data can be specified using volumetric test data to determine the compressibility constants D_i (polynomial and Ogden forms) or D (Arruda-Boyce and Van der Waals forms).

For each stress-strain data pair Abaqus generates an equation for the stress in terms of the strain invariants or stretches and the unknown hyperelastic constants, assuming incompressibility. For example, in the uniaxial deformation case the nominal stress T_U is

$$T_U = \frac{\partial U}{\partial \lambda_U} = 2\left(1 - \lambda_U^{-3}\right)\left(\lambda_U \frac{\partial U}{\partial \bar{I}_1} + \frac{\partial U}{\partial \bar{I}_2}\right),$$

where U is the strain energy potential, λ_U is the stretch in the uniaxial direction, and $\bar{I}_i$ are the deviatoric strain invariants. If the Mooney-Rivlin form (N=1) of the polynomial strain energy potential is used, then

$$U = C_{10}\left(\bar{I}_1 - 3\right) + C_{01}\left(\bar{I}_2 - 3\right)$$

and, thus,

$$T_U = \frac{\partial U}{\partial \lambda_U} = 2\left(1 - \lambda_U^{-3}\right)\left(C_{10}\lambda_U + C_{01}\right).$$

Hyperelastic Behavior of Rubberlike Materials discusses the different stress expressions used for the different deformation modes. Since the number of stress equations will be greater than the number of unknown constants, a least-squares fit must be performed to determine the hyperelastic constants. For the n stress-strain pairs that make up the test data, the following error measure E is minimized:

$$E = \sum_{i=1}^{n}\left(1 - T_i^{\text{th}}/T_i^{\text{test}}\right)^2,$$

where T_i^{test} is a stress value from the test data and T_i^{th} is a theoretical stress expression described above.

The polynomial potential is linear in the coefficients C_{ij}. Therefore, a linear least-squares procedure can be used. The Ogden potential is linear in the coefficients μ_i but strongly nonlinear in the exponents α_i. Similarly, the Arruda-Boyce and Van der Waals models are linear in the parameter μ but nonlinear in the other shear coefficients. A nonlinear least-squares procedure similar to that of Twizell and Ogden (1983) is used in Abaqus to determine the material parameters simultaneously.

Upon deriving a set of constants, Abaqus performs material stability checks along the primary deformation modes using the Drucker stability criterion:

$$d\boldsymbol{\sigma} : d\boldsymbol{\varepsilon} > 0 \qquad \Longrightarrow \qquad d\boldsymbol{\varepsilon} \cdot \mathbf{D} : d\boldsymbol{\varepsilon} > 0,$$

where $d\boldsymbol{\sigma}$ is the change in stress due to an infinitesimal change in strain $d\boldsymbol{\varepsilon}$ and $\mathbf{D}$ is the tangential material stiffness. For the stability criterion to be satisfied, $\mathbf{D}$ must be positive-definite. The analysis input file processor will issue warning messages defining the strain states at which $\mathbf{D}$ becomes singular with the potential for unstable material behavior. The deformation modes covered are the tensile and compressive cases of the uniaxial, equibiaxial, and planar modes.

Fitting case 1—using all three types of test data

The following cases are analyzed:

- Polynomial form with $N = 1$ (Mooney-Rivlin form) and $N = 2$.
- Reduced polynomial form with $N = 1$ (neo-Hookean form) and $N = 3$ (Yeoh form).
- Ogden form with $N = 2$ and $N = 3$.
- Arruda-Boyce form.
- Van der Waals form.

All three types of test data are used simultaneously in fitting the hyperelastic constants. To evaluate the hyperelastic behavior in Abaqus, a single continuum, reduced-integration, hybrid C3D8RH element with unit dimensions is subjected to uniaxial tension, equibiaxial tension, and planar tension. The deformation modes are illustrated in *Figure 2*. The Abaqus nominal stress–nominal strain results are compared with the test data in *Figure 3* to *Figure 8*.

For the polynomial potential the case $N = 1$ (Mooney-Rivlin) gives a reasonable fit at low strains but is unable to reproduce the stiffening response of the rubber at higher strains. The case $N = 2$ provides the higher-order terms to enable closer correlation to the test data at all strain levels. Similar observations apply to the reduced polynomial with $N = 1$ (neo-Hookean) and $N = 3$ (Yeoh); the neo-Hookean model offers only a linear dependence of the first invariant and, thus, fails to provide an accurate representation of the upturn. In contrast, the three-term reduced polynomial (Yeoh) provides a more accurate representation than the full polynomial with $N = 2$, which has five coefficients. In addition, the Yeoh model does not exhibit any instabilities when fitting the Treloar test data.

For the Ogden potential both the cases $N = 2$ and $N = 3$ give very close fits to all three deformation modes, with the case $N = 3$ providing the best correlation among all fits.

The Arruda-Boyce model also gives a satisfactory fit. In the uniaxial case the upturn is not as steep as in the experiment; in the middle stretch range the stresses are overestimated. Other curve fits have been reported in the literature; for example, Boyce (1996) reports $\mu = 0.27$ MPa $= 2.75$ kgf/cm^2 and $\lambda_m = 5.15$. These differ from our values, $\mu = 3.28$ kgf/cm^2 and $\lambda_m = 5.24$. The differences can be attributed to the fact that the relative error in stress is minimized. Another potential source of discrepancies could be different spacing of the Treloar test data.

The Van der Waals model gives a better fit than the Arruda-Boyce model, although not as good as the Ogden model. All stretch ranges of the stress-strain curve are fitted with high accuracy. Our fit compares favorably with those reported in the literature (Vilgis and Kilian, 1984); however, we use a more refined model since we take into account a slight dependence on the second invariant.

Fitting case 2—using uniaxial tension data only

Commonly, not all three or even two types of test data are available. *Figure 9* to *Figure 15* show the consequences of using different hyperelastic forms with only the uniaxial tension data. The following cases are analyzed:

- Polynomial form with $N = 1$ (Mooney-Rivlin form) and $N = 2$.
- Reduced polynomial form with $N = 1$ (neo-Hookean form) and $N = 3$ (Yeoh form).
- Ogden form with $N = 2$ and $N = 3$.
- Arruda-Boyce form.
- Van der Waals form with $\beta = 0$.
- Marlow form.

Except for the polynomial model with $N = 1$ (both the neo-Hookean and Mooney-Rivlin forms), the uniaxial tension results correlate very closely to the uniaxial test data. This is expected since the hyperelastic constants are fitted using the uniaxial data. However, the (general) polynomial and Ogden models show large differences between the numerical and test data for the equibiaxial tension and planar tension cases.

For the case with polynomial $N = 1$ (Mooney-Rivlin), instabilities in the equibiaxial and planar tension cases occur immediately. For the case with polynomial $N = 2$, the stress increases very rapidly at higher strains.

For the Ogden potential the case $N = 3$ starts diverging significantly at moderate strains but not as severely as the case of polynomial $N = 2$. Notably, the Ogden $N = 2$ case still gives reasonably close fits even at higher strains. Experience with additional sets of test data indicates that it may be possible to generalize these observations.

By omitting the dependence of the polynomial model on the second invariant, a much better prediction of the unmeasured stress states is obtained. This observation is in agreement with results reported in the literature; see Kaliske and Rothert (1997) or Yeoh (1993). In particular, the neo-Hookean model provides good first-order approximations to all stress states even though the coefficient C_{10} was measured from only a uniaxial test, whereas in our example the Mooney-Rivlin model is not even able to predict the qualitative tendencies correctly. The Yeoh model (or reduced polynomial, N=3) provides a good third-order approximation for all stress states without exhibiting any instabilities in the present case. Higher-order reduced polynomials, which are more likely to suffer from Drucker instability, are rarely needed, except, for example, when the stress-strain curve is "double-S-shaped."

The best fit to all three deformation modes, when the strain energy potential is derived from uniaxial data, is obtained with the Van der Waals, Arruda-Boyce, and Marlow models. If the test data in the small stretch range were more densely spaced and the S-shape were more pronounced, as is common for filled rubbers, the Van der Waals model is likely to show an even clearer superiority, since the additional parameters create enhanced flexibility in representing complex stress-strain curves.

Results and discussion

For Treloar's test data, when taking into account uniaxial, biaxial, and planar test data, the Ogden and Van der Waals forms give a closer fit than the polynomial forms. The Arruda-Boyce and Yeoh forms also provide an accurate representation. The (general) polynomial form exhibits some instabilities for $N = 2$ and provides only a first-order approximation for $N = 1$.

A completely different conclusion is reached when only limited test data are available. In this case the Van der Waals model (with $\beta = 0$) and the Arruda-Boyce model are clearly superior to the Ogden model. The polynomial model is significantly enhanced when the dependence on the second invariant is omitted. The Yeoh model gives a very good third-order representation even for the deformation modes that have not been incorporated in the curve fit. Similarly, the neo-Hookean model gives a good first-order approximation for all stress states even when the fit is based on only one deformation state.

The high quality of the Ogden fit, as opposed to the (general) polynomial, in the presence of test data for all three deformation modes can be explained by the Ogden potential's flexibility in conforming to test data—the exponents α_i can assume any real values, whereas the polynomial potential can only have integer exponents.

However, for accurate analyses with the most general models—Ogden and (general) polynomial—it is important that multiple and independent types of test data be used in fitting the hyperelastic constants if the actual elastomeric model to be analyzed will experience general stress-strain states.

According to Yeoh (1993), we suggest that the dependence on the second invariant be omitted when incomplete or limited material data are available; the curve fit for the Van der Waals model should be performed with $\beta = 0$, and the reduced polynomial form should be preferred over the (general) polynomial model. The Arruda-Boyce model is, by definition, independent of the second invariant. It is not possible to suppress the dependence on the second invariant for the Ogden model.

Figure 15 to *Figure 17* show the results for the Marlow model using different test data. It can be seen that the model can represent the material's behavior in the deformation mode for which test data are available exactly and have reasonable behavior in other modes of deformation.

Other considerations for achieving accurate and stable fits are discussed in *Hyperelastic Behavior of Rubberlike Materials*.

Input files

rubberfit_ogden_n3.inp	Treloar's test data and five static analysis steps composed of three deformation steps with two unloading steps in between the deformation steps. It is set up to use the Ogden model by specifying the OGDEN parameter in the **HYPERELASTIC* option. As an alternative procedure for postprocessing purposes, it may be more straightforward to run the three deformation modes in this example individually by using three separate input files with only a single (deformation) step each.
rubberfit_ogden_n2.inp	Ogden model with N=2.
rubberfit_mooneyrivlin.inp	Mooney-Rivlin model.
rubberfit_poly.inp	Polynomial model.
rubberfit_neohook.inp	Neo-Hookean model.
rubberfit_yeoh.inp	Yeoh model.
rubberfit_arrudaboyce.inp	Arruda-Boyce model.
rubberfit_vdwaal.inp	Van der Waals model.
rubberfit_ogden_n3_uni.inp	Ogden model with N=3, uniaxial test data only.
rubberfit_ogden_n2_uni.inp	Ogden model with N=2, uniaxial test data only.
rubberfit_mooneyrivlin_uni.inp	Mooney-Rivlin model, uniaxial test data only.
rubberfit_poly_uni.inp	Polynomial model, uniaxial test data only.
rubberfit_neohook_uni.inp	Neo-Hookean model, uniaxial test data only.
rubberfit_yeoh_uni.inp	Yeoh model, uniaxial test data only.

rubberfit_arrudaboyce_uni.inp	Arruda-Boyce model, uniaxial test data only.
rubberfit_vdwaal_uni.inp	Van der Waals model, uniaxial test data only.
rubberfit_marlow_uni.inp	Marlow model, uniaxial test data only.
rubberfit_marlow_bia.inp	Marlow model, biaxial test data only.
rubberfit_marlow_pla.inp	Marlow model, planar test data only.

References

- Boyce, M. C., “Direct Comparison of the Gent and the Arruda-Boyce Constitutive Models for Rubber Elasticity,” *Rubber Chemistry and Technology*, vol. 69, pp. 781–785, 1996.
- Kaliske, M., and H. Rothert, “On the Finite Element Implementation of Rubber-like Material at Finite Strains,” *Engineering Computations*, vol. 14, no. 2, pp. 216–232, 1997.
- Treloar, L. R. G., “Stress-Strain Data for Vulcanised Rubber under Various Types of Deformation,” *Transactions of the Faraday Society*, vol. 40, pp. 59–70, 1940.
- Twizell, E. H., and R. W. Ogden, “Non-Linear Optimization of the Material Constants in Ogden's Stress-Deformation Function for Incompressible Isotropic Elastic Materials,” *J. Austral. Math. Soc. Ser. B*, vol. 24, pp. 424–434, 1983.
- Yeoh, O. H., “Some Forms of the Strain Energy Function for Rubber,” *Rubber Chemistry and Technology*, vol. 66, pp. 754–771, 1993.
- Vilgis, Th., and H. G. Kilian, “The Van der Waals-network—A Phenomenological Approach to Dense Networks,” *Polymer*, vol. 25, pp. 71–74, January, 1984.

Figures

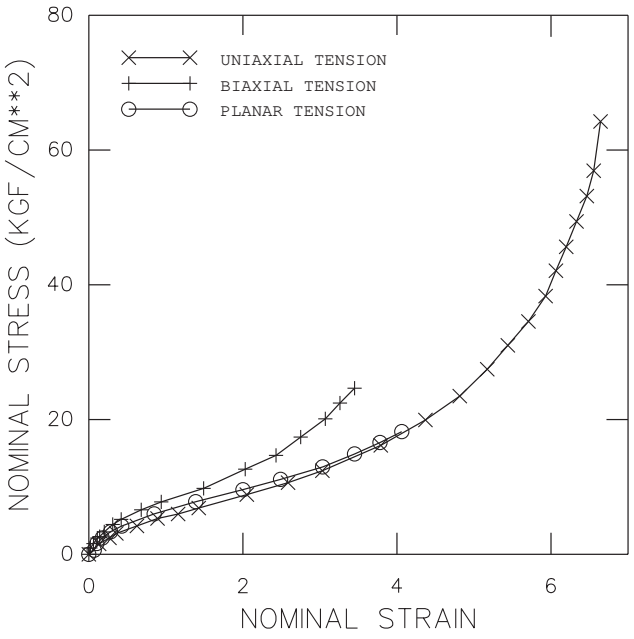


Figure 1: Treloar's experimental data.

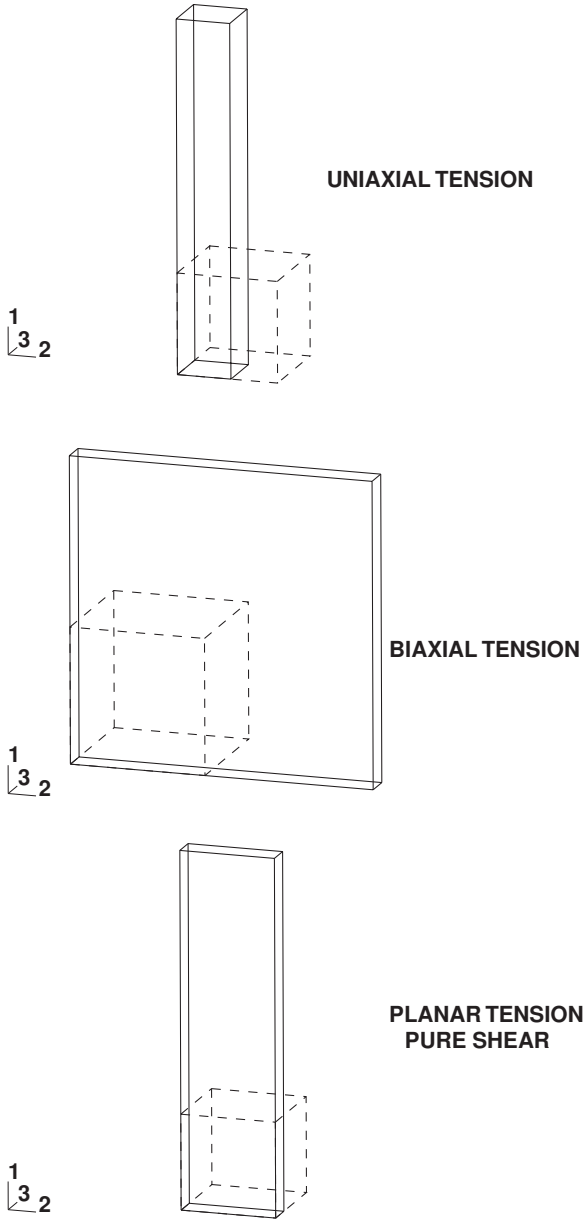


Figure 2: Three deformation modes.

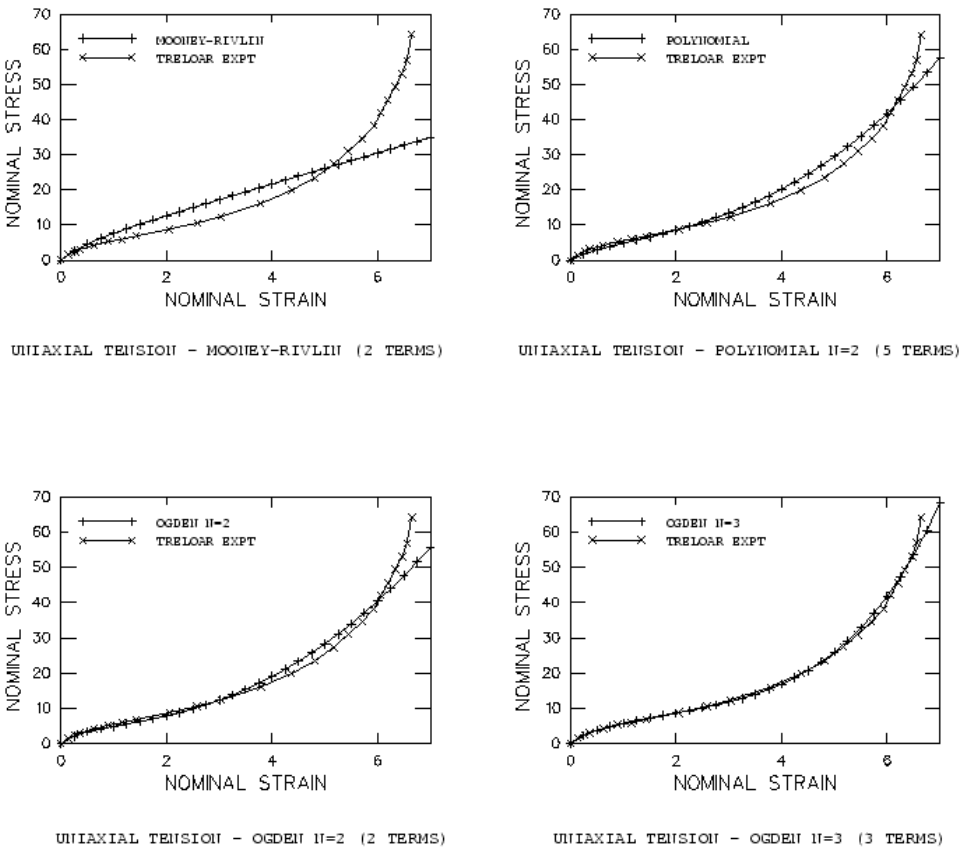


Figure 3: Uniaxial tension results using three types of test data (polynomial and Ogden models).

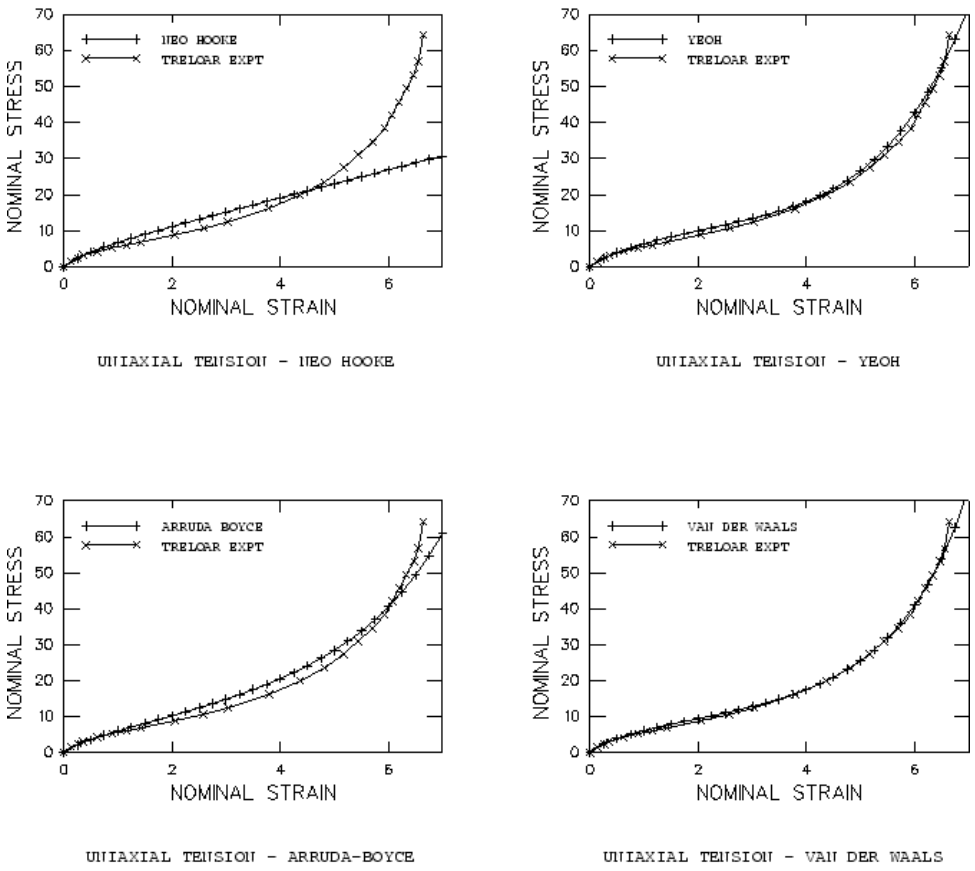


Figure 4: Uniaxial tension results using three types of test data (neo-Hookean, Yeoh, Arruda-Boyce, and Van der Waals models).

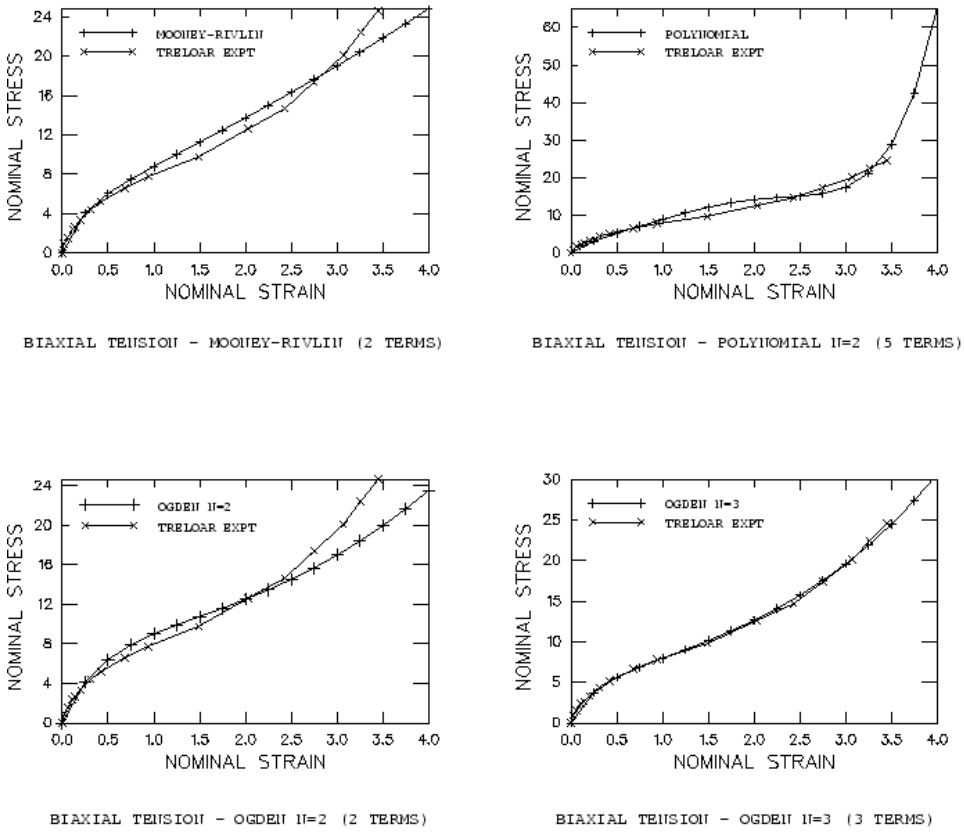


Figure 5: Equibiaxial tension results using three types of test data (polynomial and Ogden models).

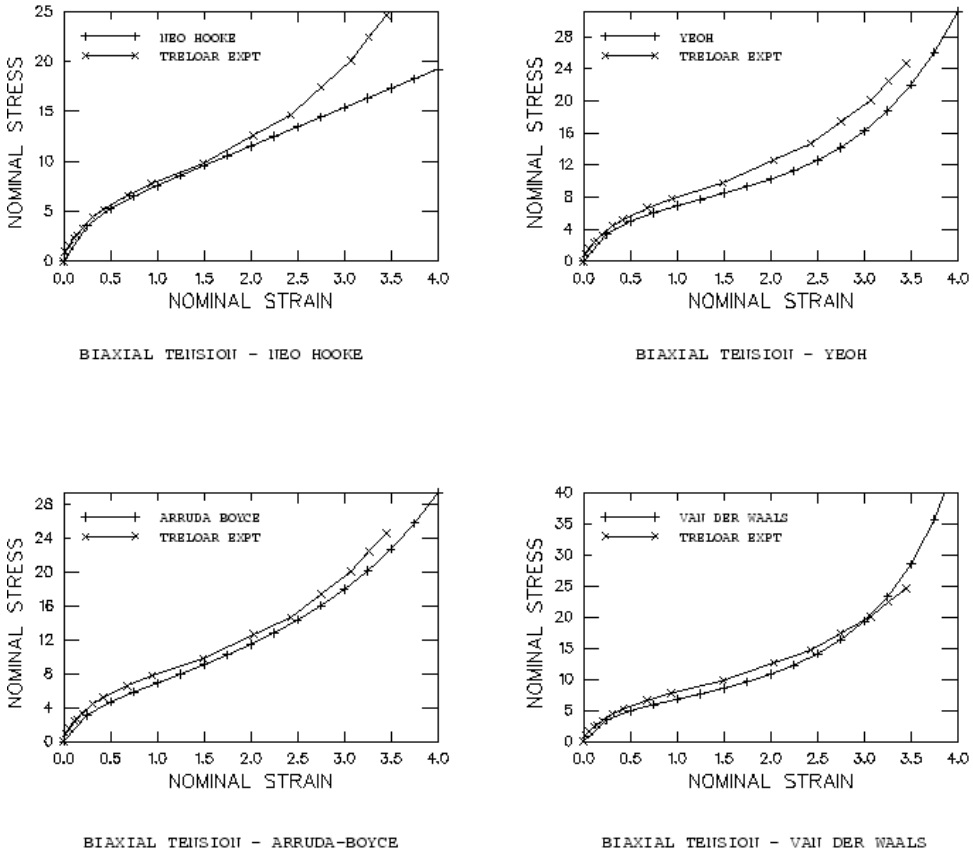
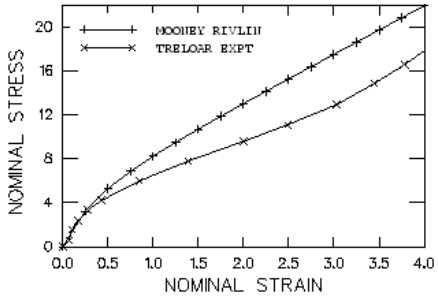
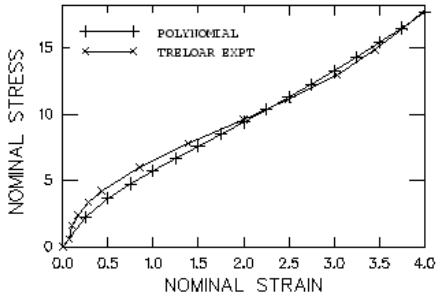


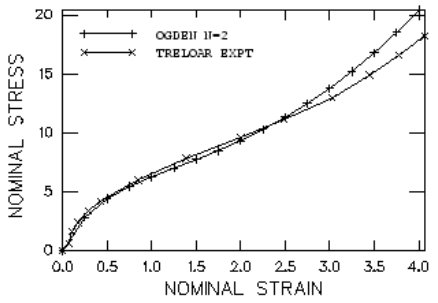
Figure 6: Equibiaxial tension results using three types of test data (neo-Hookean, Yeoh, Arruda-Boyce, and Van der Waals models).



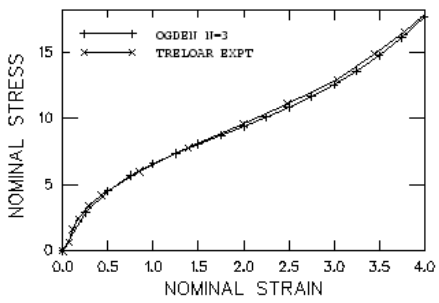
PLANAR TENSION - MOONEY-RIVLIN (2 TERMS)



PLANAR TENSION - POLYNOMIAL N=2 (5 TERMS)



PLANAR TENSION - OGDEN N=2 (2 TERMS)



PLANAR TENSION - OGDEN N=3 (3 TERMS)

Figure 7: Planar tension (pure shear) results using three types of test data (polynomial and Ogden models).

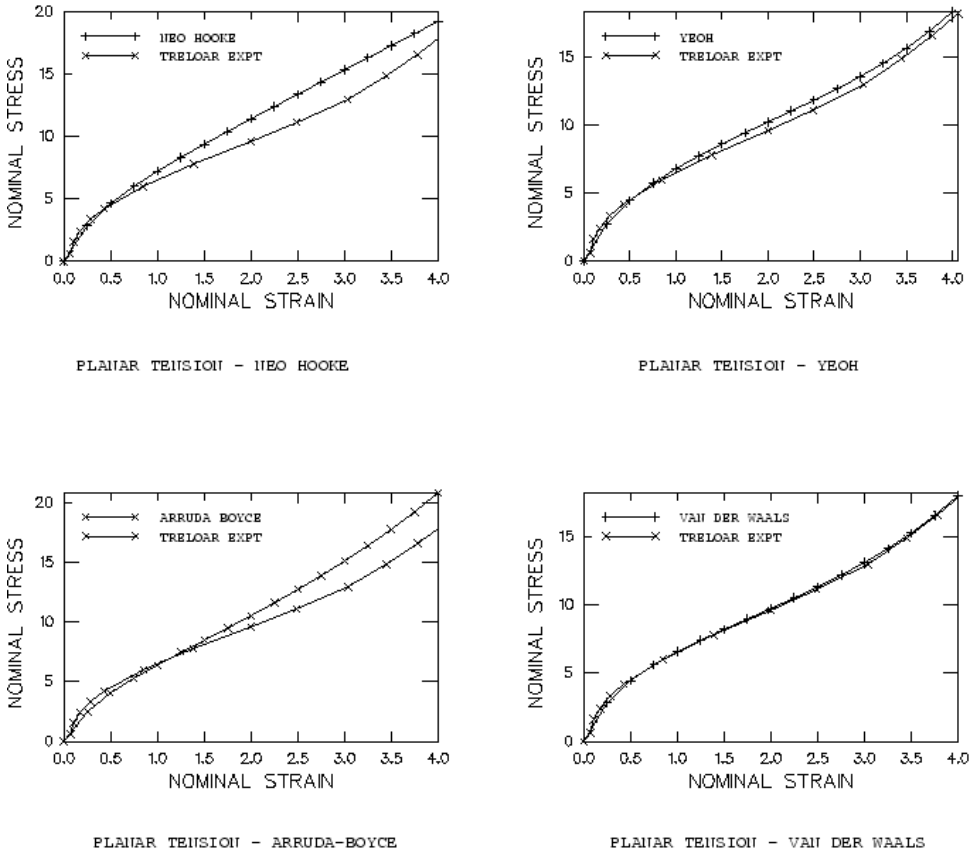


Figure 8: Planar tension (pure shear) results using three types of test data (neo-Hookean, Yeoh, Arruda-Boyce, and Van der Waals models).

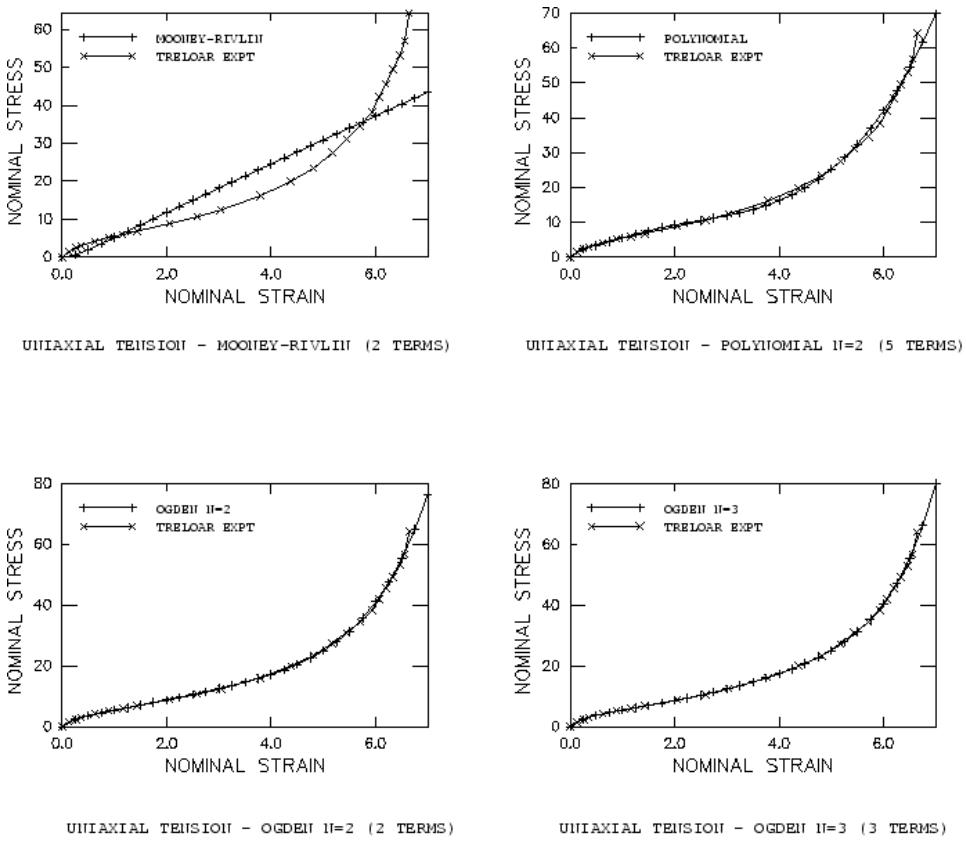


Figure 9: Uniaxial tension results using uniaxial tension test data only (polynomial and Ogden models).

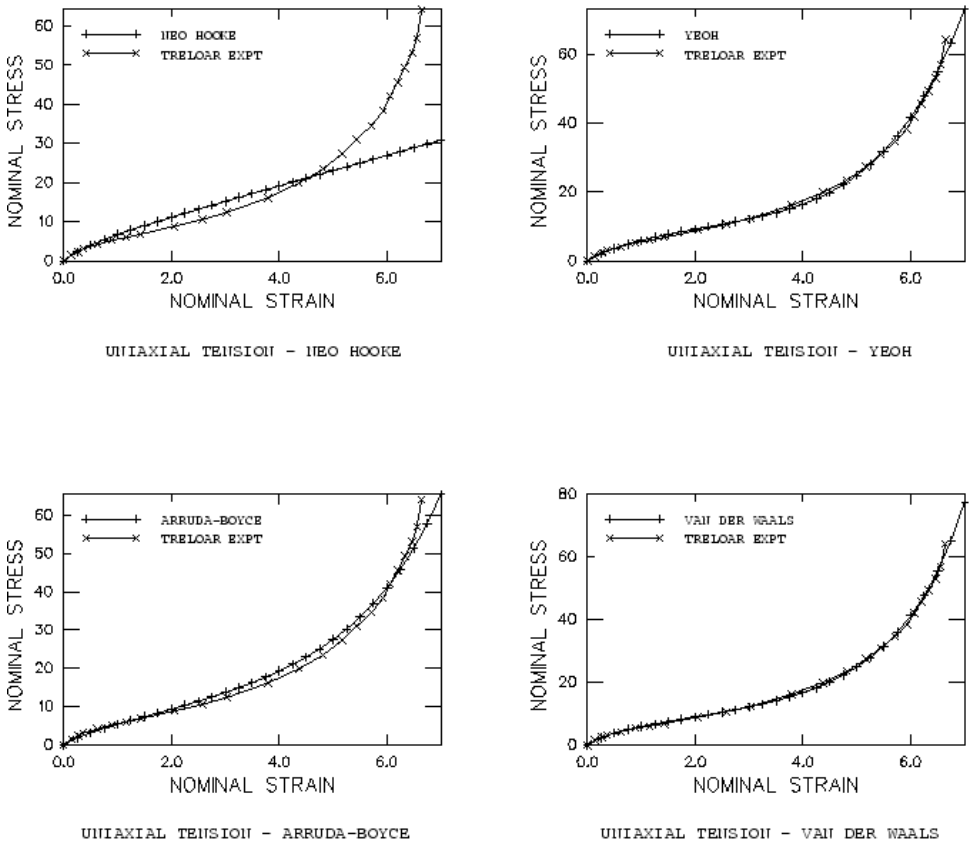
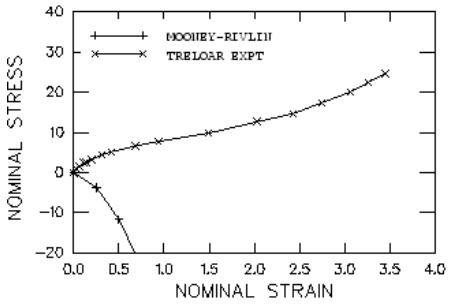
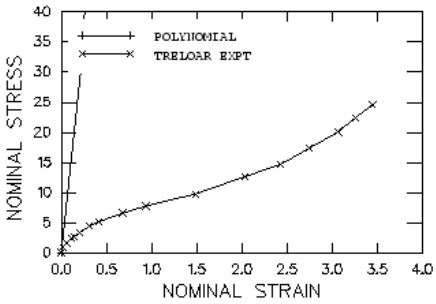


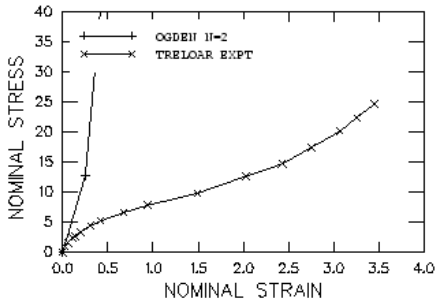
Figure 10: Uniaxial tension results using uniaxial tension test data only (neo-Hookean, Yeoh, Arruda-Boyce, and Van der Waals models).



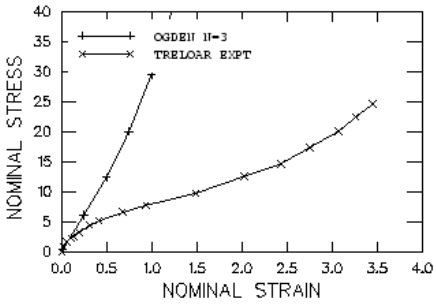
BIAXIAL TENSION - MOONEY-RIVLIN (2 TERMS)



BIAXIAL TENSION - POLYNOMIAL N=2 (5 TERMS)



BIAXIAL TENSION - OGDEN N=2 (2 TERMS)



BIAXIAL TENSION - OGDEN N=3 (3 TERMS)

Figure 11: Equibiaxial tension results using uniaxial tension test data only (polynomial and Ogden models).

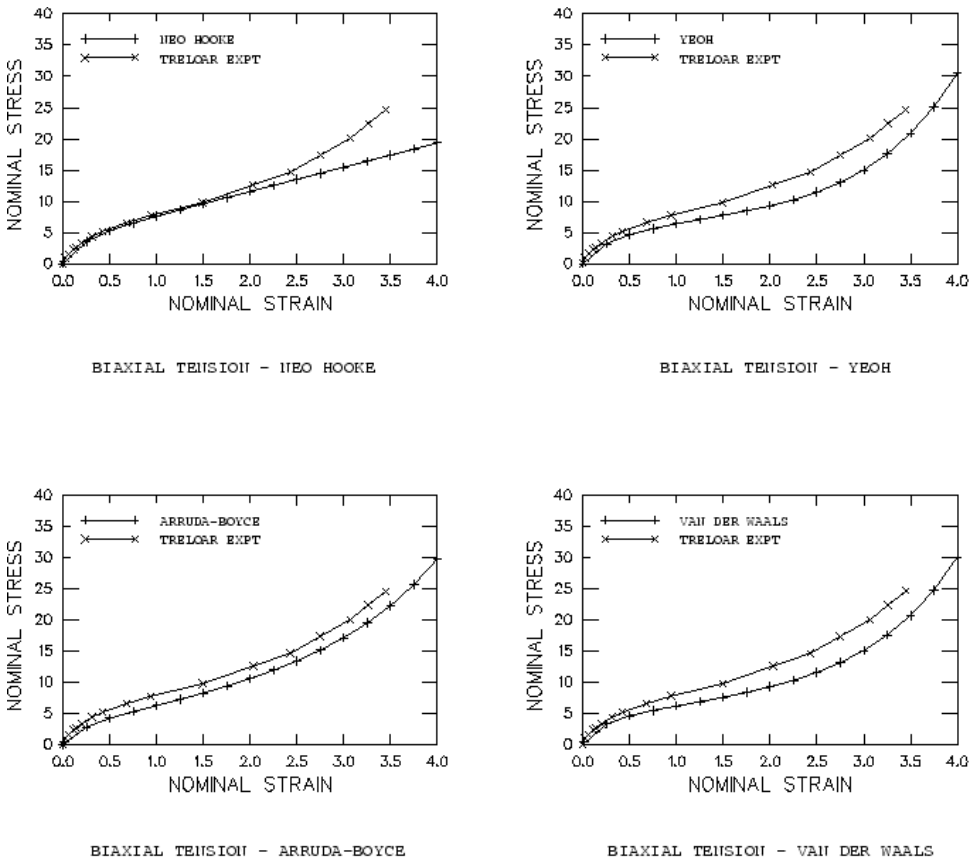
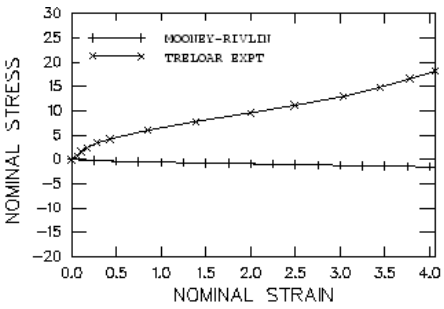
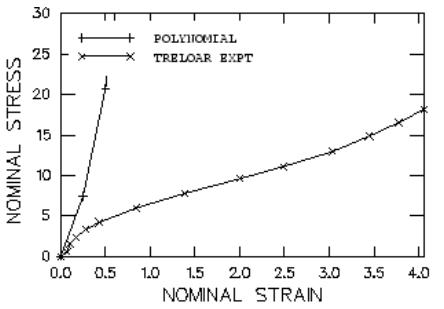


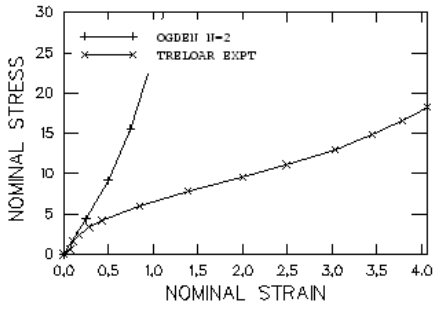
Figure 12: Equibiaxial tension results using uniaxial tension test data only (neo-Hookean, Yeoh, Arruda-Boyce, and Van der Waals models).



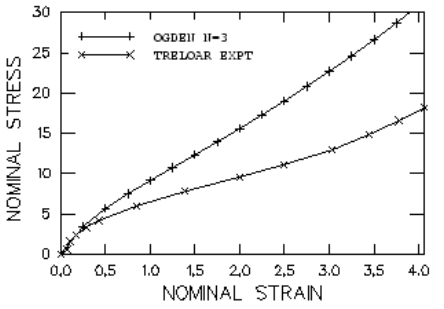
PLANAR TENSION - MOONEY-RIVLIN (2 TERMS)



PLANAR TENSION - POLYNOMIAL N=2 (5 TERMS)



PLANAR TENSION - OGDEN N=2 (2 TERMS)



PLANAR TENSION - OGDEN N=3 (3 TERMS)

Figure 13: Planar tension (pure shear) results using uniaxial tension test data only (polynomial and Ogden models).

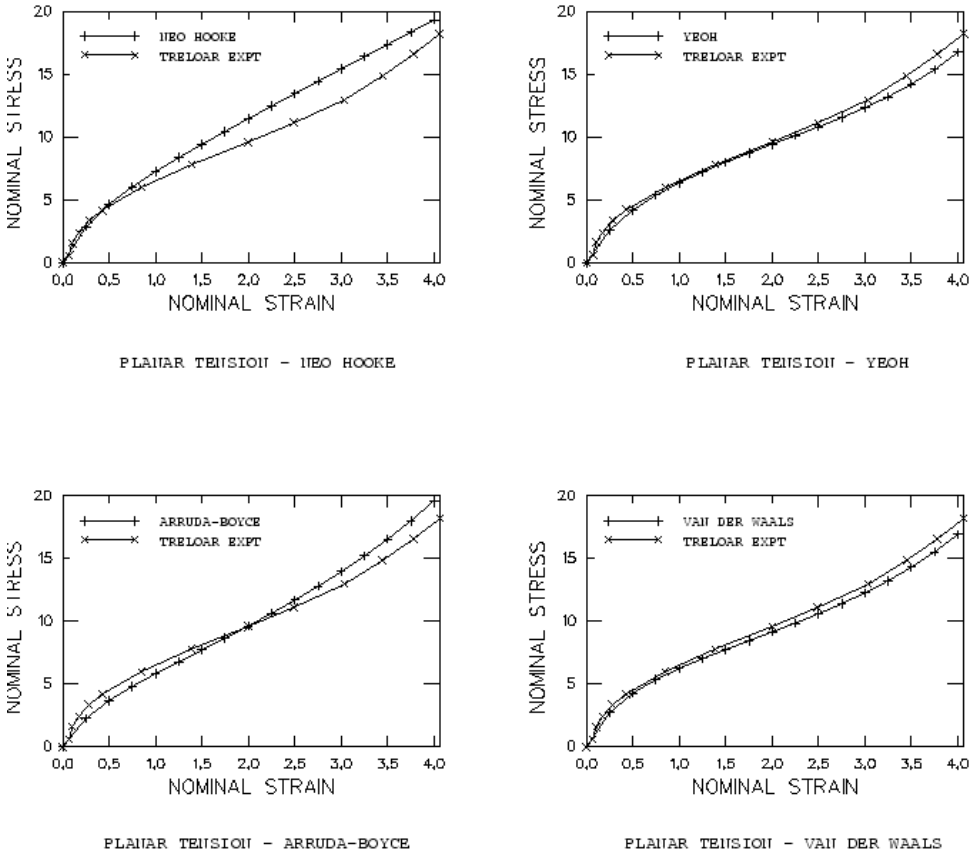
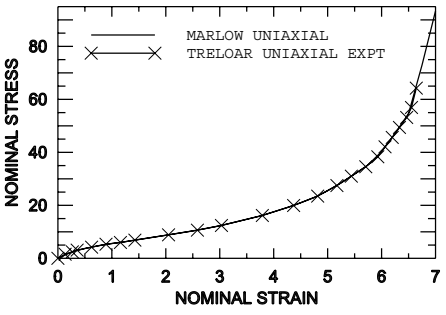
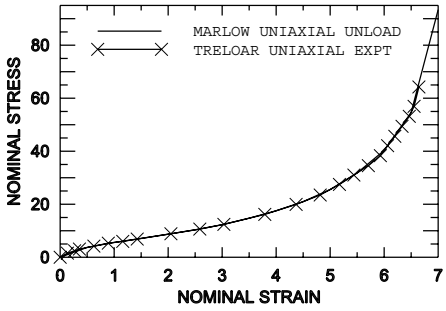


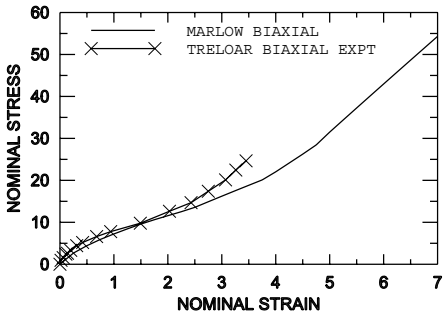
Figure 14: Planar tension (pure shear) results using uniaxial tension test data only (neo-Hookean, Yeoh, Arruda-Boyce, and Van der Waals models).



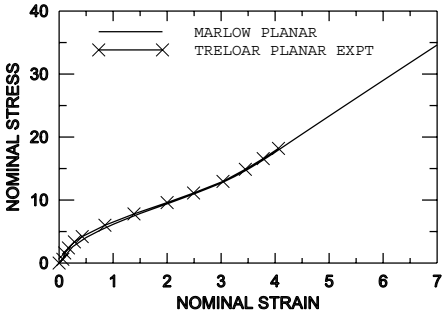
UNIAXIAL TENSION



UNIAXIAL UNLOAD



BIAXIAL TENSION



PLANAR TENSION

Figure 15: The results under different loading using uniaxial tension test data only (Marlow model).

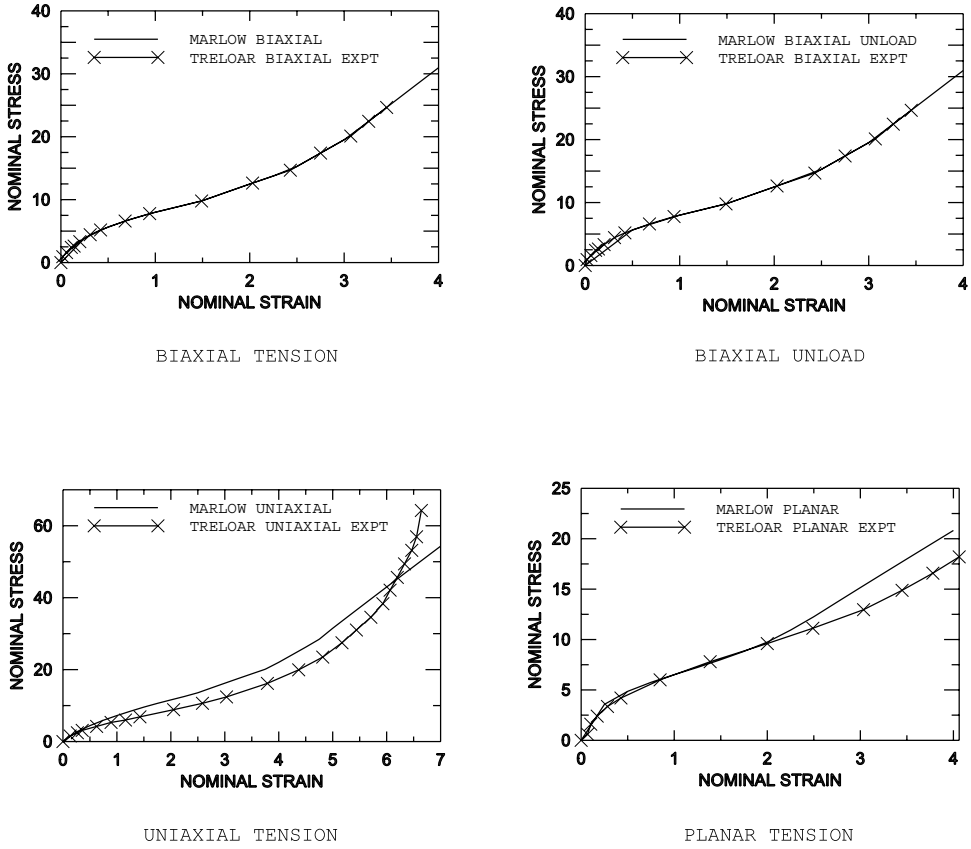
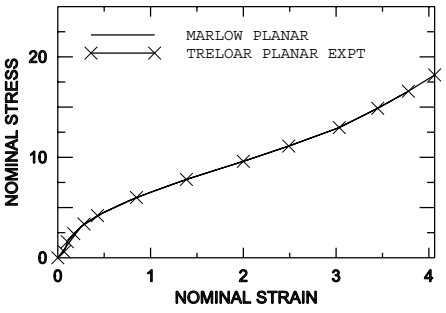
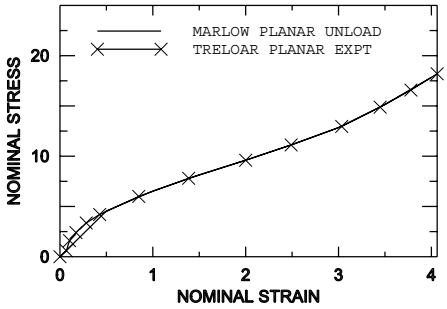


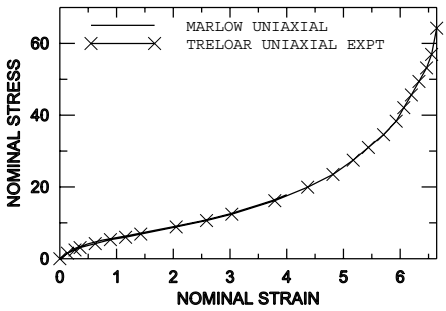
Figure 16: The results under different loading using biaxial tension test data only (Marlow model).



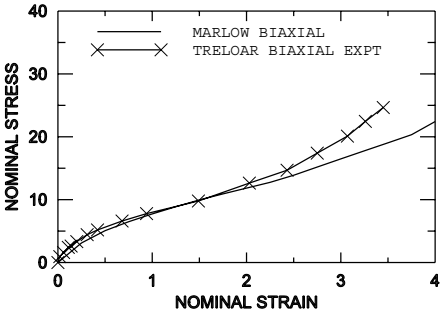
PLANAR TENSION



PLANAR UNLOAD



UNIAXIAL TENSION



BIAXIAL TENSION

Figure 17: The results under different loading using planar tension test data only (Marlow model).

Fitting of elastomeric foam test data

Products: Abaqus/Standard

Overview

This example illustrates the use of uniaxial compression data or uniaxial compression and simple shear data to fit the hyperfoam constants in the hyperelastic foam model.

Elastomeric foams are cellular materials that have the following primary mechanical characteristics:

- They can deform elastically up to 90% compression. This is their dominant mode of deformation.
- Their porosity permits very large volumetric deformations. This is in contrast to solid rubbers that are approximately incompressible.

Examples of elastomeric foam materials are cellular polymers such as cushions, padding, and packaging materials. Foams are often used for their excellent energy absorption properties—for a certain stress level, the energy absorbed by foams is substantially greater than by ordinary stiff elastic materials.

Another class of foam materials are the crushable foams that can undergo permanent (plastic) deformations. These materials are modeled using the crushable foam material model.

Elastomeric foam materials are modeled using the hyperelastic foam model (*Hyperelastic Behavior in Elastomeric Foams*), which is a nonlinear elastic model. The elastic behavior of the foams is based on the strain energy function

$$U = \sum_{i=1}^{N} \frac{2\mu_i}{\alpha_i^2} \left[\widehat{\lambda}_1^{\alpha_i} + \widehat{\lambda}_2^{\alpha_i} + \widehat{\lambda}_3^{\alpha_i} - 3 + \frac{1}{\beta_i} \left(J_{el}^{-\alpha_i \beta_i} - 1 \right) \right],$$

where

$$\widehat{\lambda}_i = J_{th}^{-\frac{1}{3}} \lambda_i \quad \rightarrow \quad \widehat{\lambda}_1 \widehat{\lambda}_2 \widehat{\lambda}_3 = J_{el}$$

and λ_i are the principal stretches. The elastic and thermal volume ratios, J_{el} and J_{th}, are

$$J_{el} = \frac{J}{J_{th}} \qquad \text{and} \qquad J_{th} = (1 + \varepsilon_{th})^3,$$

where J is the total volume ratio (current volume divided by original volume), and the thermal strain ε_{th} follows from the temperature and the isotropic thermal expansion coefficient defined in the thermal expansion material property. Time- or frequency-dependent elastic behavior is modeled through viscoelastic behavior.

The coefficients, μ_i, are related to the initial shear modulus, μ_0,

$$\mu_0 = \sum_{i=1}^{N} \mu_i;$$

the initial bulk modulus, K_0, follows from

$$K_0 = \sum_{i=1}^{N} 2\mu_i \left(\frac{1}{3} + \beta_i \right);$$

and β_i is related to Poisson's ratio ν_i,

$$\beta_i = \frac{\nu_i}{1 - 2\nu_i}.$$

This example shows how to derive the hyperfoam constants μ_i, α_i, and β_i from a set of material test data.

Problem description

For this example the test data are composed of uniaxial compression and simple shear data whose nominal stress–nominal strain curves are shown in *Figure 1*. The uniaxial compression curve (labeled "1" in the figure) can be broken down into three stages:

- At small strains ($< 5\%$) the foam deforms in a linear, elastic way due to cell wall bending.
- This is followed by a plateau of deformation with a relatively small range of stress caused by the elastic buckling of the cell walls.
- At higher strains a region of densification occurs where the cell walls crush together resulting in a rapid increase of compressive stress.

For this material the effective Poisson's ratio is zero, which is evident by the absence of lateral displacements, as seen in *Figure 2* in which a single continuum element illustrates the two deformation modes: uniaxial compression and simple shear.

The simple shear deformation results in a combination of compression and tension of the cell walls. In addition to the shear stress (labeled "2" in *Figure 1*), a transverse tensile stress (labeled "3" in the figure) is developed normal to the shear direction—this is called the Poynting effect. This transverse stress is included in the test data in addition to the shear stress.

Fitting procedure

In Abaqus the test data are specified as nominal stress–nominal strain data pairs using combinations of uniaxial test data, biaxial test data, simple shear test data, planar test data, and volumetric test data for hyperelastic foam with material constants computed by Abaqus from the test data. In addition, the effective Poisson's ratio can be specified for the hyperelastic foam.

For each stress-strain data pair, Abaqus generates an expression for the stress in terms of the stretches and the unknown hyperfoam constants. For the uniaxial, equibiaxial, planar, and volumetric deformation cases, the nominal stress T_C is

$$T_C = \frac{\partial U}{\partial \lambda_C} = \frac{2}{\lambda_C} \sum_{i=1}^{N} \frac{\mu_i}{\alpha_i} \left[\lambda_C^{\alpha_i} - J_{el}^{-\alpha_i \beta_i} \right],$$

where U is the strain energy potential and λ_C is the stretch in the primary displacement direction.

For the simple shear case, the nominal shear stress T_S is

$$T_S = \frac{\partial U}{\partial \gamma} = \sum_{j=1}^{2} \left[\frac{2\gamma}{2\left(\lambda_j^2 - 1\right) - \gamma^2} \sum_{i=1}^{N} \frac{\mu_i}{\alpha_i} \left(\lambda_j^{\alpha_i} - 1 \right) \right],$$

where γ is the shear strain and λ_j are the two principal stretches in the plane of shearing and are related to the shear strain by

$$\lambda_{1,2} = \sqrt{1 + \frac{\gamma^2}{2} \pm \gamma \sqrt{1 + \frac{\gamma^2}{4}}}.$$

For the n stress-strain pairs the following error measure E is minimized:

$$E = \sum_{i=1}^{n} \left(1 - T_i^{\text{th}} / T_i^{\text{test}} \right)^2,$$

where T_i^{test} is a stress value from the test data and T_i^{th} is one of the stress expressions described above.

As the energy potential is a nonlinear function of α_i and β_i, a nonlinear least squares procedure similar to that of Twizell and Ogden (1983) is used in Abaqus to determine μ_i, α_i, and β_i simultaneously. If the POISSON parameter is specified as the effective Poisson's ratio ν, then all the β_i constants are directly computed as

$$\beta_i = \frac{\nu}{1-2\nu}.$$

After a set of material constants is obtained, Abaqus performs material stability checks along the primary deformation modes using the Drucker stability criterion:

$$d\boldsymbol{\tau} : d\boldsymbol{\varepsilon} > 0 \qquad \Longrightarrow \qquad d\boldsymbol{\varepsilon} \cdot \mathbf{D} : d\boldsymbol{\varepsilon} > 0,$$

where $d\boldsymbol{\tau}$ is the Kirchhoff stress increment due to the logarithmic strain increment, $d\boldsymbol{\varepsilon}$, and $\mathbf{D}$ is the tangential material stiffness. For the stability criterion to be satisfied, $\mathbf{D}$ must be positive-definite. The analysis input file processor will give warning messages if $\mathbf{D}$ loses its positive-definite property, thereby defining strain states that are likely to result in unstable material behavior. The deformation modes examined are uniaxial, equibiaxial, planar, and volumetric deformation in tension and compression and the simple shear mode.

Fitting case 1—results using uniaxial compression and simple shear data

Both types of test data are used in fitting the hyperfoam constants for order $N = 2$ (with four constants: $\mu_1, \alpha_1, \mu_2, \alpha_2$; the β_1, β_2 constants are zero since the effective Poisson's ratio is zero) and order $N = 3$ (with six constants). Both the shear stress and the transverse tensile stress are used for the simple shear data. The $N = 3$ parameters fail the Drucker stability test for five deformation modes, while the $N = 2$ parameters predict no instability at all.

A single 8-node continuum element C3D8R of unit dimensions is subjected to enforced boundary conditions simulating the two deformation modes as shown in *Figure 2*. Since Abaqus outputs true (Cauchy) stress and logarithmic strain, a method of deriving nominal stress–nominal strain results is described in the listings of the data files at the end of this example.

The fits as shown in *Figure 3* for both values of *N* are accurate up to the maximum strain of 80% for the uniaxial compression case. For the simple shear case the fit is accurate up to shear strains of about 50%; beyond that, the Abaqus shear results stiffen up faster than the shear test data.

Fitting case 2—results using uniaxial compression data only

Commonly only one type of test data may be available to the user. For this example the consequences of fitting the hyperfoam constants using only the uniaxial compression data are examined. Both $N = 2$ and $N = 3$ parameters pass all the Drucker stability tests. *Figure 4* shows the results of the two deformation modes in contrast with fitting case 1.

The uniaxial compression results for both values of *N* match the test data extremely well, as expected since the hyperfoam constants are fit using only the uniaxial data. However, in the simple shear case considerable disparities are seen between the numerical and test data at the onset of finite strains—the shear results of fitting case 1 are definitely superior.

Results and discussion

For these test data the $N = 2$ model seems adequate in providing close correlations (it is stable, where the $N = 3$ model is not). However, this example also illustrates that the shear behavior is not satisfactorily reproduced when only uniaxial test data are used in fitting the hyperfoam constants. The inadequacy can be attributed to the fact that, in simple shear, certain directions are in tension. This tension behavior cannot be characterized properly

with compression data only. This observation is similar to the one made for incompressible hyperelastic materials, where there is no guarantee that other deformation modes besides the test data modes can be reproduced to an acceptable accuracy.

However, for hyperelastic foams the ample compressibility reduces the different axial deformation modes (e.g., biaxial, triaxial, etc.) into a "superposition" of several uniaxial states at different orientations. This is particularly true for compression states, where buckling of cell walls under loading in one direction is quite independent from that in perpendicular directions. Thus, it is not uncommon that a single uniaxial compression test may be sufficient to characterize the material behavior if the application is compression dominated. However, it is preferable to use test data derived from different deformation modes.

Input files

foamdatafitting_compress.inp	Uniaxial compression mode.
foamdatafitting_shear.inp	Simple shear deformation mode.

Change the value of the parameter N of the **HYPERFOAM* option to use a different order of the strain energy function. Remove the simple shear stress-strain data from the material definition to run the cases of using uniaxial compression data only.

References

- Twizell, E. H., and R. W. Ogden, "Non-Linear Optimization of the Material Constants in Ogden's Stress-Deformation Function for Incompressible Isotropic Elastic Materials," *J. Austral. Math. Soc. Ser. B*, vol. 24, pp. 424–434, 1983.

Figures

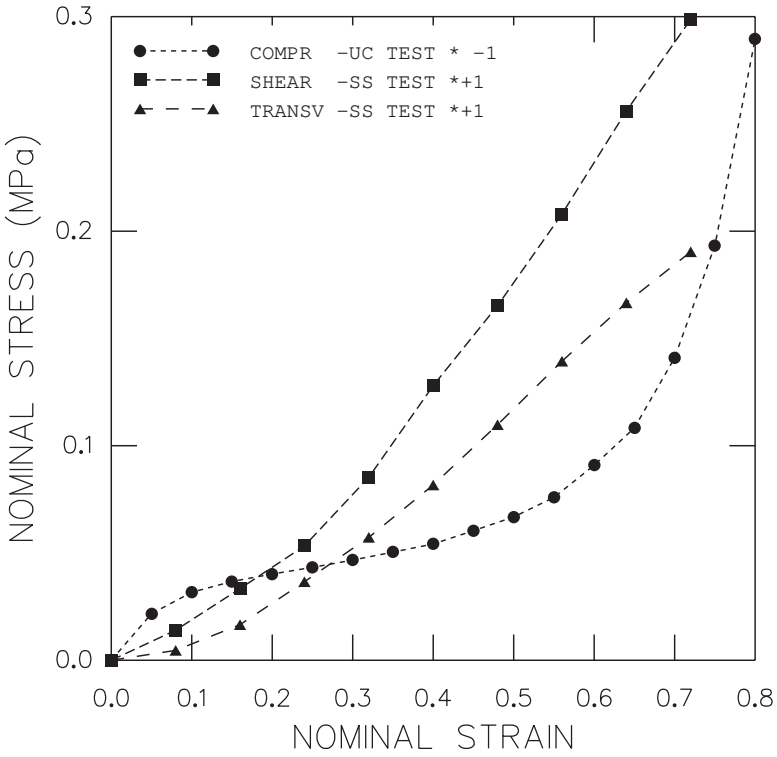


Figure 1: Uniaxial compression and simple shear test data.

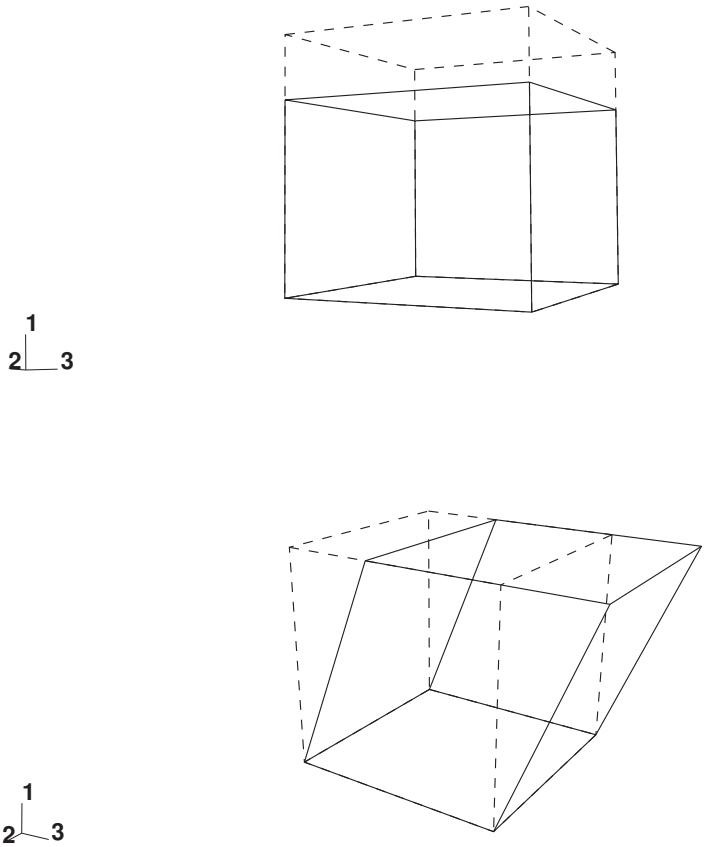


Figure 2: Uniaxial compression and simple shear deformation modes.

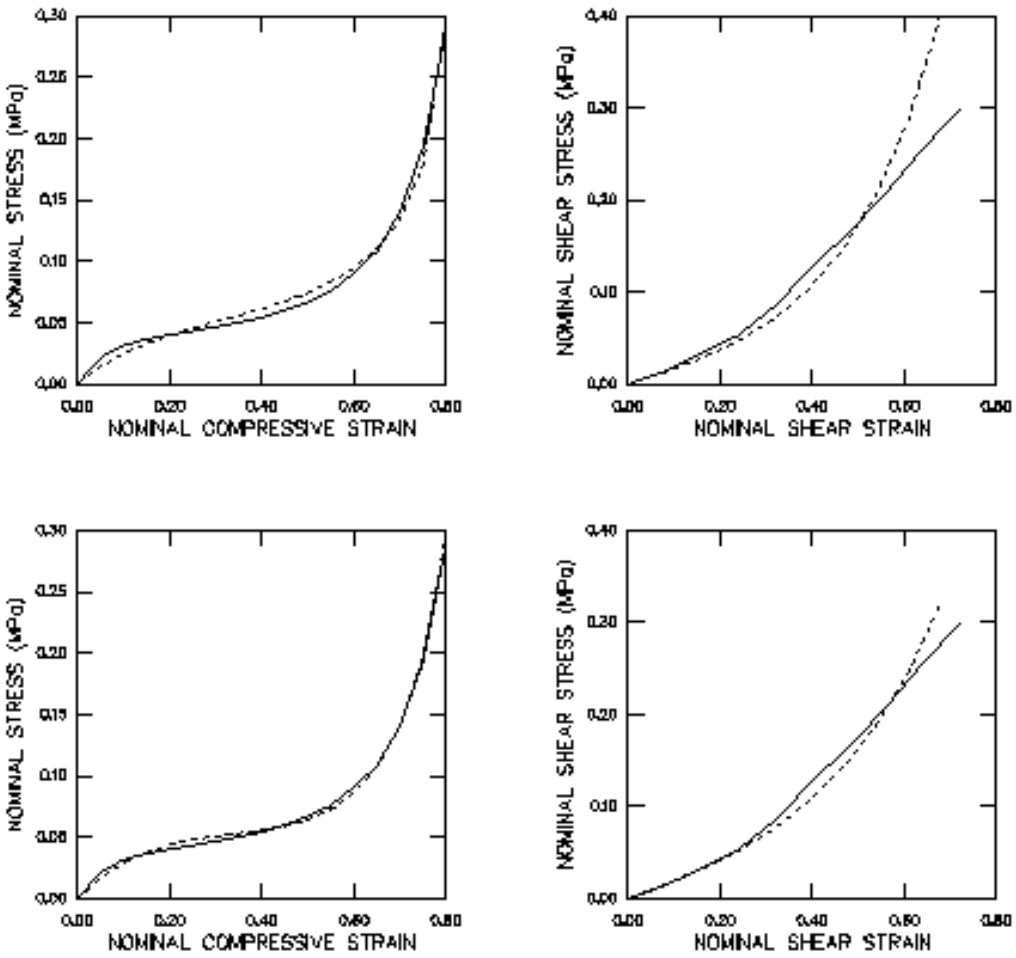


Figure 3: Results using uniaxial compression and simple shear data. Solid line: experimental data. Dashed line: Abaqus results.

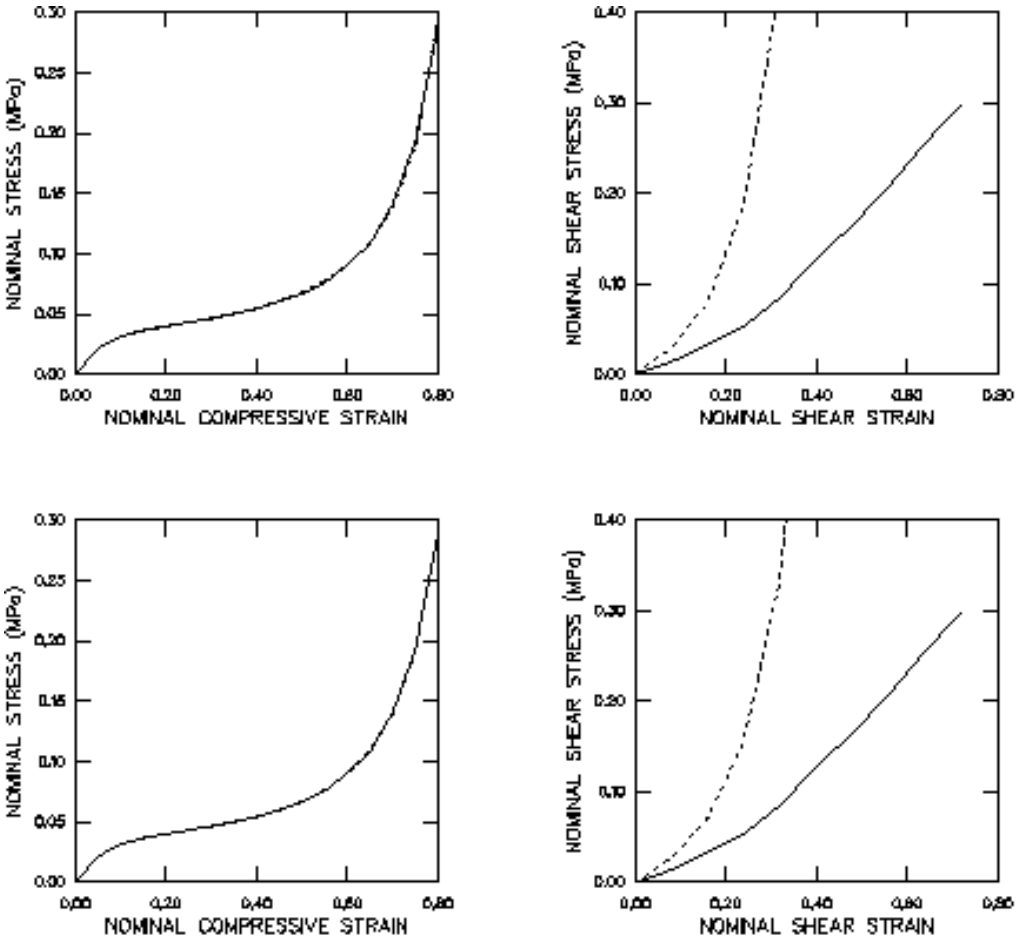


Figure 4: Results using uniaxial compression data only. Solid line: experimental data. Dashed line: Abaqus results.

Rubber under uniaxial tension

Products: Abaqus/Explicit

Overview

This example illustrates the homogeneous deformation of a cube of unit dimension.

Four types of hyperelastic strain energy potentials are used: the polynomial (including the reduced polynomial and Yeoh forms, which are the most common particular cases of the general polynomial model), Ogden, Arruda-Boyce, and Van der Waals forms with coefficients drawn from Treloar's experimental data (Treloar, 1940).

Problem description

Stress and strain data are entered using uniaxial test data, biaxial test data, planar test data, and volumetric test data for the hyperelastic material model. A least squares method is used to fit the experimental data. For the polynomial and Ogden forms the order of the series (N) is specified for the hyperelastic material model. The following cases are analyzed:

- Polynomial form with N=2 (5 deviatoric terms).
- Reduced polynomial form with N=4 (4 deviatoric terms).
- Yeoh form, which is equivalent to the reduced polynomial form with N=3.
- Ogden form with N=3 (3 deviatoric terms).
- Arruda-Boyce form.
- Van der Waals form.

The stress values are given in pascals. The density of the material is 1000 kg/m^3. A state of simple uniaxial tension is induced in the cube up to a strain of 600%. The stretching velocity is ramped up from zero to 6.0 m/s within 2.0 seconds. The coefficients fitted by using the polynomial strain energy potential with N=2 may lead to unstable material behavior when the nominal strain in a uniaxial tensile test reaches approximately 440%. (Unstable regimes for hyperelastic materials are discussed in *Hyperelastic Behavior of Rubberlike Materials*.) Therefore, for that particular analysis the final deformation was set to be 400% by ramping the velocity from zero to 4.2 m/s within 2.0 seconds.

Results and discussion

Figure 1 shows the initial and deformed shapes. *Figure 2* through *Figure 4* show comparisons of the computed nominal stress and strain in the stretching direction with Treloar's experimental data. The Abaqus results are seen to match the experimental results closely for the entire range of deformation, especially when the Ogden model or the Van der Waals potential is used. Analogous results were obtained for all the material models using C3D10M elements.

Input files

hypertest.inp	Polynomial hyperelasticity case; C3D8R elements.
hyper_reducedpoly.inp	Reduced polynomial hyperelasticity case; C3D8R elements.
hyper_yeoh.inp	Yeoh hyperelasticity case; C3D8R elements.
hyper_ogden.inp	Ogden hyperelasticity case; C3D8R elements.

hyper_ab_all.inp	Arruda-Boyce hyperelasticity case; C3D8R elements.
hyper_vw_all.inp	Van der Waals hyperelasticity case; C3D8R elements.
hypertest_c3d10m.inp	Polynomial hyperelasticity case; C3D10M elements.
hyper_reducedpoly_c3d10m.inp	Reduced polynomial hyperelasticity case; C3D10M elements.
hyper_yeoh_c3d10m.inp	Yeoh hyperelasticity case; C3D10M elements.
hyper_ogden_c3d10m.inp	Ogden hyperelasticity case; C3D10M elements.
hyper_ab_all_c3d10m.inp	Arruda-Boyce hyperelasticity case; C3D10M elements.
hyper_vw_all_c3d10m.inp	Van der Waals hyperelasticity case; C3D10M elements.

Two additional input files are also included with the Abaqus release for the purpose of testing the implementation of the Arruda-Boyce and Van der Waals strain energy potentials (file names: *hyper_ab_uni.inp* and *hyper_vw_uni.inp*).

References

- Treloar, L. R. G., “Stress-Strain Data for Vulcanised Rubber under Various Types of Deformation,” *Transactions of the Faraday Society*, vol. 40, pp. 59–70, 1940.

Figures

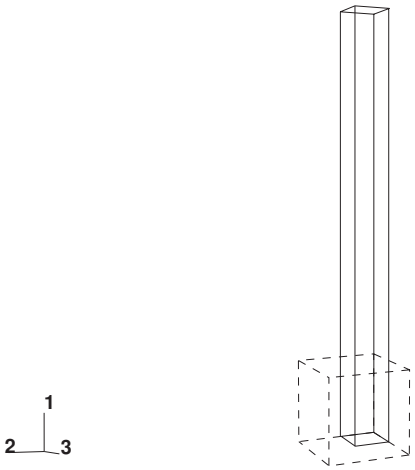


Figure 1: Deformed and initially undeformed element.

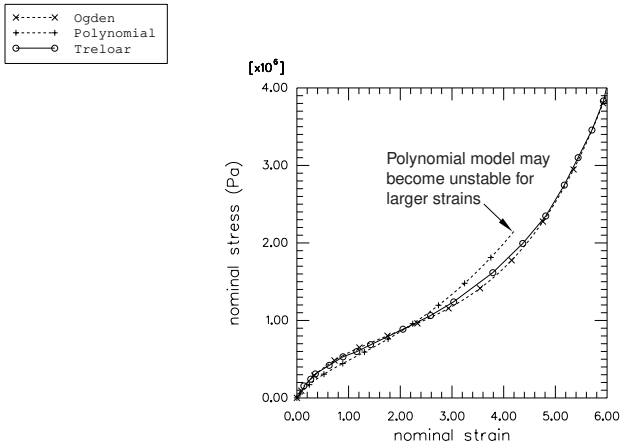


Figure 2: Stresses vs. strains in the stretching direction for the polynomial and Ogden forms.

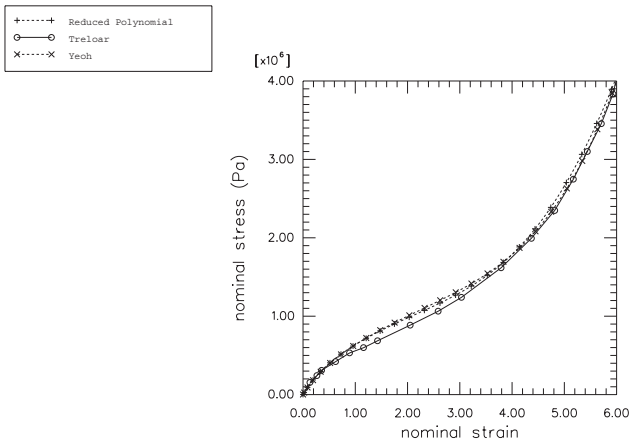


Figure 3: Stresses vs. strains in the stretching direction for the reduced polynomial and Yeoh forms.

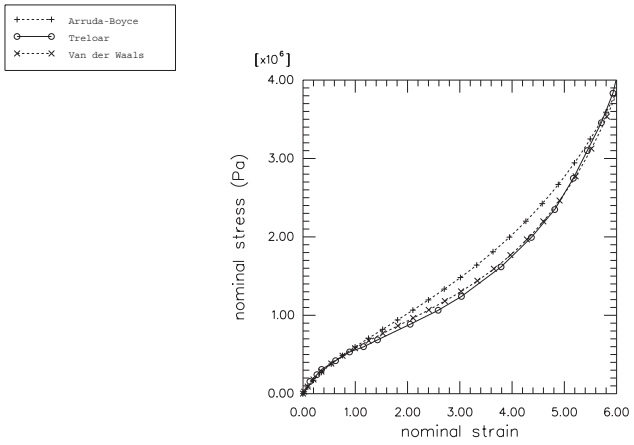


Figure 4: Stresses vs. strains in the stretching direction for the Arruda-Boyce and Van der Waals forms.

Anisotropic hyperelastic modeling of arterial layers

Products: Abaqus/Standard Abaqus/Explicit

Overview

This problem illustrates the use of the anisotropic hyperelastic capabilities in Abaqus to model soft biological tissue. More specifically, the problem shows how these capabilities can be used to model the mechanical response of the adventitial layer of human iliac arteries.

Numerical examples are provided for simple tension tests of iliac adventitial strips cut along the axial and circumferential directions of the artery. An example of a strip cut at an angle of 15° with respect to the circumferential direction is also included. The numerical study demonstrates the significant effect that dispersion of the collagen fiber orientations can have on the mechanical response of soft tissue. The problem has been analyzed numerically by Gasser, Holzapfel, and Ogden (2006).

Problem description

We consider the numerical analysis of simple tensile tests of adventitial strips cut along the axial and circumferential directions of the artery, as well as a strip cut at an angle of 15° with respect to the circumferential direction, as illustrated in *Figure 1*. Following the work of Gasser, Holzapfel, and Ogden (2006), the adventitial strips considered in the study have referential dimensions of 10.0 mm length × 3.0 mm width × 0.5 mm thickness and are assumed to be stress free in the reference configuration. It is assumed that two families of collagen fibers are embedded in the specimens, symmetrically arranged with respect to the axial and circumferential directions of the artery and with no component in the radial (thickness) direction, as shown in *Figure 1*. The angle between the mean orientation of the fibers and the circumferential direction is γ=49.98°. The specimens are loaded in the longitudinal direction, and their end faces are not allowed to deform. Using appropriate symmetry boundary conditions, only one-eighth of the geometry is modeled in the case of strips cut along the axial and circumferential directions. A full scale simulation is carried out for strips cut at a 15° angle with respect to the circumferential direction. The finite element model of the specimens consists of a 20 × 10 × 2 mesh for the simulations considering one-eighth symmetry and a 40 × 20 × 4 mesh for the full scale simulations.

The numerical analyses are conducted using the static analysis procedure in Abaqus/Standard. Linear solid hybrid elements (C3D8H) are used to model the incompressible deformation of the arterial layers. For comparison, the solution is also computed in Abaqus/Explicit assuming quasi-static loading conditions. Since Abaqus/Explicit has no mechanism for imposing the incompressibility constraint, some amount of compressibility is introduced in the material response in the Abaqus/Explicit simulations and linear solid elements (C3D8R) are used.

Material

The mechanical response of the adventitial layer is modeled using the anisotropic hyperelastic strain energy function proposed by Gasser, Holzapfel, and Ogden (2006) to model arterial layers with distributed collagen fiber orientations. Details of the modeling materials are given in *Holzapfel-Gasser-Ogden Form*, and a discussion of the theory behind anisotropic hyperelastic material behavior is presented in *Holzapfel-Gasser-Ogden form*.

It is assumed that the arterial layer is composed of two families of collagen fibers embedded in a soft incompressible ground matrix. The families of fibers have mean orientations characterized by the vectors $\mathbf{A_1}$ and $\mathbf{A_2}$ in the reference configuration, but the orientations of the fibers within each family are dispersed. The model incorporates a scalar structure parameter, κ ($0 \le \kappa \le 1/3$), that characterizes the level of dispersion of the collagen orientations. When κ=0, the fibers are perfectly aligned (no dispersion). When κ=1/3, the fibers are

randomly distributed and the material becomes isotropic. A value of κ=0.226 is used in the numerical simulations. For comparison, numerical tests are also carried out assuming ideal alignment of the collagen fibers (κ=0).

The material properties are taken from Gasser, Holzapfel, and Ogden (2006) and are shown in *Table 1*. They are based on least-squares fitting of longitudinal and circumferential tension tests of adventitial strips of nine iliac arteries carried out by Holzapfel, Sommer, and Regitnig (2004).

Loading and controls

The mounting of the specimen in the testing machine is modeled by constraining both ends of the strip. The strips are loaded in the tensile direction, and their end faces are not allowed to deform.

In the Abaqus/Standard simulations the static procedure is used with a prescribed load of 2.0 N. Geometric nonlinearities are considered in this step to account for the large deformations of the adventitial strip.

Since Abaqus/Explicit is a dynamic analysis program and we are interested in a static solution to the problem, care must be taken to avoid significant inertia effects as the adventitial strip is loaded. A smooth step amplitude curve is used to prescribe the uniaxial displacement of the strip and to promote a quasi-static solution in the Abaqus/Explicit simulations. The simulations are run in double precision.

Results and discussion

Results for the Abaqus/Standard and Abaqus/Explicit analyses are discussed in the following sections.

Abaqus/Standard results

Figure 2 shows the computed stress in the tensile direction for the axial (left) and circumferential (right) specimens with distributed fibers (κ=0.226) for a tensile load of 2.0 N. The thickness of the specimens remains approximately constant during loading, with small transition zones at the ends of the strips. The corresponding results for the case of perfectly aligned fibers (κ=0) are shown in *Figure 3*. In this case the embedded collagen fibers need to rotate significantly toward the loading direction before they can carry significant load. The combined effect of the large rotation of the fibers and the incompressibility constraint causes the thickness of the specimen to increase (and the width to decrease) in the middle region of the strip, away from the restrained boundaries. The transition areas at the end of the strip resemble deformation patterns similar to those observed in woven fabrics.

Figure 4 shows the computed load versus displacement curves for the circumferential and axial specimens. The dashed curves correspond to the simulations with ideally aligned fibers, and the continuous curves correspond to the simulations that include dispersion. As seen in the figure, the material response is very soft at low stretches; only a small force is needed to achieve significant extension. Once the collagen fibers are approximately aligned with the loading direction, the material stiffens rapidly. This is particularly evident in the case of the circumferential specimen with κ=0; the alignment requires very large average stretches, and the specimen stiffens at a displacement of about 4 mm. In contrast, when dispersion is included in the simulation, the collagen fibers need to rotate less before they carry load compared with the ideally aligned case. Therefore, the dispersion of the collagen fibers leads to a stiffer macroscopic response of the specimens. Specifically, the dispersion parameter κ controls the elongation at which the specimen stiffens.

These numerical results for axial and circumferential specimens are in agreement with the results reported in Gasser, Holzapfel, and Ogden (2006).

Figure 5 shows the stress in the loading direction in a specimen cut at an angle of 15° with respect to the circumferential direction. The plot on the left corresponds to the case of distributed fibers, and the plot on the right corresponds to the case of ideally aligned fibers. Again, we observe that significantly more rotation is required for ideally aligned fibers before they carry load.

Abaqus/Explicit results

For comparison, the axial and circumferential specimens with distributed fibers are also analyzed using Abaqus/Explicit. *Figure 6* shows the stress in the tensile direction for the axial (left) and circumferential (right) specimens at the end of the Abaqus/Explicit simulations. As illustrated in *Figure 7*, the load-displacement response computed in Abaqus/Explicit compares well with that obtained in Abaqus/Standard. Small discrepancies in the results can be attributed to the use of different element types as well as minor dynamic effects and some amount of material compressibility in the Abaqus/Explicit simulations.

Input files

Abaqus/Standard input files

adventitia_axial.inp	Simple tension of adventitial strip cut along axial direction; distributed fibers (κ=0.226); 1/8 symmetry model; C3D8H elements.
adventitia_axial_k0.inp	Simple tension of adventitial strip cut along axial direction; ideally aligned fibers (κ=0); 1/8 symmetry model; C3D8H elements.
adventitia_circ.inp	Simple tension of adventitial strip cut along circumferential direction; distributed fibers (κ=0.226); 1/8 symmetry model; C3D8H elements.
adventitia_circ_k0.inp	Simple tension of adventitial strip cut along circumferential direction; ideally aligned fibers (κ=0); 1/8 symmetry model; C3D8H elements.
adventitia_15deg.inp	Simple tension of adventitial strip cut at an angle of 15° with respect to the circumferential direction; distributed fibers (κ=0.226); C3D8H elements.
adventitia_15deg_k0.inp	Simple tension of adventitial strip cut at an angle of 15° with respect to the circumferential direction; ideally aligned fibers (κ=0); C3D8H elements.

Abaqus/Explicit input files

adventitia_axial_xpl.inp	Simple tension of adventitial strip cut along axial direction; distributed fibers (κ=0.226); 1/8 symmetry model; compressible; C3D8R elements.
adventitia_circ_xpl.inp	Simple tension of adventitial strip cut along circumferential direction; distributed fibers (κ=0.226); 1/8 symmetry model; compressible; C3D8R elements.

Additional Abaqus/Standard input files not used for discussion

adventitia_axial_cps4r.inp	Simple tension of adventitial strip cut along axial direction; distributed fibers (κ=0.226); plane stress 1/4 symmetry model; incompressible; CPS4R elements.
adventitia_axial_m3d4r.inp	Simple tension of adventitial strip cut along axial direction; distributed fibers (κ=0.226); plane stress 1/4 symmetry model; incompressible; M3D4R elements.
adventitia_axial_s4r.inp	Simple tension of adventitial strip cut along axial direction; distributed fibers (κ=0.226); plane stress 1/4 symmetry model; incompressible; S4R elements.

Additional Abaqus/Explicit input files not used for discussion

adventitia_axial_m3d4_xpl.inp	Simple tension of adventitial strip cut along axial direction; distributed fibers (κ=0.226); plane stress 1/4 symmetry model; incompressible; M3D4 elements.
adventitia_axial_k0_m3d4_xpl.inp	Simple tension of adventitial strip cut along axial direction; ideally aligned fibers (κ=0); plane stress 1/4 symmetry model; incompressible; M3D4 elements.

adventitia_circ_m3d4_xpl.inp	Simple tension of adventitial strip cut along circumferential direction; distributed fibers (κ=0.226); plane stress 1/4 symmetry model; incompressible; M3D4 elements.
adventitia_circ_k0_m3d4_xpl.inp	Simple tension of adventitial strip cut along circumferential direction; ideally aligned fibers (κ=0); plane stress 1/4 symmetry model; incompressible; M3D4 elements.
adventitia_15deg_m3d4_xpl.inp	Simple tension of adventitial strip cut at an angle of 15° with respect to the circumferential direction; distributed fibers (κ=0.226); plane stress 1/4 symmetry model; incompressible; M3D4 elements.
adventitia_15deg_k0_m3d4_xpl.inp	Simple tension of adventitial strip cut at an angle of 15° with respect to the circumferential direction; ideally aligned fibers (κ=0); plane stress 1/4 symmetry model; incompressible; M3D4 elements.

References

- Gasser, T. C., G. A. Holzapfel, and R. W. Ogden, "Hyperelastic Modelling of Arterial Layers with Distributed Collagen Fibre Orientations," *Journal of the Royal Society Interface*, vol. 3, pp. 15–35, 2006.
- Holzapfel, G. A., T. C. Gasser, and R. W. Ogden, "A New Constitutive Framework for Arterial Wall Mechanics and a Comparative Study of Material Models," *Journal of Elasticity*, vol. 61, pp. 1–48, 2000.
- Holzapfel, G. A., G. Sommer, and P. Regitnig, "Anisotropic Mechanical Properties of Tissue Components in Human Atherosclerotic Plaques," *Journal of Biomechanical Engineering*, vol. 126, pp. 657–665, 2004.

Tables

Table 1: Assumed material properties for iliac adventitial layer.

Holzapfel-Gasser-Ogden energy function coefficients:

C_{10} = 3.82 kPa

k_1 = 996.6 kPa

k_2 = 524.6

κ = 0.226

D = 0 ($D = 1\times 10^{-6}$ for compressible case)

Fiber directions (for strips cut along circumferential direction):

$$\mathbf{A}_1 = (\cos\gamma, \sin\gamma, 0),$$

$$\mathbf{A}_2 = (\cos\gamma, -\sin\gamma, 0),$$

with γ = 49.98°.

Figures

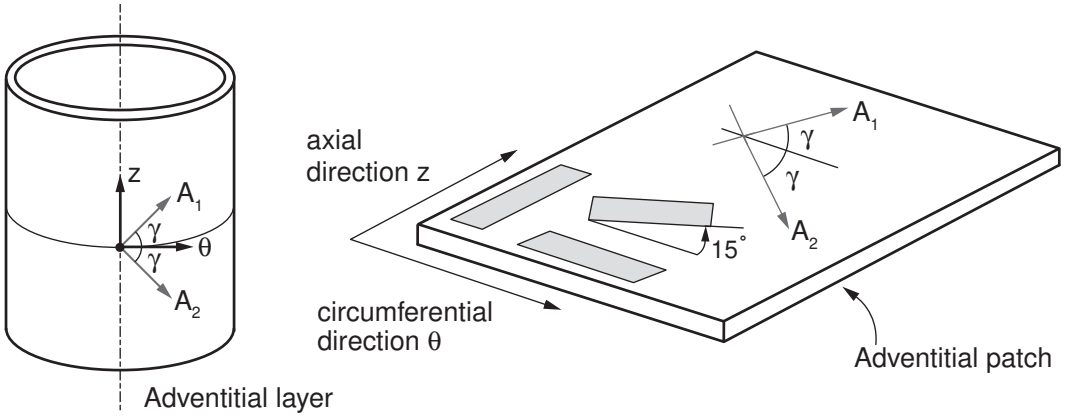


Figure 1: Adventitial layer with two embedded families of fibers with mean orientations $\mathbf{A_1}$ and $\mathbf{A_2}$ (left). Definition of circumferential, axial, and 15° specimens for the tensile tests (right).

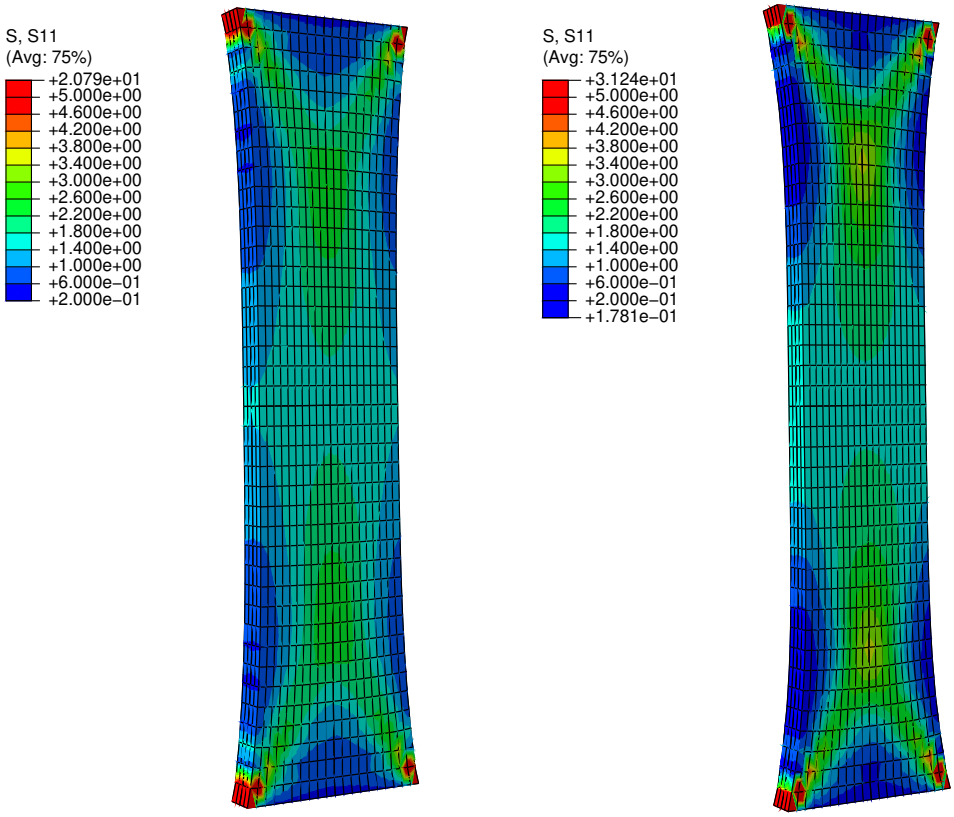


Figure 2: Stress in the direction of applied load for iliac adventitial strips cut in the axial (left) and circumferential (right) directions. Results correspond to an applied load of 2.0 N; dispersion of collagen fibers is included (κ=0.226).

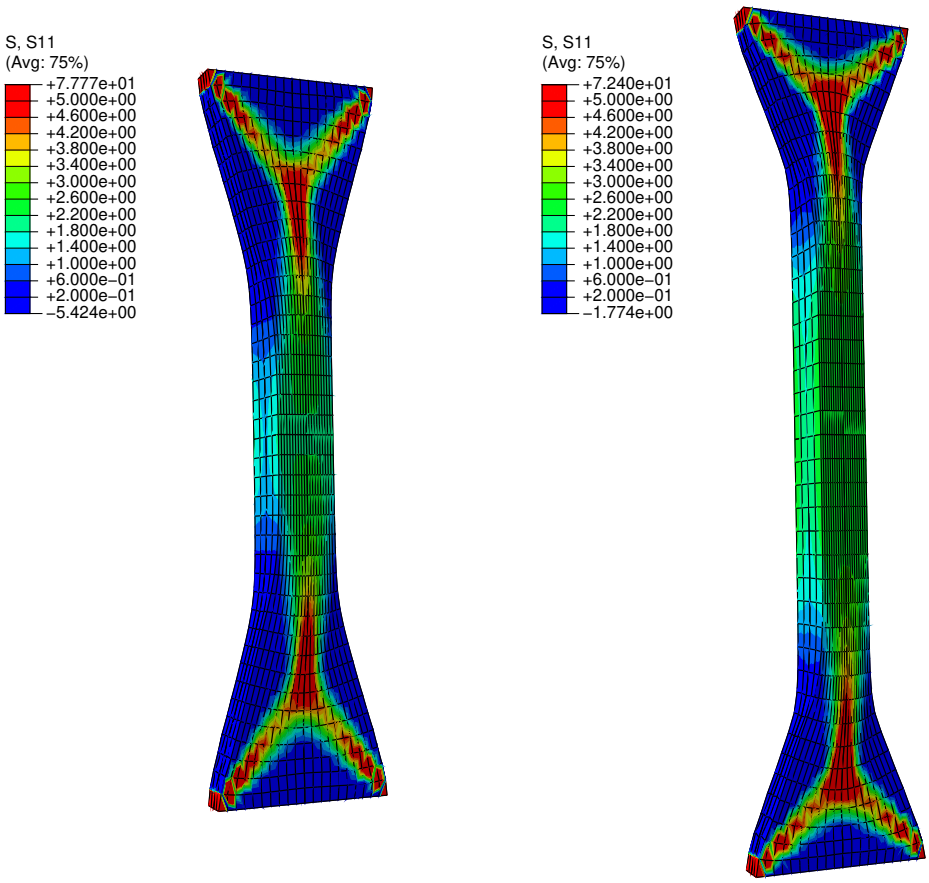


Figure 3: Stress in the direction of applied load for iliac adventitial strips cut in the axial (left) and circumferential (right) directions. Results correspond to an applied load of 2.0 N; collagen fibers are perfectly aligned (κ=0).

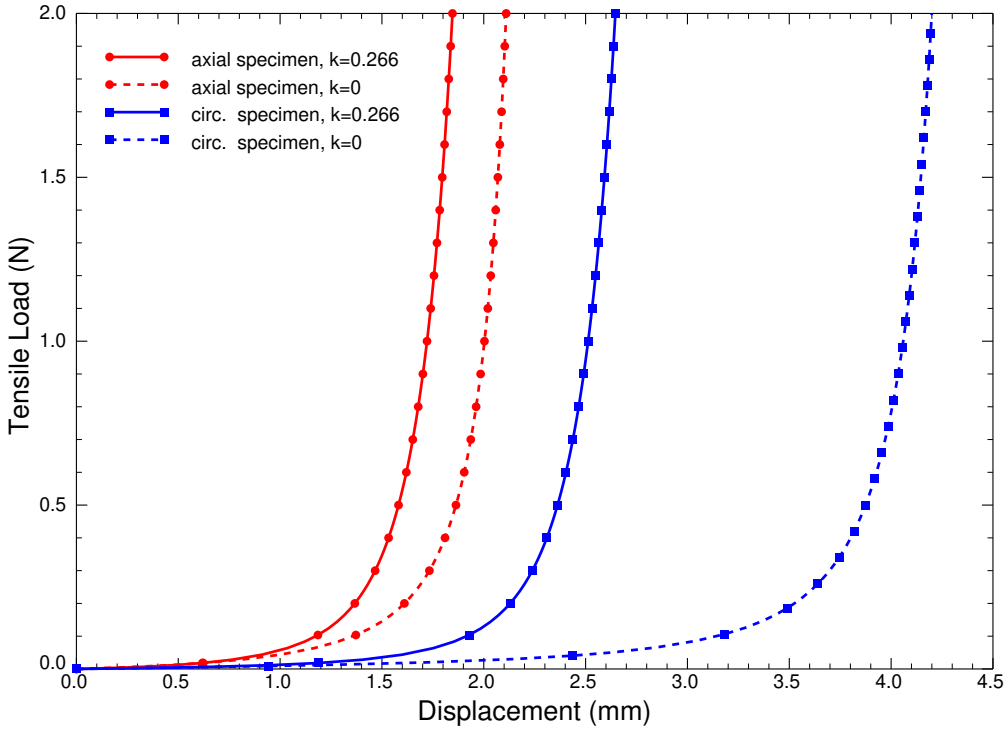


Figure 4: Load-displacement response of circumferential and axial specimens.

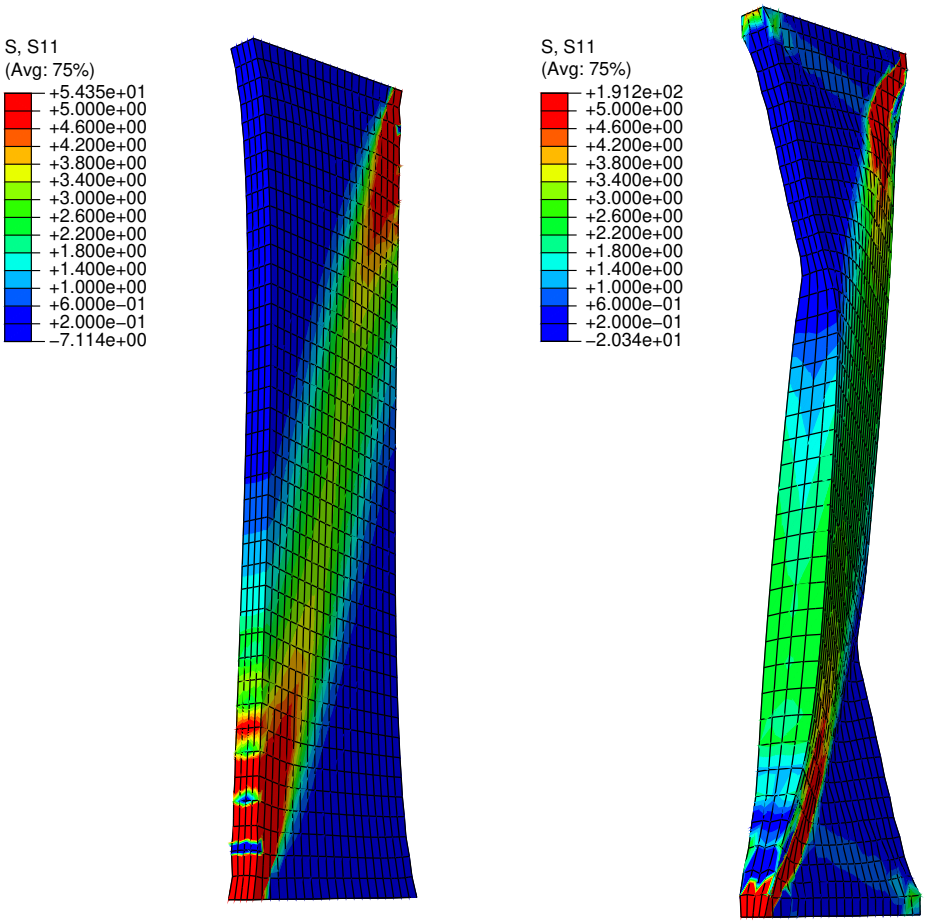


Figure 5: Stress in the direction of applied load for iliac adventitial strips cut at an offset angle of 15° with respect to the circumferential direction; with dispersion of collagen fibers included (left) and without dispersion (right).

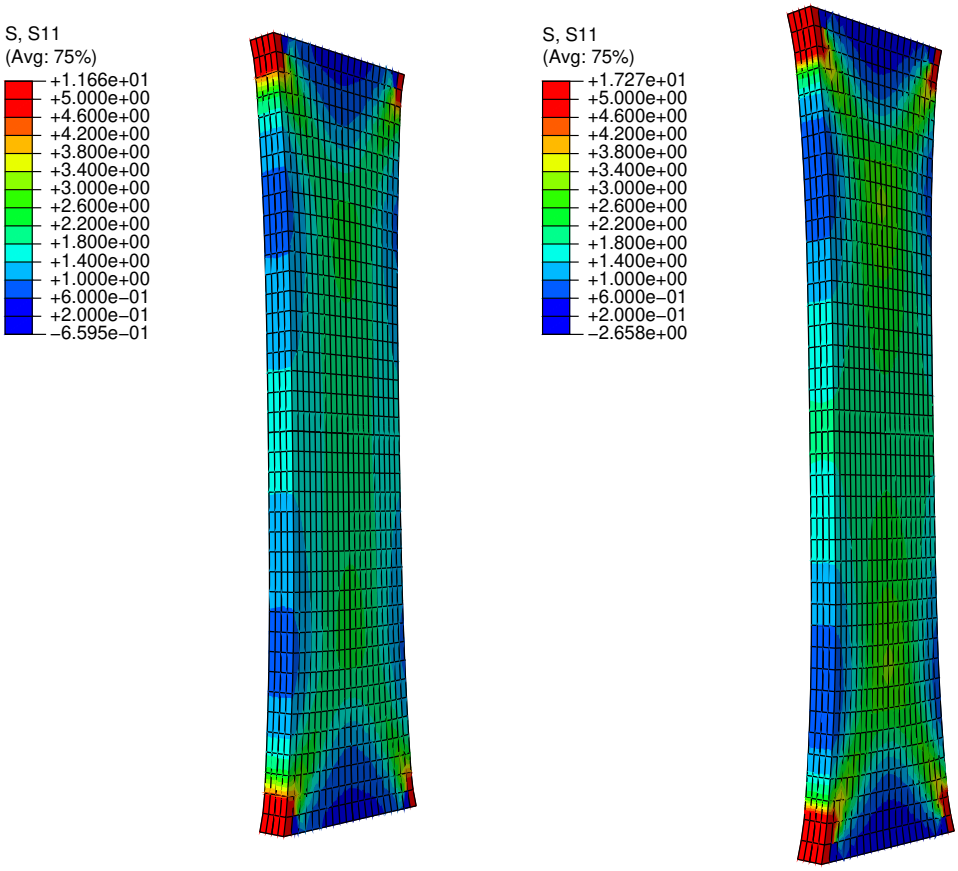


Figure 6: Abaqus/Explicit results for the stress in the direction of applied load for iliac adventitial strips with distributed fibers (κ=0.226) cut in the axial (left) and circumferential (right) directions. Results correspond to the end of the Abaqus/Explicit simulations.

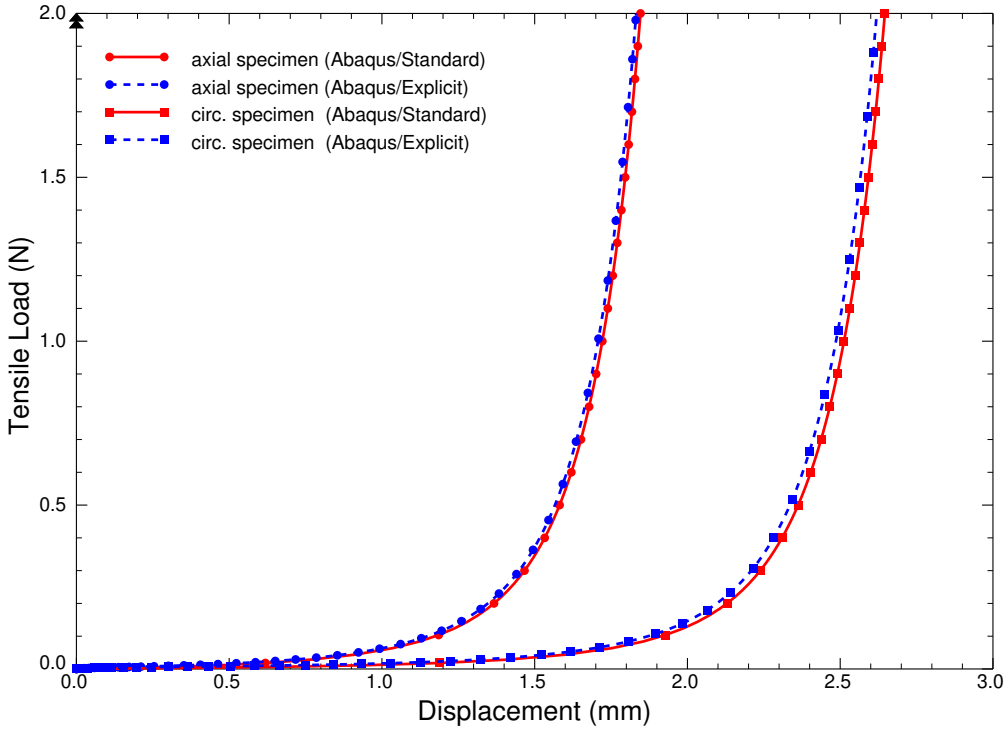


Figure 7: Comparison of Abaqus/Standard and Abaqus/Explicit results for the load-displacement response of circumferential and axial specimens with distributed fibers (κ=0.226).

Plasticity and creep

In this section:

- *Uniformly loaded, elastic-plastic plate*
- *Test of ORNL plasticity theory under biaxial loading*
- *One-way reinforced concrete slab*
- *Triaxial tests on a saturated clay*
- *Uniaxial tests on jointed material*
- *Verification of creep integration*
- *Simple tests on a crushable foam specimen*
- *Simple proportional and nonproportional cyclic tests*
- *Biaxial tests on gray cast iron*
- *Indentation of a crushable foam plate*
- *Notched unreinforced concrete beam under 3-point bending*
- *Mixed-mode failure of a notched unreinforced concrete beam*
- *Slider mechanism with slip-rate-dependent friction*
- *Cylinder under internal pressure*
- *Creep of a thick cylinder under internal pressure*
- *Pressurization of a thick-walled cylinder*
- *Stretching of a plate with a hole*
- *Pressure on infinite geostatic medium*

Uniformly loaded, elastic-plastic plate

Products: Abaqus/Standard

Overview

This example verifies the application of a standard rate-independent plasticity theory for metals and assesses the accuracy of the integration of the plasticity equations, especially in the case of nonproportional stressing.

Integration of elastic-plastic material models is a potential source of error in numerical structural analysis. See, for example, the discussions by Krieg and Krieg (1977) and Schreyer et al. (1979). Usually the error is most severe when kinematic hardening is used in plane stress with nonproportional stressing (perhaps because of the complexity of the motion of the stress point and yield surface in stress space in this theory). This example contains two such problems for which the exact solutions are available (Foster Wheeler report, 1972). Experience with a number of other computer programs has suggested that the second example, in particular, is a severe test of the numerical implementation of the plasticity theory. Both problems involve states of uniform plane stress and, hence, are done here by using a single plane stress element.

Problem description

The material models for the uniaxially and biaxially loaded cases are described below.

Case 1—Uniaxial loading

Figure 1 shows the material model for this case. The elastic modulus is 68.94 GPa (10.0×10^6 lb/in^2), the yield stress is 68.9 MPa (10.0×10^3 lb/in^2), and the work hardening slope is 68.9 GPa (10.0×10^6 lb/in^2). This is specified by giving a yield stress of 34.57 GPa (5.01×10^6 lb/in^2) at a plastic strain of 0.5. The total force and the total moment on the loaded face of the model are output to the results file.

Case 2—Biaxial loading

Figure 1 shows the material model for this case. The elastic modulus is 207 GPa (30.0×10^6 lb/in^2), the yield stress is 207 MPa (30.0×10^3 lb/in^2), and the work hardening slope is 11 GPa (1.59×10^6 lb/in^2). This is specified by giving a yield stress of 10.62 GPa (1.53×10^6 lb/in^2) at a plastic strain of 0.95.

Model and loading

The geometries and loading distributions for the uniaxial and biaxial cases are described below.

Case 1—Uniaxial loading

Figure 1 shows the geometry for this case. Two types of meshes are provided: a single-element mesh using higher-order plane stress and shell elements (CPS8R, S8R5, S9R5, and STRI65) and a mesh using linear shell and continuum shell elements (S4R and SC8R). Two edges have simple support. The load history is shown in *Figure 2* and is prescribed with an amplitude curve (*Amplitude Curves*). The load distribution is a uniform, direct stress on the element edge. Since the strain should be uniform, the edge nodes are constrained using an equation constraint (*Linear Constraint Equations*) to move together in the direction normal to the edge. Then the total load on the edge is simply given on one of the edge nodes.

Case 2—Biaxial loading

The case is set up with the same geometric model (*Figure 1*). However, the loading is more complex (see *Figure 2*).

First, the plate is loaded into the plastic range in uniaxial tension in the x-direction, unloaded slightly, and reloaded. Biaxial loading then follows, with σ_x and σ_y prescribed, as shown in *Figure 2*, so that the quantity $\sqrt{\sigma_x^2 + \sigma_y^2 - \sigma_x\sigma_y}$ remains constant at 276 MPa (40000 lb/in^2). This loading is defined by an amplitude curve by reading in a file of values previously calculated in the small program AMP (see *elasticplasticplate_amplitude.f*).

Results and discussion

Exact solutions for these two problems have been developed by Chern in a Foster Wheeler report (1972), where they are documented as Problems 8 and 9. These solutions provide a basis for the comparison of the Abaqus results.

Case 1—Uniaxial loading

The plastic strains are the basic solution in these cases (since stress is prescribed). The results for this case are summarized in *Table 1*. The Abaqus results agree with the exact solution. *Table 1* also records the number of iterations required to achieve equilibrium.

Case 2—Biaxial loading

The results in this case are best represented by the σ_x versus ε_x plot shown in *Figure 3*. The agreement with the exact solution is again very close.

Input files

elasticplasticplate_cps8r_uni.inp	Uniaxial loading case using the CPS8R element.
elasticplasticplate_cps8r_bi.inp	Biaxial loading case using the CPS8R element.
elasticplasticplate_amplitude.f	Program used to generate the amplitude data records.
elasticplasticplate_s8r5_uni.inp	Uniaxial loading case using the S8R5 element.
elasticplasticplate_s8r5_bi.inp	Biaxial loading case using the S8R5 element.
elasticplasticplate_s9r5_uni.inp	Uniaxial loading case using the S9R5 element.
elasticplasticplate_s9r5_bi.inp	Biaxial loading case using the S9R5 element.
elasticplasticplate_stri65_uni.inp	Uniaxial loading case using the STRI65 element.
elasticplasticplate_stri65_bi.inp	Biaxial loading case using the STRI65 element.
elasticplasticplate_s4r_uni.inp	Uniaxial loading case using the S4R element.
elasticplasticplate_s4r_bi.inp	Biaxial loading case using the S4R element.
elasticplasticplate_sc8r_uni.inp	Uniaxial loading case using the SC8R element.
elasticplasticplate_sc8r_bi.inp	Biaxial loading case using the SC8R element.

References

- Foster Wheeler Corporation, “Intermediate Heat Exchanger for Fast Flux Test Facility: Evaluation of the Inelastic Computer Programs,” report prepared for Westinghouse ARD, Foster Wheeler Corporation, Livingston, NJ, 1972.

- Krieg, R. D., and D. B. Krieg, "Accuracies of Numerical Solution Methods for the Elastic-Perfectly Plastic Model," *ASME Journal of Pressure Vessel Technology*, vol. 99, no. 4, pp. 510–515, 1977.
- Schreyer, H. L., R. F. Kulak, and J. M. Kramer, "Accurate Numerical Solutions for Elastic-Plastic Models," *ASME Journal of Pressure Vessel Technology*, vol. 101, no. 3, pp. 226–234, 1979.

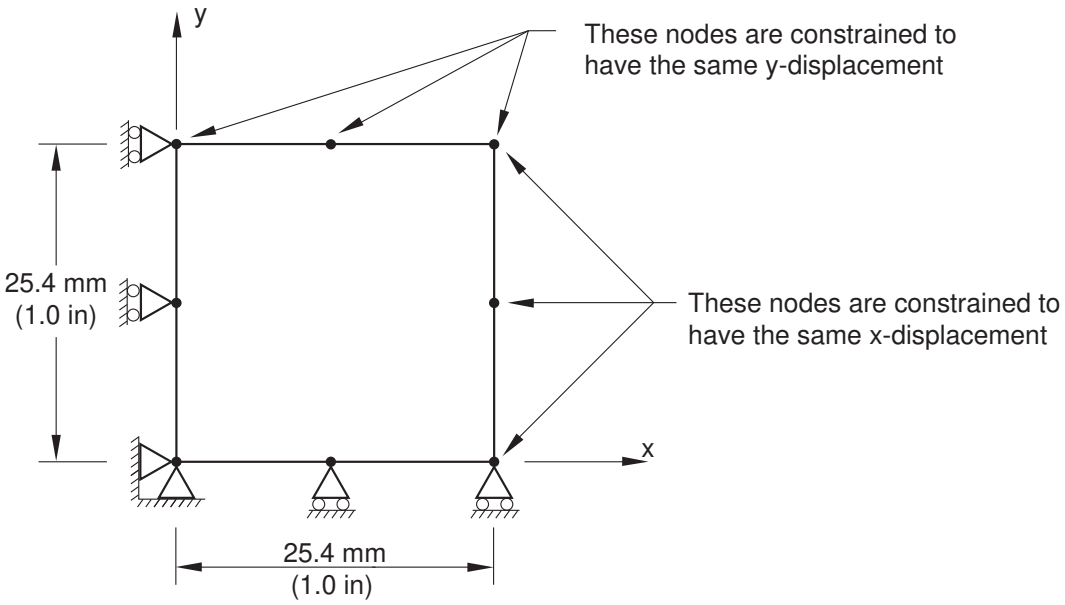
Tables

Table 1: Some results for uniaxial load.

Load increment	Number of iterations	σ_x		ε_x^{pl} (10^{-3})	
		(MPa)	(lb/in^2)	(Abaqus)	(exact)
1	1	68.947	10000	0	0
2	1	103.422	15000	0.500	0.500
3	1	137.895	20000	1.000	1.000
4	1	172.369	25000	1.500	1.500
5	3	86.529	12550	1.500	1.500
6	2	0.69	100	1.010	1.010
7	3	103.77	15050	1.010	1.010
8	2	206.83	30000	2.000	not shown
9	3	103.77	15050	2.000	not shown
10	2	0.69	100	1.010	1.010

Figures

Geometry



Material models

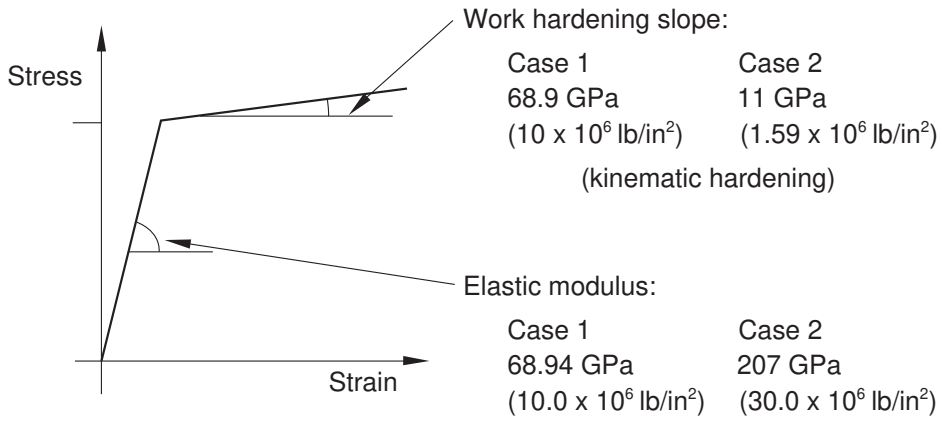
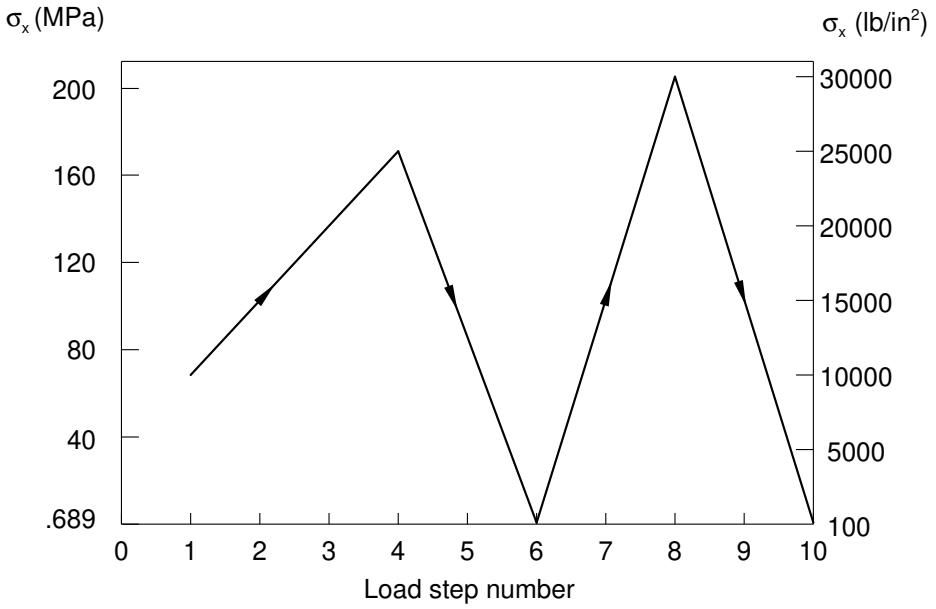


Figure 1: Geometry and material models for plasticity test cases.

Load history, Case 1



Load history, Case 2

Load step	σ_x	
	MPa	lb/in^2
1	206.843	30000
2	224.080	32500
3	241.317	35000
4	258.554	37500
5	275.791	40000
6	279.238	40500
7	275.791	40000
7-84	$(\sigma_x^2+\sigma_y^2-\sigma_x\sigma_y)^{1/2}$ =	276 MPa (40000 lb/in^2)

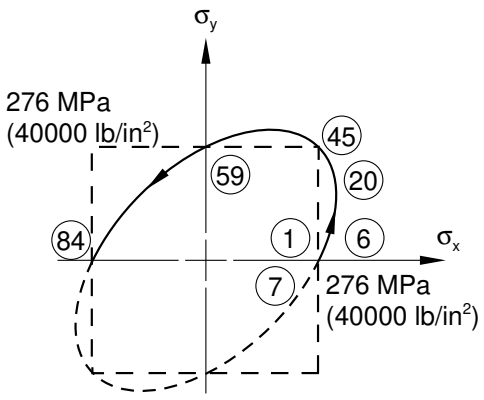


Figure 2: Load histories.

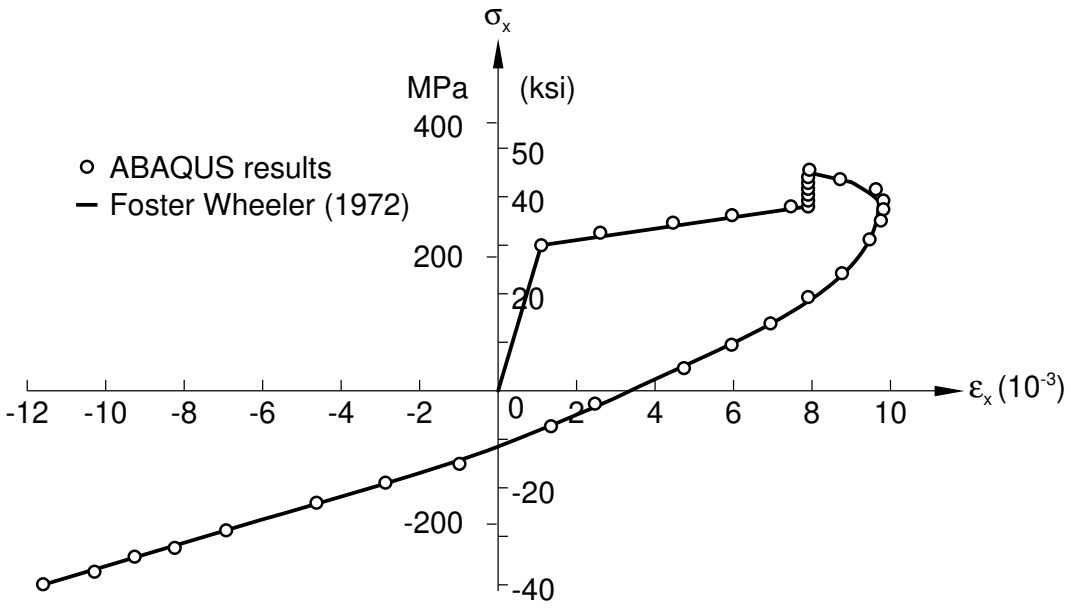


Figure 3: σ_x versus ε_x, biaxially loaded plate.

Test of ORNL plasticity theory under biaxial loading

Products: Abaqus/Standard

Overview

This example verifies the ORNL plasticity theory (Oak Ridge, 1981) model in Abaqus under conditions of plane stress with biaxial stressing.

An exact solution is developed to verify the Abaqus results. The problem involves a state of uniform plane stress, so the geometric model is a single element, constrained to respond uniformly (*ORNL – Oak Ridge National Laboratory Constitutive Model*).

Problem description

The virgin material properties are given by:

Young's modulus	207 GPa (30×10^6 lb/in^2)
Virgin yield stress	207 MPa (30000 lb/in^2)
Virgin work hardening slope	10.3 GPa (1.5×10^6 lb/in^2)
10th cycle yield stress	234 MPa (34000 lb/in^2)
10th cycle work hardening slope	10.3 GPa (1.5×10^6 lb/in^2)

Biaxial loading

The case is set up with the same geometric and virgin material model as in Case 2 in *Uniformly loaded, elastic-plastic plate*. The plate is first loaded elastically to the virgin yield surface in the x-direction and then loaded into the plastic range in uniaxial tension in the x-direction to a stress, σ_0, of 276 MPa (40000 lb/in^2).

Biaxial loading then follows, with σ_x and σ_y prescribed, as shown in *Figure 1*, so that $\sigma_y = \sigma_0 - \sigma_x$. This loading is defined by an amplitude curve (*Amplitude Curves*). Abaqus reads in two files (`ORNL2.AMP` and `ORNL3.AMP`) of values, which are calculated in the small program `AMP` (see *ornlbiaxialload_ampdata.f*).

"Exact" solution

An "exact" solution is developed by first defining the total strain rates, $d\varepsilon_x$ and $d\varepsilon_y$, as functions of the stress rates $d\sigma_x$ and $d\sigma_y$. The resulting rate equations are then integrated numerically with high accuracy to give a reference solution.

Ziegler's kinematic hardening gives, under isothermal conditions,

$$d\boldsymbol{\sigma} = \left[\mathbf{D}^{el} - \frac{1}{d}\mathbf{D}^{el} : \mathbf{n}\,\mathbf{n} : \mathbf{D}^{el}\right] : d\boldsymbol{\varepsilon},$$

where

$$\mathbf{n} = \frac{3}{2\sigma^0}\left(\boldsymbol{\sigma} - \frac{1}{3}\mathbf{I}\ \text{trace}\ (\boldsymbol{\sigma}) - \boldsymbol{\alpha} + \frac{1}{3}\mathbf{I}\ \text{trace}\ (\boldsymbol{\alpha})\right),$$

where σ^0 is the yield stress and

$$d = \mathbf{n}^T : \mathbf{D} : \mathbf{n} + C,$$

where C is the slope of the stress versus plastic strain curve under uniaxial loading conditions.

Under plane stress conditions ($\sigma_z = 0$) and with $\sigma_{xy} = 0$,

$$\mathbf{n} = \begin{Bmatrix} n_x \\ n_y \end{Bmatrix} = \begin{Bmatrix} (\sigma_x - \sigma_y/2 - \alpha_x + \alpha_y)/2\sigma^0 \\ (\sigma_y - \sigma_x/2 - \alpha_y + \alpha_x)/2\sigma^0 \end{Bmatrix}$$

and

$$\mathbf{D} = \frac{E}{1-\nu^2}\begin{bmatrix} 1 & \nu \\ \nu & 1 \end{bmatrix} = \begin{bmatrix} A & B \\ B & A \end{bmatrix},$$

where $A = E/\left(1-\nu^2\right)$ and $B = E\nu/\left(1-\nu^2\right)$.

Hence,

$$d = \mathbf{n}^T : \mathbf{D} : \mathbf{n} + C = An_x^2 + 2Bn_xn_y + An_y^2 + C$$

and

$$\mathbf{D} : \mathbf{n}\mathbf{n}^T : \mathbf{D} = \begin{Bmatrix} P \\ Q \end{Bmatrix}(P\ Q) = \begin{bmatrix} P^2 & PQ \\ PQ & Q^2 \end{bmatrix},$$

where $P = An_x + Bn_y$, $Q = Bn_x + An_y$.

The stress rate-strain rate relation is, therefore,

$$\begin{Bmatrix} d\sigma_x \\ d\sigma_y \end{Bmatrix} = \begin{bmatrix} \widetilde{A} & \widetilde{B} \\ \widetilde{B} & \widetilde{C} \end{bmatrix}\begin{Bmatrix} d\varepsilon_x \\ d\varepsilon_y \end{Bmatrix},$$

where $\widetilde{A} = A - P^2/d$, $\widetilde{B} = B - PQ/d$, and $\widetilde{C} = A - Q^2/d$.

Inverting this relationship gives the total strain rates as

$$\begin{Bmatrix} d\varepsilon_x \\ d\varepsilon_y \end{Bmatrix} = \frac{1}{\widetilde{B}^2 - \widetilde{A}\widetilde{C}}\begin{bmatrix} -\widetilde{C} & \widetilde{B} \\ \widetilde{B} & -\widetilde{A} \end{bmatrix}\begin{Bmatrix} d\sigma_x \\ d\sigma_y \end{Bmatrix}.$$

The center of the yield surface translates according to Ziegler's kinematic hardening rule, so that

$$d\boldsymbol{\alpha} = \frac{Cd\bar{\varepsilon}^{pl}}{\sigma^0}\left(\boldsymbol{\sigma} - \boldsymbol{\alpha}\right),$$

where

$$d\bar{\varepsilon}^{pl} = \frac{\mathbf{n}^T : \mathbf{D} : d\boldsymbol{\varepsilon}}{d}.$$

Hence,

$$d\bar{\varepsilon}^P = \frac{1}{d}(P\ Q)\begin{Bmatrix} d\varepsilon_x \\ d\varepsilon_y \end{Bmatrix} = \left(Pd\varepsilon_x + Qd\varepsilon_y\right)/d,$$

and the translation rate at the center of the yield surface is given in components by

$$\begin{Bmatrix} d\alpha_x \\ d\alpha_y \end{Bmatrix} = \begin{Bmatrix} C\left(Pd\varepsilon_x + Qd\varepsilon_y\right)\left(\sigma_x - \alpha_x\right)/\sigma^0 d \\ C\left(Pd\varepsilon_x + Qd\varepsilon_y\right)\left(\sigma_y - \alpha_y\right)/\sigma^0 d \end{Bmatrix}.$$

Given the values of the variables σ_x, σ_y, ε_x, ε_y, α_x and α_y, at the beginning of the increment, together with the prescribed stress increments $d\sigma_x$ and $d\sigma_y$, the total strain rate equation and the translation rate equation for the center of the yield surface provide the values of $d\varepsilon_x$, $d\varepsilon_y$, $d\alpha_x$, and $d\alpha_y$.

A small program for calculating the required variables is given in *ornlbiaxialload_exact.f*. The main program provides prescribed stress increments $\Delta\sigma_x$ and $\Delta\sigma_y$ equal to those used in the finite element analysis. Each of these increments is then split into 1000 subincrements, and the total strain rate equation and the translation rate equation for the center of the yield surface are integrated over each subincrement to provide virtually exact values of $\Delta\varepsilon_x$, $\Delta\varepsilon_y$, $\Delta\alpha_x$, and $\Delta\alpha_y$, corresponding to the prescribed values of $\Delta\sigma_x$ and $\Delta\sigma_y$ used in the analysis. In each of the subincrements, a test is made to determine if $\boldsymbol{\alpha : d\varepsilon^{pl} < 0}$. When this test is satisfied, the yield surface is expanded from the virgin properties to the 10th cycle properties so that σ^0 is increased from its virgin value of 207 MPa (30000 lb/in^2) to its 10th cycle value of 234 MPa (34000 lb/in^2). This value of σ^0 is used in each subincrement following the initial satisfaction of the test $\boldsymbol{\alpha : d\varepsilon^{pl} < 0}$, according to the ORNL plasticity algorithm.

Results and discussion

The loading path in stress space is shown in *Figure 1*. When the stress contacts point *A*, the yield surface starts to translate so that at point *B* the yield surface occupies the position shown by the dashed curve. At point *B* the stress path changes direction, and elastic loading along path $\boldsymbol{BC}$ occurs. At point *C* the stress point pierces the yield surface, and since $\boldsymbol{\alpha : d\varepsilon^{pl} < 0}$, the ORNL algorithm prescribes an expansion of the yield surface from the virgin properties to the 10th cycle properties. The expanded yield surface is indicated in *Figure 1* by the dashed and dotted curve. Continuing loading along path $\boldsymbol{CD}$ produces an elastic response since point *C* lies inside the 10th cycle yield surface. At point *D* the stress point contacts the expanded yield surface, and active plastic yielding occurs along path $\boldsymbol{DE}$. A comparison between the "exact" results and the finite element results in *Table 1* and *Table 2* shows very close agreement.

Input files

ornlbiaxialload.inp	Biaxial loading test.
ornlbiaxialload_ampdata.f	Program used for generating the data records for the **AMPLITUDE* option.
ornlbiaxialload_exact.f	Program used for generating the "exact" solution.

References

- Nuclear Standard NE F9–5T, "Guidelines and Procedures for Design of Class 1 Elevated Temperature Nuclear System Components," USDOE Technical Information Center, Oak Ridge, Tennessee, March 1981.

Tables

Table 1: Comparison of "exact" and numerical results for biaxial plate using ORNL plasticity theory—stresses and strains in the x-direction.

Numerical solution		"Exact" solution		Increment
σ_x, MPa (10^3 lb/in^2)	ε_x(%)	σ_x, MPa (10^3 lb/in^2)	ε_x(%)	type
206.84 (30.00)	0.1000	206.84 (30.00)	0.1000	Elastic
224.08 (32.50)	0.2656	224.08 (32.50)	0.2655	Plastic
241.32 (35.00)	0.4312	241.32 (35.00)	0.4311	Plastic
258.55 (37.50)	0.5968	258.55 (37.50)	0.5968	Plastic
275.79 (40.00)	0.7624	275.79 (40.00)	0.7624	Plastic
258.55 (37.50)	0.7516	258.55 (37.50)	0.7516	Elastic
68.95 (10.00)*	0.6324	68.95 (10.00)	0.6324	Elastic
51.71 (7.50)	0.6216	51.71 (7.50)	0.6214	Elastic
34.48 (5.00)	0.4702	34.47 (5.00)	0.4744	Plastic
17.24 (2.50)	0.2999	17.24 (2.50)	0.3076	Plastic
3.65 (0.53)	0.1191	0.00 (0.00)	0.1292	Plastic
−17.24 (−2.50)	−0.0709	−17.24 (−2.50)	−0.0591	Plastic
−34.48 (−5.00)	−0.2687	−34.47 (−5.00)	−0.2558	Plastic
−51.71 (−7.50)	−0.4731	−51.71 (−7.50)	−0.4598	Plastic

*The yield surface is pierced at $\sigma_x =$ 68.95 MPa (10000 lb/in^2), $\sigma_y =$ 206.84 MPa (30000 lb/in^2. The next increment is elastic due to the expansion of the yield surface.

Table 2: Comparison of "exact" and numerical results for biaxial plate using ORNL plasticity theory—stresses and strains in the y-direction.

Numerical solution		"Exact" solution		Increment
σ_y, MPa (10^3 lb/in^2)	ε_y(%)	σ_y, MPa (10^3 lb/in^2)	ε_y(%)	type
0.00 (0.00)	−0.0300	0.00 (0.00)	−0.0300	Elastic
0.04 (0.01)	−0.1111	0.00 (0.00)	−0.1108	Plastic
0.04 (0.01)	−0.1923	0.00 (0.00)	−0.1922	Plastic
0.04 (0.01)	−0.2734	0.00 (0.00)	−0.2734	Plastic
0.04 (0.01)	−0.3545	0.00 (0.00)	−0.3545	Plastic
17.24 (2.50)	−0.3437	17.24 (2.50)	−0.3437	Elastic
206.84 (30.00)*	−0.2245	206.84 (30.00)	−0.2245	Elastic
224.08 (32.50)	−0.2137	224.08 (32.50)	−0.2135	Elastic
241.32 (35.00)	0.0313	241.32 (35.00)	0.0319	Plastic
258.55 (37.50)	0.2914	258.55 (37.50)	0.2930	Plastic
275.79 (40.00)	0.5537	275.79 (40.00)	0.5564	Plastic
293.03 (42.50)	0.8172	293.03 (42.50)	0.8211	Plastic

Numerical solution		"Exact" solution		Increment
σ_y, MPa (10^3 lb/in^2)	ε_y(%)	σ_y, MPa (10^3 lb/in^2)	ε_y(%)	type
310.26 (45.00)	1.0811	310.26 (45.00)	1.0860	Plastic
327.51 (47.50)	1.3450	327.50 (47.50)	1.3508	Plastic
*The yield surface is pierced at $\sigma_x =$ 68.95 MPa (10000 lb/in^2), $\sigma_y =$ 206.84 MPa (30000 lb/in^2. The next increment is elastic due to the expansion of the yield surface.				

Figures

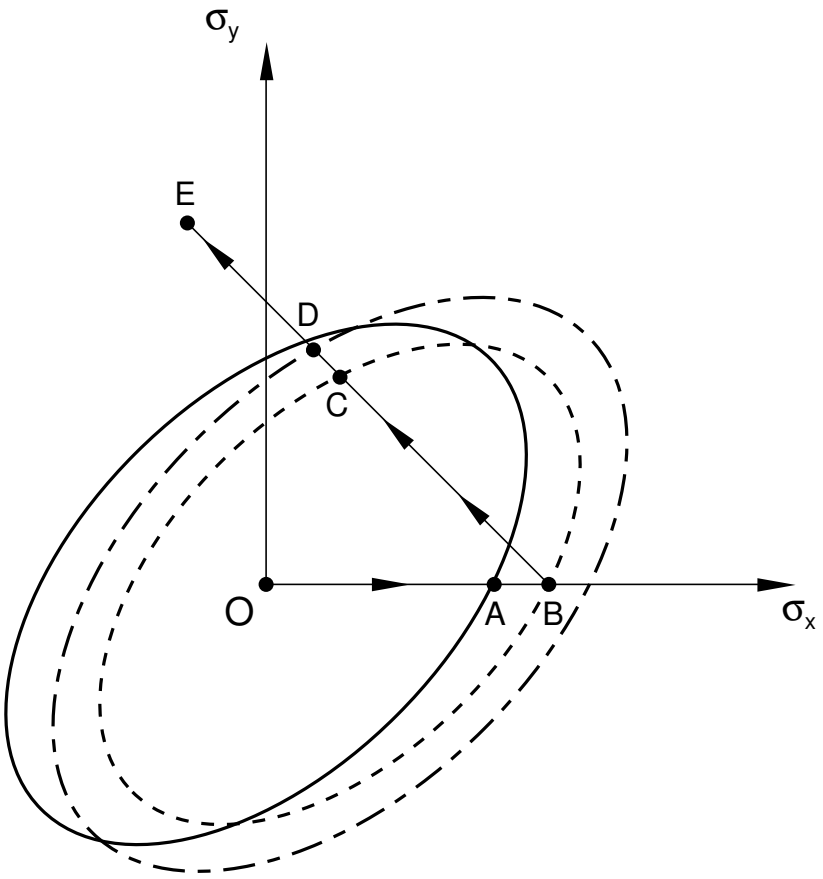


Figure 1: Biaxial stress path in σ_x–σ_y plane for ORNL plasticity solution.

One-way reinforced concrete slab

Products: Abaqus/Standard Abaqus/Explicit

Overview

This problem illustrates the use of the smeared crack model in Abaqus/Standard and the brittle cracking model in Abaqus/Explicit to model reinforced concrete, including cracking the concrete, rebar/concrete interaction using the "tension stiffening" concept, and rebar yield.

Problem description

The structure modeled is a simply supported slab, reinforced in one direction only. The slab is subjected to four-point bending. The local energy release and the concrete-rebar interaction that occur as the concrete begins to crack are of major importance in determining the structure's response between its initial, recoverable deformation and its collapse. The problem is based on an experiment by Jain and Kennedy (1974) and has been analyzed numerically by others (Gilbert and Warner, 1978, and Crisfield, 1982).

The dimensions of the slab and the layout of the reinforcements are shown in *Figure 1*. The symmetry of the problem suggests that only half the slab needs to be modeled.

We assume that the response is essentially one-dimensional but model the slab in Abaqus/Standard as a beam, as a shell, as a continuum, and as a continuum shell to provide verification of the reinforced-concrete modeling capabilities. The response will be uniform in the central section of the slab, so a simple mesh will suffice. The beam and shell models use five elements in the half-slab. The number of concrete integration points through the thickness of the slab is set to nine instead of the default of five points. This provides a smoother response as the cracks propagate through the thickness.

The solid element models use second-order elements or reduced-integration linear elements, because this is a bending problem and the first-order fully integrated elements do a poor job of modeling bending. Two second-order elements are used through the thickness of the slab so there will be enough stress calculation points through the thickness for the response to be reasonably smooth (as in the beam and shell models). Five elements are again used along the half-slab. Because bending is the primary mode of deformation, a minimum of four reduced-integration linear elements (C3D8R or CPS4R) are needed through the thickness of the model to capture the response adequately. Four different CPS4R meshes are used to assess the sensitivity of the results to mesh refinement: a 4×10 mesh, a 4×20 mesh, an 8×10 mesh, and a 4×40 mesh.

Material

The material properties are taken from Gilbert and Warner (1978) and are shown in *Table 1*. The concrete cracking model in Abaqus/Explicit allows unlimited strength in compression. This is a reasonable assumption in this problem, because the behavior of the structure is dominated by cracking due to tension in the slab under bending.

The effects of the concrete rebar interaction and the energy release during cracking are modeled indirectly in Abaqus by adding tension stiffening to the plain concrete model, as illustrated in *Figure 2*. This approach is described in detail in *An inelastic constitutive model for concrete* and *Concrete Smeared Cracking* for Abaqus/Standard and in *A cracking model for concrete and other brittle materials* and *Cracking Model for Concrete* for Abaqus/Explicit. The simplest tension stiffening model, a linear reduction in the tensile strength beyond cracking failure of the concrete, is used in this problem, following Crisfield (1982). To illustrate the effect of tension stiffening parameters on the explicit dynamic response, three different values (5×10^{-4}, 8×10^{-4}, and 11×10^{-4}) are used in the Abaqus/Explicit analysis for the strain beyond failure at which all the tensile strength of the concrete is lost. The Abaqus/Standard analysis uses a value of 5.7×10^{-4} (about 10 times the

failure strain), a typical assumption for standard reinforced-concrete designs that gives a reasonable match to the experimentally measured response of the slab. For illustration purposes the Abaqus/Standard analyses are also run without tension stiffening effects, although this is not recommended as a model for practical cases.

Since the explicit dynamic problem involves pure bending, the response 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. Thus, the choice of shear retention in Abaqus/Explicit has a minimal influence on the results, provided that a reasonable value is used. We have chosen to use a shear retention that is exhausted at a value of crack opening that is 100 times the value at which the tension stiffening is exhausted.

Solution control parameters and loading

Reinforced concrete solutions involve regimes where the load-displacement response is unstable. The Riks procedure in Abaqus/Standard, described in *Modified Riks algorithm*, is designed to overcome difficulties associated with obtaining solutions during unstable phases of the response. It assumes proportional loading and develops the solution by stepping along the load-displacement equilibrium line with the load magnitude included as an unknown. When the Riks method is used, the relative magnitudes of the various loads given on the data lines specify the loading pattern. The actual magnitudes are computed as part of the solution. The user must prescribe loads and provide solution parameters that will give a reasonable estimate of the initial increment of load. If the response is linear, this first increment of load will be the ratio of the initial time increment to the time period, multiplied by the actual load magnitude. If the response is nonlinear, the initial load increment will be somewhat different, depending on the degree of nonlinearity. The termination condition for the analysis is set in this case by specifying a maximum required displacement in the middle of the step as 9 mm (.35 in). This is enough to ensure that a limit condition is reached.

Since Abaqus/Explicit is a dynamic analysis program and in this case we are interested in a static solution, care must be taken that the slab is loaded such that significant inertia effects are avoided. For analyses such as this one, in which the static load-displacement response is unstable, it may be difficult to avoid inertia effects with a dynamic procedure if force-controlled loading is used (even if the forces are ramped on slowly). Displacement-controlled loading is often a viable alternative. In this problem the slab is loaded by applying a velocity that increases linearly from 0.0 to 5.0 in/second over 0.1 seconds. This loading causes a midspan deflection of approximately 0.3 in. The loading is slow enough to ensure that quasi-static solutions are obtained.

The boundary conditions are symmetric about $x = 0$ (all nodes along $x = 0$ have $u_x = 0$ prescribed) and, for the C3D8R models, symmetric about $z = -1.5$ in (all nodes along $z = -1.5$ in have $u_z = 0$ prescribed). All the nodes along the bottom edge ($y = -0.75$ in) at $x = 15$ in have $u_y = 0$.

Results and discussion

Results for all analyses are discussed in the following sections.

Abaqus/Standard results

The Abaqus/Standard analyses are compared with the experimental response on the basis of the deflection at the middle of the slab plotted versus the moment per unit width on that section of the slab. *Figure 3* shows the analyses that do not include tension stiffening, and *Figure 4* shows those that do include tension stiffening in the manner described above for the beam, shell, and continuum models. The experimental data obtained by Jain and Kennedy (1974) are also plotted on these figures. In the analysis without tension stiffening the initial cracking of the concrete causes a loss of strength in the slab, while the inclusion of tension stiffening eliminates this drop in load even though the concrete is cracking. The cracks propagate rapidly through the slab, until collapse occurs as the rebar yields. The collapse load is well predicted by all the models, and the various geometric models are reasonably consistent both with and without tension stiffening. The improvement in predicting the actual response obtained from including tension stiffening is obvious when the two figures are compared, graphically illustrating the need for including this effect in the model.

The results for the continuum shell element analysis are similar to results obtained from the S8R model.

Abaqus/Explicit results

Figure 5 shows the 4 × 20 mesh that was used in the Abaqus/Explicit analysis. *Figure 6* shows the deformed shape at $t = 0.1$, which is the point of full load application.

The load-deflection response of the slab for the four different mesh densities using a tension stiffening value of 8×10^{-4} and CPS4R elements is shown in *Figure 7*. Meshes with 10 elements along the length predict a slightly higher limit load than the mesh with 20 elements along the length. The mesh with 40 elements along the length of the slab gives results that are nearly identical to those given by the mesh with 20 elements. The tension stiffening study described next is, therefore, performed using the 4 × 20 mesh.

The results using the 4 × 20 mesh of CPS4R elements are compared to the experimental data in *Figure 8* for three different values of tension stiffening. It is clear that the less tension stiffening used, the softer the load-deflection response will be during the cracking of the concrete. The middle value of tension stiffening appears to match the experimental data best. The load-deflection responses during the latter part of the analyses are almost entirely governed by the yield in the rebar and are, therefore, nearly independent of the tension stiffening.

The results using the 4 × 20 mesh of C3D8R elements with the various values of tension stiffening are compared with the experimental data in *Figure 9*. The results using a 2 × 10 mesh of S4R elements with the various values of tension stiffening are compared with the experimental data in *Figure 10*. The results for both C3D8R and S4R elements are similar to those obtained with the CPS4R elements.

Input files

Abaqus/Standard input files

onewayconcreteslab_b21.inp	Slab modeled as a beam with tension stiffening.
onewayconcreteslab_s8r.inp	Slab modeled with shell elements of type S8R with tension stiffening.
onewayconcreteslab_cps8.inp	Slab using element type CPS8 (plane stress) with tension stiffening.
onewayconcreteslab_cpe8.inp	Slab using element type CPE8 (plane strain) with tension stiffening.
onewayconcreteslab_c3d20.inp	Slab using element type C3D20 with tension stiffening.
onewayconcreteslab_sc8r.inp	Slab modeled with shell elements of type SC8R without tension stiffening.

Abaqus/Explicit input files

jainkennedy1.inp	Slab modeled with 40 CPS4R elements (4 × 10 mesh) using a tension stiffening value of 8.0×10^{-4}.
jainkennedy2.inp	Slab modeled with 80 CPS4R elements (4 × 20 mesh) using a tension stiffening value of 8×10^{-4}.
jainkennedy3.inp	Slab modeled with 80 CPS4R elements (8 × 10 mesh) using a tension stiffening value of 8×10^{-4}.
jainkennedy4.inp	Slab modeled with 80 CPS4R elements (4 × 20 mesh) using a tension stiffening value of 5×10^{-4}.
jainkennedy5.inp	Slab modeled with 80 CPS4R elements (4 × 20 mesh) using a tension stiffening value of 11×10^{-4}.

jainkennedy6.inp	Slab modeled with 80 C3D8R elements (4 × 20 mesh) using a tension stiffening value of 5×10^{-4}.
jainkennedy7.inp	Slab modeled with 80 C3D8R elements (4 × 20 mesh) using a tension stiffening value of 8×10^{-4}.
jainkennedy8.inp	Slab modeled with 80 C3D8R elements (4 × 20 mesh) using a tension stiffening value of 11×10^{-4}.
jainkennedy9.inp	Slab modeled with 160 CPS4R elements (4 × 40 mesh) using a tension stiffening value of 8×10^{-4}.
jainkennedy10.inp	Slab modeled with 20 S4R elements (2 × 10 mesh) using a tension stiffening value of 5×10^{-4}.
jainkennedy11.inp	Slab modeled with 20 S4R elements (2 × 10 mesh) using a tension stiffening value of 8×10^{-4}.
jainkennedy12.inp	Slab modeled with 20 S4R elements (2 × 10 mesh) using a tension stiffening value of 11×10^{-4}.

References

- Crisfield, M. A., "Variable Step-Lengths for Nonlinear Structural Analysis," *Report 1049, Transport and Road Research Lab, Crowthorne, England*, 1982.
- Gilbert, R. J., and R. F. Warner, "Tension Stiffening in Reinforced Concrete Slabs," *Journal of Structural Division, American Society of Civil Engineering*, vol. 104, ST12, pp. 1885–1900, 1978.
- Jain, S. C., and J. B. Kennedy, "Yield Criterion for Reinforced Concrete Slabs," *Journal of Structural Division, American Society of Civil Engineering*, vol. 100, ST3, pp. 631–644, 1974.

Tables

Table 1: Assumed material properties for one-way slab. Reinforcement ratio (volume of steel: volume of concrete) 7.2×10^{-3}.

Concrete properties	
Young's modulus:	29 GPa (4.2×10^6 lb/in^2)
Poisson's ratio:	0.18
Yield stress:	18.4 MPa (2670 lb/in^2)
Failure stress:	32 MPa (4640 lb/in^2)
Plastic strain at failure:	1.3×10^{-3}
Ratio of uniaxial tensile to compressive failure stress:	6.25×10^{-2}
Density:	2400 kg/m^3 (2.246×10^{-4} lbf s^2/in^4)
Cracking failure stress:	2 MPa (290 lb/in^2)
In the Abaqus/Explicit analyses "tension stiffening" is assumed as a linear decrease of the stress to zero stress at a direct cracking strain of 5×10^{-4}, 8×10^{-4}, or 11×10^{-4}.	
Steel (rebar) properties	
Young's modulus:	200 GPa (29×10^6 lb/in^2)
Yield stress:	220 MPa (31900 lb/in^2) (Perfectly plastic)

Figures

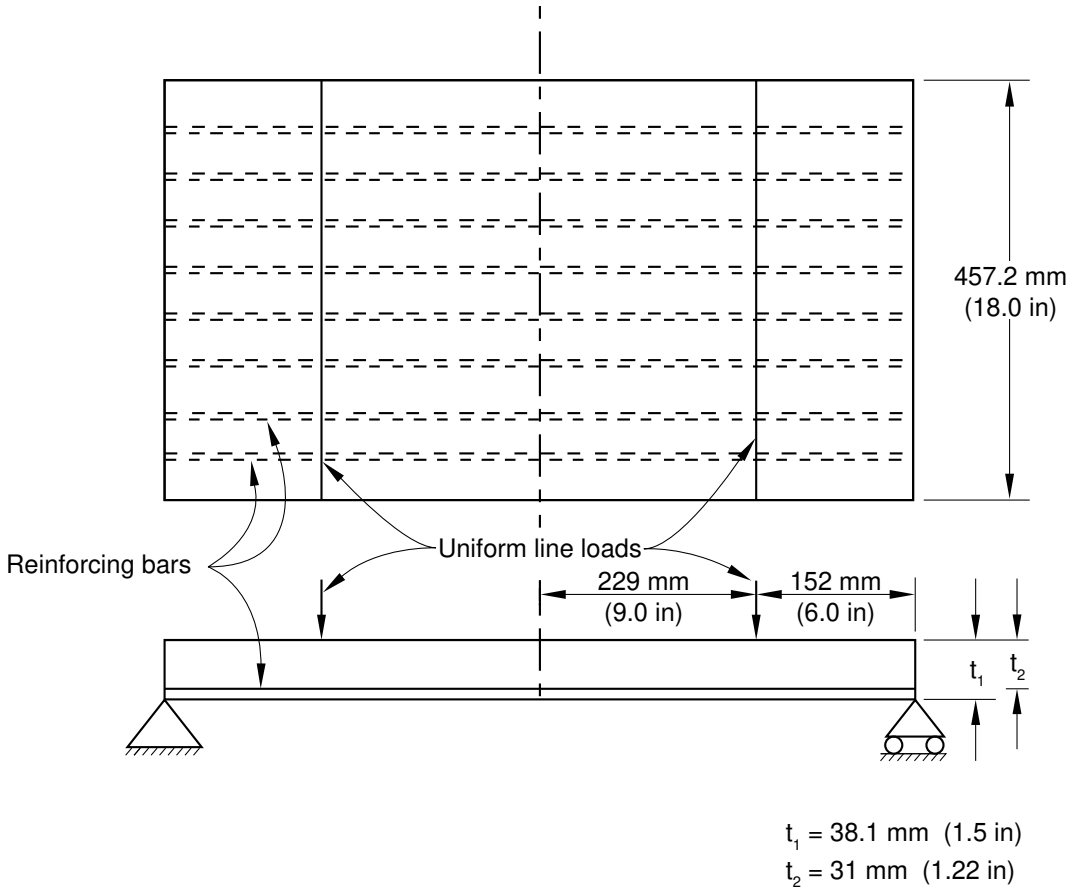
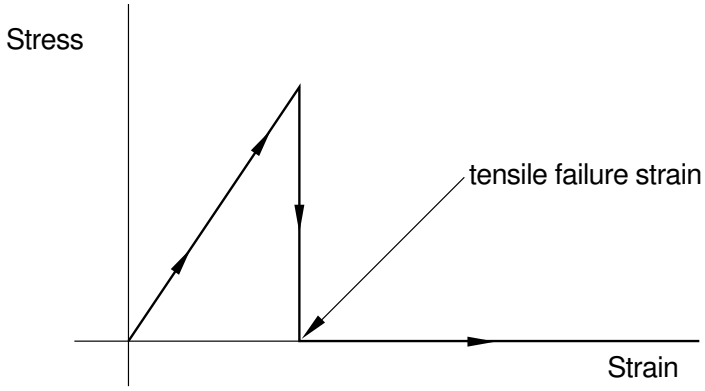
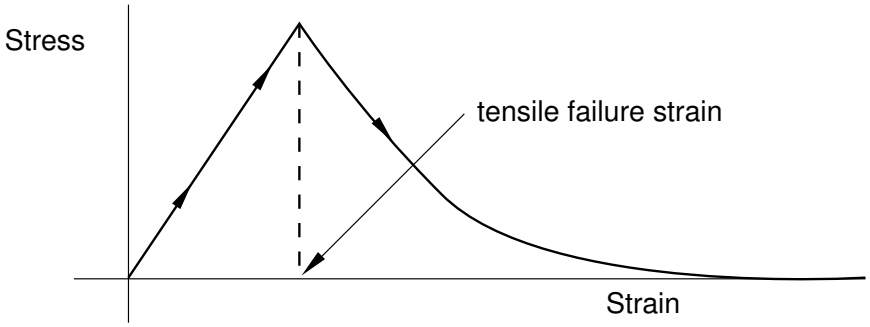


Figure 1: One-way reinforced concrete slab.



a) Plain concrete model in tension.



b) Modified plain concrete model in tension, including "tension stiffening."

Figure 2: Tension stiffening effect.

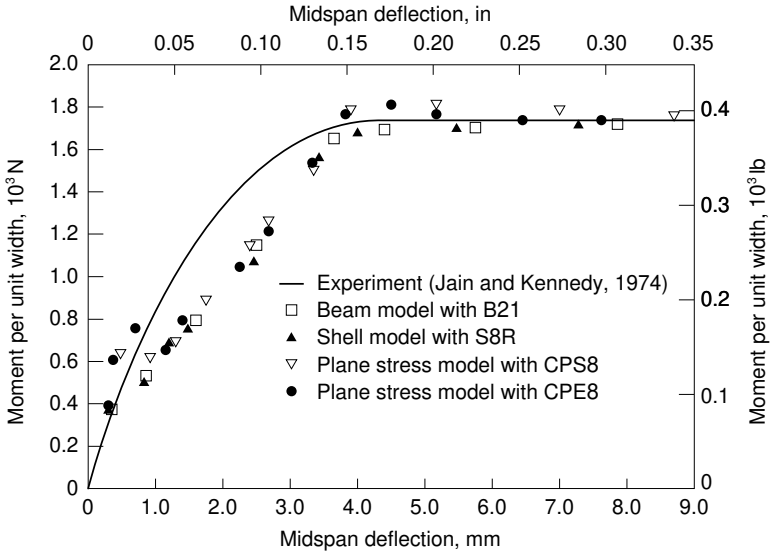


Figure 3: Moment-deflection response with no tension stiffening (Abaqus/Standard).

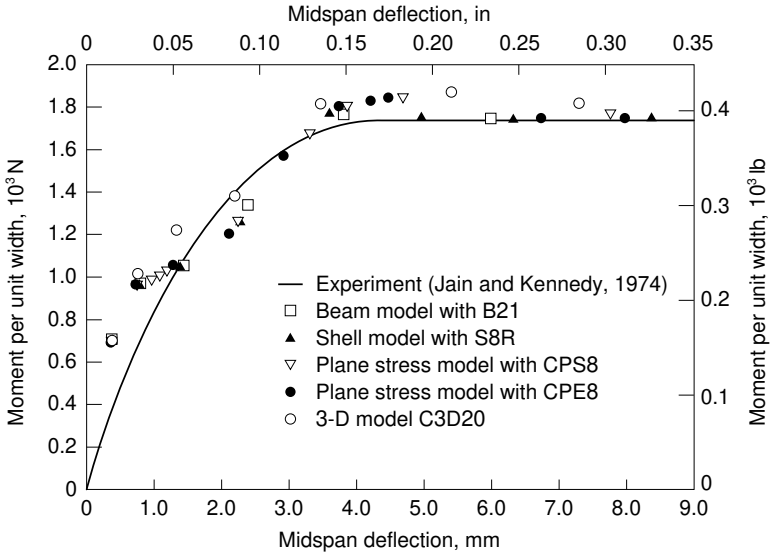


Figure 4: Moment-deflection response with tension stiffening (Abaqus/Standard).

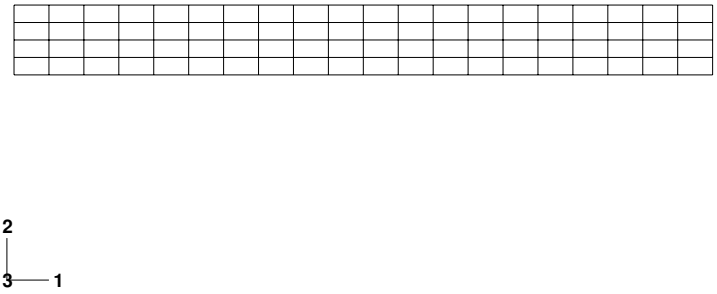


Figure 5: Undeformed CPS4R 4 × 20 mesh (Abaqus/Explicit).

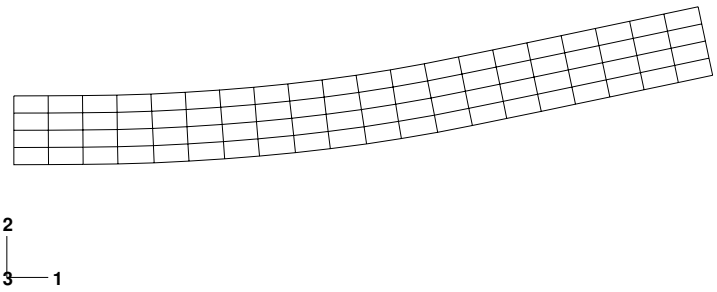


Figure 6: Deformed CPS4R mesh (Abaqus/Explicit). Deformation is magnified by a factor of 5.

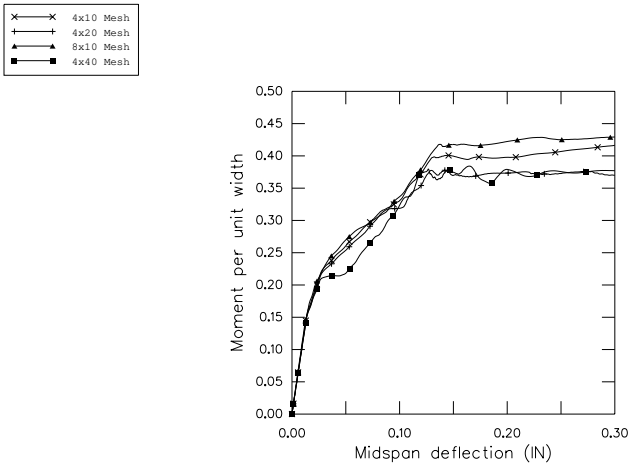


Figure 7: Moment-deflection response of Jain and Kennedy slab; influence of mesh refinement. CPS4R elements (Abaqus/Explicit).

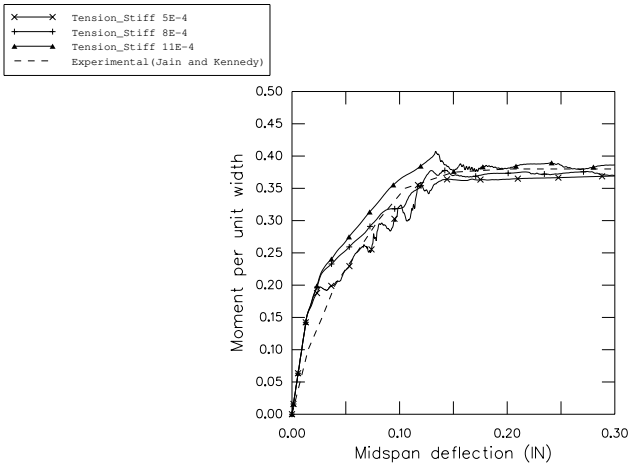


Figure 8: Moment-deflection response of Jain and Kennedy slab; influence of tension stiffening on 4 × 20 mesh. CPS4R elements (Abaqus/Explicit).

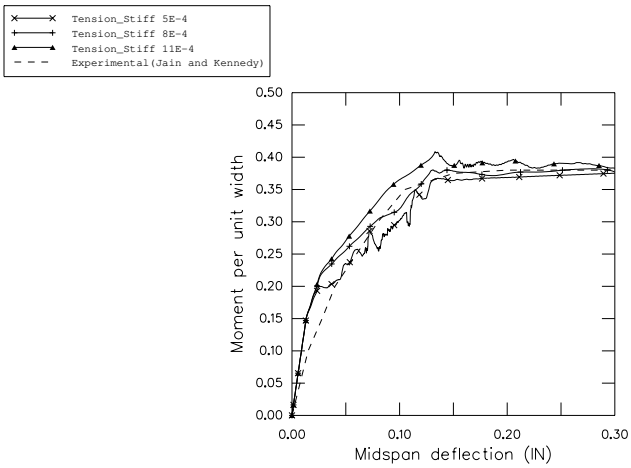


Figure 9: Moment-deflection response of Jain and Kennedy slab; influence of tension stiffening on 4 × 20 mesh. C3D8R elements (Abaqus/Explicit).

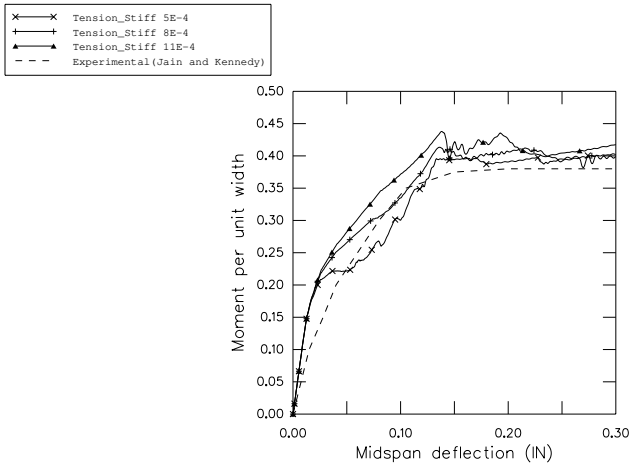


Figure 10: Moment-deflection response of Jain and Kennedy slab; influence of tension stiffening on 2 × 10 mesh. S4R elements (Abaqus/Explicit).

Triaxial tests on a saturated clay

Products: Abaqus/Standard

Overview

This example demonstrates the modified Cam-clay plasticity model provided in Abaqus.

Cam-clay theory provides a reasonable match to the experimentally observed behavior of saturated clays and belongs to the family of critical state plasticity models developed by Roscoe and his colleagues (see Roscoe and Burland—1968—and Schofield and Wroth—1968).

The Cam-clay model in Abaqus permits two extensions of the original Roscoe model: "capping" of the yield ellipse on the wet side of critical state, and consideration of the third stress invariant in the yield function. Both of these extensions to the modified Cam-clay theory are documented in *Plasticity for nonmetals* and are included in this example.

The general modified Cam-clay yield function (for isotropic plasticity and zero tensile strength) used in Abaqus is

$$\frac{1}{\beta^2}\left(\frac{p}{a}-1\right)^2+\left(\frac{q}{gMa}\right)^2-1=0,$$

where the three stress invariants are the equivalent pressure stress given by

$$p=-\frac{1}{3}\,\mathrm{trace}\,(\boldsymbol{\sigma});$$

the equivalent shear stress given by

$$q=\sqrt{\frac{3}{2}\mathbf{S}:\mathbf{S}},$$

where $\mathbf{S}$ is the deviatoric stress ($\mathbf{S}=\boldsymbol{\sigma}+p\mathbf{I}$); and the third stress invariant,

$$r=\left(\frac{9}{2}\,\mathbf{S}:\mathbf{S}\cdot\mathbf{S}\right)^{\frac{1}{3}}.$$

The other parameters in the function are a, the value of the equivalent pressure stress at critical state; M, a material parameter defining the slope of the critical state lines; β, a "capping" parameter used to provide a different shaped yield ellipse on the wet side of critical state; and g, a function that is dependent on the third stress invariant, used to define different yield surface sizes in compression and extension:

$$g=\frac{2K}{1+K+(1-K)r^3/q^3},$$

where K is a material parameter.

The "standard" Cam-clay yield function has $\beta=g=1$. Including these parameters in the yield surface expression generalizes that expression to allow closer matching of data under various conditions of loading.

Problem description

The material parameters used in this example are as follows:

Elasticity:

Logarithmic bulk modulus, κ:	0.026
Poisson's ratio, ν:	0.3

Plasticity:

Logarithmic hardening modulus, λ:	0.174
Critical state ratio, M:	1.0
Wet cap parameter, β:	0.5
Third stress invariant parameter, K:	0.75
Initial overconsolidation parameter, α_0:	58.3 kN/m^2 (8.455 lb/in^2)

The example studies a simple triaxial test: an axisymmetric soil sample contained between two smooth platens, one of which is held fixed and the other of which has prescribed vertical motion, positive for extension and negative for compression. The soil specimen is first loaded by constant pressure. Then the top platen is moved, either downward to test triaxial compression or upward to test triaxial extension. *Figure 1* defines the problem geometry. The analyses are meant to simulate drained triaxial tests; therefore, they can be run with the pure displacement elements in Abaqus.

Since the platens are assumed to be smooth and the soil is homogeneous, the stress will be constant throughout the model. For simplicity, large displacement effects are ignored.

Results and discussion

For both cases the initial pressure stress is given via initial conditions, and an initial geostatic step (*Geostatic Stress State*) is included in which the confining pressure is applied to the outer surface of the specimen. At the start of a soils analysis with initial stresses, Abaqus checks to see that the stress specified does not violate the initial yield surface. If it does, the hardening value (a in the yield surface definition above) is modified to make the yield surface consistent with the stress state. To test this part of the code, in the present example the initial stress state lies within the initial yield surface when the "standard" Cam-clay plasticity theory is used, but it violates the yield criterion with the given initial overconsolidation parameter, α_0, when the "capped" plasticity theory is used. The adjustment to the value of α_0 is shown in *Figure 2*.

It is recommended that a geostatic procedure always be included at the start of a soils analysis to ensure compatibility between the initially prescribed stress state and the initial loading.

Compression of a drained triaxial specimen

In this case, during the second step of the analysis the top platen is moved down by half the soil sample height. The material response is shown in *Figure 3*. Depending on the theory used, the soil yields more or less gradually as the displacement increases until critical state is reached (that is, when $q = Mp$: see *Figure 2*) when the response is perfectly plastic. "Capping" has a strong effect on the material response: for the load path specified (line OS on *Figure 2*) the "capped" theory predicts that critical state will be reached at a normalized vertical displacement of 0.18, whereas the "standard" Cam-clay theory predicts that critical state will not be reached until the soil sample has been reduced in height by half. It should be emphasized that these results have been obtained with the small-displacement assumption; although the stress-strain response is accurate, the load-displacement response is not because the strains are well beyond the range where linearized strain-displacement relations are reasonable.

Extension of a drained triaxial specimen

In this case, in the second step the top platen is moved vertically upward. This decreases the confining pressure in the soil, so critical state is reached at a lower value of equivalent shear stress than in the compression case. This is clearly seen in *Figure 3*. Of interest here is the effect of the third stress invariant on the plasticity solution: this dependence is specified via the parameter K (see the *Introduction: general* for a complete discussion). Since the present case is pure triaxial extension, the critical state condition becomes $q = KMp$. As seen in *Figure 4*, this has the effect of lowering the achievable equivalent shear stress states by flattening the yield surface in p–q

space. For the load path specified here, the solution follows line OS on *Figure 4* for the "standard" Cam-clay theory and line OS_1 for the case that includes dependence on the third stress invariant.

Input files

triaxialtestclay_cax8r.inp	Compression of a drained triaxial soil specimen modeled with "standard" modified Cam-clay theory. A single CAX8R element is used in this model.
triaxialtestclay_caxa8r1.inp	Same analysis using the CAXA element. A single CAXA8R1 element is used.

Input files for the other cases described in this example are created by including the parameters β and K in the definition of the Cam-clay model.

References

- Roscoe, K. H., and J. B. Burland, Stress-strain Behavior of `Wet Clay', Engineering Plasticity, J. Heyman and F. A. Leckie, Editors, Cambridge University Press, 1968.
- Schofield, A., and C. P. Wroth, Critical State Soil Mechanics, McGraw-Hill, 1968.

Figures

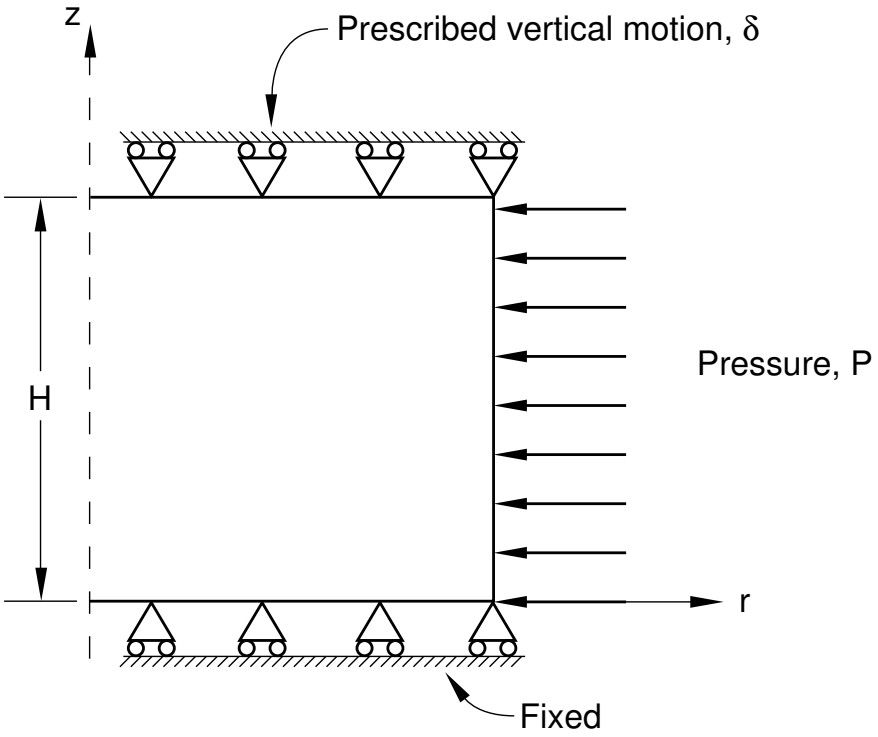


Figure 1: Triaxial test with smooth platens.

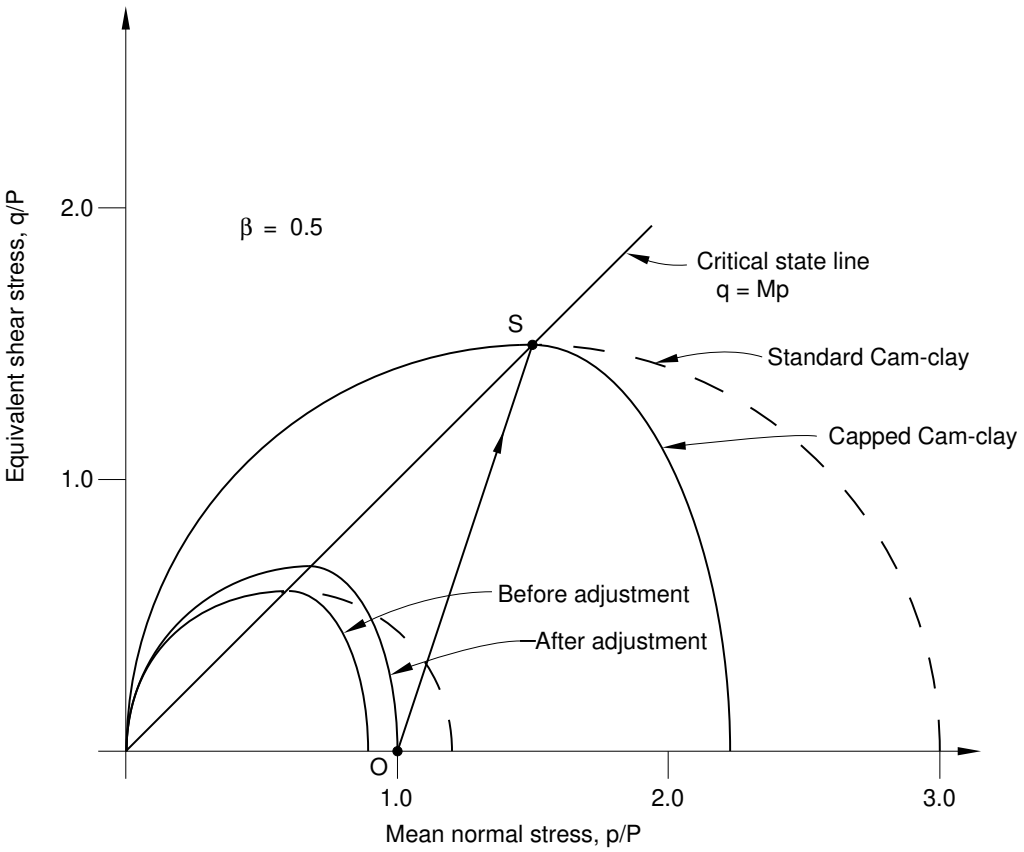


Figure 2: Yield surface profiles for triaxial compression solutions.

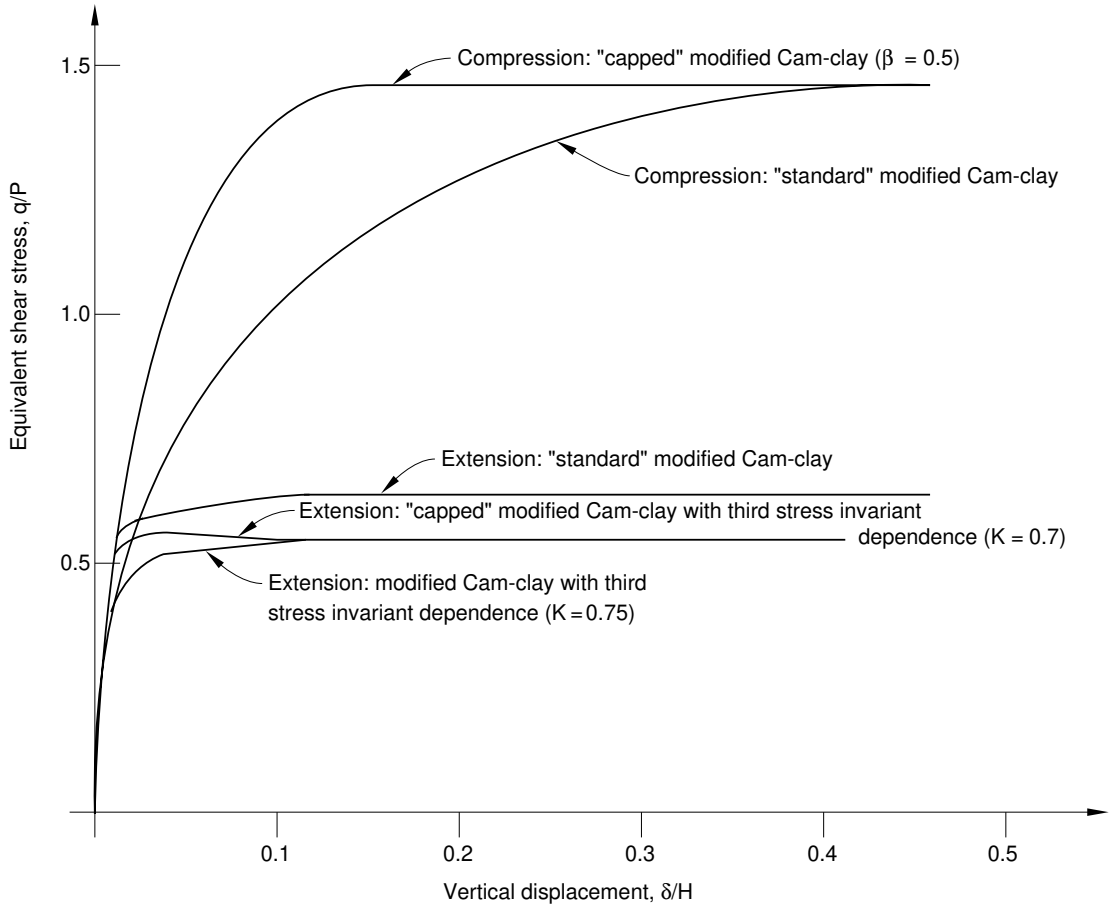


Figure 3: Modified Cam-clay plasticity response.

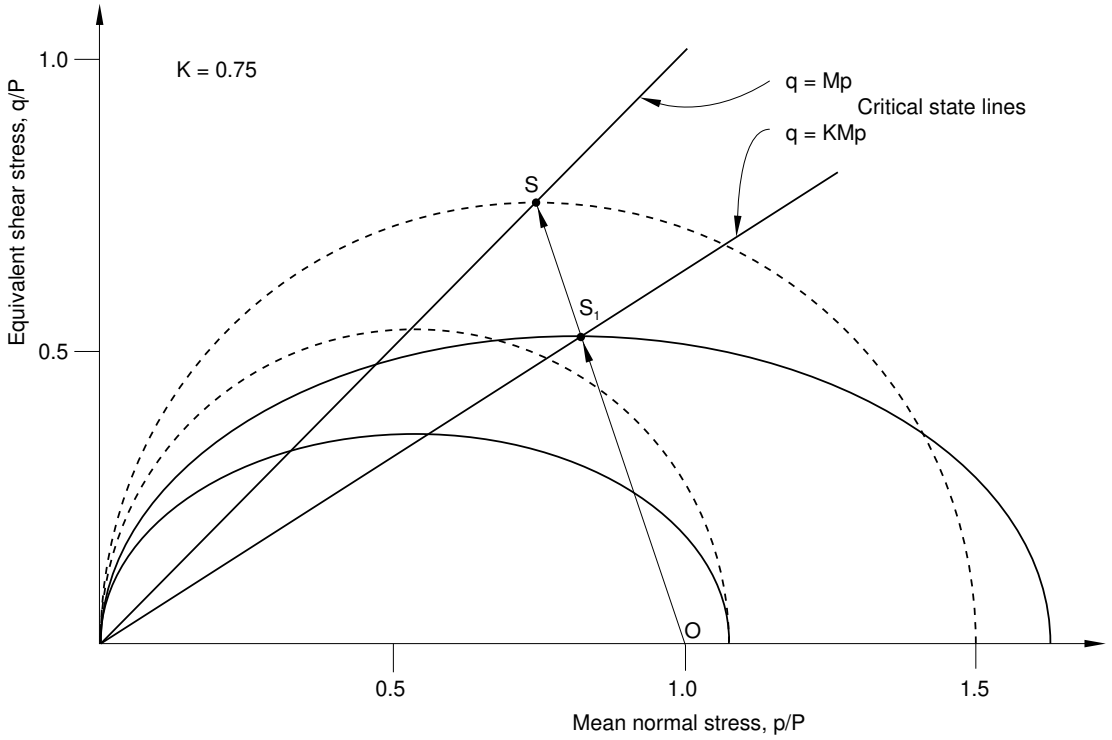


Figure 4: Yield surface profiles for triaxial extension solutions.

Uniaxial tests on jointed material

Products: Abaqus/Standard

Overview

This example illustrates the fundamental material behavior obtained with the jointed material model in Abaqus. We construct a failure envelope for a material containing two sets of joints and subjected to uniaxial stress conditions. A complete description of the model is given in *Constitutive model for jointed materials*.

Problem description

We consider a sample of material subjected to uniaxial compression/tension. The material has two sets of planes of weakness having an included angle of 2α. We seek to construct the failure envelope of the material as the orientation (θ in *Figure 1*) of the planes of weakness is varied.

In the Abaqus model the failure surface for sliding on the joint systems is defined as

$$f_a = \tau_a - p_a \tan\beta_a - d_a = 0,$$

where p_a is the pressure stress across the joint, τ_a is the shear stress magnitude in the joint, β_a is the friction angle of the joint, and d_a is its cohesion. For this problem we assume that for both joints $d_a = 1000$ (the units are not important), $\beta_a = 45°$, and that plastic flow in the joints is associated.

The behavior of the bulk material is based on the Drucker-Prager failure criterion

$$q - p\tan\beta_b - d_b = 0,$$

where $q \overset{\text{def}}{=} \sqrt{\frac{3}{2}\mathbf{S}\mathbf{S}}$ is the Mises equivalent deviatoric stress (here $\mathbf{S}$ is the deviatoric stress $\mathbf{S} \overset{\text{def}}{=} \boldsymbol{\sigma} + p\mathbf{I}$), $p \overset{\text{def}}{=} -\frac{1}{3}\mathbf{I} : \boldsymbol{\sigma}$ is the equivalent pressure stress, β_b is the friction angle of the bulk material, and d_b is the cohesion of the bulk material. For this problem we assume that $d_b = 8000$, $\beta_b = 45°$, and that plastic flow of the bulk material is associated.

When all the joints are closed, the material is assumed to be isotropic and linear elastic with a Young's modulus of 3×10^5 and a Poisson's ratio of 0.3. When a joint opens, the material is assumed to have no elastic stiffness with respect to direct strain across the joint system or with respect to shearing associated with this direction. Open joints, thus, create anisotropic elastic response.

Each test performed in this example is carried out using a cube (one C3D8 element) of unit dimensions. Displacements are prescribed at the nodes of the cube to simulate homogeneous deformation and stress conditions.

Results and discussion

Figure 2 shows the variation of the compressive failure stress $P/2d_a$ with θ, the angle which the bisector of the joints forms with the direction of the load. Compression failure envelopes are developed for $\alpha = 0°, 20°, 30°$. It is clear that for certain ranges of orientation of the joints with respect to the loading direction, failure along the planes of weakness becomes increasingly improbable and failure of the bulk material takes place first. It may be noted that the case when $\alpha = 0°$ corresponds to the theory of a single plane of weakness proposed by Jeager (1960).

When the load is applied in tension, the material cannot carry any stress since the joints open readily.

Input files

jointedmat_comp_alpha20thete0.inp	Compression test with $\alpha = 20$ and $\theta = 0$.
jointedmat_comp_alpha20thete20.inp	Compression test with $\alpha = 20$ and $\theta = 20$.
jointedmat_tens_alpha0theta10.inp	Tension test with $\alpha = 0$ and $\theta = 10$.
jointedmat_tens_alpha20theta20.inp	Tension test with $\alpha = 20$ and $\theta = 20$.
jointedmat_comp_pert.inp	A version of jointedmat_comp_alpha20thete0.inp including linear perturbation steps.
jointedmat_tens_pert.inp	A version of jointedmat_tens_alpha0theta10.inp including linear perturbation steps.

The remaining cases analyzed in this example problem can be generated by changing the orientation of the joints.

References

- Jeager, J. C., "Shear Failure of Anisotropic Rocks," *Geological Magazine*, vol. 97, pp. 65–72, 1960.

Figures

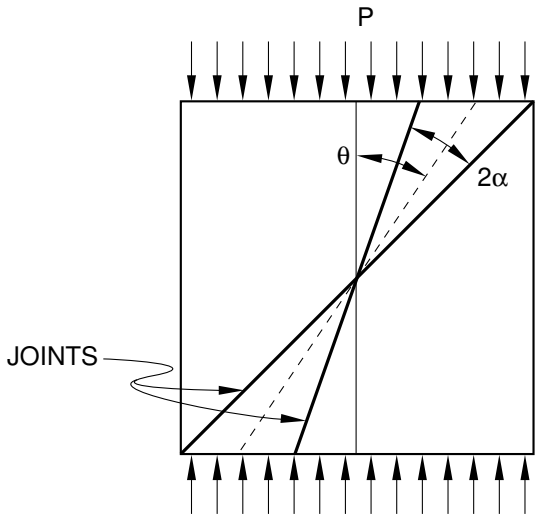


Figure 1: Problem geometry.

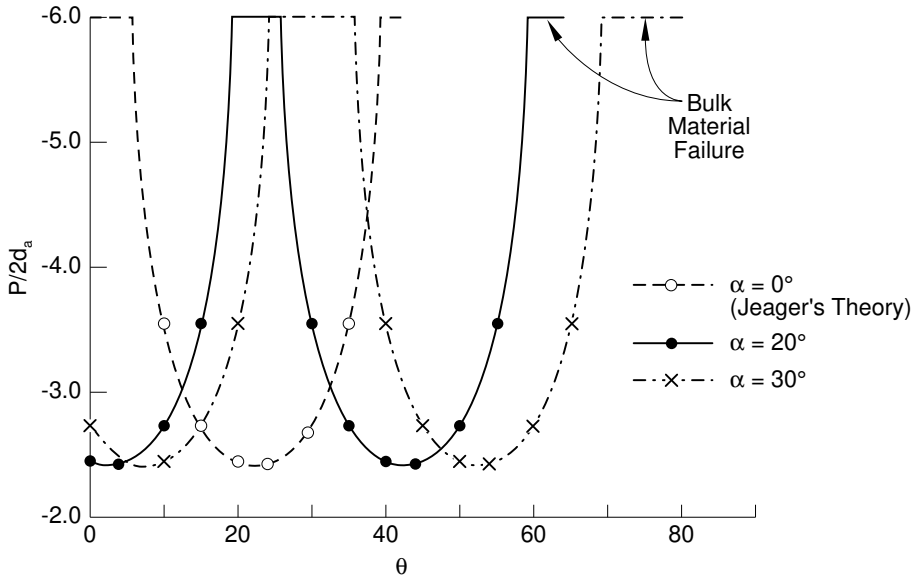


Figure 2: Uniaxial compression failure envelopes.

Verification of creep integration

Products: Abaqus/Standard

Overview

This example verifies the algorithms used to integrate creep constitutive behavior by comparison with closed-form solutions.
In the first part of the example the constitutive creep behavior for simple creep and relaxation tests are verified. In the second part solutions for the Mises, the Drucker-Prager, and the extended Drucker-Prager/Cap creep and plasticity models are verified for tests in which the load is ramped from zero over a given period of time.

Problem description

Model for simple creep and relaxation tests

The model for the simple creep and relaxation tests contains eight independent, single-element specimens, as shown in *Figure 1*. The input files for both the creep and relaxation tests are given in *creep_usr_creep.inp* and *creep_relax_usr_creep.inp*. All of the elements are plane stress elements. The plane stress case provides the most rigorous verification because plane stress is usually the most difficult case for integrating the type of rate constitutive model that arises in classical metal creep theories. The eight elements are divided into two groups of four. One group is subjected to a creep test, and the other group models a relaxation experiment. Creep behavior can be defined in Abaqus directly on data lines (for simple creep laws) or by a user subroutine. In either case time or strain hardening can be used. To test all of these possibilities, the material definitions for the elements are set up as follows:

Material behavior and method of input	Element number for creep test	Element number for relaxation test
Data defined time hardening	1	5
Data defined strain hardening	2	6
User subroutine defined time hardening	3	7
User subroutine defined strain hardening	4	8

Models for coupled creep and plasticity tests

The models used to verify the integration of coupled creep and plasticity constitutive behavior consist of a single element of unit dimension. The test case employing the Mises model (*creep_mises.inp*) uses a plane stress element simulating a tensile test. The test cases using the Drucker-Prager model (*creep_druckercap_ramp.inp*) and extended Drucker-Prager/Cap model (*creep_druckercap_ramp.inp*) use a solid element. The first simulates a tensile test, and the second, a compression test. A time hardening creep law is used for the Mises and the Drucker-Prager test cases; a Singh-Mitchell type creep law is defined for the modified Drucker-Prager/Cap model.

Material

For all test cases the elastic material properties use a Young's modulus of 138 GPa (20×10^6 psi) and a Poisson's ratio of 0.3. In the creep and relaxation tests (first part of the example) the material definition uses creep behavior

with the Mises stress potential and the equivalent uniaxial creep strain rate defined by $\dot{\varepsilon}^{cr} = Aq^n t^m$, where q is the Mises equivalent stress, t is time in the time hardening case, or—in the strain hardening case—

$$t = \left(\frac{m+1}{A}\frac{\varepsilon^{cr}}{q^n}\right)^{1/(m+1)}.$$

A, n, and m are constants, which are defined here as $A = 1.6 \times 10^{-16}$ MPa^{-5} sec$^{-0.8}$ (2.5×10^{-27} psi^{-5}sec$^{-0.8}$), $n = 5$, and $m = -0.2$.

For the strain hardening case t can be eliminated from the creep strain rate definition, giving

$$\dot{\varepsilon}^{cr} = (Aq^n)^{1/(m+1)}((1+m)\,\varepsilon^{cr})^{m/(m+1)}.$$

The time hardening creep law defined above with $A = 1.6 \times 10^{-16}$ MPa^{-5} sec$^{-0.8}$ (2.5×10^{-27} psi^{-5}sec$^{-0.8}$) is specified for the coupled Mises and the coupled Drucker-Prager models. The plasticity hardening curve is given by

$$\sigma = \sigma_0\left(1 + \left(\bar{\varepsilon}^{pl}\right)^{\mu}\right),$$

where $\bar{\varepsilon}^{pl}$ is the equivalent plastic strain, $\sigma_0 = 69$ MPa (1×10^4 psi), and $\mu = 0.2$. The Drucker-Prager model is reduced to a Mises model by specifying the material angle of friction, $\beta = 0.0$, and the dilation angle, $\psi = 0.0$. No intermediate principal stress effect is used (i.e., $K = 1.0$), as is required by this type of model.

A Singh-Mitchell type creep law is used for the case employing the modified Drucker-Prager/Cap model, activating the cohesion mechanism only. This creep strain rate is defined by

$$\dot{\varepsilon}^{cr} = Ae^{\alpha\sigma_u}\left(\frac{t_1}{t}\right)^m,$$

where σ_u is the equivalent uniaxial compression creep stress, and the constants $A = 2.5 \times 10^{-5}$ sec^{-1} (2.5×10^{-5} sec^{-1}), $\alpha = 1.45 \times 10^{-2}$ MPa^{-1} ($\alpha = 0.0001$ psi^{-1}), $t_1 = 1.0$, and $m = 0.0$.

Solution control

Abaqus begins the analysis with explicit integration and continues to use that method unless its stability limit appears to be too severe a restriction on the size of the time increment or if plasticity occurs. If one of these conditions occurs, Abaqus switches to the backward difference method; thus, the integration is unconditionally stable, and the only limitation on time increment size is solution accuracy. This approach is usually the most economic method for applications involving this type of material behavior.

The accuracy of the time integration of the creep behavior is determined by the size of the time increments chosen by the automatic time incrementation scheme, which is controlled by a tolerance specified in the quasi-static procedure (*Quasi-Static Analysis*). This tolerance limits the difference between the creep strain increments computed from the creep strain rates calculated from conditions at the beginning and at the end of the increment. In a case such as this, where the creep strain rate depends strongly on the stress, the usual guideline for setting this tolerance is to decide on a value that represents a small error in the stress and then divide that value by the elastic modulus to determine the setting. For the creep and relaxation tests we have used 0.69 MPa (100 lb/in^2) as a small stress; hence, the tolerance is chosen as 5×10^{-6}. For the coupled Mises and coupled Drucker-Prager tests a tolerance of 1×10^{-4} showed sufficient accuracy, and for the modified Drucker-Prager/Cap test a tolerance of 5×10^{-6} was selected.

Exact solutions

For the one-dimensional cases observed here, the uniaxial stress is equal to the effective stress, q, and also equal to the equivalent creep test stress, σ_u. In the creep test the creep law can be integrated directly to give

$$\varepsilon^{cr} = \frac{A}{m+1} q^n t^{m+1}.$$

This solution is the same for both time and strain hardening. It is plotted in *Figure 2*.

In the relaxation test the strain is constant, so

$$\dot{\varepsilon}^{cr} = -\dot{\varepsilon}^{el} = -\frac{\dot{q}}{E}.$$

For the time hardening assumption this gives

$$-\frac{\dot{q}}{E} = A q^n t^m,$$

which integrates to give

$$q = \left(\sigma_0^{1-n} + \frac{A}{m+1} E\,(n-1)\, t^{m+1}\right)^{1/(1-n)},$$

where σ_0 is the stress at the start of the event. This solution is shown in *Figure 3*.

For the strain hardening assumption the governing equation becomes

$$-\frac{\dot{q}}{E} = A q^n \left(\frac{m+1}{A q^n} \varepsilon^{cr}\right)^{m/(m+1)}.$$

Since the strain is constant, $\varepsilon^{cr} + \varepsilon^{el} = \varepsilon = \sigma_0 / E$, and $\varepsilon^{el} = q/E$ at any time. Thus, the governing equation defines the stress by

$$\dot{q} = -(AE)^{1/(m+1)} q^{n/(m+1)} \left((m+1)\ (\sigma_0 - q)\right)^{m/(m+1)},$$

so q is defined by

$$\int_{\sigma_0}^{q} q^{-n/(m+1)} (\sigma_0 - q)^{-m/(m+1)}\, dq = -(AE)^{1/(m+1)} (m+1)^{m/(m+1)} t.$$

This equation is integrated numerically, using a fourth-order Runge-Kutta scheme, with a time increment of one second, which should provide a solution of high accuracy. The solution is plotted in *Figure 3*. For the relaxation test the solutions provided by the time hardening and strain hardening models are slightly different.

In the second part of this example, the closed-form solution for the creep strain of both the Mises and the Drucker-Prager models can be obtained by integrating the strain rate. The following is the exact solution for the total strain prior to yielding:

$$\varepsilon^{tot}(t) = \frac{\sigma_0}{t_1 E} t + A \left(\frac{\sigma_0}{t_1}\right)^n \frac{t^{n+m+1}}{n+m+1},$$

where σ_0 is the initial yield stress occurring at time t_1. After the onset of yield, the exact solution for the total strain is as follows:

$$\varepsilon^{tot}(t) = \frac{\sigma_0}{E} + \frac{A \sigma_0^n}{m+n+1} t_1^{m+1} \left(\frac{t}{t_1}\right)^{m+n+1} + \frac{\sigma_0}{t_1 E}(t - t_1) + \left(\frac{t}{t_1} - 1\right)^{\frac{1}{\mu}}.$$

For the case employing the modified Drucker-Prager/Cap model, a very high value of yield stress is specified to prevent yielding. Thus, a closed-form solution of the total strain can easily be obtained:

$$\varepsilon^{tot}(t) = -\frac{\sigma_0}{E t_1} t - \frac{A t_1}{\alpha \sigma_0} \left(e^{\alpha \left(\frac{\sigma_0}{t_1}\right) t} - 1\right).$$

Results and discussion

In the creep test (*creep_usr_creep.inp*) Abaqus uses explicit integration for the first 2.57 sec. When the time increment is 1.28 sec and is being restricted to that value by the conditional stability of the explicit operator, the switch to implicit integration occurs. Toward the end of the time period, time increments of more than 40,000 seconds are being used. For this case the Abaqus results all agree with the exact solution, which is to be expected. The creep test is a constant stress test; and when creep behavior is defined directly on data lines, the exact integration defined above for constant stress is used in Abaqus for each increment. The same technique has been used in the user subroutine *CREEP* for use with*creep_usr_creep.inp* and *creep_relax_usr_creep.inp*.

Figure 3 shows the Abaqus results for the relaxation test (*creep_relax_usr_creep.inp*), compared to the exact solutions described above. The agreement is quite good. Accuracy can be improved by using a smaller value for the accuracy tolerance, which causes Abaqus to use smaller time increments initially. Generally this is not done because the creep data from which the material behavior are defined show considerable scatter, so it is not worthwhile to attempt to obtain very high accuracy in the response prediction.

Figure 4 shows the results for the coupled creep and plasticity tests. The closed-form solutions for the total strain (solid line), the plastic strain (short dashed line), the creep strain (long dashed line), and the elastic strain (dotted line) are shown as lines. The Abaqus solution is shown as symbols. Triangles are plotted for the Mises case, and circles are plotted for the Drucker-Prager case. The symbols are plotted at various intervals and show excellent agreement with the closed-form solution.

For the case using the modified Drucker-Prager/Cap model the creep strain in the direction of the loading (variable CE22) and the equivalent creep strain (variable CEEQ) deviate slightly from the theoretical values of the creep strain (see *Figure 5*). The discrepancy develops while the stress is low, after which no additional deviations occur. This behavior is a shortcoming of the creep potentials used by Abaqus, as described in *Models for granular or polymer behavior*. This kind of discrepancy is not observed in more typical creep analyses that have a short-term ("static") preloading, followed by long-term creep. Such behavior is highlighted if the ramp amplitude of the load is replaced by a step function. In that case the closed-form solution becomes

$$\varepsilon^{cr} = Ate^{\alpha\sigma_p},$$

where σ_p represents the prescribed stress at $t = 0$. *Figure 6* shows the same three curves thus obtained.

Input files

creep_usr_creep.inp	Creep test. The load is applied in Step 1, and the response is obtained in Step 2.
creep_usr_creep.f	User subroutine *CREEP* used in creep_usr_creep.inp and creep_postoutput.inp.
creep_relax_usr_creep.inp	Relaxation test. The displacement is prescribed in Step 1, and the response obtained in Step 2.
creep_relax_usr_creep.f	User subroutine *CREEP* used in creep_relax_usr_creep.inp.
creep_exact.f	Runge-Kutta integration of the equation that defines the solution to the strain hardening relaxation test.
creep_mises.inp	Mises creep and plasticity test case.
creep_postoutput.inp	**POST OUTPUT* analysis.
creep_misescurve.inp	Hardening data for the Mises creep and plasticity test case given in creep_mises.inp.
creep_drucker.inp	Drucker-Prager creep and plasticity test case.
creep_druckercap_ramp.inp	Modified Drucker-Prager/Cap creep and plasticity test case. The stress is applied as a ramp function with time.

creep_druckercap_step.inp	Modified Drucker-Prager/Cap creep and plasticity test case. The stress is applied as a step function and held constant with time.
creep_exact_closedform.f	Fortran program for the closed-form solutions to verify the coupled creep and plasticity option.

Figures

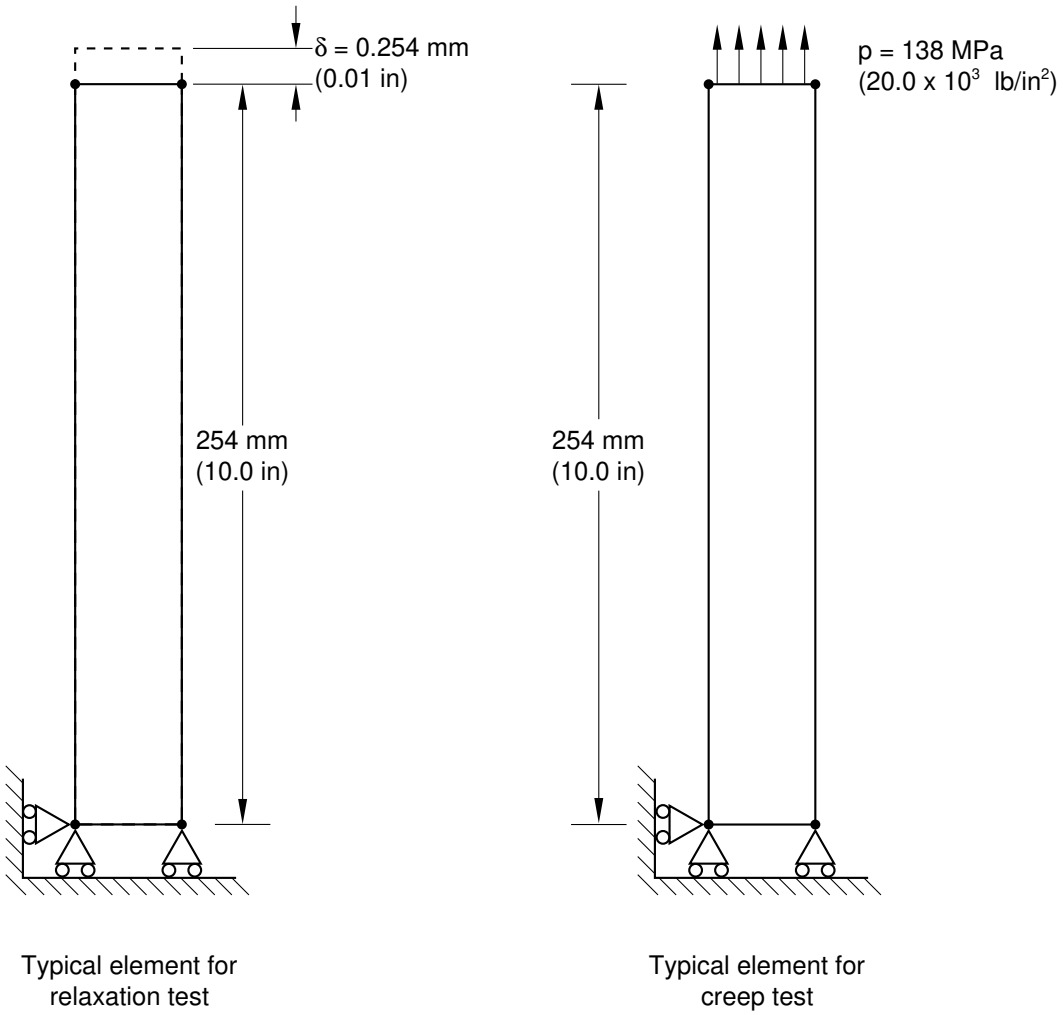


Figure 1: Typical elements for creep and relaxation tests.

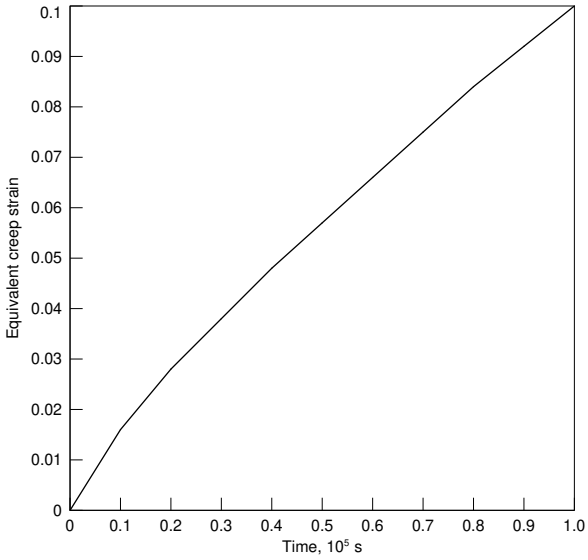


Figure 2: Creep test history.

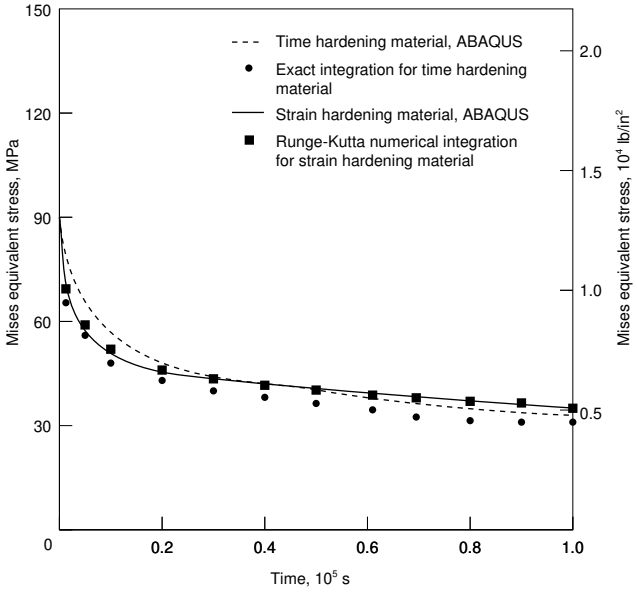


Figure 3: Relaxation test history.

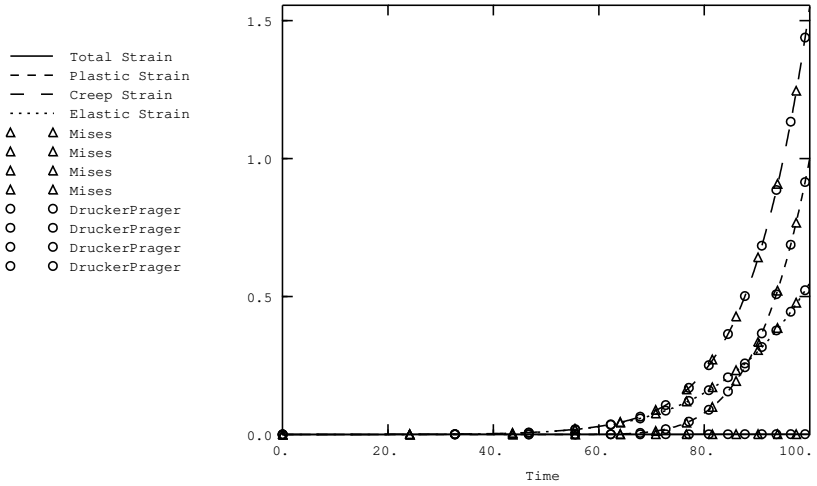


Figure 4: Mises and Drucker-Prager creep and plasticity models.

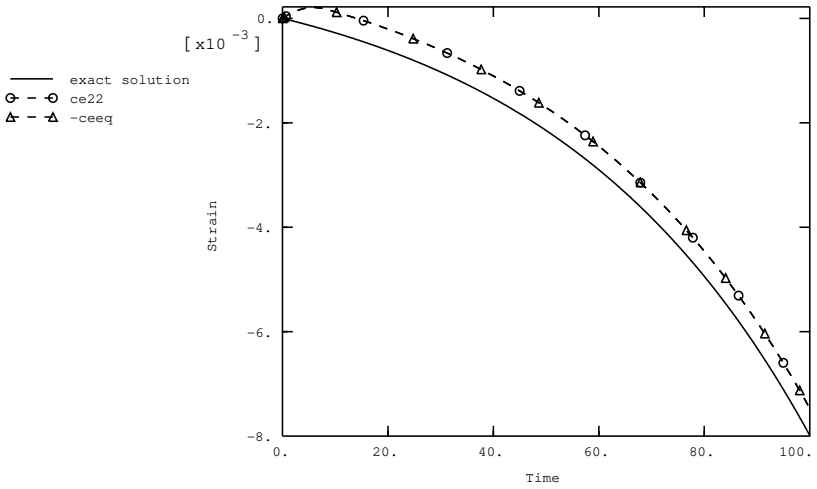


Figure 5: Modified Drucker Prager/Cap model (stress applied as a ramp function).

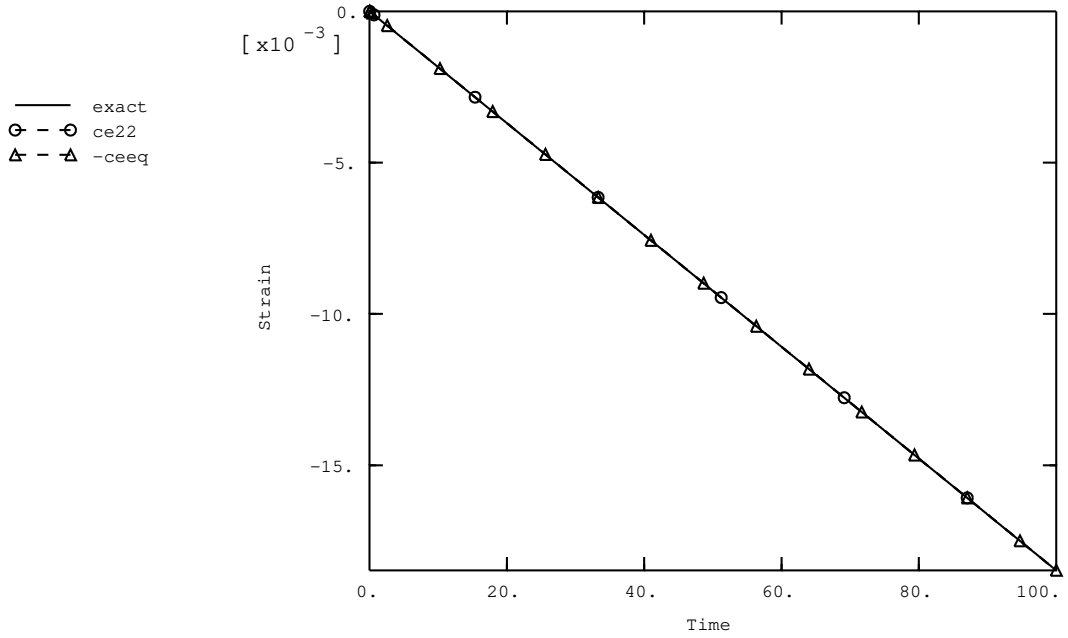


Figure 6: Modified Drucker Prager/Cap model (stress applied as a step function).

Simple tests on a crushable foam specimen

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example illustrates the fundamental material behavior obtained with the crushable foam plasticity model in Abaqus and outlines a simple calibration procedure for the foam material parameters.

Crushable Foam Plasticity Models contains a summary of the model. A complete description of the model is given in *Models for crushable foams*.

Problem description

The simple tests performed in this example are carried out using a cube (one C3D8 element for Abaqus/Standard and one C3D8R element for Abaqus/Explicit) of unit dimensions. Displacements are prescribed at the nodes of the cube to simulate homogeneous deformation and stress conditions. Four different load cases are analyzed in this example using Abaqus/Standard: hydrostatic compression, uniaxial compression, uniaxial tension, and pure shear. These load cases are analyzed using both the volumetric hardening and isotropic hardening models. Since the foam material undergoes very large deformations, geometric nonlinearities are considered in this large-deformation analysis. As the foam model assumes nonassociated flow, the unsymmetric storage and solution scheme is activated. The cases of uniaxial compression and hydrostatic compression are also analyzed using the Abaqus/Explicit volumetric hardening model and the isotropic hardening model, respectively.

Calibration of material parameters

The elasticity is defined by the Young's modulus, E, and elastic Poisson's ratio, ν. For the crushable foam model with volumetric hardening, the initial yield surface is defined by k, the ratio of yield stress in uniaxial compression, σ_c^0, to the yield stress in hydrostatic compression, p_c^0 (this is the initial value of p_c); and k_t, the ratio of yield stress in hydrostatic tension, p_t, to the yield stress in hydrostatic compression, p_c^0. For the isotropic hardening model, the initial yield surface is defined by k only, but the user needs to provide the plastic Poisson's ratio, which is the ratio of the transverse to the longitudinal plastic strain under uniaxial compression. The evolution of the yield surface is defined by the yield stress versus plastic strain curve in uniaxial compression. For the rate-dependent version of the model we also need the inverse viscosity, D, and the viscous power law coefficient, p.

The model is calibrated for a foam (Dytherm 2.5) for which we have only hydrostatic compression and uniaxial compression data at slow (static) strain rates. The rate-independent calibration is done as follows. The Young's modulus and elastic Poisson's ratio can be obtained easily from the experimental data. From the hydrostatic compression test we immediately obtain the initial yield pressure, p_c^0. From the uniaxial compression test we obtain the initial yield stress in uniaxial compression, σ_c^0. For the volumetric hardening model, we also need the strength in hydrostatic tension. In the absence of tensile test data we assume the tensile strength, p_t, to be one-tenth of the hydrostatic compression yield strength p_c^0. For the isotropic hardening model we choose zero plastic Poisson's ratio based on the experimental observation that uniaxial compression of a specimen shows almost no lateral strain. The subsequent yield surface is defined by the hardening curve characterized using uniaxial compression data. The calibration as described above yields the following material parameters for Dytherm 2.5:

- $E =$ 3.0 MPa

- $\nu = 0.2$
- $\sigma_c^0 = 0.22$ MPa
- $p_t = 0.02$ MPa
- $p_c^0 = 0.2$ MPa (this is the initial value of p_c)
- $k = \sigma_c^0 / p_c^0 = 1.1$
- $k_t = p_t / p_c^0 = 0.1$ (for the volumetric hardening model)
- $\nu_p = 0$ (for the isotropic hardening model)

To calibrate the rate dependence parameters D and p, it would be necessary to have a number of similar configuration tests performed at different strain rates. Such data are not available for this material.

Results and discussion

The results obtained with the above calibration are compared to the static experiments of Bilkhu (1987) in *Figure 1* (hydrostatic compression) and *Figure 2* (uniaxial compression). The Abaqus/Explicit results shown above are obtained from the volumetric hardening model. Good agreement is observed for both tests. *Figure 3* and *Figure 4* show comparisons of the response of the volumetric hardening model and isotropic hardening model, respectively, for three different monotonic loading paths: uniaxial compression, pure shear, and uniaxial tension. The strain axis represents the deformation in the direction of the major direct stress. The volumetric hardening model correctly predicts a perfectly plastic response for pure shear, as well as any loading condition that gives rise to a negative pressure stress state, such as uniaxial tension, while hardening takes place for any loading condition that gives rise to a positive pressure stress state. The isotropic hardening model, on the other hand, predicts hardening behavior under all loading conditions.

Input files

crushablefoam_hydrostatic.inp	Abaqus/Standard hydrostatic compression test using the volumetric hardening model.
crushablefoam_unicomp.inp	Abaqus/Standard uniaxial compression test using the volumetric hardening model.
crushablefoam_unitens.inp	Abaqus/Standard uniaxial tension test using the volumetric hardening model.
crushablefoam_shear.inp	Abaqus/Standard pure shear test using the volumetric hardening model.
crushablefoam_iso_hydrostatic.inp	Abaqus/Standard hydrostatic compression test using the isotropic hardening model.
crushablefoam_iso_unicomp.inp	Abaqus/Standard uniaxial compression test using the isotropic hardening model.
crushablefoam_iso_unitens.inp	Abaqus/Standard uniaxial tension test using the isotropic hardening model.
crushablefoam_iso_shear.inp	Abaqus/Standard pure shear test using the isotropic hardening model.
crushfoamvol_hydro.inp	Abaqus/Explicit hydrostatic compression test using the volumetric hardening model.
crushfoamvol_ucomp.inp	Abaqus/Explicit uniaxial compression test using the volumetric hardening model.
crushfoamiso_hydro.inp	Abaqus/Explicit hydrostatic compression test using the isotropic hardening model.

crushfoamiso_ucomp.inp	Abaqus/Explicit uniaxial compression test using the isotropic hardening model.
crushfoamvol_hydro_c3d8.inp	Abaqus/Explicit hydrostatic compression test using the volumetric hardening model for the sole purpose of testing the performance of the C3D8 element.
crushfoamvol_ucomp_c3d8.inp	Abaqus/Explicit uniaxial compression test using the volumetric hardening model for the sole purpose of testing the performance of the C3D8 element.
crushfoamiso_hydro_c3d8.inp	Abaqus/Explicit hydrostatic compression test using the isotropic hardening model for the sole purpose of testing the performance of the C3D8 element.
crushfoamiso_ucomp_c3d8.inp	Abaqus/Explicit uniaxial compression test using the isotropic hardening model for the sole purpose of testing the performance of the C3D8 element.
crushfoamvol_hydro_c3d8i.inp	Abaqus/Explicit hydrostatic compression test using the volumetric hardening model for the sole purpose of testing the performance of the C3D8I element.
crushfoamvol_ucomp_c3d8i.inp	Abaqus/Explicit uniaxial compression test using the volumetric hardening model for the sole purpose of testing the performance of the C3D8I element.
crushfoamiso_hydro_c3d8i.inp	Abaqus/Explicit hydrostatic compression test using the isotropic hardening model for the sole purpose of testing the performance of the C3D8I element.
crushfoamiso_ucomp_c3d8i.inp	Abaqus/Explicit uniaxial compression test using the isotropic hardening model for the sole purpose of testing the performance of the C3D8I element.

References

- Bilkhu, S., Private Communication, 1987.

Figures

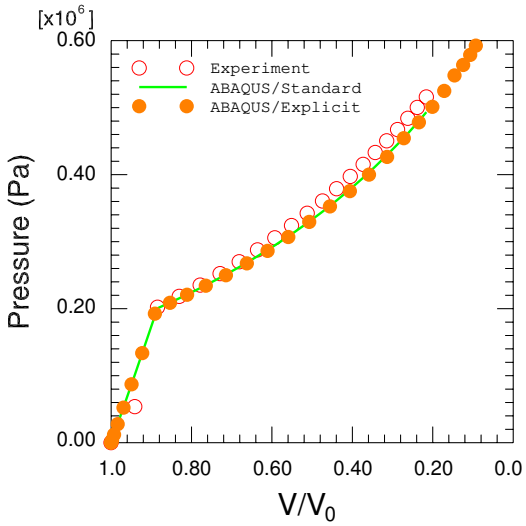


Figure 1: Hydrostatic compression test.

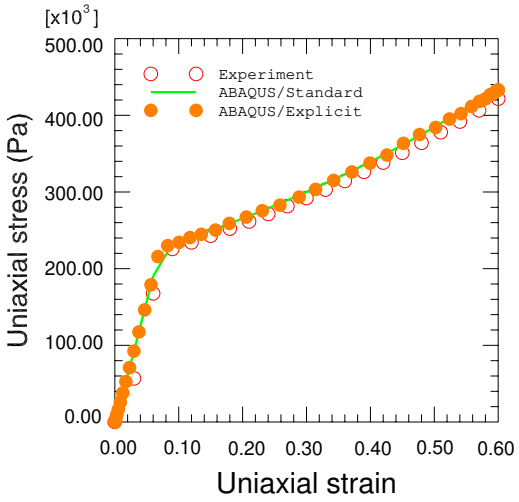


Figure 2: Uniaxial compression test.

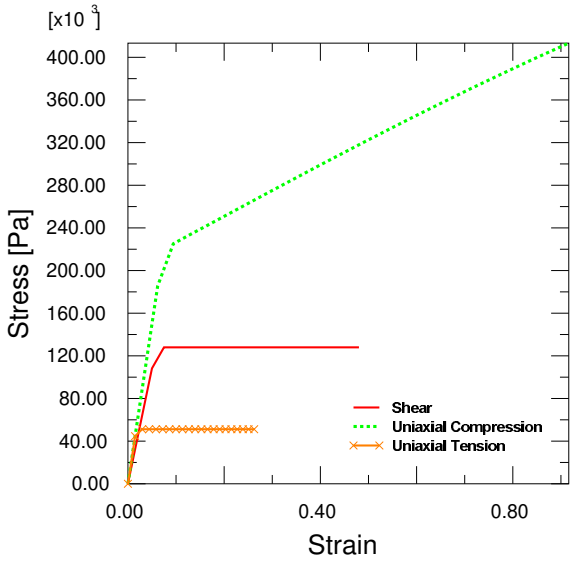


Figure 3: Comparison of the volumetric hardening model response for three different loading paths.

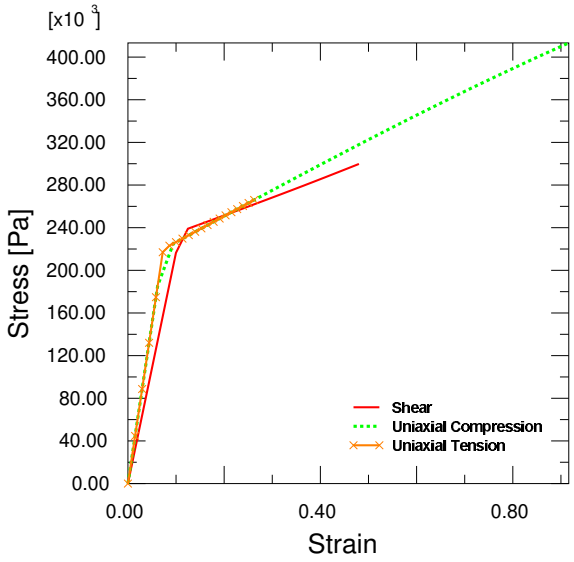


Figure 4: Comparison of the isotropic hardening model response for three different loading paths.

Simple proportional and nonproportional cyclic tests

Products: Abaqus/Standard

Overview

This example illustrates the process of calibrating the nonlinear isotropic/kinematic hardening model using test data from a uniaxial, symmetric strain-controlled, cyclic experiment. It also illustrates the limitations of the model under multiaxial loading conditions when the material properties are calibrated with uniaxial test data.
Models for Metals Subjected to Cyclic Loading contains a description of the model and its use. A mathematical description of the model is presented in *Models for metals subjected to cyclic loading*.

Three different simulations are performed: a uniaxial, symmetric strain-controlled experiment; a uniaxial, unsymmetric strain-controlled experiment; and a multiaxial tension-torsion experiment. The model predictions are compared with experimental test data for OFHC copper (Anand, 1996). The simulations show that the model captures the response of the material accurately when the experiment that is used to calibrate the model is simulated. However, it only approximates the behavior of the material when the loading does not correspond to the loading of the calibration experiment.

Problem description

The simple tests performed in this example are carried out using a tube (one PIPE31 element) of unit length and unit midsurface radius. This element is chosen so that the simulation of a tension-torsion cyclic experiment can be performed easily.

This example uses test data obtained from a well-annealed OFHC copper with a Young's modulus, E, of 104 GPa and a Poisson's ratio, ν, of 0.3.

Calibration of the model

The model is calibrated using test data from a uniaxial experiment (*Figure 1*) obtained at a strain range $\Delta\varepsilon =$ 1.5%. Both the kinematic component and isotropic hardening component of the model are calibrated.

The shape of the first cycle differs from the shape of subsequent cycles, suggesting that the kinematic hardening component is a function of the cycle number. Since the model does not allow for such a dependency, a representative shape must be chosen. The objective in this example is to compare the model predictions with test data over many cycles. The stabilized cycle is, therefore, chosen for calibration. If the model were being used to simulate only one or two load cycles, it would be more appropriate to use the first loading cycle for calibration.

The second half of the saturated cycle used for calibrating the kinematic hardening material parameters C and γ is shown in *Figure 2*. The data are entered as values of yield stress, σ_i^0, versus plastic strain, ε_i^{pl}, to define the stabilized cycle of the metal plasticity model, where

$$\varepsilon_i^{pl} = \varepsilon_i - \frac{\sigma_i^0}{E} - \varepsilon_p^0,$$

with ε_i the total strain for data point i, and $\varepsilon_p^0 = \varepsilon_1 - \sigma_1^0/E$. The onset of yield is taken as $\sigma_1^0 =$ 46.9 MPa. The calibration yields $C =$ 33.55 GPa and $\gamma =$ 701.3; these quantities are output to the printed output (`.dat`) file using a model definition data request.

The isotropic hardening component is calibrated next. Isotropic hardening defines the evolution of the elastic range as a function of equivalent plastic strain. The size of the elastic range can be determined easily at points where the loading is reversed as half the difference between the yield stress in tension and compression. For the stabilized cycle the size of the elastic range is 96.2 MPa. The corresponding values of equivalent plastic strain are obtained by assuming that the test is approximately performed as a symmetric plastic strain-controlled experiment, where

$$\Delta\varepsilon^{pl} = \Delta\varepsilon - 2\frac{\overline{\sigma}}{E},$$

and $\overline{\sigma}$ is an averaged yield stress over all the cycles. $\overline{\sigma}$ is taken as 75.0 MPa for this material. With this assumption the equivalent plastic strain is obtained as

$$\overline{\varepsilon}^{pl} = \frac{1}{2}(4i - 3)\Delta\varepsilon^{pl},$$

where i is the cycle number. This approximation yields a value of $\overline{\varepsilon}^{pl} = 25.16\%$ for the last cycle ($i = 10$). The resulting data are used to define the size of the elastic range. The change in elastic range during the first half-cycle is specified as zero to compensate for the difference in the shape of this cycle compared to subsequent cycles.

Results and discussion

The predictions of the calibrated model for the three experiments are shown in *Figure 3* through *Figure 5*.

Figure 3 compares the predictions of the model with the test data obtained from the symmetric strain-controlled cyclic experiment. The figure shows a close match between the simulation and the test data except for the first cycle, which is to be expected since the calibration of the kinematic hardening component is based on stabilized test data. The equivalent plastic strain reported by Abaqus at the end of the analysis is equal to 23.67%; therefore, the assumption used to calibrate the isotropic hardening component that the test is approximately plastic strain-controlled is justified.

Next, the calibrated model is used to simulate the behavior of the material during an unsymmetric strain-controlled cyclic experiment with strain varying between 0.25% and 1.75%. *Figure 4* compares the simulation with the experimental test data. Again, a poor match is observed over the first cycle, since the model is calibrated using the stabilized stress-strain curve. The figure further shows that the model predicts less cyclic hardening (approximately 8%) than that obtained in the experiment.

Figure 5 compares the predicitions of the model with test data for a tension-torsion cyclic experiment. The experiment is conducted as follows: first, the specimen is stretched to a strain of $\varepsilon_0 = 1\%$; then it is cycled with an axial strain $\varepsilon = \varepsilon_0 \cos(\omega t)$ and a shear strain $\gamma = \sqrt{3}\varepsilon_0 \sin(\omega t)$. Since the material is assumed to be rate independent, it is convenient to choose $\omega = \pi$ so that each strain cycle runs over two time units.

The model predicts a maximum normal stress of 143.1 MPa, which is also the saturated value that the model predicts during a uniaxial experiment. Saturation is reached after only four cycles. The experiment, on the other hand, shows that the maximum (saturated) normal stress of 218.0 MPa is reached after approximately 10 cycles. The large difference between the maximum values of the normal stress and the rate at which saturation is reached is the result of the different responses of the material under different loading conditions and different strain ranges. For example, the tension-torsion experiment is conducted over a strain range of 2.0%, whereas the isotropic hardening component is calibrated using test data from a uniaxial experiment conducted over a 1.5% strain range.

These experiments show that the cyclic response of the material depends on the strain range over which the cycles occur and on the type of loading conditions. The experiments further show that the kinematic hardening component may change from cycle to cycle. Some materials also show different kinematic hardening behavior at different strain ranges, although this effect is negligible in this case. Typically, isotropic hardening properties are more sensitive to loading conditions than kinematic hardening properties. The model does not allow for such dependencies. Therefore, it is important to perform different types of cyclic experiments at different strain ranges

to establish the sensitivity of the isotropic and kinematic hardening properties to strain range and loading conditions.

Acknowledgment

SIMULIA would like to thank Professor L. Anand of the Massachusetts Institute of Technology for providing the experimental test data.

Input files

cyclictests_sym.inp	Symmetric strain-controlled cyclic simulation.
cyclictests_kinematic.inp	Kinematic hardening data obtained from the symmetric strain-controlled cyclic experiment.
cyclictests_unsym.inp	Unsymmetric strain-controlled cyclic simulation.
cyclictests_tensiontorsion.inp	Tension-torsion cyclic simulation.

References

- Anand, L., “Test Data for a Well-Annealed OFHC Copper Material,” *Massachusetts Institute of Technology, Cambridge, MA*, 1996.

Figures

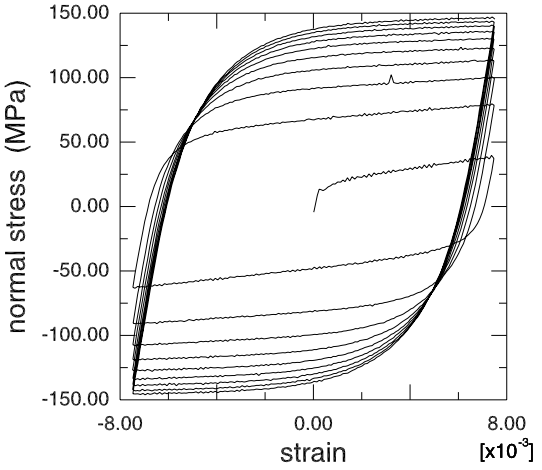


Figure 1: Symmetric strain cyclic test data.

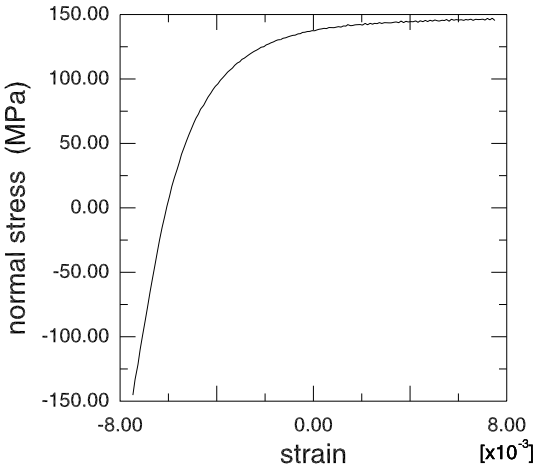


Figure 2: The last half cycle of test data is used to calibrate the kinematic hardening component.

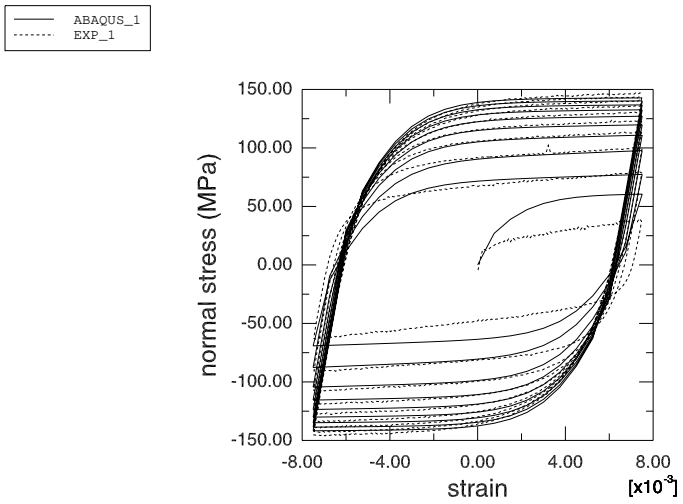


Figure 3: Comparison of the calibrated model and the test data for the symmetric strain cycle experiment.

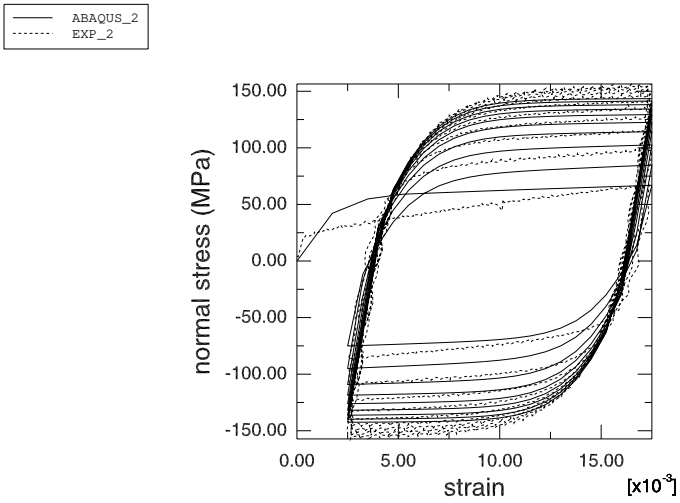


Figure 4: Comparison of the calibrated model and the test data for the unsymmetric strain cycle experiment.

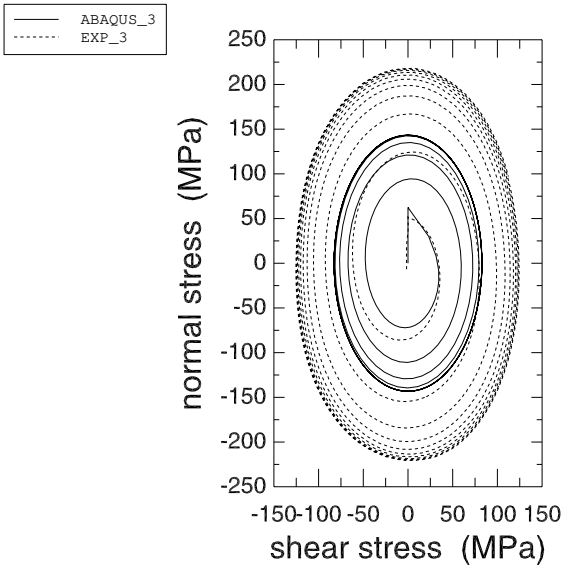


Figure 5: Comparison of the calibrated model and the test data for the tension-torsion cycle experiment.

Biaxial tests on gray cast iron

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example illustrates the fundamental material behavior obtained with the cast iron plasticity material model in Abaqus. It also compares the model predictions under multiaxial loading conditions with the experimental test data given by Coffin (1950).

The model is calibrated with uniaxial tension and uniaxial compression test data, and the predictions are compared with test data under different loading conditions. *Cast Iron Plasticity* contains a summary of the model; and a complete description of the model is given in *Cast iron plasticity*.

Problem description

The tests performed in this example are carried out using a cube (one C3D8 element) of unit dimensions. Pressure loads are applied to appropriate faces of the element to model six different loading paths in stress space. These loading paths are uniaxial tension, uniaxial compression, equibiaxial tension, pure shear, biaxial tension (with the magnitude of the loading in the two directions being unequal), and biaxial tension/compression (with the magnitude of the loading in the two directions being unequal), respectively. The loading in all the test cases results in homogeneous deformation. Since the cast iron plasticity model assumes nonassociated flow, the unsymmetric matrix storage and solution scheme is used in the Abaqus/Standard analyses.

Material parameters

In the cast iron plasticity model the elastic behavior is assumed to be linear and isotropic. The Young's modulus, E, and Poisson's ratio, ν, are assumed to be the same in tension and compression. For calibrating the plastic behavior, the model requires the value of the "plastic Poisson's ratio," ν_{pl}; the hardening curve in uniaxial tension; and the hardening curve in uniaxial compression. The plastic Poisson's ratio is an average measure of the transverse to the longitudinal plastic strain under uniaxial tension. The material data used in this example were obtained from Coffin (1950). *Figure 1* shows the uniaxial tension and compression curves (the first data point in each case corresponds to the onset of plastic deformation) that are used to calibrate the hardening of the model. The plastic Poisson's ratio is taken to be 0.039, based on Coffin's data for permanent volumetric strain under uniaxial tension. The units for the stresses and Young's modulus are psi.

Results and discussion

The results for each test are presented in two plots: one showing the stress/strain response and the other showing the variation of the permanent volumetric strain with the applied stress. For each plot two sets of data are presented: one corresponds to Coffin's experimental data, and the other corresponds to the Abaqus simulation.

Figure 2, *Figure 3*, and *Figure 4* compare the prediction of the model with the experimental test data in uniaxial tension and uniaxial compression, respectively. All the figures show good agreement between the test data and experimental results. This behavior is to be expected since the model is calibrated using uniaxial test data. *Figure 5* shows that small (compared to the strain in *Figure 4*) permanent volume changes were observed in the uniaxial compression experiment. The model in Abaqus predicts zero permanent volume change under uniaxial compression and higher confining stresses.

The results for equibiaxial tension are presented in *Figure 6* and *Figure 7*. At high stress values Abaqus predicts a stiffer response. The error in the stress/strain response is about 20% at 70% of the fracture stress and higher

for higher stress values. The error in the stress/strain response at high stresses is probably because the Rankine yield criterion is only an approximation to the real material behavior under equibiaxial loading conditions.

The results for pure shear are shown in *Figure 8* and *Figure 9*. Again, Abaqus predicts a stiffer response at high stress values. The error in the stress/strain response is about 20% at 70% of the fracture stress and higher for higher stress values. The stress/strain response of the simulation indicates a change in the rate of hardening at very high stresses. This behavior is due to a change in the yielding mechanism from Rankine to Mises.

Figure 10 and *Figure 11* show the results for unequal biaxial tension, where the applied loading in one direction is twice of that in the other. In both the figures the strains are plotted against the maximum principal stress. In *Figure 10* the curves labeled “Abaqus–1” and “COFFIN-1” correspond to the maximum principal stress versus the maximum principal strain, and the curves labeled “Abaqus–2” and “COFFIN-2” correspond to the maximum principal stress versus the intermediate principal strain. The stress/strain response predicted by Abaqus is in good agreement with the experimental results. The difference between the numerical and experimental results for the permanent volume strain under high stresses suggests that the minimum principal strain has not been captured as accurately as the other principal strains in the numerical simulation. The reason for the better agreement in the stress/strain results for unequal biaxial tension as compared to those for equibiaxial tension and shear may be that the loading path is, in relative terms, closer to the uniaxial tension loading path and that the model is calibrated with uniaxial tension results.

Finally, the case of biaxial tension/compression is shown in *Figure 12* and *Figure 13*. The loading for this test consists of tension in one direction and compression, with twice the magnitude of the tensile load, in the other direction. The stress/strain response as predicted by Abaqus agrees well with the experimental results. The loading path for this case is close to the loading path for uniaxial compression, and the model is calibrated with uniaxial compression results. In *Figure 13* the experimental results indicate a very high value for the maximum permanent volume change. Given that the loading is predominantly compressive, such a high value of the permanent volume change is somewhat surprising. It is possible that such a permanent volume change may be related to effects such as microbuckling of the graphite flakes; the Abaqus model does not capture this behavior.

Conclusions

These simulations show that the Abaqus model generally matches the experiments reasonably well. As expected, the match is better for stress paths close to the ones that are used for calibration. However, the model is only a first approximation to the real material behavior, and it would need more features to match the experimental results well for all stress paths. For stress paths that represent equibiaxial tension and pure shear, respectively, the simulations indicate that about 20% error may be expected at 70% of the fracture stress (such high stresses are unlikely to be acceptable in a design).

Input files

Abaqus/Standard input files

castiron_unitension.inp	Uniaxial tension test.
castiron_unicompress.inp	Uniaxial compression test.
castiron_equitension.inp	Equibiaxial tension test.
castiron_shear.inp	Shear test.
castiron_bitension.inp	Biaxial tension case.
castiron_tensioncompress.inp	Biaxial tension/compression case.

Abaqus/Explicit input files

castiron_unitension_xpl.inp	Uniaxial tension test.
castiron_unicompress_xpl.inp	Uniaxial compression test.

castiron_equitension_xpl.inp	Equibiaxial tension test.
castiron_shear_xpl.inp	Shear test.
castiron_bitension_xpl.inp	Biaxial tension case.
castiron_tensioncompress_xpl.inp	Biaxial tension/compression case.

References

- Coffin, L. F., “The Flow and Fracture of a Brittle Material,” *Journal of Applied Mechanics*, vol. 72, pp. 233–248, 1950.

Figures

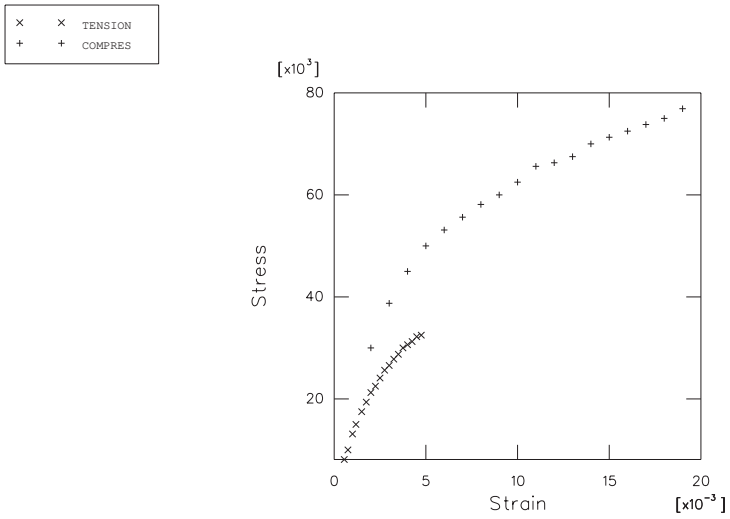


Figure 1: Uniaxial stress/strain curves for gray cast iron.

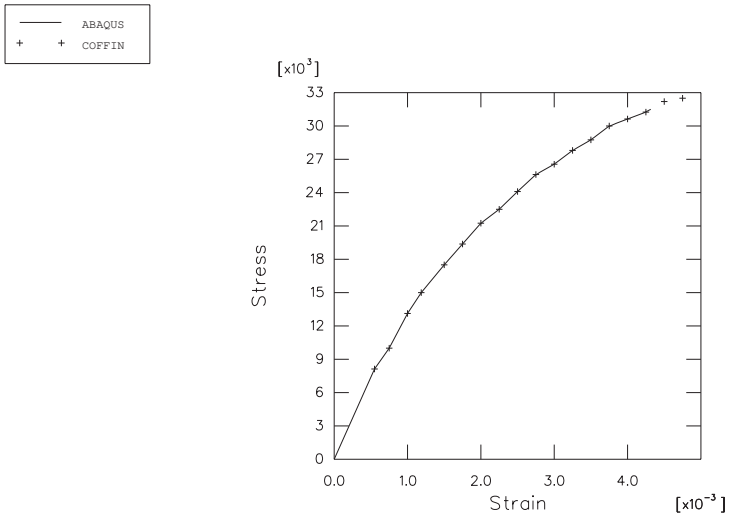


Figure 2: Stress versus strain under uniaxial tension.

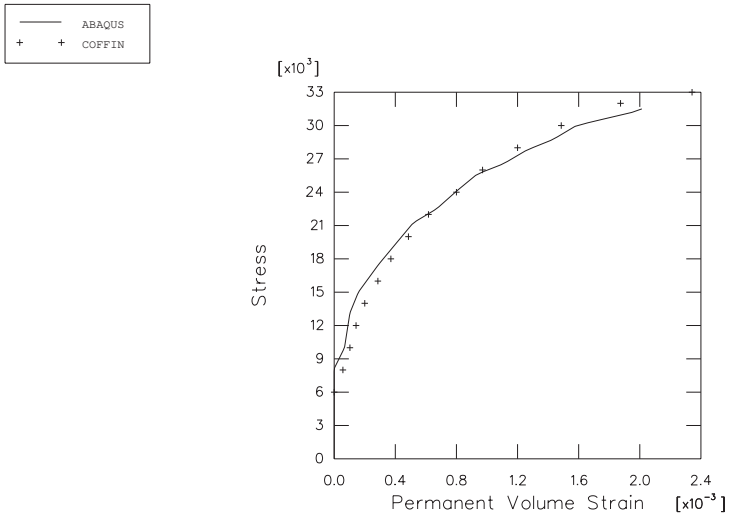


Figure 3: Stress versus permanent volume strain under uniaxial tension.

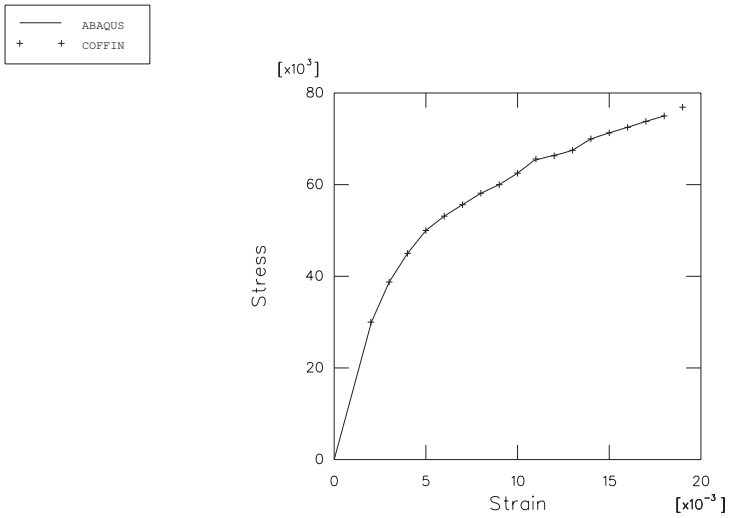


Figure 4: Stress versus strain under uniaxial compression.

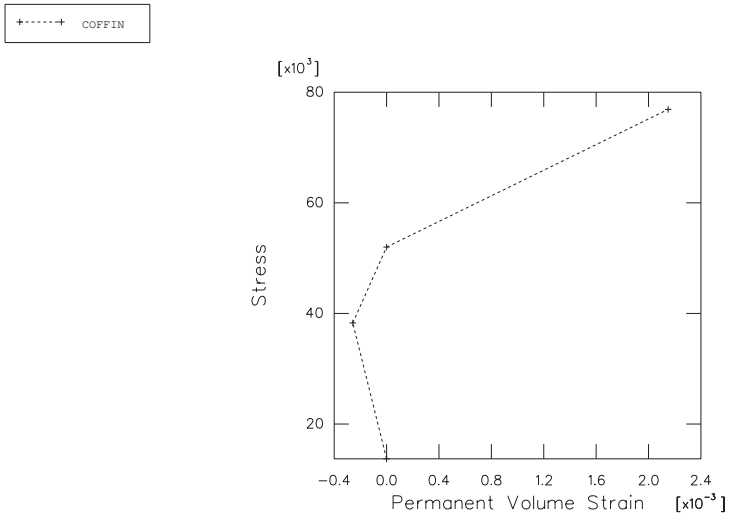


Figure 5: Stress versus permanent volume strain under uniaxial compression.

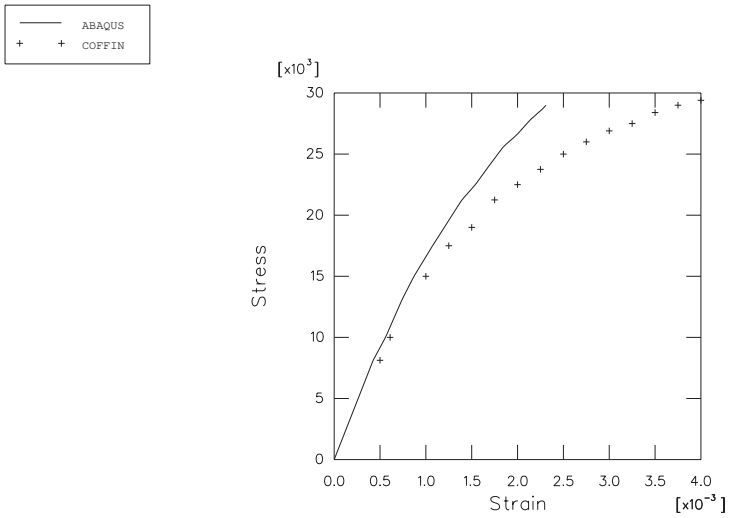


Figure 6: Stress versus strain under equibiaxial tension.

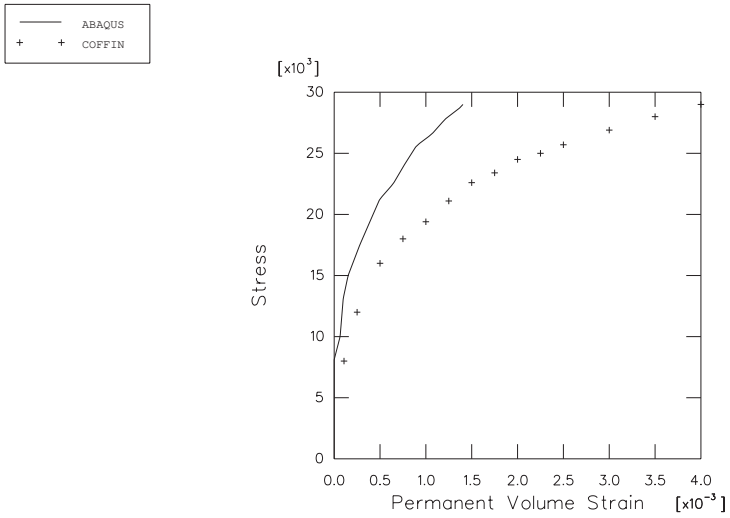


Figure 7: Stress versus permanent volume strain under equibiaxial tension.

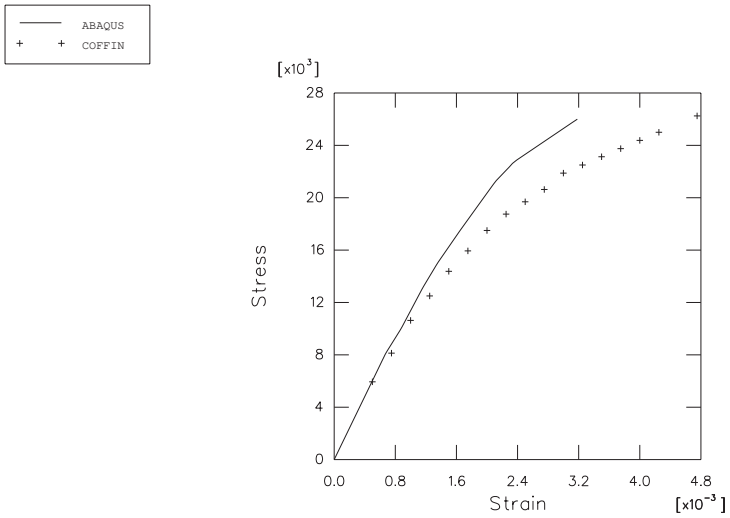


Figure 8: Stress versus strain under pure shear.

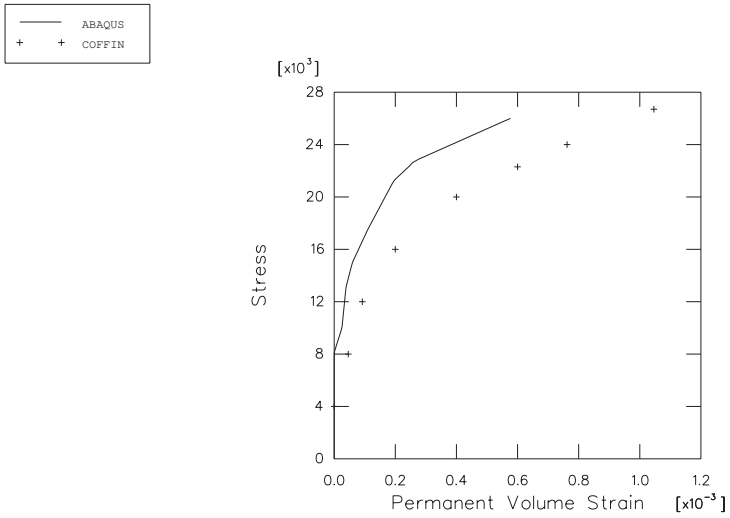


Figure 9: Stress versus permanent volume strain under pure shear.

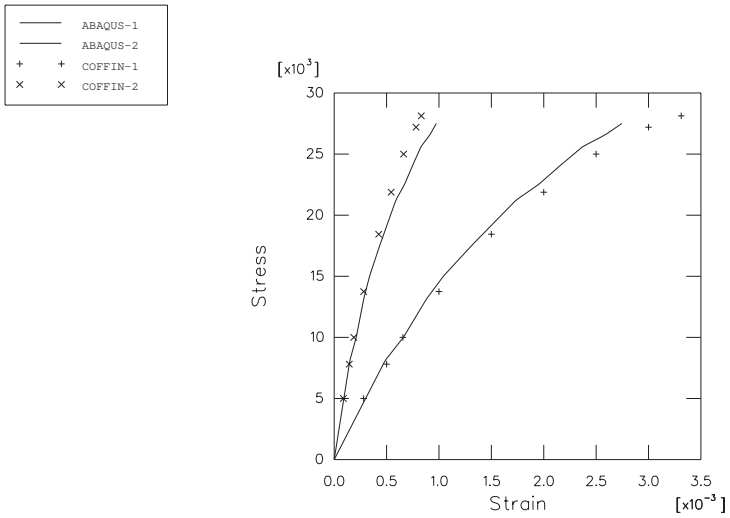


Figure 10: Stress versus strain under unequal biaxial tension.

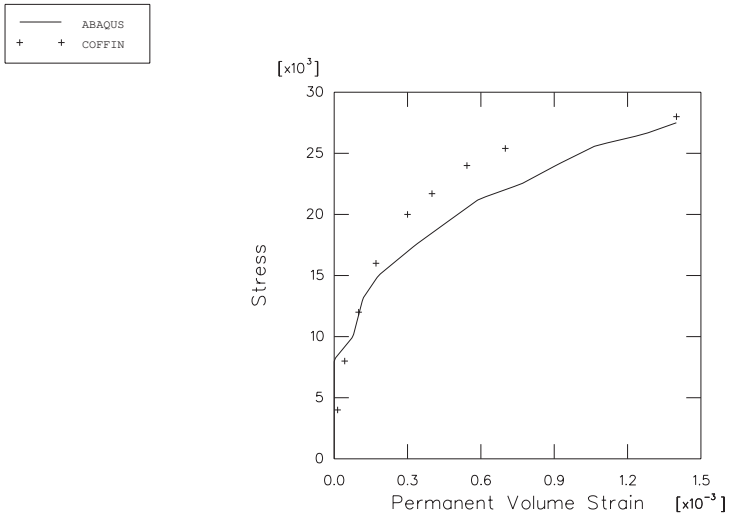


Figure 11: Stress versus permanent volume strain under unequal biaxial tension.

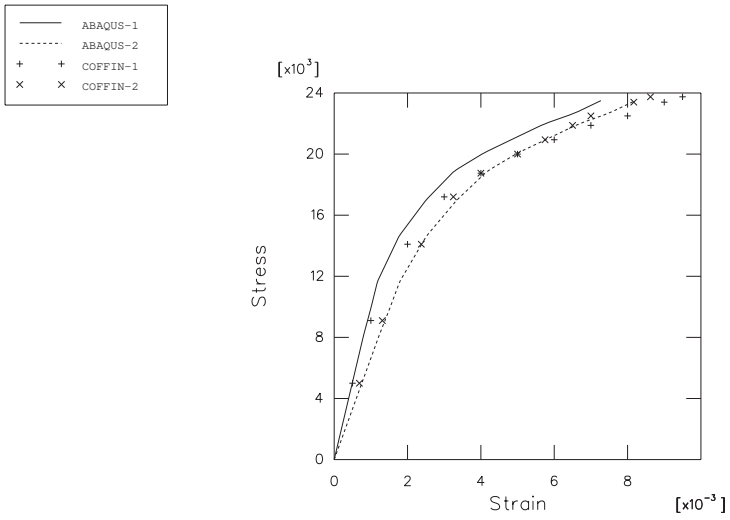


Figure 12: Stress versus strain under biaxial tension/compression.

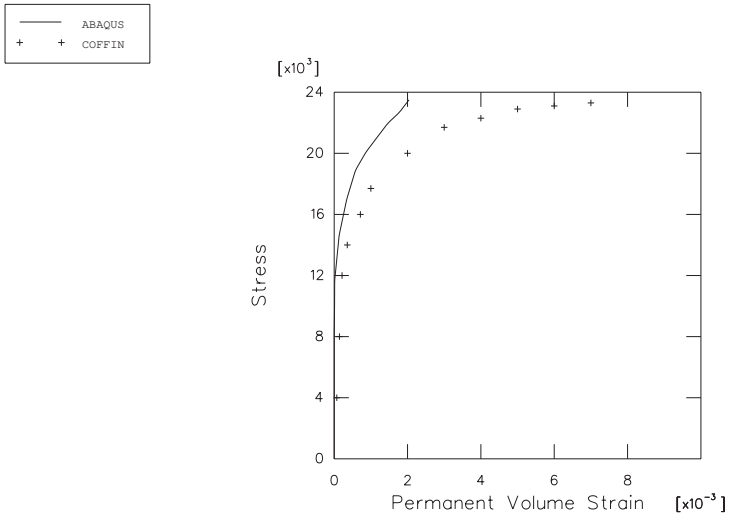


Figure 13: Stress versus permanent volume strain under biaxial tension/compression.

Indentation of a crushable foam plate

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example illustrates a typical application of crushable foam materials used as energy absorption devices.

Two indentation problems are considered: a square plate of polyurethane foam indented by a rigid, cylindrical punch and a cylindrical plate of the same kind of foam indented by a rigid, hemispherical punch. The effect of rate dependence of the foam is shown. Results are presented for both the isotropic and volumetric hardening foam models.

Problem description

The model consists of a rigid impactor and a deformable plate made of polyurethane foam. The undeformed square plate is 30 mm thick and extends 180 mm on each side. The plate is assumed to deform in a symmetric manner, so only half of it is discretized, as shown in *Figure 1*. The half plate is modeled with 10 × 30 CPE4 elements in Abaqus/Standard and 10 × 30 and 15 × 45 CPE4R elements in Abaqus/Explicit.

The undeformed cylindrical plate has a radius of 90 mm and a thickness of 30 mm, as shown in *Figure 1*. It is modeled with 10 × 30 CAX4 elements in Abaqus/Standard and 10 × 30 and 15 × 45 CAX4R elements in Abaqus/Explicit. In both cases the impactors are assumed to have a radius of 82.5 mm. The bottom nodes of the mesh are fixed, while the outer boundary is free to deform.

Material

Uniaxial and hydrostatic compression tests have been conducted on a block of sample polyurethane foam material by Schluppkotten (1999). The yield stress in uniaxial compression is plotted against the axial plastic strain in *Figure 2*. Insignificant lateral deformation is observed during uniaxial compression. The hydrostatic compression test results show that the initial yield stress in hydrostatic compression, p_c^0, is almost the same as that in uniaxial compression, σ_c^0. The elastic response is approximated by the following constants:

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

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

The material parameters for the isotropic hardening foam model are

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

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

and the material parameters for the volumetric hardening foam model are

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

$k_t = p_t/p_c^0 =$ 0.1 (tension yield strength ratio).

The density for the polyurethane foam analyzed in this example is

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

In addition, the experimental results provide the following material properties for the rate-dependent case:

$D =$ 4638.0 per sec,

$n =$ 2.285.

Contact interaction

The contact between the top exterior surface of the foam plate and the rigid punch is modeled with a contact pair. Both the cylindrical and hemispherical rigid punches are modeled as analytical rigid surfaces using a surface definition in conjunction with a rigid body constraint. Coulomb friction is modeled between the punch and the plate with a friction coefficient of 0.2. The maximum shear traction due to friction is assumed to be $\sigma_c^0/\sqrt{3}$, or 0.115 MPa.

Loading and controls

The impactor is fully constrained except in the vertical direction, in which motion is prescribed such that the maximum indentation depth is about 90% of the thickness of the plate. In most of the tests, the load is applied in one analysis step. A few tests also verify the import capability. In these simulations the load is applied over two analysis step. Tests are included where the solution obtained by Abaqus/Standard at the end of the first load step is transferred and the rest of the simulation completed in Abaqus/Explicit, as well as tests where the solution starts in Abaqus/Explicit and is then transferred and completed in Abaqus/Standard.

Abaqus/Standard

The impactor is displaced statically to indent the foam. To model the large deformations of the foam, geometric nonlinearities are taken into account in the step. For nonassociated flow cases the unsymmetric storage and solution scheme is activated. This is important to obtain an acceptable rate of convergence during the equilibrium iterations, since the nonassociated flow plasticity model used for the foam has a nonsymmetric stiffness matrix.

The accuracy of the equilibrium solution within a time increment is controlled by iterating until the out-of-balance forces reduce to a small fraction of an average force magnitude calculated internally by Abaqus. The rough punch causes an inhomogeneous stress state: stresses are higher in the region of the mesh near the punch. This tends to cause an underestimation of the average force magnitude since the reference force magnitude is averaged over the entire mesh. To avoid an excessive number of iterations, solution controls for field equations are used to relax the convergence tolerance.

Abaqus/Explicit

The plate is indented quasi-statically when the foam is modeled without rate dependence. An amplitude curve with smoothing is used to specify the displacement of the punch and to promote a quasi-static solution. The plate is indented dynamically when the foam is modeled with rate effects. For this case a ramped velocity profile is prescribed such that the maximum velocity is 5.4 m/sec.

Results and discussion

The same response is obtained in Abaqus/Explicit using the coarse mesh and the fine mesh. The overall load-deflection response of the foam plate is plotted in *Figure 3* for indentation with the cylindrical punch and in *Figure 4* for indentation with the hemispherical punch. In both cases the simulated load-deflection responses are in good agreement with the experimental results by Schluppkotten (1999). The deformed configuration of the mesh at the end of the loading step (showing actual displacements) and the contour plots of the equivalent plastic strain (for the isotropic hardening foam model) or the volumetric compacting plastic strain (for the

volumetric hardening foam model) are shown in *Figure 5* through *Figure 10*. The figures show that the plastic strain magnitude in the vicinity of the punch approaches 180%.

The import analysis can be verified by comparing the results from the zero increment of the imported analysis to the last increment of the previous analysis. In all cases the response of the structure is continuous between the first analysis to the second analysis and compares very closely with solutions obtained using one simulation module. As an example, see *Figure 11* which compares load-deflection responses of the impactor using four different modeling approaches.

Input files

Abaqus/Standard input files

cyl_volstd_reg.inp	Rate-independent case with cylindrical impactor, coarse mesh of the plate, and the volumetric hardening foam model.
sph_volstd_reg.inp	Rate-independent case with hemispherical impactor, coarse mesh of the plate, and the volumetric hardening foam model.
cyl_isostd_reg.inp	Rate-independent case with cylindrical impactor, coarse mesh of the plate, and the isotropic hardening foam model. Base problem for carrying out import from Abaqus/Standard to Abaqus/Explicit.
cyl_isostd_regimport.inp	Import into Abaqus/Standard from base problem cyl_isoexp_reg.inp.
sph_isostd_regrate.inp	Rate-dependent case with hemispherical impactor, coarse mesh of the plate, and the isotropic hardening foam model.

Abaqus/Explicit input files

cyl_isoexp_reg.inp	Rate-independent case with cylindrical impactor, coarse mesh of the plate, and the isotropic hardening foam model. Base problem for carrying out import from Abaqus/Explicit to Abaqus/Standard.
cyl_isoexp_regimport.inp	Import into Abaqus/Explicit from base problem cyl_isostd_reg.inp.
cyl_isoexp_fin.inp	Rate-independent case with cylindrical impactor, fine mesh of the plate, and the isotropic hardening foam model.
cyl_isoexp_regrate.inp	Rate-dependent case with cylindrical impactor, coarse mesh of the plate, and the isotropic hardening foam model.
cyl_volexp_reg.inp	Rate-independent case with cylindrical impactor, coarse mesh of the plate, and the volumetric hardening foam model.
cyl_volexp_fin.inp	Rate-independent case with cylindrical impactor, fine mesh of the plate, and the volumetric hardening foam model.
cyl_volexp_regrate.inp	Rate-dependent case with cylindrical impactor, coarse mesh of the plate, and the volumetric hardening foam model.
sph_isoexp_reg.inp	Rate-independent case with hemispherical impactor, coarse mesh of the plate, and the isotropic hardening foam model.
sph_isoexp_fin.inp	Rate-independent case with hemispherical impactor, fine mesh of the plate, and the isotropic hardening foam model.
sph_isoexp_regrate.inp	Rate-dependent case with hemispherical impactor, coarse mesh of the plate, and the isotropic hardening foam model.
sph_volexp_reg.inp	Rate-independent case with hemispherical impactor, coarse mesh of the plate, and the volumetric hardening foam model.
sph_volexp_fin.inp	Rate-independent case with hemispherical impactor, fine mesh of the plate, and the volumetric hardening foam model.

sph_volexp_regrate.inp	Rate-dependent case with hemispherical impactor, coarse mesh of the plate, and the volumetric hardening foam model.

References

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

Figures

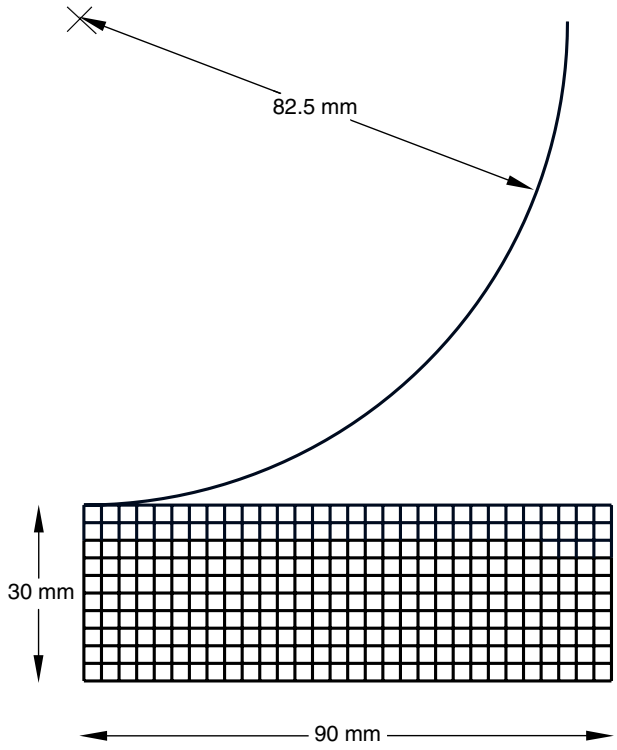


Figure 1: Model for foam indentation by cylindrical or hemispherical punch.

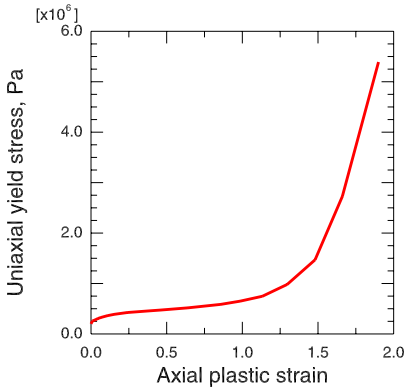


Figure 2: Uniaxial compression test of a sample material.

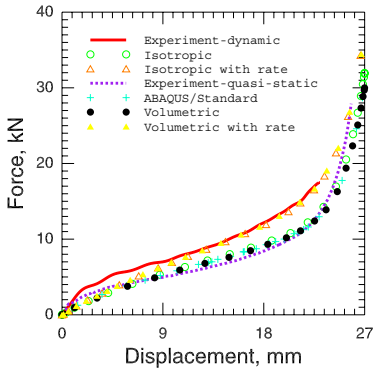


Figure 3: Cylindrical punch force versus penetration response.

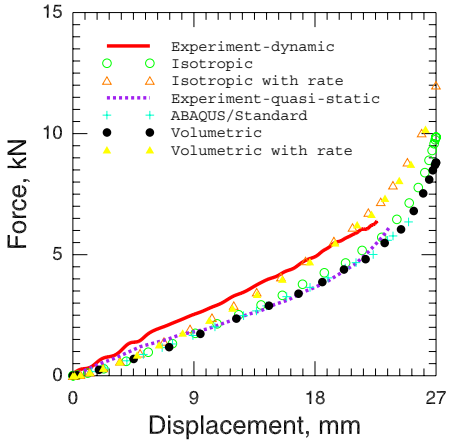


Figure 4: Hemispherical punch force versus penetration response.

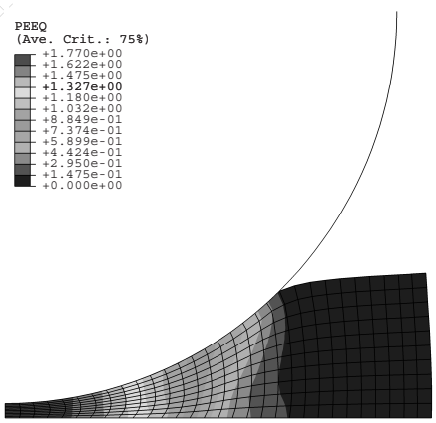


Figure 5: Deformed configuration and contours of the equivalent plastic strain for indentation with cylindrical impactor and the isotropic hardening foam model in Abaqus/Explicit.

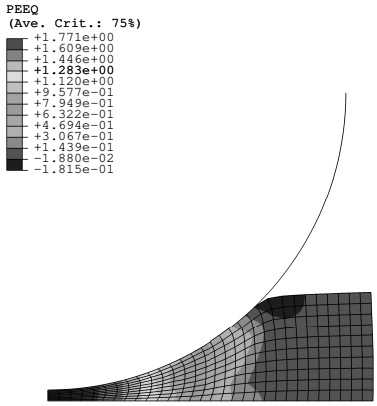


Figure 6: Deformed configuration and contours of the volumetric compacting plastic strain for indentation with cylindrical impactor and the volumetric hardening foam model in Abaqus/Explicit.

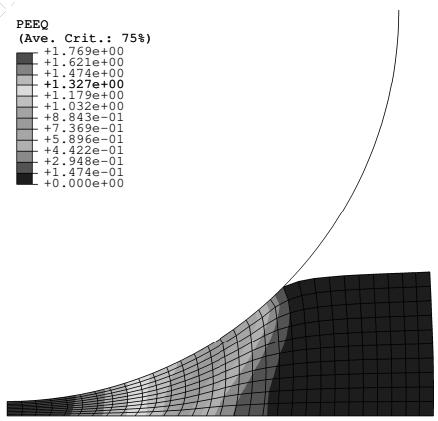


Figure 7: Deformed configuration and contours of the equivalent plastic strain for indentation with hemispherical impactor and the isotropic hardening foam model in Abaqus/Explicit.

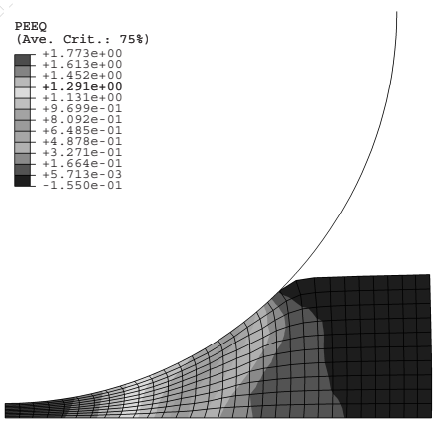


Figure 8: Deformed configuration and contours of the volumetric compacting plastic strain for indentation with hemispherical impactor and the volumetric hardening foam model in Abaqus/Explicit.

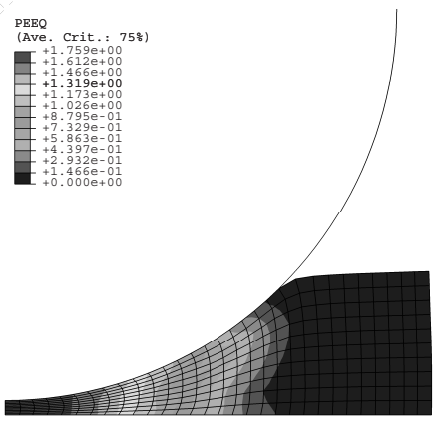


Figure 9: Deformed configuration and contours of the volumetric compacting plastic strain for indentation with cylindrical impactor in Abaqus/Standard.

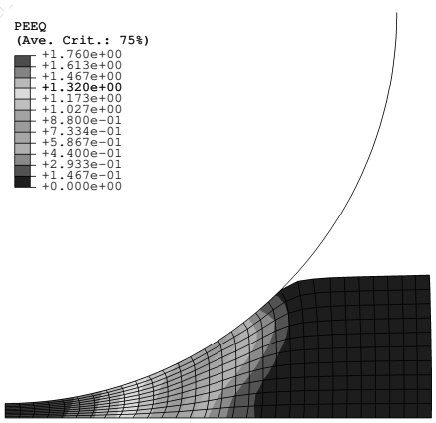


Figure 10: Deformed configuration and contours of the volumetric compacting plastic strain for indentation with hemispherical impactor in Abaqus/Standard.

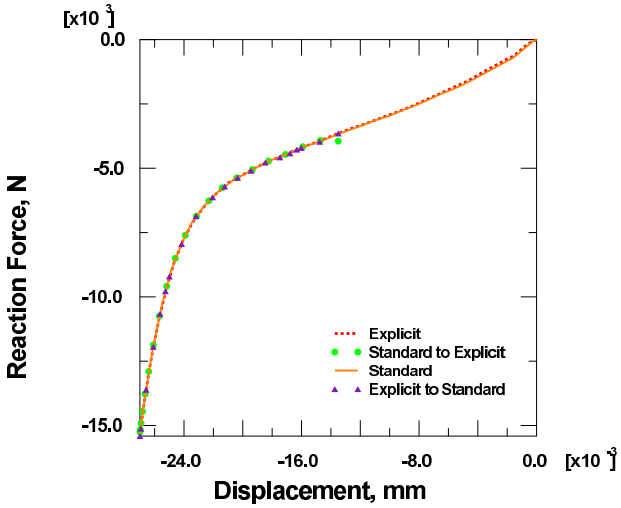


Figure 11: Load-deflection response of the impactor using the import capability to transfer the solution between Abaqus/Standard and Abaqus/Explicit.

Notched unreinforced concrete beam under 3-point bending

Products: Abaqus/Standard Abaqus/Explicit

Overview

This example illustrates the use of the concrete damaged plasticity model in both Abaqus/Standard and Abaqus/Explicit for the analysis of an unreinforced notched concrete beam under 3-point bending.

This problem is chosen because it has been studied extensively both experimentally by Petersson (1981) and analytically by Rots et al. (1984, 1985), de Borst (1986), and Meyer et al. (1994), among others. The predominant behavior is Mode I cracking, so the example provides good verification of this aspect of the constitutive model. We also have the advantage that this beam experiment has been repeated by a number of different researchers, and there is good material information about important parameters, such as the Mode I fracture energy, G_f^I. Thus, we can directly compare the numerical results with the experimental results with minimal uncertainty. We also investigate the sensitivity of the numerical results to the finite element discretization and to the choice of cracking material properties.

Abaqus provides constitutive models suitable for brittle materials such as concrete in which cracking is important. These models are intended for unreinforced as well as reinforced concrete structures.

The concrete damaged plasticity model in Abaqus provides a general capability for modeling plain or reinforced concrete in the applications of monotonic, cyclic, and/or dynamic loading. This model can be used to simulate the irreversible damage involved in the fracturing process and the recovery of stiffness as loads change from tension to compression or vice versa. In addition, this model can include strain rate dependency. For more details on this model, see *Concrete Damaged Plasticity*.

In addition to the concrete damaged plasticity model, Abaqus provides the smeared cracking concrete model in Abaqus/Standard and the brittle cracking model in Abaqus/Explicit. For a description of these models, see *Concrete Smeared Cracking* and *Cracking Model for Concrete*.

Problem description

The notched beam is shown in *Figure 1*. Because of symmetry, only one half of the beam is modeled. *Figure 2* shows the three meshes used for this problem: a coarse mesh of 70 elements, a medium mesh of 280 elements, and a fine mesh of 1120 elements. We model the beam using plane stress (CPS4R) elements and three-dimensional (C3D8R) elements to provide verification of both element types.

The beam has a Young's modulus of 30 GPa (4.35×10^6 lb/in^2), a Poisson's ratio of 0.20, a density of 2400 kg/m^3 (0.225×10^{-3} lb s^2/in^4), a cracking failure stress of 3.33 MPa (482.96 lb/in^2), and a Mode I fracture energy G_f^I of 124 N/m (0.708 lb/in). The fracture energy value, G_f^I, defines the area under the postcracking stress-displacement curve. The effect of different postcracking softening behavior is the subject of one of the studies carried out in this example.

Loading

The beam is loaded by prescribing the vertical displacement at the center of the beam until it reaches a value of 0.0015 m.

Solution control

The Riks method is used in Abaqus/Standard since the behavior of the beam is quite unstable when cracking progresses.

Abaqus/Explicit is a dynamic analysis program. In this case we are interested in static solutions; hence, care must be taken that the beam is loaded slowly enough to eliminate significant inertia effects. For problems involving brittle failure, this is especially important since the sudden drops in load-carrying capacity that normally accompany brittle behavior generally lead to increases in the kinetic energy content of the response. Therefore, the beam is loaded by applying a velocity that increases linearly from 0 to 0.06 m/s over a period of 0.05 seconds to obtain the final displacement of 0.0015 m at the center of the beam. This ensures a quasi-static solution (the kinetic energy in the beam is small throughout the response) in a reasonable number of time increments. Nevertheless, oscillations in the load-displacement response caused by inertia effects are still visible, mainly after the concrete has cracked significantly.

The speed of application of the loading in Abaqus/Explicit is the subject of another study in this problem.

Results and discussion

Results are described below for each analysis variation.

Mesh refinement study

The three finite element meshes described earlier are used to show the influence of mesh refinement on the load-displacement response of the concrete beam.

Since there is no reinforcement in this problem, the postfailure behavior is specified in terms of the stress-displacement response to minimize mesh sensitivity. We can also specify the postfailure behavior directly in terms of the fracture energy, G_f^I. The fracture energy method assumes a linear loss of strength after cracking. Thus, if we specify the tension softening behavior in terms of stress versus cracking displacement and assume a linear curve (σ_{tu}^I, 0), (0, $2G_f^I/\sigma_{tu}^I$) as shown in *Figure 3*, the above two methods will give the same results.

Tensile damage is specified in terms of the tension damage variable, d_t, versus the cracking displacement. A linear dependence—(0, 0), (0.9, $2G_f^I/\sigma_{tu}^I$)—is assumed for this study, as shown in *Figure 4*. For the constitutive calculations, Abaqus automatically converts the cracking displacement values to "plastic" displacement values using the relationship

$$u_t^{pl} = u_t^{ck} - \frac{d_t}{(1-d_t)}\frac{\sigma_t l_0}{E_0},$$

where the specimen length, l_0, is assumed to be one unit; (i.e., $l_0 = 1$). Care must be taken in specifying the tension damage to ensure that the calculated plastic strain (or displacement) is positive and monotonically increasing with increasing cracking strain (or displacement).

The load-displacement response of the notched beam obtained for the three meshes with Abaqus/Standard is shown in *Figure 5* for the three-dimensional models and in *Figure 6* for the plane stress models. The load-displacement response obtained with Abaqus/Explicit is shown in *Figure 7* for the three-dimensional models and in *Figure 8* for the plane stress models. These figures show that the three-dimensional and plane stress models in Abaqus/Standard are in close agreement. Minor differences are observed in the results obtained with Abaqus/Explicit; these can be attributed primarily to dynamic effects. Three-dimensional models have a relatively higher level of mesh sensitivity due to the effect of possible cracking in the out-of-plane direction. For the two-dimensional models, although a small amount of mesh sensitivity remains between the coarse mesh and the other two meshes, the medium and fine meshes give similar results. Based on these observations, all subsequent studies are done using the plane stress medium mesh. All the curves shown are smoothed. Displaced shapes

obtained with Abaqus/Standard for the three plane stress meshes are shown in *Figure 9*. The three-dimensional meshes and the Abaqus/Explicit meshes show essentially the same deformation. The expected Mode I fracture pattern is observed consistently in all meshes.

Influence of tension softening

The results described above are obtained using linear tension softening. Such a choice of softening leads to a response that is too stiff compared with the experimental observations of Petersson. In this study we use three different evolutions of the stress as a function of the cracking displacement. We compare the linear variation used previously to two tension softening functions where the cracking stress is reduced more rapidly as the crack initiates. These functions are shown in *Figure 10*: one consists of a two-segment representation of softening, and the other is a four-segment representation. The area under the softening curve is the same in all cases so that the value of the Mode I fracture energy of the material is preserved. Different linear tension damage curves are used for each tension softening model in this study to ensure that the plastic displacement is positive and monotonically increasing with increasing cracking displacement for all three tension softening curves.

The load-displacement responses obtained with Abaqus/Standard for the three tension softening representations are shown in *Figure 11* for the plane stress medium mesh. For Abaqus/Explicit the responses are shown in *Figure 12* for the plane stress medium mesh. It is clear that more rapid reductions of the cracking stress after initial cracking lead to less stiff responses. The modeling of tension softening is a key determinant of the peak/failure response. The two-segment and four-segment softening models provide peak/failure responses that agree well with the experimental observations of Petersson. The initial linear responses of the calculated results are slightly softer than the experimental results. This small difference is because a relatively blunt notch is used in this study, while a much sharper cast notch was used in Petersson (1981). All the curves shown have been smoothed.

Influence of speed of application of the load and curve smoothing in Abaqus/Explicit

The quasi-static solutions obtained in the previous Abaqus/Explicit studies still show some oscillations due to inertia effects, albeit somewhat hidden by the fact that curve smoothing is used. This additional exercise is intended to show the difference between the unsmoothed and smoothed responses obtained at the loading speed used thus far (0.06 m/s) and an analysis where the loading is applied at a much lower speed (0.005 m/s).

Figure 13 shows the results obtained for the plane stress medium mesh with four-segment tension softening. Smoothing of the faster load-displacement response (19635 analysis increments) is shown to match reasonably well the load-displacement response obtained at the slower speed (235830 analysis increments). Since the slower response does not provide much more useful information, we conclude that we are justified to run at the faster speed and to use smoothing to present the quasi-static response.

Input files

Abaqus/Standard input files

Three-dimensional mesh:

notchedconcbeam_3d_coarse_std.inp	Coarse mesh response.
notchedconcbeam_3d_medium_std.inp	Medium mesh response.
notchedconcbeam_3d_fine_std.inp	Fine mesh response.

Plane stress mesh:

notchedconcbeam_2d_coarse_std.inp	Coarse mesh response.
notchedconcbeam_2d_medium_std.inp	Medium mesh response.
notchedconcbeam_2d_fine_std.inp	Fine mesh response.
notchedconcbeam_2d_gfi_std.inp	Medium mesh response with **CONCRETE TENSION STIFFENING*, TYPE=GFI.

notchedconcbeam_2d_1seg_std.inp	Medium mesh, one-segment tension softening response with **TENSION STIFFENING*, TYPE=DISPLACEMENT.
notchedconcbeam_2d_2seg_std.inp	Medium mesh, two-segment tension softening response with **TENSION STIFFENING*, TYPE=DISPLACEMENT.
notchedconcbeam_2d_4seg_std.inp	Medium mesh, four-segment tension softening response with **TENSION STIFFENING*, TYPE=DISPLACEMENT.

Abaqus/Explicit input files

Three-dimensional mesh:

notchedconcbeam_3d_coarse_xpl.inp	Coarse mesh response.
notchedconcbeam_3d_medium_xpl.inp	Medium mesh response.
notchedconcbeam_3d_fine_xpl.inp	Fine mesh response.
notchedconcbeam_3d_fine_del_xpl.inp	Fine mesh response with element removal.

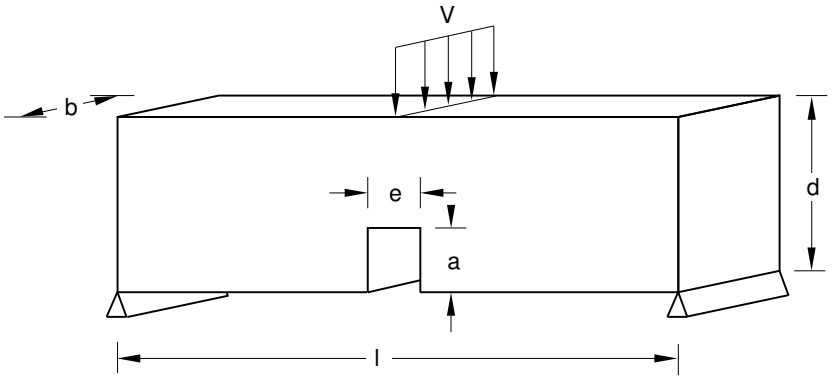
Plane stress mesh:

notchedconcbeam_2d_coarse_xpl.inp	Coarse mesh response.
notchedconcbeam_2d_medium_xpl.inp	Medium mesh response.
notchedconcbeam_2d_fine_xpl.inp	Fine mesh response.
notchedconcbeam_2d_fine_del_xpl.inp	Fine mesh response with element removal.
notchedconcbeam_2d_1seg_xpl.inp	Medium mesh, one-segment tension softening response.
notchedconcbeam_2d_2seg_xpl.inp	Medium mesh, two-segment tension softening response.
notchedconcbeam_2d_4seg_xpl.inp	Medium mesh, four-segment tension softening response.
notchedconcbeam_2d_speed2_xpl.inp	Medium mesh, 0.005 m/s speed response.

References

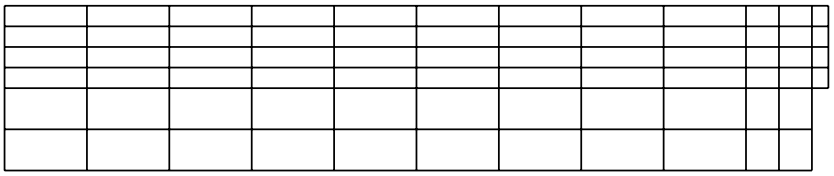
- de Borst, R., Ph. D. thesis, Delft University of Technology, The Netherlands, 1986.
- Meyer, R., H. Ahrens, and H. Duddeck, "Material Model for Concrete in Cracked and Uncracked States," *Journal of Engineering Mechanics Division, ASCE*, vol. 120, EM9, pp. 1877–1895, 1994.
- Petersson, P. E., "Crack Growth and Development of Fracture Zones in Plain Concrete and Similar Materials," *Report No. TVBM-1006, Division of Building Materials, University of Lund, Sweden*, 1981.
- Rots, J. G., G. M. A. Kusters, and J. Blaauwendraad, "The Need for Fracture Mechanics Options in Finite Element Models for Concrete Structures," *Computer-Aided Analysis and Design of Concrete Structures, Pineridge Press, Swansea, United Kingdom*, pp. 19–32, 1984.
- Rots, J. G., P. Nauta, G. M. A. Kusters, and J. Blaauwendraad, "Smeared Crack Approach and Fracture Localization in Concrete," *HERON, Delft University of Technology, The Netherlands*, vol. 30, no. 1, 1985.

Figures

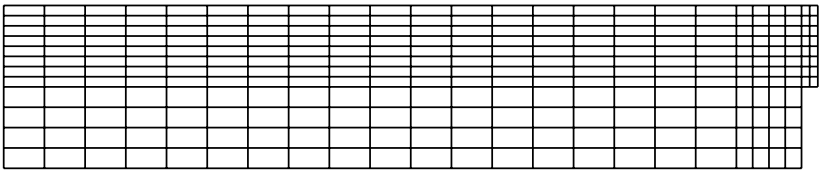


a/d = 0.5
l = 2 m, d = 0.2 m,
b = 0.05 m, e = 0.04 m

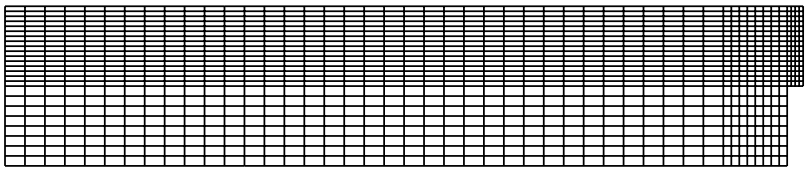
Figure 1: Notched beam: geometry and dimensions.



coarse mesh



medium mesh



fine mesh

Figure 2: Finite element meshes of half of the notched beam.

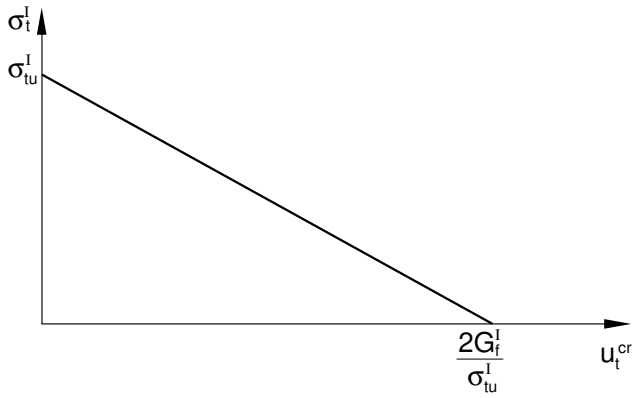


Figure 3: Tension softening model used for mesh refinement study.

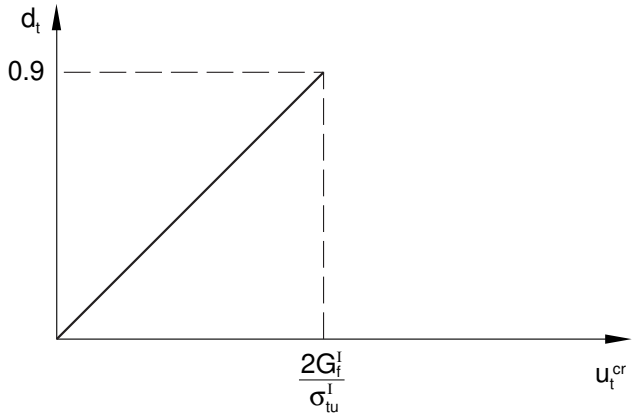


Figure 4: Tension damage curve used for mesh refinement study.

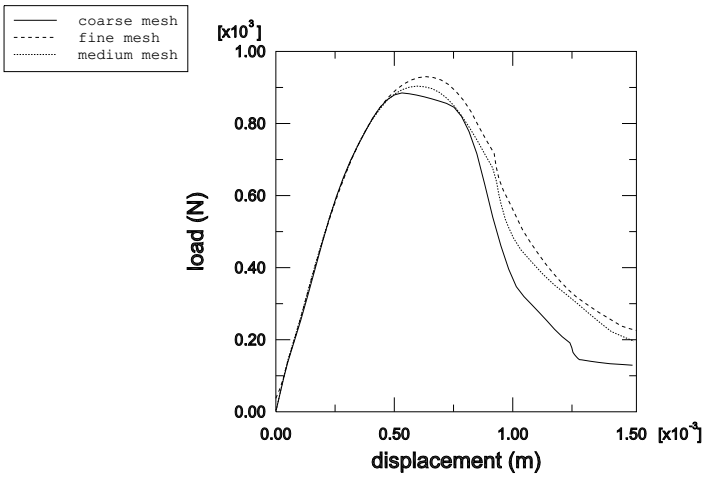


Figure 5: Three-dimensional Abaqus/Standard mesh refinement study.

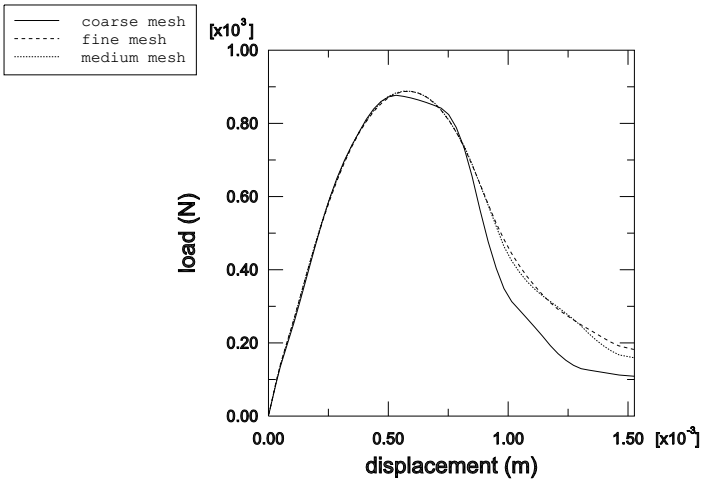


Figure 6: Plane stress Abaqus/Standard mesh refinement study.

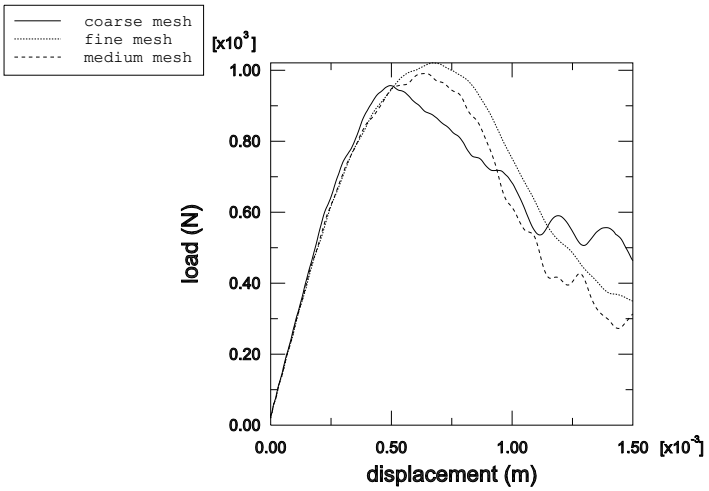


Figure 7: Three-dimensional Abaqus/Explicit mesh refinement study.

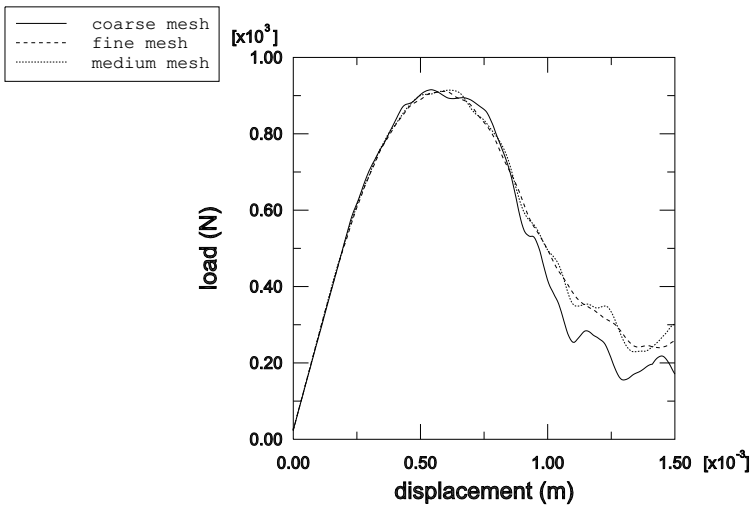


Figure 8: Plane stress Abaqus/Explicit mesh refinement study.

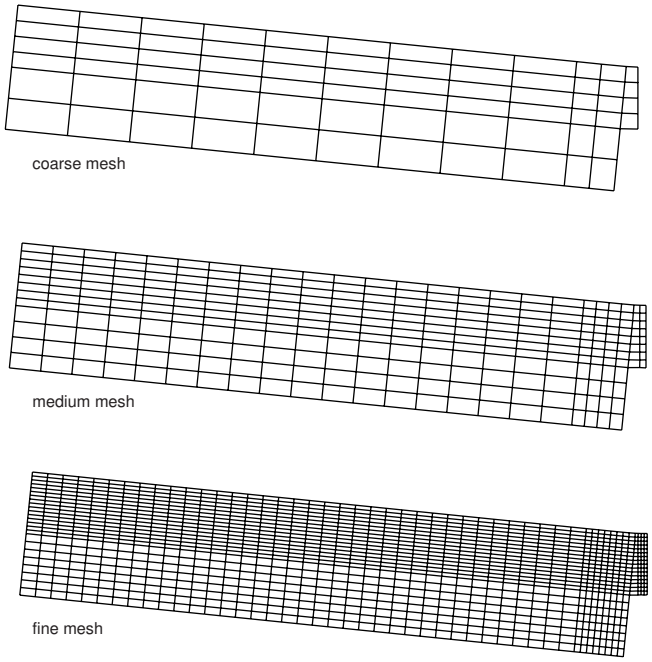


Figure 9: Displaced shapes obtained in the plane stress Abaqus/Standard mesh refinement study (magnification factor 100).

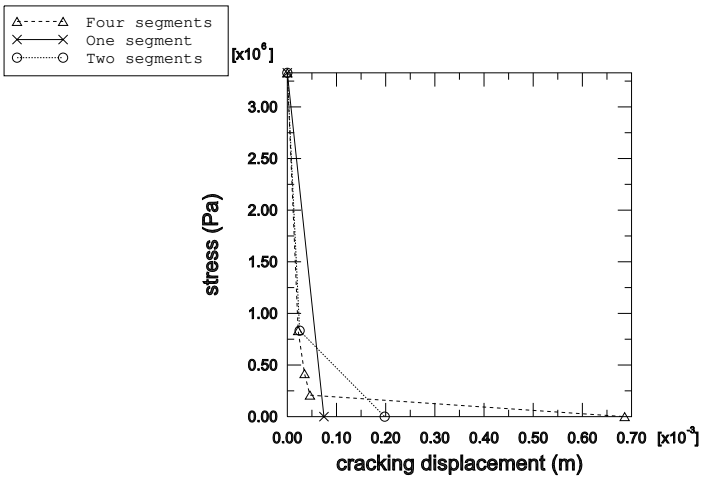


Figure 10: Tension softening models.

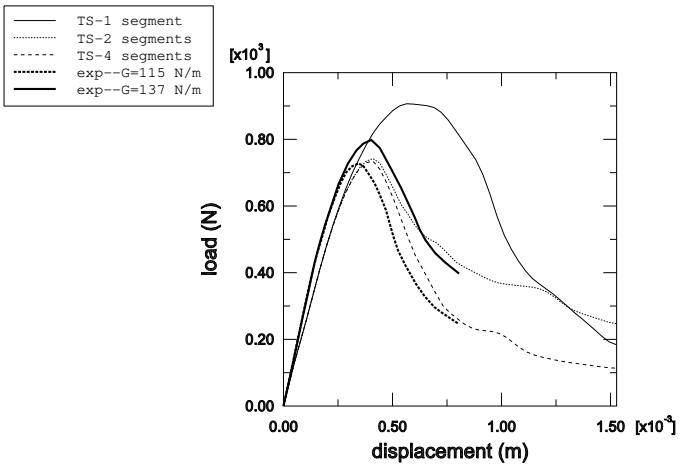


Figure 11: Abaqus/Standard tension softening study: plane stress medium mesh.

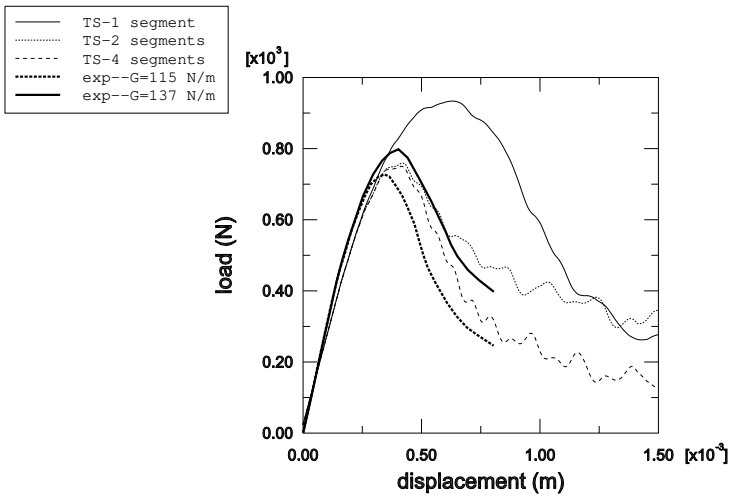


Figure 12: Abaqus/Explicit tension softening study: plane stress medium mesh.

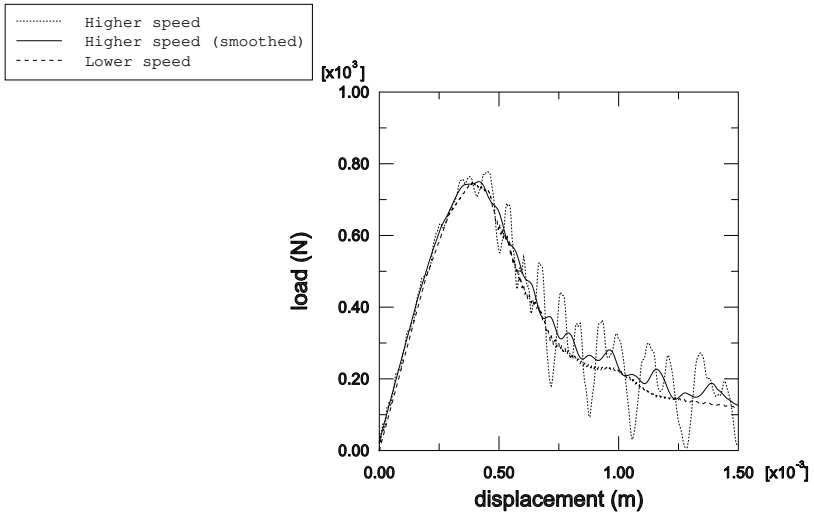


Figure 13: Abaqus/Explicit speed and curve smoothing study: plane stress medium mesh.

Mixed-mode failure of a notched unreinforced concrete beam

Products: Abaqus/Explicit

References:

- *Cracking Model for Concrete*

Overview

This example illustrates the use of the cracking constitutive model for the analysis of an unreinforced notched concrete beam subject to loading that causes mixed-mode cracking.

This problem was chosen because it has been studied extensively both experimentally by Arrea and Ingraffea (1982) and analytically by Rots et al. (1984, 1985, 1987, 1989, 1991, 1992), de Borst (1986, 1987), and Meyer et al. (1994), among others. The behavior in this problem is a combination of Mode I and Mode II cracking. It, therefore, provides verification of the model for general mixed-mode loading. We also have the advantage that this beam experiment has been repeated by a number of different researchers, and there is good material information about important parameters

such as the Mode I fracture energy, G_f^I. We investigate the sensitivity of the numerical results to the finite element discretization as well as the choice of cracking material properties.

Problem description

The notched beam is shown in *Figure 1*. *Figure 2* shows the two meshes used for this problem: a coarse mesh of 210 elements, and a fine mesh of 840 elements. The beam is assumed to be in a state of plane stress, so CPS4R elements are used. The basic concrete material properties used in the beam are given in *Table 1*. The fracture

energy value G_f^I does not completely define the evolution of the postcracking stress; this is the subject of one of the studies carried out in this example. The shear retention properties, given later, are the subject of the other material property study.

Loading and solution control

Since Abaqus/Explicit is a dynamic analysis program, and in this case we are interested in static solutions, care must be taken that the beam is loaded slowly enough to eliminate any significant inertia effects. For problems involving brittle failure, this is especially important since the sudden drops in load-carrying capacity that normally accompany brittle behavior generally lead to increases in the kinetic energy content of the response.

The beam is loaded by applying a velocity that increases linearly from zero to 0.75 mm/second over a period of 0.38 seconds. The velocity is applied at point C and transmitted to the notched beam through the rigid beam AB. The beam itself is not modeled since its kinematic motion can easily be modeled using an equation constraint. The load transmitted at points D and B is distributed over a 30 mm length to avoid hourglassing of the elements in the vicinity of these points where the highest loads are transmitted. The velocity chosen ensures that a quasi-static solution is obtained. The kinetic energy in the beam is small until the crack has propagated across the entire depth of the beam. Nevertheless, oscillations in the load-displacement response caused by inertia effects are still visible, mainly after the concrete has cracked significantly.

Results and discussion

Results are described below for each analysis variation.

Mesh refinement study

Two finite element meshes are used to show the influence of mesh refinement on the load-displacement response

of the concrete beam. The value of the Mode I fracture energy, G_f^I, can be specified directly for brittle cracking properties to define tension softening behavior that gives approximately mesh insensitive results. However, this is not done here for two reasons: first, this specification restricts the postcracking normal stress evolution to a linear variation, and we want to be more flexible than that in some of our studies; second, by specifying the postfailure stress-strain relationship directly, we show how Abaqus/Explicit converts fracture energy data into cracking stress versus cracking strain data.

If we specify the tension softening behavior in terms of stress versus cracking strain and we assume a linear dependence of stress on cracking strain, as shown in *Figure 3*, the cracking strain at which the stress reaches a

zero value, ε_0^{ck}, can be calculated as $2G_f^I / (\sigma_{tu}^I h)$, where σ_{tu}^I is the cracking failure stress and h is a characteristic element length. This characteristic length represents the size of the element that cracks and has values of 15 and 7.5 mm for the coarse and fine meshes, respectively. This method of calculating the cracking strain at which the stress reaches a zero value provides material data that will give approximately mesh insensitive results and is essentially what Abaqus/Explicit does when the parameter TYPE=GFI is used. This is discussed in more detail in *Cracking Model for Concrete* and *A cracking model for concrete and other brittle materials*.

The shear retention properties used for the two meshes are shown in *Figure 4*. The evolution of the shear retention factor, ρ, is chosen such that the shear resistance of the material is reduced drastically as soon as the crack initiates.

The response of the load transmitted at point B or D versus the crack mouth sliding displacement (CMSD) of the notched beam obtained with the two meshes is shown in *Figure 5*. This figure shows that the coarse and fine meshes give similar results. Based on this observation, all subsequent studies are performed using only the fine mesh. Displaced shapes and crack patterns obtained at the end of the analysis are shown for the two meshes in *Figure 6* and *Figure 7*. The crack propagation path tends to curve away from the original crack tip and move toward point B. This behavior is typical for a crack subjected to mixed-mode loading.

Influence of tension softening

The previous results were obtained using linear tension softening. The maximum load-carrying capacity of the beam compares well with the experimental observations of Arrea and Ingraffea. However, the postcracking behavior is somewhat stiff compared to the experiments. In the following study we use three different evolutions of the stress as a function of cracking strain. We compare the linear variation used previously to two tension softening functions where the stress is reduced more rapidly as the crack initiates. These functions are shown in *Figure 8*: one consists of a two-segment representation of softening, and the other is a four-segment representation. The area under the softening curve is the same in all cases so that the value of the Mode I fracture energy of the material is preserved.

The load-CMSD responses obtained for the three tension softening representations are shown in *Figure 9*. Although the analyses were performed over the same duration (0.38 seconds), the end value of the crack mouth sliding displacement increases as tension softening is lowered. This is to be expected, since the crack faces are likely to slide more with respect to each other as tension softening is lowered. The peculiar behavior observed at a CMSD value of about 0.15 mm in the case of the four-segment tension softening simply shows that the response is no longer quasi-static because the crack has propagated completely through the depth of the beam. It is clear that more rapid reductions of the stress after initial cracking lead to less stiff responses. Although the simulation predicts the trend of the experimental results, the decrease in the simulated load-carrying capacity in the softening region is not as great as the experimental results suggest. The effect of shear retention is, therefore, addressed next in an attempt to bring the numerical results closer to the experimental observations.

Influence of shear retention

Two different evolutions of shear retention are used to show the influence of shear retention on the load-CMSD response of the beam. One is the evolution of shear retention that was used in all previous analyses. The other is a lower shear retention model, as shown in *Figure 10*. This lower shear retention model corresponds to practically no shear carrying capability in the cracked elements once cracking initiates.

The load-CMSD responses obtained for these two cases are shown in *Figure 11* for the fine mesh with the two-segment tension softening model and in *Figure 12* for the fine mesh with the four-segment tension softening model. Although we still apply the same linearly varying velocity at point C (0.75 mm/second at 0.38 seconds), the analyses for the lower shear retention model were stopped at 0.36 seconds and 0.34 seconds for the mesh with the two- and four-segment tension softening models, respectively. These times roughly correspond to times at which the crack has propagated across the entire depth of the beam. Responses obtained after these times are no longer meaningful in the context of this problem, since the beam no longer has any static load-carrying capacity, and the applied velocity loading causes the beam to respond dynamically.

The results show that, even using zero shear retention, the numerical simulation is not able to predict both a peak load of about 140 kN and the sharp reduction of that load observed in the experiments. This can be explained by the bias introduced when using a rectangular mesh, which tends to promote crack propagation along vertical lines of elements instead of the more curved crack path observed in the experiments. Rots et al. (1989) have indeed shown numerical results that match the softening response of the beam better by using a mesh designed with elements aligned along the experimentally observed curved crack path. This can be done in a case such as this one where good experimental data exist, but it is not possible in general. Results obtained for plain concrete should, therefore, be treated as only relatively coarse approximations of actual behavior.

Effect of element removal

Abaqus/Explicit provides a brittle failure criterion that allows elements to be removed when any local direct cracking strain (or displacement) reaches a failure strain (or displacement). This option is intended primarily to avoid analyses that end prematurely because cracked elements undergo too severe distortion. However, as discussed later, by setting the failure strain for element removal to a relatively low value, the removal of cracked elements can also create a significantly weaker postfailure behavior.

Figure 13 and *Figure 14* show the effect of element removal. In *Figure 13* the two- and four-segment tension softening curves of *Figure 8* are used, respectively, and the failure strain is chosen as 0.4%. The load-CMSD responses obtained for these two simulations are plotted compared to the corresponding responses without element removal. In *Figure 14* the two-segment tension softening curve is used. Two levels of failure strain—i.e., 0.2% and 0.4%, respectively—are considered. The resulting load-CMSD responses are plotted along with the corresponding responses without element removal. As expected, the use of this brittle failure model produces a large drop in the load after the peak load is reached.

Input files

mixedmodeconcbeam_1.inp	Input data used to obtain the coarse mesh response shown in *Figure 5*.
mixedmodeconcbeam_2.inp	Input data used to obtain the fine mesh response shown in *Figure 5*.
mixedmodeconcbeam_2_subcyc.inp	Input data used to obtain the fine mesh response shown in *Figure 5* with subcycling.
mixedmodeconcbeam_3.inp	Input data used to obtain the fine mesh, two-segment tension softening response shown in *Figure 9*.
mixedmodeconcbeam_4.inp	Input data used to obtain the fine mesh, four-segment tension softening response shown in *Figure 9*.
mixedmodeconcbeam_5.inp	Input data used to obtain the fine mesh, two-segment tension softening, zero shear retention response shown in *Figure 11*.

mixedmodeconcbeam_6.inp	Input data used to obtain the fine mesh, four-segment tension softening, zero shear retention response shown in *Figure 12*.
mixedmodeconcbeam_7.inp	Input data used to obtain the fine mesh, 0.4% failure strain, and the four-segment tension softening response shown in *Figure 13*.
mixedmodeconcbeam_8.inp	Input data used to obtain the fine mesh, 0.4% failure strain, and the two-segment tension softening response shown in *Figure 13* and *Figure 14*.
mixedmodeconcbeam_9.inp	Input data used to obtain the fine mesh, 0.2% failure strain, and the two-segment tension softening response shown in *Figure 14*.

References

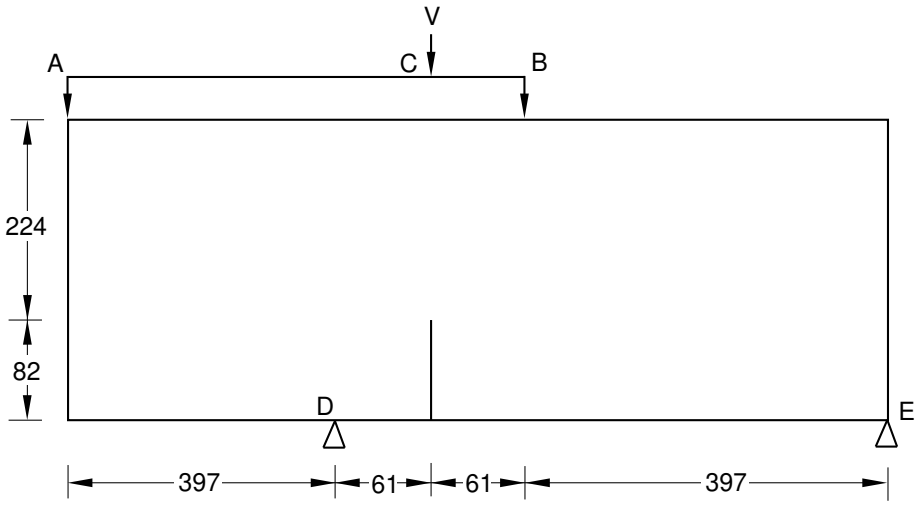
- Arrea, M., and A. R. Ingraffea, "Mixed-Mode Crack Propagation in Mortar and Concrete," *Report No. 81–13, Dept. of Structural Engineering, Cornell University, Ithaca, N.Y.*, 1982.
- de Borst, R., Ph.D. thesis, Delft University of Technology, The Netherlands, 1986.
- de Borst, R., "Computation of Post-Bifurcation and Post-Failure Behavior of Strain-Softening Solids," *Computers and Structures*, vol. 25, no. 2, pp. 211–224, 1987.
- Meyer, R., H. Ahrens, and H. Duddeck, "Material Model for Concrete in Cracked and Uncracked States," *Journal of Engineering Mechanics Division, ASCE*, vol. 120, EM9, pp. 1877–1895, 1994.
- Rots, J. G., "Removal of Finite Elements in Smeared Crack Analysis," *Proceeding of the Third Conference on Computational Plasticity, Fundamentals and Applications, Part I, Pineridge Press, Swansea, United Kingdom*, pp. 669–680, 1992.
- Rots, J. G., "Smeared and Discrete Representations of Localized Fracture," *International Journal of Fracture*, vol. 51, pp. 45–59, 1991.
- Rots, J. G., and J. Blaauwendraad, "Crack Models for Concrete: Discrete or Smeared? Fixed, Multi-Directional or Rotating?," *HERON, Delft University of Technology, The Netherlands*, vol. 34, no. 1, 1989.
- Rots, J. G., and R. de Borst, "Analysis of Mixed-Mode Fracture in Concrete," *ASCE Journal of Engineering Mechanic*, vol. 113, EM11, pp. 1739–1758, 1987.
- Rots, J. G., G. M. A. Kusters, and J. Blaauwendraad, "The Need for Fracture Mechanics Options in Finite Element Models for Concrete Structures," *Computer-Aided Analysis and Design of Concrete Structures, Pineridge Press, Swansea, United Kingdom*, pp. 19–32, 1984.
- Rots, J. G., P. Nauta, G. M. A. Kusters, and J. Blaauwendraad, "Smeared Crack Approach and Fracture Localization in Concrete," *HERON, Delft University of Technology, The Netherlands*, vol. 30, no. 1, 1985.

Tables

Table 1: Concrete material properties.

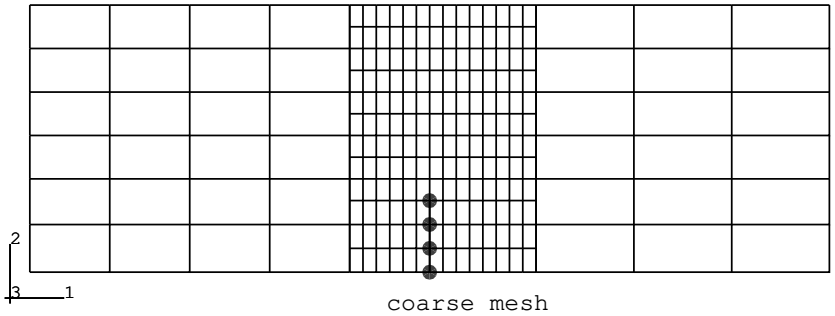
Young's modulus:	24800 N/mm^2 (3.60 × 10^6 lb/in^2)
Poisson's ratio:	0.18
Cracking failure stress:	2.8 N/mm^2 (406.09 lb/in^2)
Mode I fracture energy G_f^I:	0.055 N/mm (0.314 lb/in)
Density:	2.4 × 10^{-6} kg/mm^3 (0.225 × 10^{-3} lb s^2/in^4)

Figures



thickness : 156
dimensions in mm

Figure 1: Notched, mixed-mode beam: geometry and dimensions.



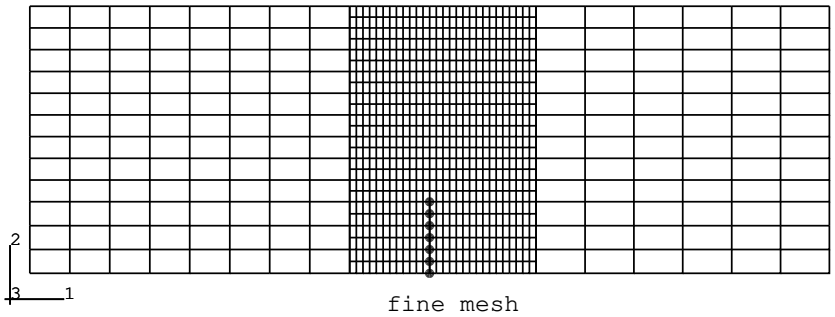


Figure 2: Finite element meshes used for notched, mixed-mode concrete beam.

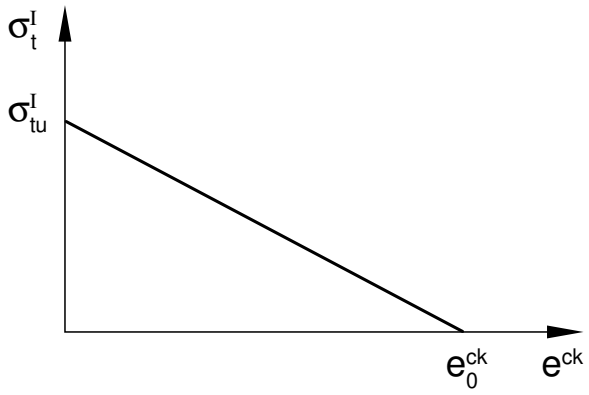


Figure 3: Tension softening model used for mesh refinement study.

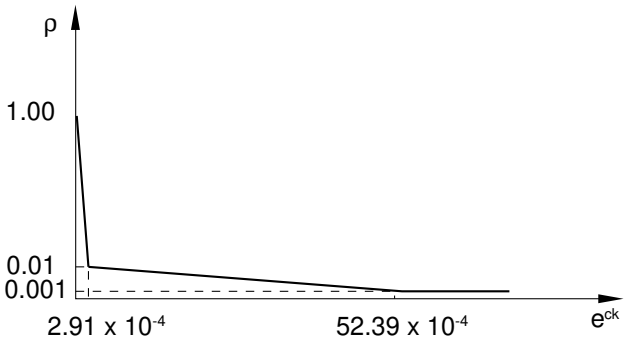


Figure 4: Shear retention model used for mesh refinement study.

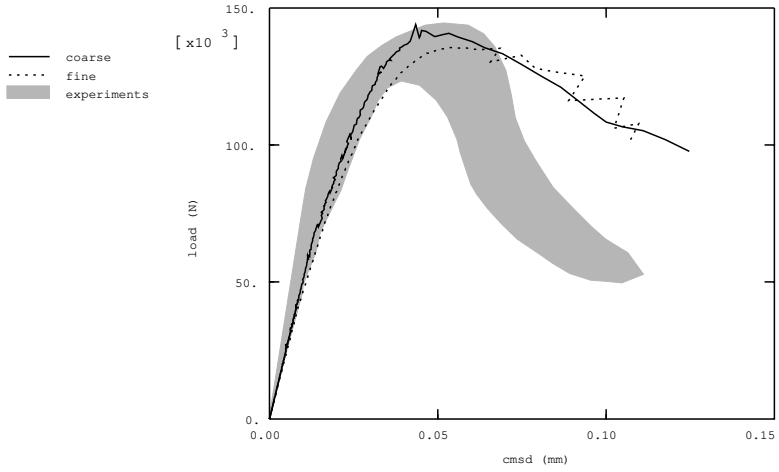


Figure 5: Mesh refinement study: load-CMSD responses.

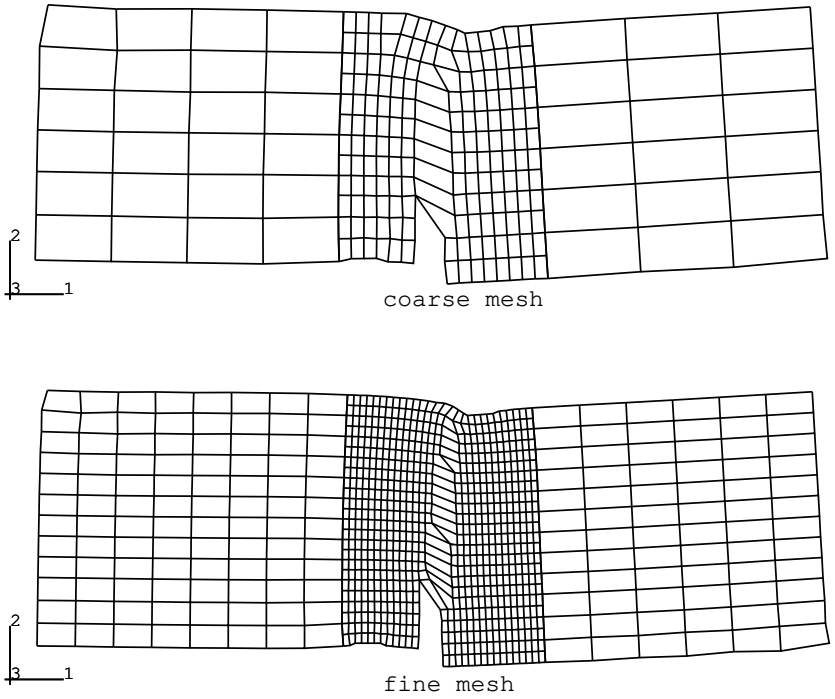


Figure 6: Displaced shapes obtained in mesh refinement study (magnification factor 200).

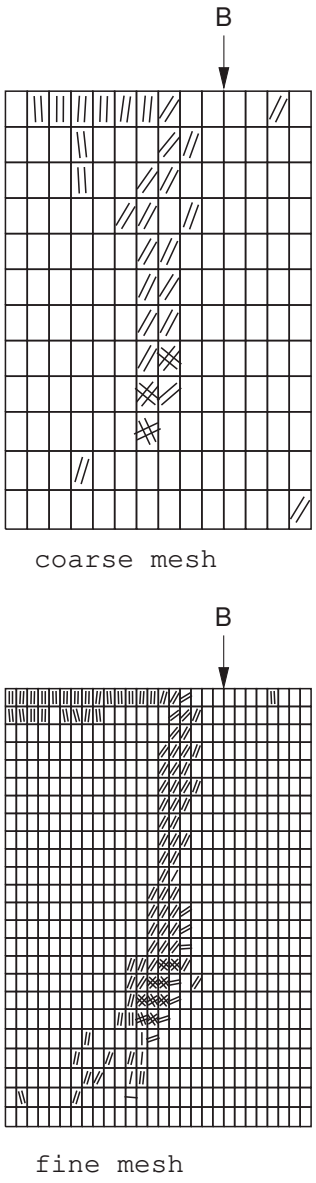


Figure 7: Crack patterns obtained in mesh refinement study (detail of the concrete beam around its notch).

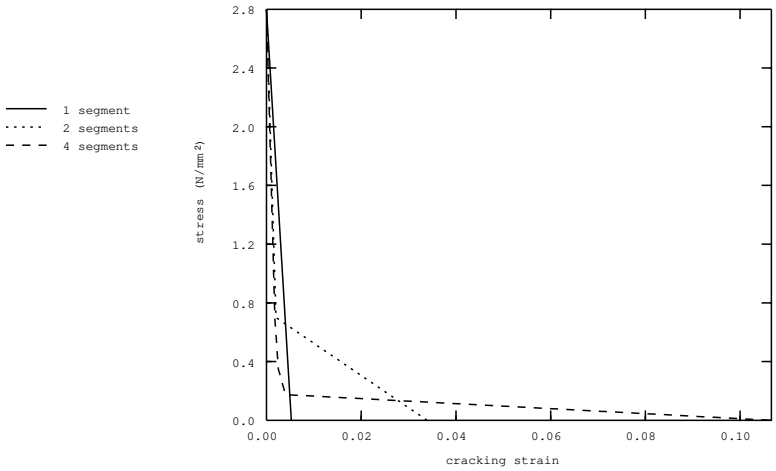


Figure 8: Tension softening models.

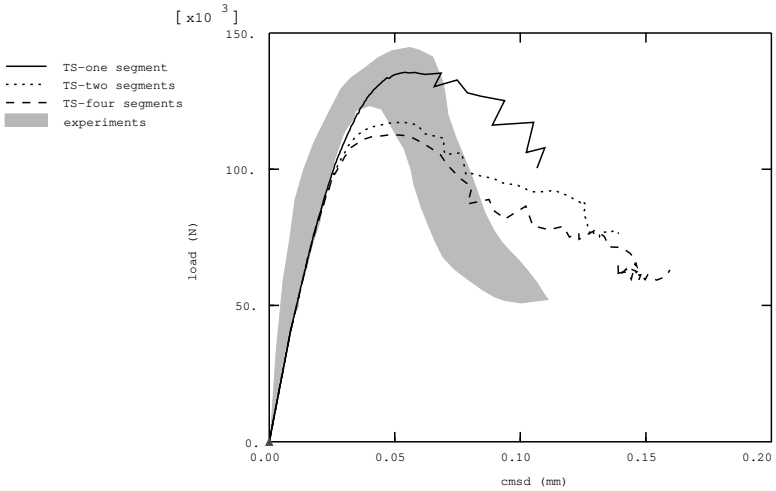


Figure 9: Tension softening study; fine mesh.

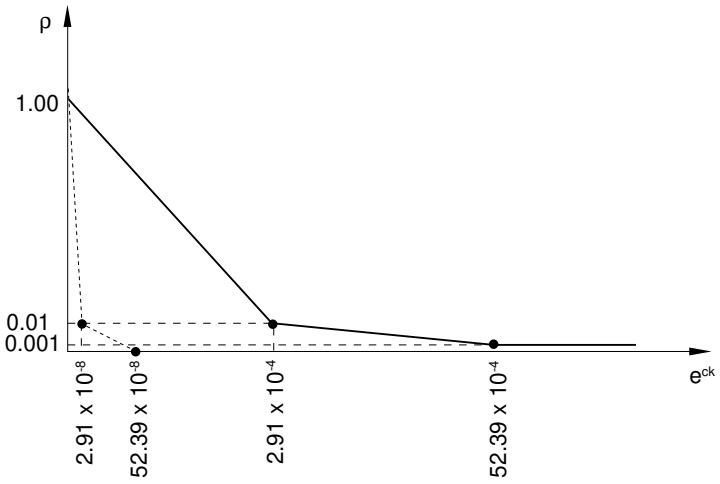


Figure 10: Shear retention models.

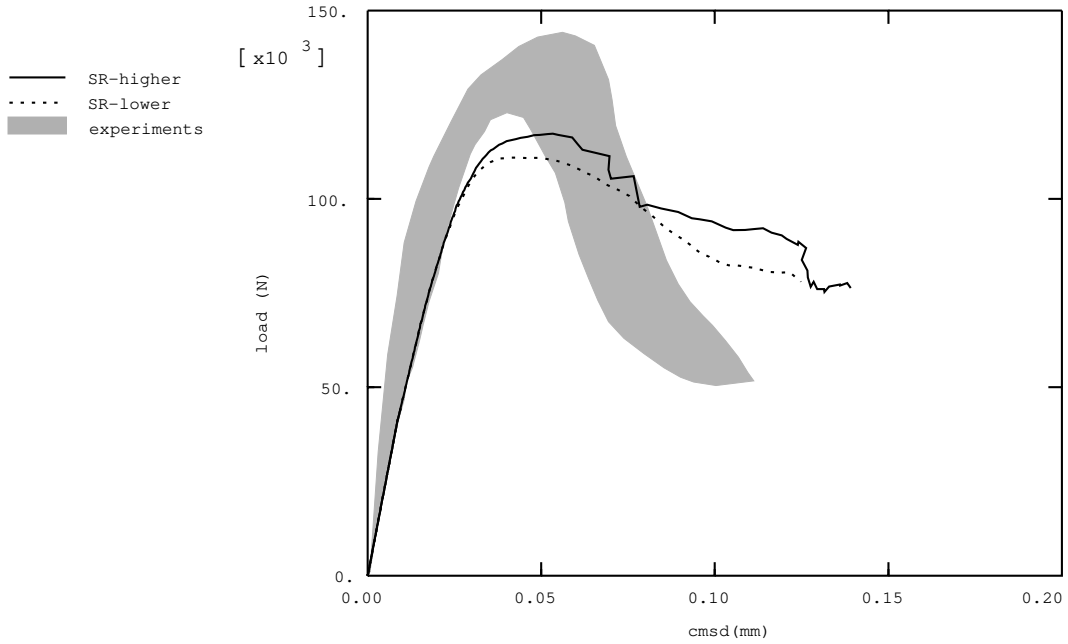


Figure 11: Shear retention study; fine mesh with two-segment tension softening.

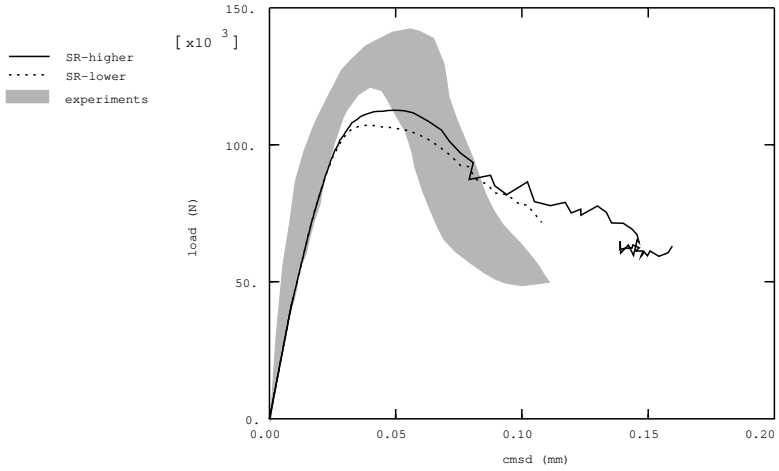


Figure 12: Shear retention study; fine mesh with four-segment tension softening.

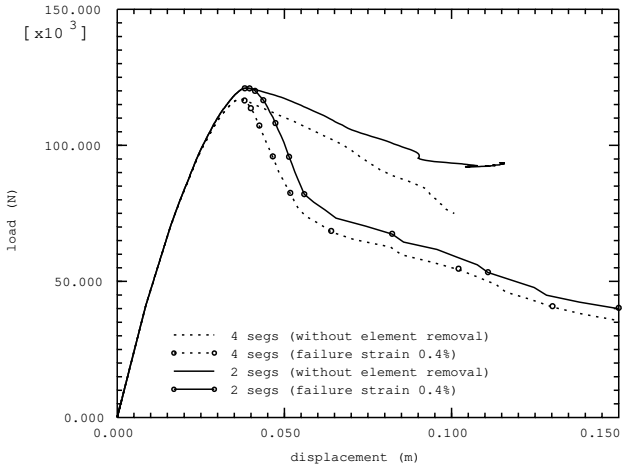


Figure 13: Element removal: tension softening study for plane stress fine mesh.

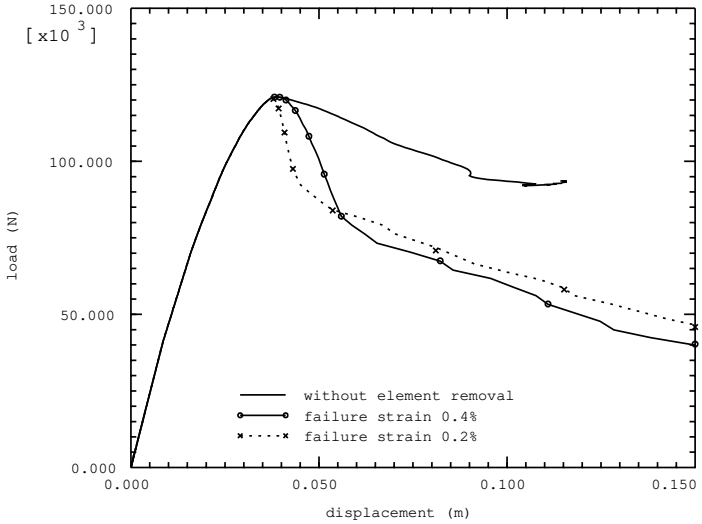


Figure 14: Element removal: plane stress fine mesh with a two-segment tension softening curve.

Slider mechanism with slip-rate-dependent friction

Products: Abaqus/Standard

Overview

This example verifies the slip-rate-dependent friction models in Abaqus for static and dynamic analysis. The example also illustrates specifying an allowable contact interference and changing friction properties.

Two slip-rate-dependent friction models are implemented. One is an extended form of the classical Coulomb friction model in which the friction coefficient can be defined in terms of slip rate, contact pressure, surface temperature, and field variables. In the second model the user provides a static friction coefficient, a kinetic friction coefficient, and a decay parameter. The static friction coefficient decays exponentially to the kinetic friction coefficient. This model is referred to as the exponential decay friction model.

Problem description

The model consists of a rod, a sliding cylinder, and a compound that is tightly fit between the rod and the cylinder. Both axisymmetric and three-dimensional models are created. *Figure 1* shows the axisymmetric model. A detail of the compound between the rod and the cylinder is shown in *Figure 2*. The inner radius of the rod is 19 mm (3/4 inch), and the outer radius is 25.4 mm (1 inch). The rod is 304.8 mm (12 inches) long and fixed at both ends. The inner radius of the sliding cylinder is 27 mm (1 1/16 inches), has a thickness of 12.7 mm (1/2 inch), and is 50.8 mm (2 inches) long. The initial thickness of the compound is larger than the 1.6 mm (1/16 inch) gap; the compound is confined between the rod and the cylinder.

Material

All parts of the model are elastic. The Young's modulus, Poisson's ratio, and density for the rod and the cylinder are 207 GPa (30.0×10^6 psi), 0.3, and 7800 kg/m^3 (0.73×10^{-3} lbf s^2/ in^4), respectively. The compound has a Young's modulus of 6.9 GPa (1.0×10^6 psi), a Poisson's ratio of 0.2, and a density of 1069 kg/m^3 (0.1×10^{-3} lbf s^2/ in^4).

It is assumed that the interface between the slider and the compound is rough; i.e., no slip can occur when contact is established. The rough surface interface is modeled with the Lagrange friction model and a high friction coefficient. It is assumed that the interface between the rod and the compound is polished and has a static friction coefficient $\mu_1 = 0.25$. Experimental tests show that the dynamic friction coefficient, μ_2, is 0.1 for a slip rate equal to 2.5 inches per second. Furthermore, the static coefficient decays exponentially to the kinetic friction coefficient, μ_∞, according to $\mu = \mu_\infty + (\mu_1 - \mu_\infty)\, e^{-d_c \dot{\gamma}_{eq}}$, where d_c is the decay coefficient. The dynamic coefficient at higher slip rates is not known; hence, the default Abaqus assumption that the ratio $\mu_2 - \mu_\infty$ to $\mu_1 - \mu_\infty$ is 5% is used. The idealized friction model is illustrated in *Figure 3* and is specified using test data to fit the exponential model for frictional behavior. Abaqus calculates the kinetic friction coefficient and the decay parameter. For the cases that use the Coulomb friction model, the data for the friction coefficient and the corresponding slip rate have been provided in tabular form.

Loading

The compound material is tightly fit between the rod and the slider in the first step of the analysis. The initial overclosure is resolved by specifying an allowable contact interference.

Friction is introduced in the second step by changing the friction properties. The contact interference allowance is removed. No loads are specified in this step to ensure that contact and equilibrium are established.

A harmonic sliding motion of the form $u(t) = A\cos(\omega t) - A$ is applied to the cylinder. The amplitude, A, is equal to 101.6 mm (4.0 inches), and the frequency, ω, is equal to $\pi/5$ rad/second. This form of harmonic motion is selected since it produces a zero velocity and avoids an intantaneous acceleration jump at the beginning of the dynamic step. A dynamic analysis is performed for 10 seconds to complete one full cycle of harmonic load in Step 3. Another harmonic cycle is completed using a static analysis in Step 4.

Results and discussion

The contact pressure distribution between the compound and the rod is nonuniform. This can be attributed to the deformation of the rod when the compound material is clamped between the rod and the cylinder. The Mises contour plot is shown in *Figure 4*.

Figure 5 shows the time history of the total normal force along the interface between the compound and the rod for the static step. The coarse main surface mesh is responsible for the oscillations in the curve. *Figure 6* shows the time history of the frictional shear forces that develop along this interface. The exponential form of the friction model is apparent as the slider completes one cycle of the harmonic motion. During this cycle the slip rate varies according to $\dot{\gamma}_{eq}(t) = -wA\sin(\omega t)$. The slider starts at the top. At 5 seconds the slider reaches the bottom, the velocity of the slider is zero, and the slider goes from “slip” to “stick.” It reverses its direction and slips again. At 10 seconds the slider is back at the top. This motion is repeated for the static analysis.

Input files

sliderslipfric_cax4_expon.inp	Axisymmetric model with the exponential decay friction model.
sliderslipfric_cax4_coulomb.inp	Axisymmetric model with the Coulomb friction model.
sliderslipfric_c3d8_expon.inp	Three-dimensional model with the exponential decay friction model.
sliderslipfric_c3d8_coulomb.inp	Three-dimensional model with the Coulomb friction model.

Figures

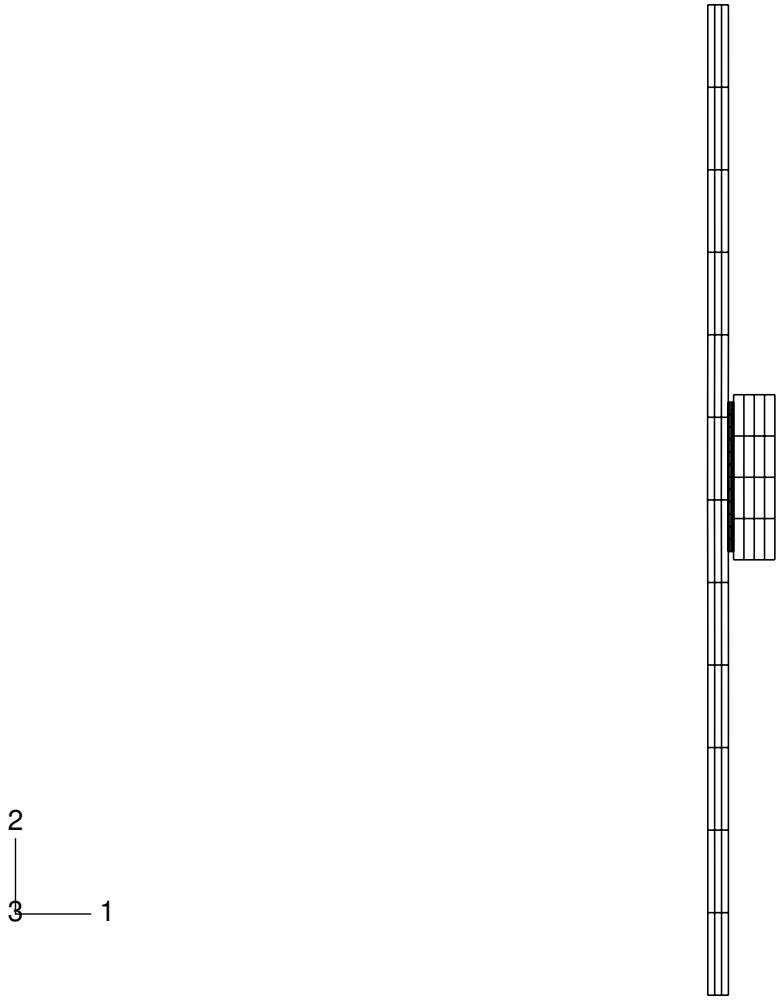


Figure 1: Axisymmetric model of the slider mechanism.

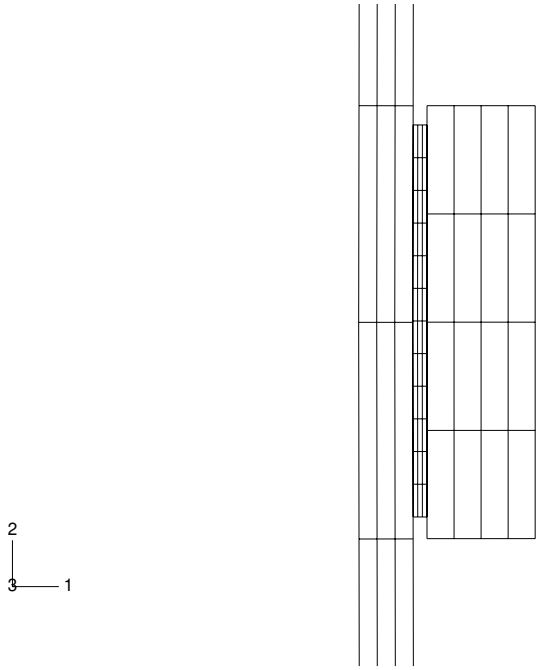


Figure 2: Detail of the compound between the rod and the sliding cylinder.

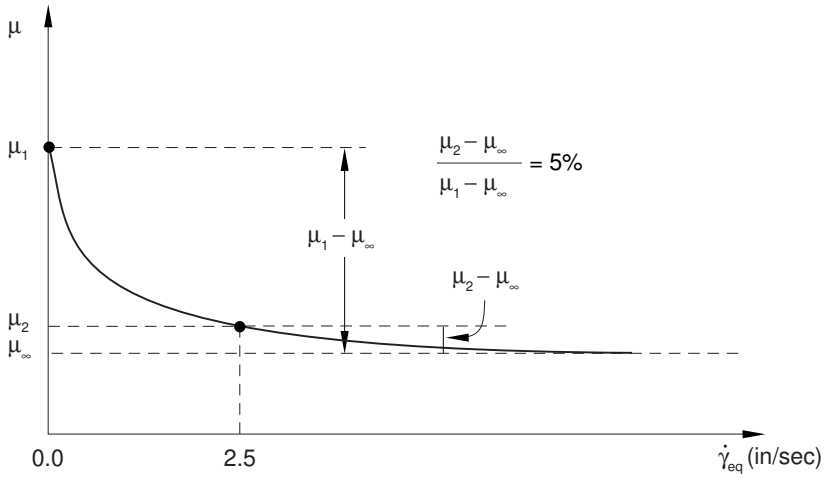


Figure 3: Idealized friction model for the rod–compound surface interface.

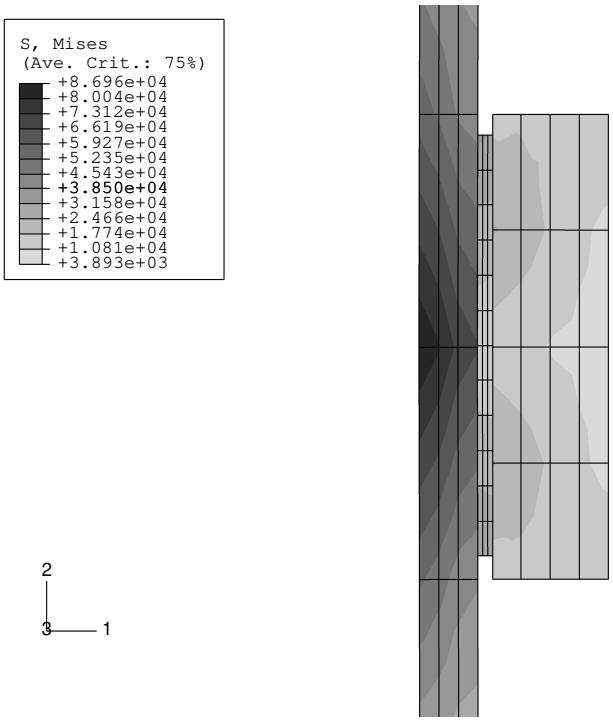


Figure 4: Mises stresses.

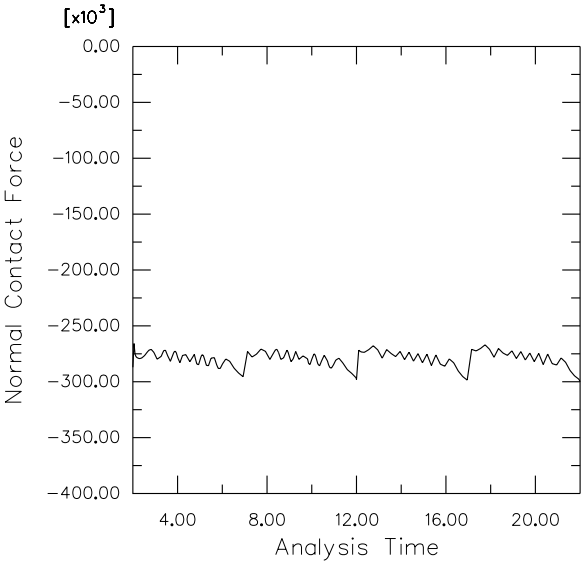


Figure 5: Normal contact forces across the rod–compound surface interface.

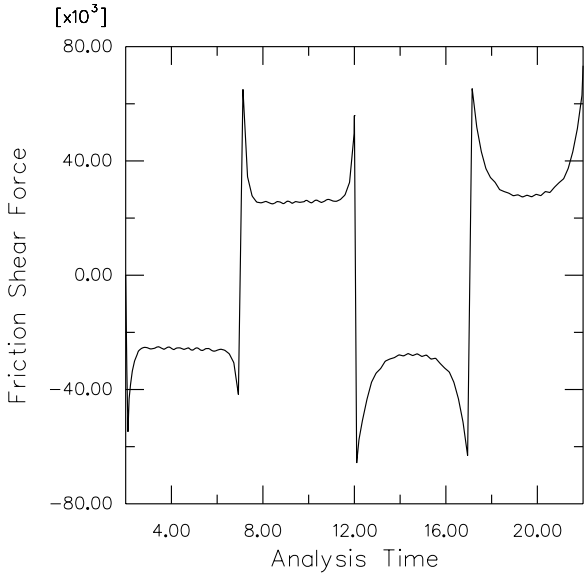


Figure 6: Shear forces across the rod–compound surface interface.

Cylinder under internal pressure

Products: Abaqus/Standard

Overview

This example verifies the finite-strain, elastic-plastic capability in Abaqus.

A cylinder made of elastic-plastic material is subjected to internal pressure under plane strain conditions, and a large change in the cylinder's inner radius (a factor of three) is prescribed. This problem is one of the best-known simple examples of elastic-plastic behavior and has been discussed extensively (see Prager and Hodge, 1951). Both axisymmetric and plane strain models are used to verify both of these kinematic formulations.

Problem description

The problem is illustrated in *Figure 1*. The cylinder is assumed to be stress-free, with an inside radius of 254 mm (10 in) and an outside radius of 508 mm (20 in). It is modeled both as an axisymmetric structure and in plane strain, as shown in the figure. Boundary conditions are symmetry about the z = constant faces (axisymmetric case) or symmetry about the θ = constant faces (plane strain case), the latter requiring the use of a local coordinate system to impose the appropriate conditions.

For each type of model, four meshes are used: ten regular 4-node elements (type CAX4, CPE4), ten hybrid 4-node elements (type CAX4H, CPE4H), five regular 8-node elements (type CAX8, CPE8), and five hybrid 8-node elements (type CAX8H, CPE8H). While no mesh convergence studies have been performed, the comparison of the numerical results with the analytic solution shows that, with an exception that is readily explained, all of these models give quite accurate results. The axisymmetric analysis with the corresponding CAXA elements are repeated for verification purposes.

The cylinder is assumed to be made of an elastic, perfectly plastic, Mises material, with the following properties:

Young's modulus	207 GPa (30×10^6 lb/in^2)
Poisson's ratio	0.3
Yield stress in pure tension	207 MPa (30×10^3 lb/in^2)

Loading

The cylinder is expanded by applying internal pressure. Following initial yielding, the cylinder reaches a limit state, after which the pressure decreases rapidly as the cylinder expands. This load-displacement behavior is unstable (softening) and, therefore, requires use of the modified Riks algorithm for solution under load control. Another approach, followed here, is to load the cylinder by prescribing the radial displacement at the innermost nodes. The pressure is then computed from the reaction forces conjugate to these prescribed radial displacements. (*Snap-through buckling analysis of circular arches* and *Snap-through of a shallow, cylindrical roof under a point load*, among others, illustrate the use of the modified Riks algorithm.)

The cylinder is expanded to three times its initial radius in a small number of increments. This requires very large strain increments and would probably be too large for a more complicated problem that involves shear and rotation as well as direct straining. However, large strain increments are suitable for this simple case.

Results and discussion

As the strains are so large, the results should compare very closely with the exact, rigid-plastic solution of Prager and Hodge (1951). That exact solution gives the stresses as follows:

$$\sigma_r/k = \ln\left[\left(\left(\frac{r_0}{a_0}\right)^2 + \left(\frac{a}{a_0}\right)^2 - 1\right) \Big/ \left(\left(\frac{b_0}{a_0}\right)^2 + \left(\frac{a}{a_0}\right)^2 - 1\right)\right],$$

$$\sigma_\theta/k = \sigma_r/k + 2,$$

$$\sigma_z/k = \sigma_r/k + 1,$$

where k is the yield stress in pure shear ($1/\sqrt{3}$ times the yield stress in pure tension); a_0 is the initial inside radius; b_0 is the initial outside radius; r_0 is the radius, in the initial configuration, of the material point at which the stresses are being calculated; and a is the current value of the inside radius. The form of the solution shows that we need only compare the radial stress, since the other stresses are obtained directly from that component.

The results for the axisymmetric element models are summarized in *Figure 2* and *Figure 3*, while *Figure 4* and *Figure 5* show the results for the plane strain models. *Figure 2* compares the pressure versus inside radius given by the CAX8H and CAX8 finite element models to that given by the exact, rigid-plastic solution. All of the axisymmetric models agree very closely with the exact solution with the exception of that using the fully integrated 8-node (CAX8) element. Recall that the solution is obtained by prescribing the motion of the inside surface of the cylinder, so the pressure is calculated for the finite element models from the reaction forces conjugate to these prescribed displacements.

Figure 3 compares the stress calculated by Abaqus at the point initially halfway through the cylinder wall ($r_0/a_0 = 1.5$) to the exact, rigid-plastic solution. Again, with the exception of the CAX8 element model, there is excellent agreement with the analytical solution.

Figure 4 and *Figure 5* show similar results to *Figure 2* and *Figure 3* for the plane strain models. Again, with the exception of the fully integrated 8-node (CPE8) element model, all of the plane strain models show excellent agreement with the exact, rigid-plastic solution.

The pure displacement 8-node elements (CAX8 and CPE8) give poor results because the strains are calculated directly from the interpolation functions at each integration point, and the incompressibility requirement causes a severe oscillation in the mean pressure stress throughout each element. However, in the hybrid, 8-node elements the mean pressure stress is interpolated independently, so an accurate value is obtained for this variable. In addition, the 4-node elements in Abaqus are constant strain/stress elements for this case (because these elements are coded with a constant hoop strain value and use "selective reduced integration," in which the volume strain is computed at the centroid only), and so also provide accurate pressure stress values.

Results for models using the fully integrated versions of axisymmetric and plane strain elements are shown here to caution the user. With rare exceptions the fully integrated 8-node quadrilaterals are not as effective as the reduced integration versions of the same elements; the reduced integration 8-node quadrilaterals are, hence, almost always recommended over their fully integrated counterparts. This particular problem gives a dramatic illustration of a difficulty encountered with full integration in a problem in which the bulk behavior of the material is very much stiffer than the shear behavior, a type of behavior commonly encountered.

Input files

cylinderunderpress_cax4.inp	CAX4 element model.
cylinderunderpress_cax4h.inp	CAX4H element model.
cylinderunderpress_cax4i.inp	CAX4I element model.
cylinderunderpress_cax4ih.inp	CAX4IH element model.

cylinderunderpress_cax8.inp	CAX8 element model.
cylinderunderpress_cax8h.inp	CAX8H element model.
cylinderunderpress_caxa41.inp	CAXA41 element model.
cylinderunderpress_caxa4h1.inp	CAXA4H1 element model.
cylinderunderpress_caxa81.inp	CAXA81 element model.
cylinderunderpress_caxa8h1.inp	CAXA8H1 element model.
cylinderunderpress_cpe4.inp	CPE4 element model.
cylinderunderpress_cpe4h.inp	CPE4H element model.
cylinderunderpress_cpe4i.inp	CPE4I element model.
cylinderunderpress_cpe4ih.inp	CPE4IH element model.
cylinderunderpress_cpe8.inp	CPE8 element model.
cylinderunderpress_cpe8h.inp	CPE8H element model.

References

- Prager, W., and P. G. Hodge, Theory of Perfectly Plastic Solids, John Wiley and Sons, New York, 1951.

Figures

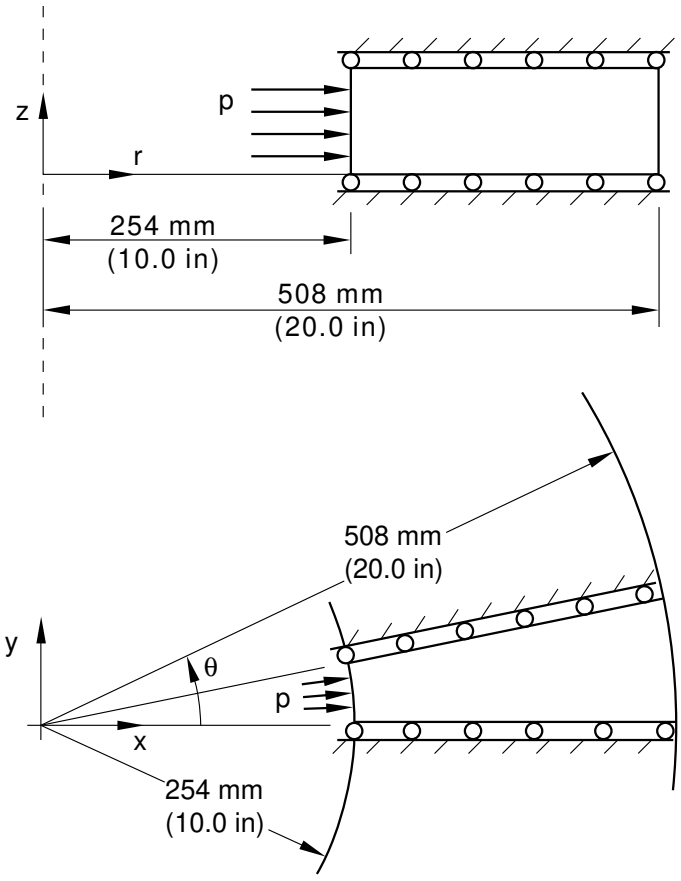


Figure 1: Thick cylinder under internal pressure.

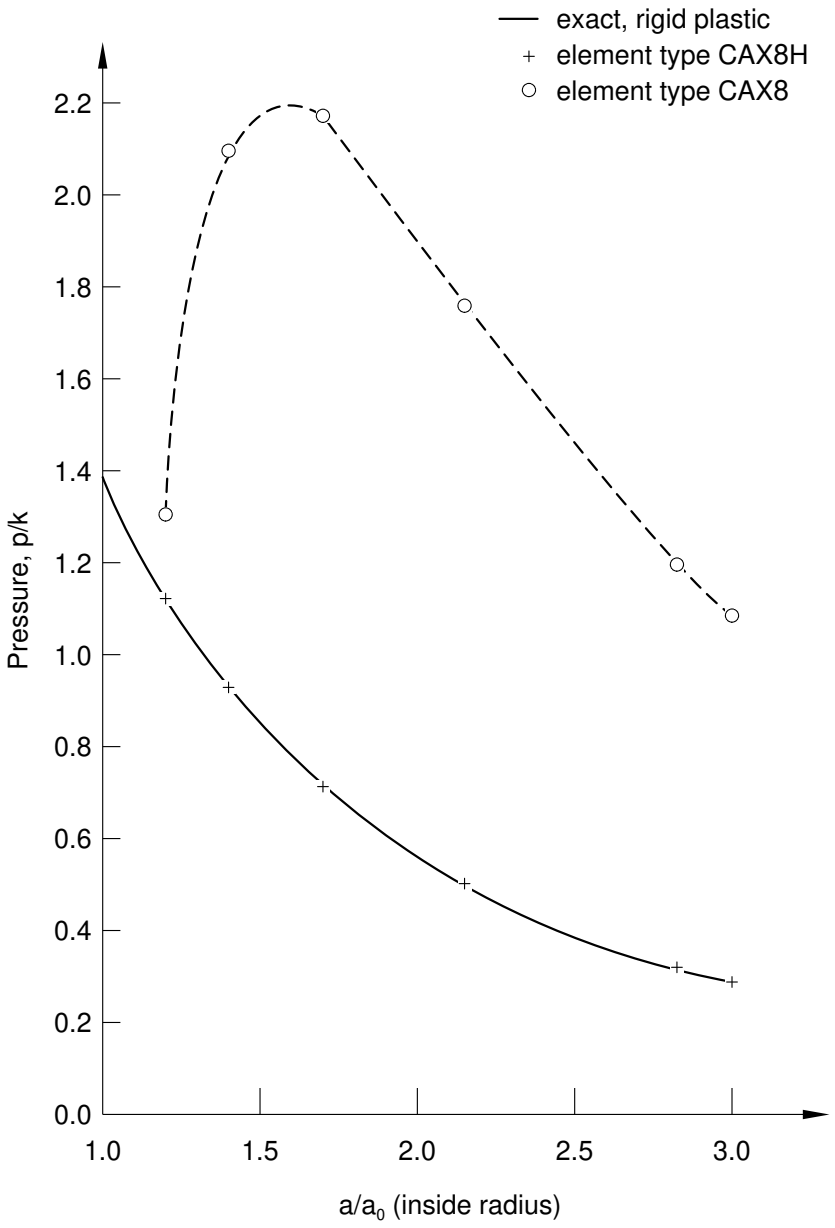


Figure 2: Internal pressure versus inside radius, axisymmetric models.

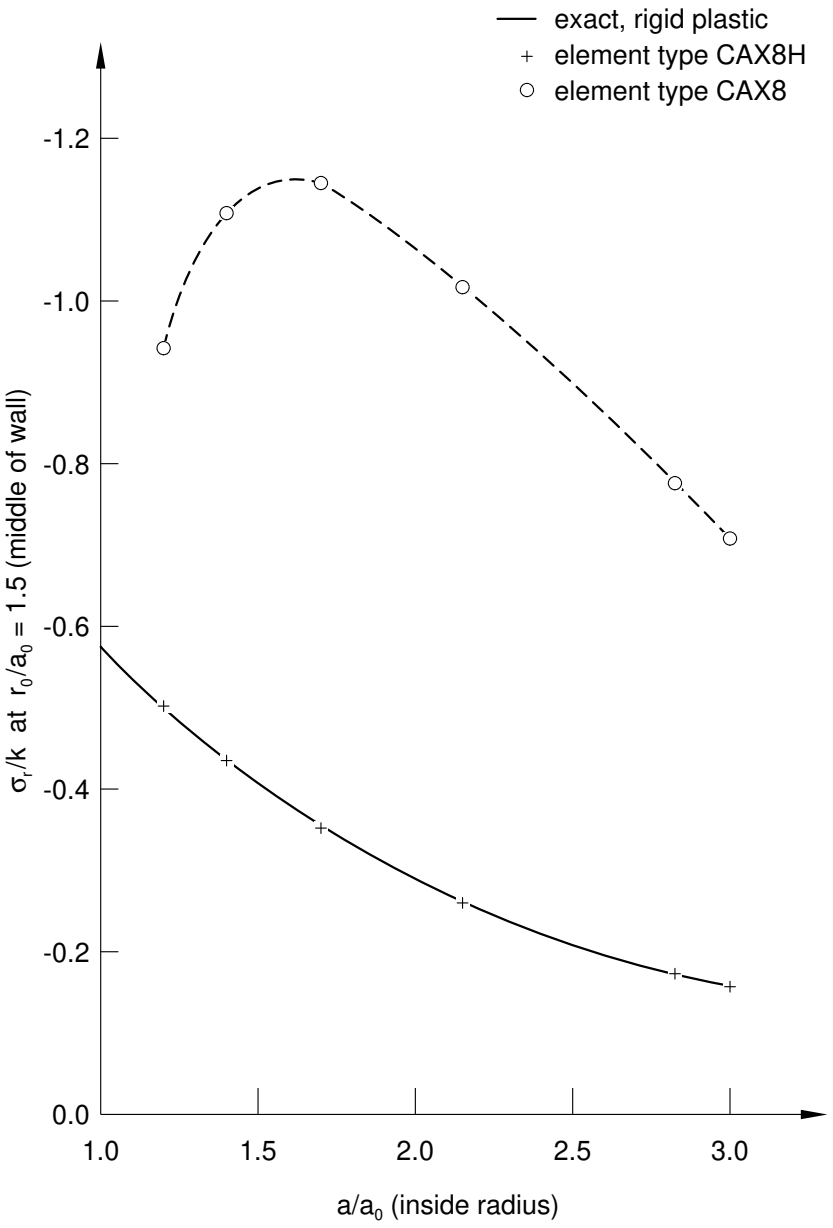


Figure 3: Stress at $r_0/a_0 = 1.5$ versus inside radius, axisymmetric models.

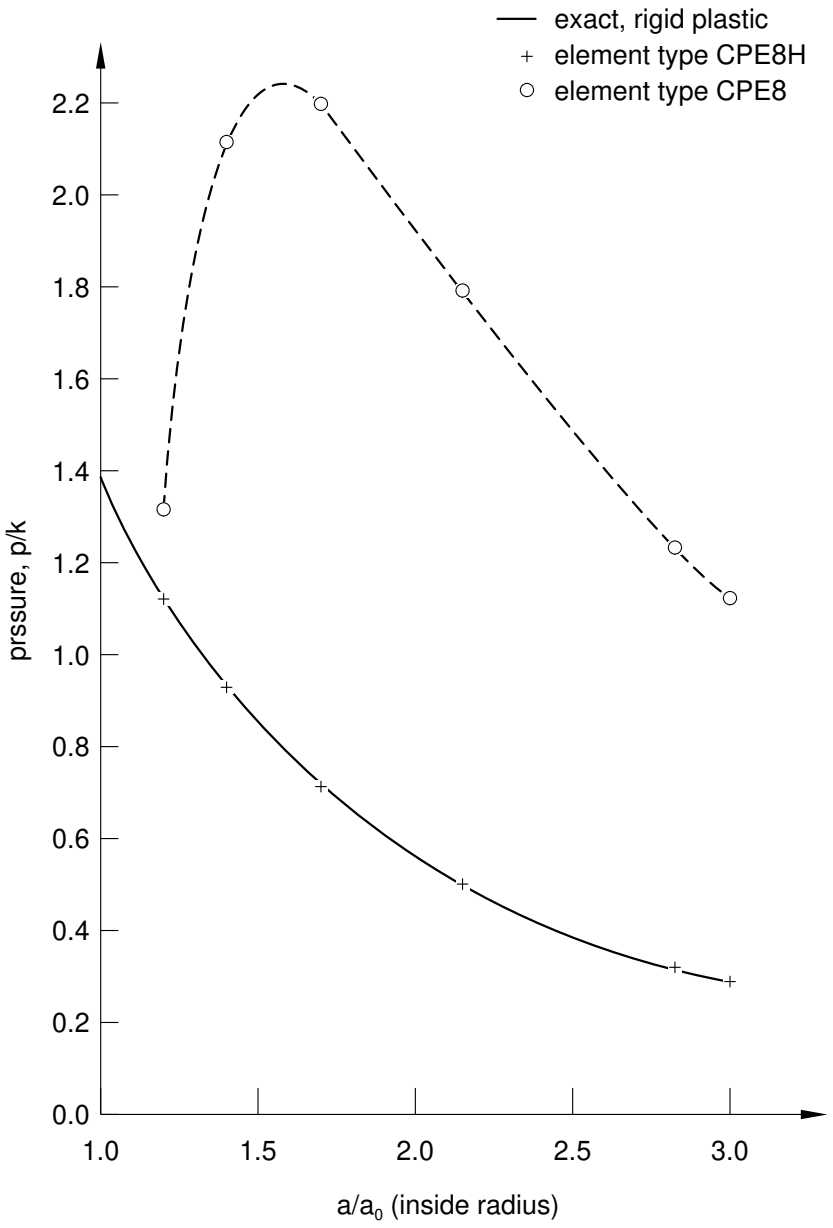


Figure 4: Internal pressure versus inside radius, plane strain models.

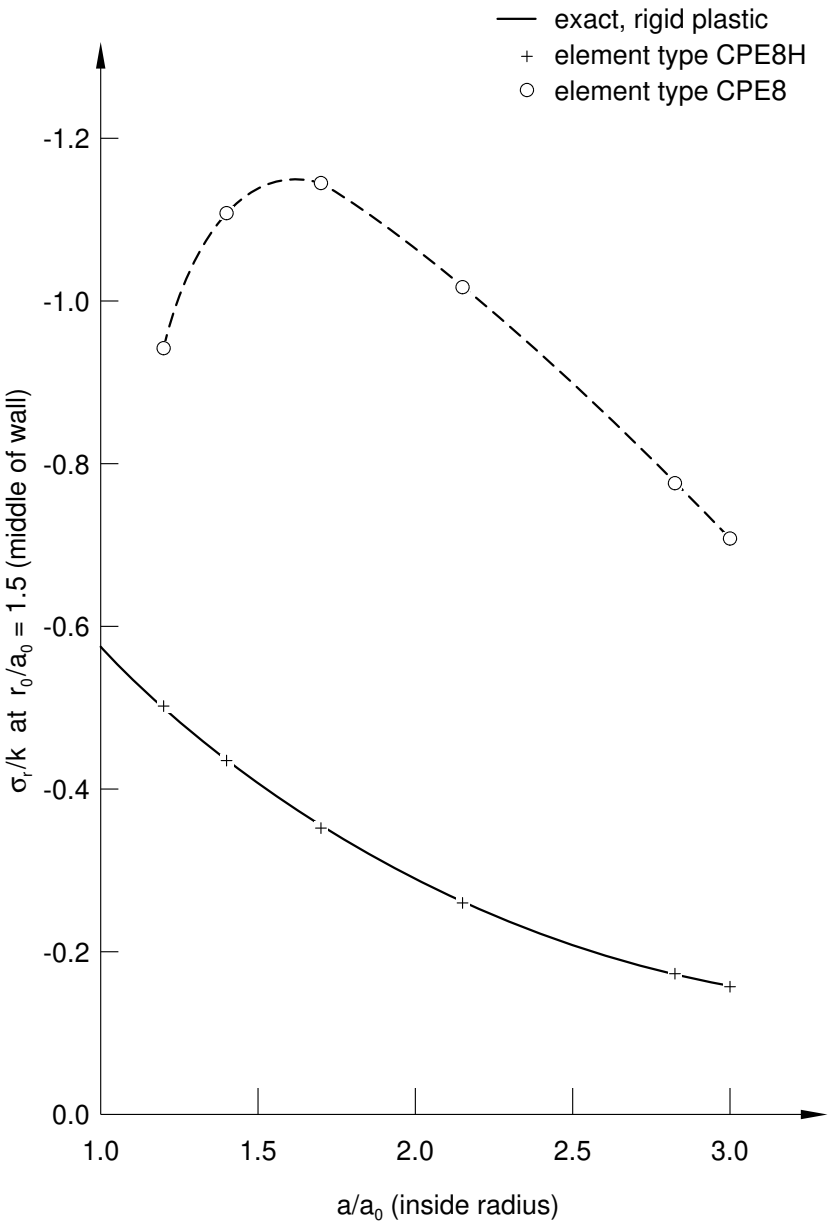


Figure 5: Stress at $r_0/a_0 = 1.5$ versus inside radius, plane strain models.

Creep of a thick cylinder under internal pressure

Products: Abaqus/Standard

Overview

This example verifies the Abaqus capability for high-temperature creep analysis.
An exact solution is available for the steady-state part of the response.

Problem description

The problem is shown in *Figure 1*. The structure is a cylinder, with an inside radius of 25.4 mm (1 in) and an outside radius of 50.8 mm (2 in). The cylinder is assumed to be under plane strain conditions (the axial strain is zero), and the solution is one-dimensional (independent of axial position) and axisymmetric. Therefore, a single row of axisymmetric elements is sufficient. Five equal-sized CAX8R elements are used. No mesh convergence studies have been done, but a comparison with the exact elasticity and steady-state solutions shows that this discretization provides accurate stress predictions.

The material is assumed to be isotropic elastic, with Young's modulus of 138 GPa (20×10^6 lb/in^2) and Poisson's ratio of 0.3, with a Mises creep potential and uniaxial creep behavior defined by

$$\dot{\varepsilon}^{cr} = A\sigma^n,$$

where A is 1.7828×10^{-17} per sec (stress in MPa) (10^{-24} per hour with stress in lb/in^2) and $n = 5$. These values are typical of structural steel at a fairly high temperature.

Loading and control

The cylinder is subjected to a rapidly applied internal pressure of 60 MPa (8700 lb/in^2) that is held constant for a long period of time, so that the steady-state creep conditions are reached.

The initial application of the pressure is assumed to occur so quickly that it involves purely elastic response, which is obtained by using the static procedure. The creep response is then developed in a second step, using the quasi-static procedure. A response of 180,000 seconds (50 hours) is requested, which is sufficient to reach steady-state conditions. During the quasi-static step a tolerance is required to control the time increment choice and, hence, the accuracy of the transient creep solution. In this case we assume that moderate accuracy is required. Errors in stress of about 0.7 MPa (100 lb/in^2) will make a small difference to the creep strain added within an increment. Converting this stress error to a strain error by dividing it by the elastic modulus gives a tolerance of 5×10^{-6}. Higher accuracy in the integration of the creep constitutive model can be obtained by reducing this tolerance, at the expense of using more time increments. Alternately, using a large tolerance value will allow Abaqus to use the largest possible time increments, so that low accuracy will result during the transient, but the steady-state solution will be reached at minimum cost. Thus, if the steady-state solution is the only part of the solution of interest, it is effective to set the tolerance to a large number.

With the tolerance specified in the quasi-static procedure, Abaqus uses automatic time incrementation. The scheme is rather simple and aims at increasing the time increments gradually as the solution progresses toward steady state. In a small-displacement case such as this, explicit integration of the creep constitutive model is usually efficient because the method is inexpensive per time increment (since no new stiffness matrix needs to be formed and solved), and its stability limit is usually quite large compared to times of interest in the solution. The automatic time stepping scheme includes an internal calculation of the stability limit, and the time increment is controlled to remain within this limit. If this is too restrictive—if it results in a sequence of time increments that are all much smaller than the remaining part of the time period requested on the data line associated with the quasi-static procedure (10 successive increments where the time increment is less than 2% of the remaining

time period)—Abaqus automatically switches to an implicit time integration scheme that is unconditionally stable. The only limit at all on the time increment selection is then accuracy as specified by the tolerance. This switch to implicit integration can be suppressed by the user in the quasi-static procedure. In this example the switch to implicit integration occurs at increment 44, after 27,108 seconds (7.53 hours) of creep. This allows Abaqus to choose large time increments (up to 39,600 seconds, or 11 hours) toward the end of the solution.

Results and discussion

The steady-state solution to this problem is given by Odquist and Hult (1962). The steady-state stresses are the radial stress

$$\sigma_r = -p\left(\frac{r^{-2/n}-b^{-2/n}}{a^{-2/n}-b^{-2/n}}\right),$$

the circumferential stress

$$\sigma_\theta = p\left(\frac{-(1-2/n)r^{-2/n}+b^{-2/n}}{a^{-2/n}-b^{-2/n}}\right),$$

and the axial stress

$$\sigma_z = p\left(\frac{-(1-1/n)r^{-2/n}+b^{-2/n}}{a^{-2/n}-b^{-2/n}}\right),$$

where p is the pressure, r is the radial position of the point at which the stresses are given, a is the inside radius of the cylinder, b is the outside radius, and n is the exponent in the uniaxial creep law.

Figure 2 to *Figure 4* show the computed results compared to this exact solution, as well as to the initial elastic solution (which is available in standard textbooks, such as Timoshenko and Goodier, 1951). The plots show that the computed stresses agree closely with these solutions.

Figure 5 shows a time history plot of the hoop stress at the inside and outside radius (obtained as nodal stress values) and illustrates the way the solution evolves from the initial elastic response to the steady-state creep response.

Input files

creepthickcylinder_cax8r.inp	Example using CAX8R elements.
creepthickcylinder_cax4i.inp	Example using CAX4I elements.

References

- Odquist, F. K. G., and J. Hult, Kriechfestigkeit Metalischer Werkstoffe, Springer-Verlag, Berlin, 1962.
- Timoshenko, S., and J. N. Goodier, Theory of Elasticity, McGraw-Hill, New York, 1951.

Figures

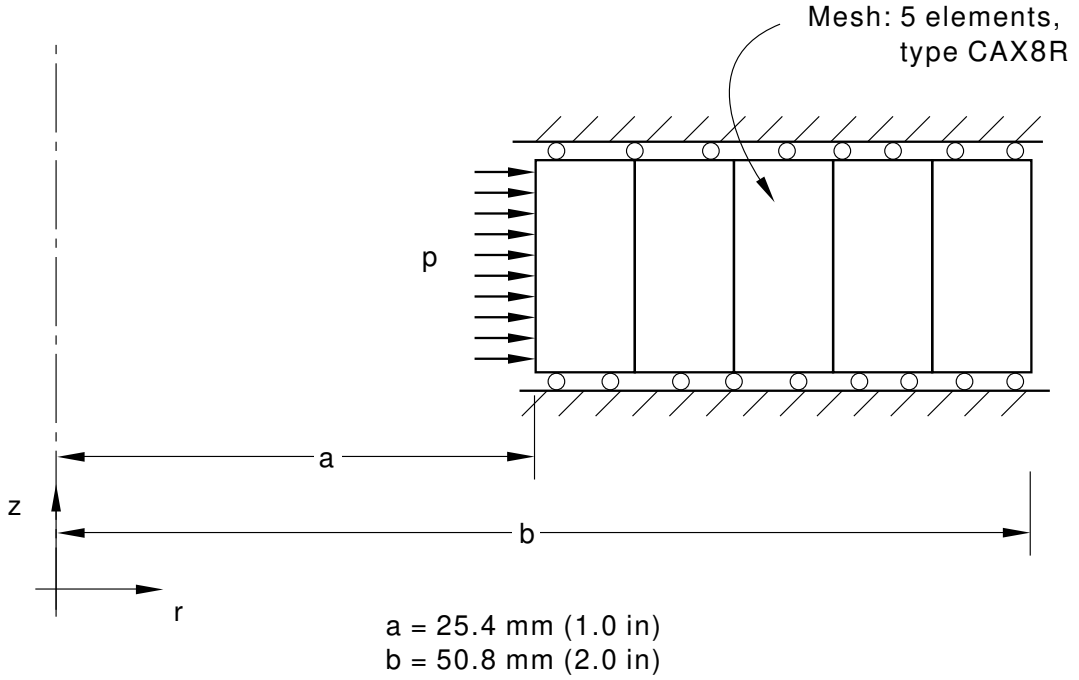


Figure 1: Thick cylinder creep example.

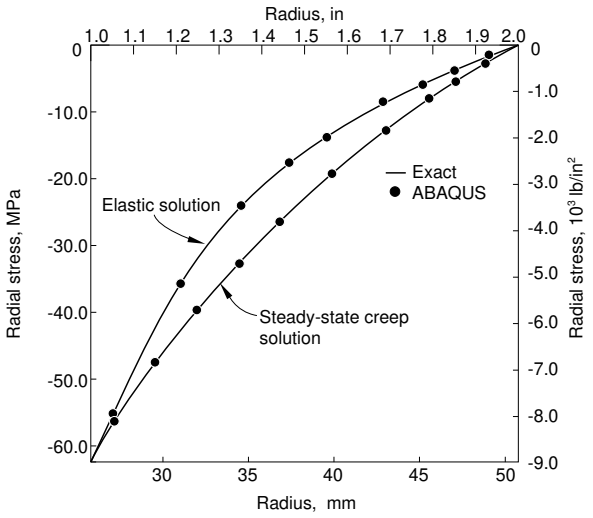


Figure 2: Radial stress versus radial position.

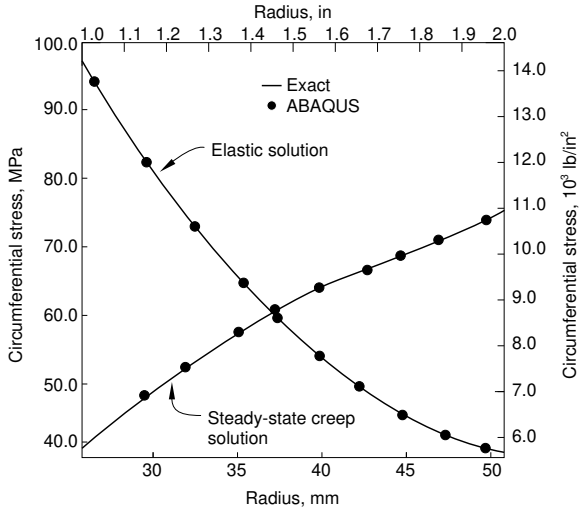


Figure 3: Circumferential stress versus radial position.

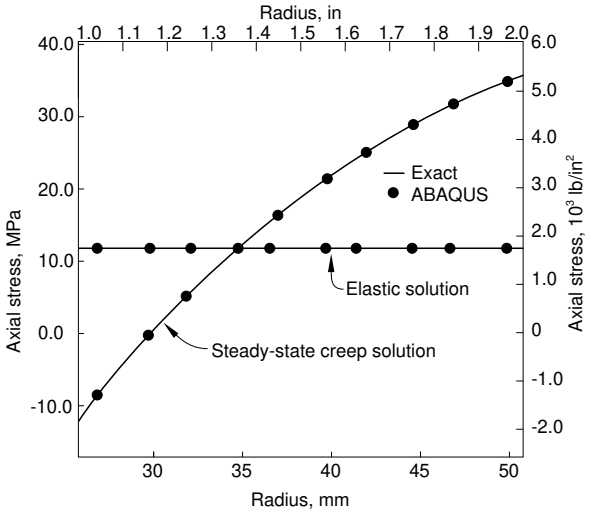


Figure 4: Axial stress versus radial position.

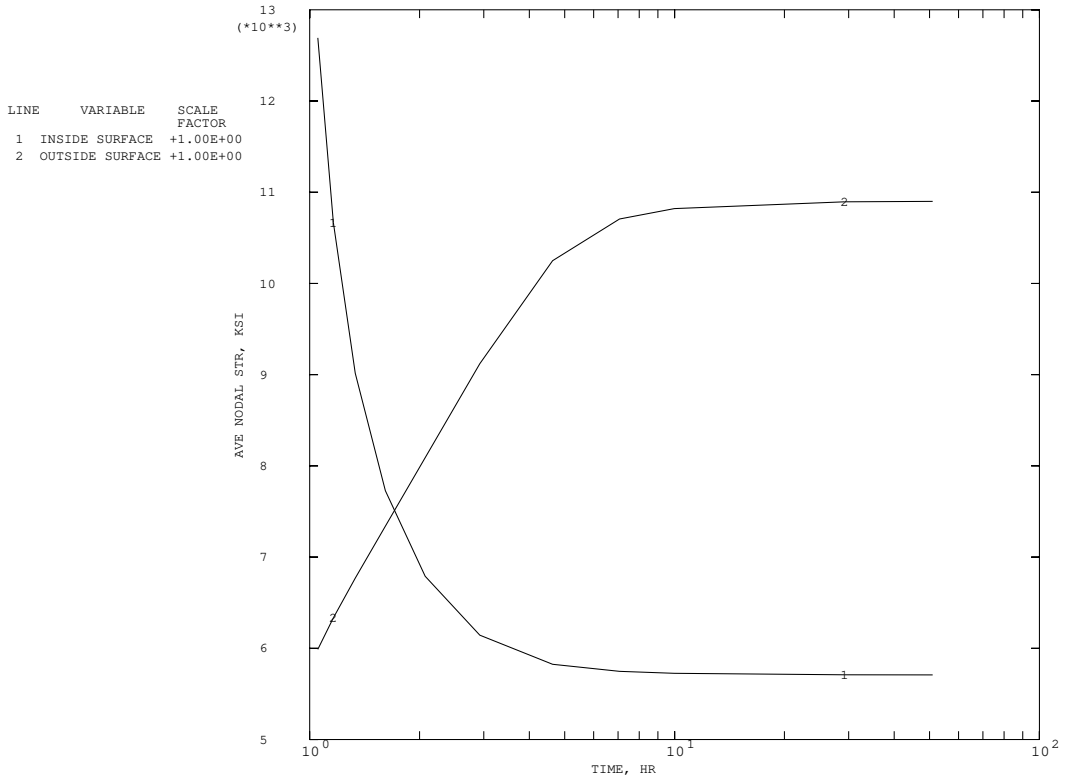


Figure 5: Hoop stress histories at inside and outside surfaces.

Pressurization of a thick-walled cylinder

Products: Abaqus/Explicit

Overview

This example verifies the use an axisymmetric model, a plane strain model, and a three-dimensional model for a thick-walled cylinder that is loaded with an internal pressure beyond the limit load for the cylinder.
The quasi-static solution for this problem is given in Nagtegaal and De Jong (1981).

Problem description

The cylinder has an inner radius of 1.0 and outer radius of 2.0. The material characterization for the cylinder is assumed to be elastic-plastic with constant isotropic hardening. The material properties are Young's modulus = 1000., Poisson's ratio = 0.3, yield stress = 1., hardening slope = 3., and density = .001.

The meshes used in the analysis are shown in *Figure 1*. In each case there are 20 elements through the thickness of the cylinder.

The loading is in the form of displacement control of the nodes on the inner radius of the cylinder. The displacements are of sufficient magnitude that the cylinder becomes fully plastic and exceeds the limit load for the structure. It is not possible to capture the pressure versus displacement curve using pressure loading because of the instability of the structure under the load. Using displacement control, the pressures can be recovered at each (prescribed) displacement point of the inner radius of the structure. To mitigate the dynamic effects in this problem (they cannot be eliminated entirely) the radial velocity of the inner radius nodes is specified as a linear ramp from a velocity of zero to a velocity of 5 over the 0.2 second duration of the steps. The time period of the loading is much longer than the period of any natural mode of vibration of the structure, and the peak velocity is two orders of magnitude lower than the wave speed of the material.

Results and discussion

Abaqus/Explicit models this problem as a transient dynamic analysis. *Figure 2* shows the pressure versus radial displacement curves for the three element types used in the analysis. The pressure is inferred from the σ_{xx}-component of stress in the first element in each of the meshes. All three cases show some dynamic effects as the cross-section becomes fully plastic. *Figure 3* shows the energy balance for this analysis.

Input files

prcyl.inp	Input data used for this analysis.

References

- Nagtegaal, J. C., and J. E. De Jong, “Some Computational Aspects of Elastic-Plastic Large Strain Analysis,” *International Journal of Numerical Methods in Engineering*, vol. 17, pp. 15–41, 1981.

Figures

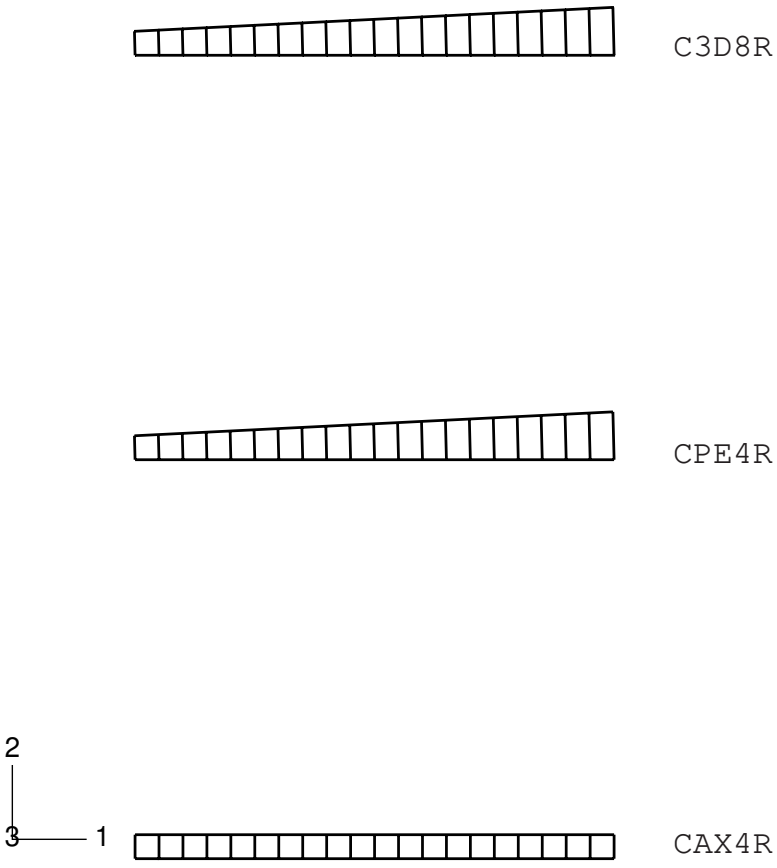


Figure 1: Meshes for pressurized cylinder problem.

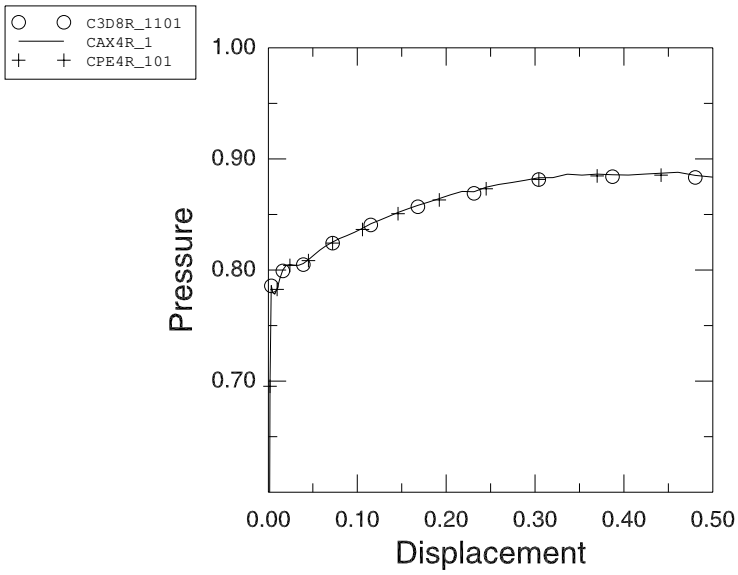


Figure 2: Displacement of inner radius versus pressure.

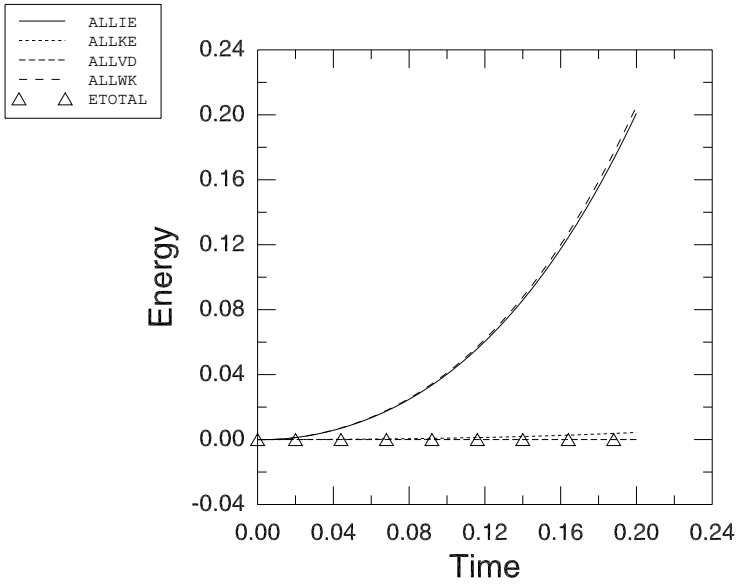


Figure 3: Energy balance as a function of time.

Stretching of a plate with a hole

Products: Abaqus/Standard Abaqus/Explicit

Overview

This problem verifies the anisotropic plasticity model in Abaqus/Explicit and the transfer of material properties into Abaqus/Standard using the results transfer capability.

Problem description

In the first case the entire process is analyzed in Abaqus/Explicit as a quasi-static analysis for a total time period of $t = 1.0$. In the second case part of the analysis is conducted in Abaqus/Explicit and the remainder of the analysis is conducted in Abaqus/Standard.

A square 30×30 plate containing a hole of radius 4 is stretched in the y-direction, while displacements in the x-direction are restrained along its outer perimeter. *Figure 1* shows the initial quarter symmetry CPS4R meshes with enhanced hourglass control used in this analysis. There are three identical meshes shown with three plasticity cases: isotropic Mises plasticity, anisotropic plasticity with a ratio of yield stresses of 3:2, and anisotropic plasticity with a ratio of yield stresses of 2:3. The material orthotropic axes are taken as the coordinate basis. Only yield stresses in the x-direction are altered. Other components of the yield stress are taken to be the same as the reference yield stress that is specified for the isotropic Mises plasticity model.

The elastic material properties of the plate are a Young's modulus of 1×10^9 and a Poisson's ratio of 0.3. The density is 2500.

The isotropic Mises plasticity specification uses constant isotropic hardening with an initial yield of 1×10^6 and a hardening modulus of 4×10^5. Anisotropic plasticity is used with two of the meshes to define a ratio of yield stress in each of the two in-plane directions. For the anisotropic cases the reference yield and hardening is defined to be the same as for the isotropic Mises case. It can be verified that the choice of the ratios does not violate the requirement that Hill's yield surface be convex in the deviatoric plane.

In the analysis that is performed entirely in Abaqus/Explicit, the plate is stretched by ramping the velocity at the top nodes to 5 for the first half of the step time and then keeping a constant velocity of 5 at these nodes for the rest of the analysis. The results of the explicit analysis obtained for the first half of the step time are also imported into Abaqus/Standard using the import capability. The stretching of the plate is continued in Abaqus/Standard by prescribing a displacement of 2.5 at the top nodes; this displacement corresponds to the velocity boundary conditions in Abaqus/Explicit.

The import capability allows for the analysis to be continued with or without updating the reference configuration to be the imported configuration. When the reference configuration is updated, the deformed model with its material state at the end of the Abaqus/Explicit analysis is imported into Abaqus/Standard. The deformed configuration is used as the reference configuration in the import analysis. When the reference configuration is not updated, the deformed model with its material state, displacements, and strains at the end of the Abaqus/Explicit analysis is imported into Abaqus/Standard. The original configuration is used as the reference configuration for the import analysis. In the import analyses both cases are used. To ensure the final deformed configurations are similar in both cases, a displacement of 3.75 is prescribed at the top nodes for the case when the reference configuration is not updated, as opposed to a displacement of 2.5 for the case when the reference configuration is updated.

Results and discussion

The contours of the equivalent plastic strain in each of the plates, obtained from the analysis performed exclusively in Abaqus/Explicit, are shown in *Figure 2*. Inspection of the deformed shapes and regions of high plastic strain show that the anisotropic plasticity has a large effect on the manner in which the hole enlarges, or rather, the necking of the ligament. In the case of a 3:2 ratio the high flow stress in the x-direction inhibits the straining in the direction across the ligament. There is not much difference in the deformed shape and the plastic strain distribution between the first and second cases. When the yield stress ratio is changed to 2:3 (yield stress in the x-direction is two-thirds of that in the y-direction), it is easier for the material in the ligament to flow in the x-direction. Therefore, the plate necks faster than in the other cases. The smaller neck in the ligament would subsequently render less resistance to material elongation in the y-direction. Inclined plastic shear bands start to develop in all three cases, with the last case being the most severe.

Contours of equivalent plastic strain at the end of the import analysis are shown in *Figure 3* (no configuration update) and *Figure 4* (updated configuration). The equivalent plastic strains obtained when the analysis is performed exclusively in Abaqus/Explicit and those obtained in the import analyses show differences in the region near the hole where maximum straining occurs. The differences in the results are due to the differences in the computation of thickness in plane stress conditions in the two analyses. In Abaqus/Standard the thickness is computed based on the assumption that the volume of the element remains the same during the analysis, whereas in Abaqus/Explicit thickness is computed by considering the strains in the out-of-plane direction. If the Poisson's ratio is allowed to approach 0.5, the results from the two analyses agree well.

This problem tests the features listed but does not provide independent verification of them.

Input files

hole.inp	Input data used in this analysis.
xs_s_hole.inp	Input data for the import analysis when UPDATE=NO.
xs_s_hole1.inp	Input data for the import analysis when UPDATE=YES.

Figures

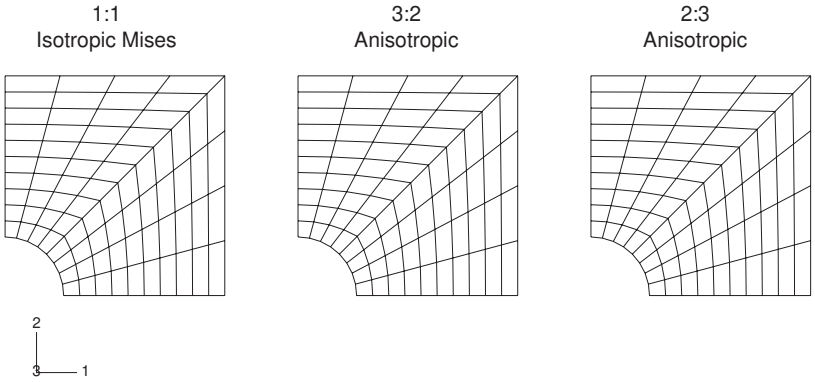


Figure 1: Original meshes for stretching of perforated plates.

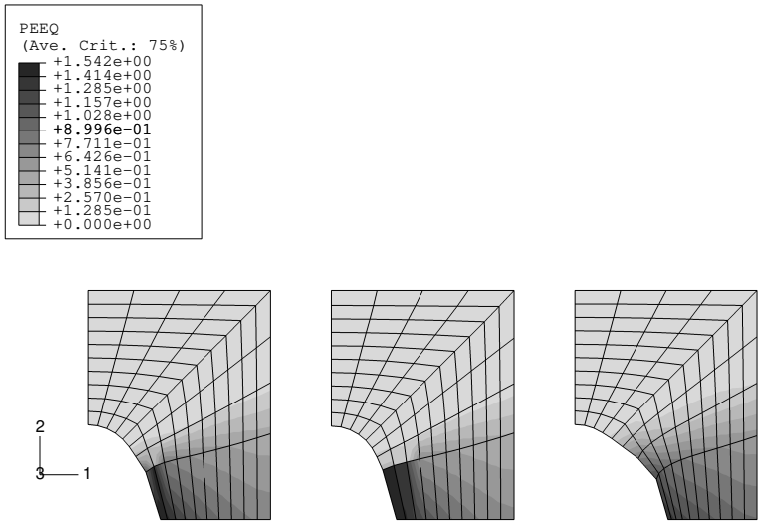


Figure 2: Contours of equivalent plastic strain.

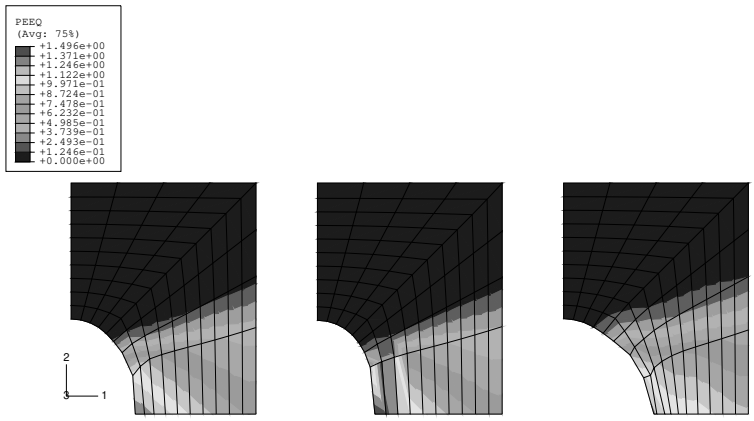


Figure 3: Contours of equivalent plastic strain (reference configuration not updated).

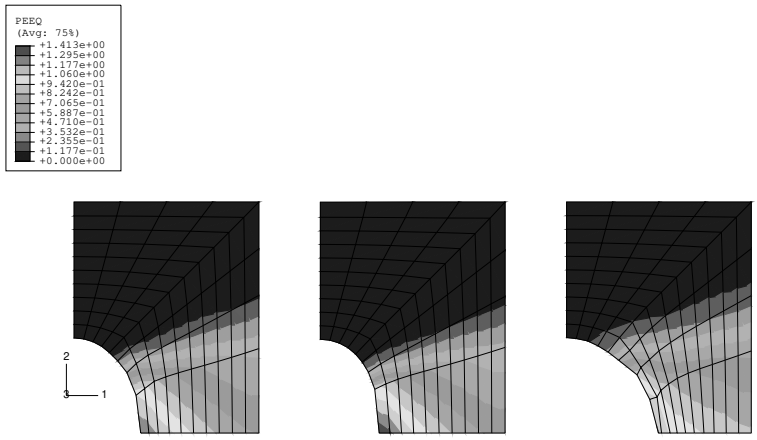


Figure 4: Contours of equivalent plastic strain (reference configuration updated).

Pressure on infinite geostatic medium

Products: Abaqus/Explicit

Overview

This example simulates a semi-infinite granular medium under initial geostatic stress, subject to pressure suddenly applied to part of its surface.

Problem description

To model a semi-infinite half-space, one option is to generate a mesh that extends far away from the region of interest so that there are no reflections from the farthest boundaries of the model back into the region of interest. However, this is computationally expensive, because the solution must be computed in a large part of the model in which the user has no interest. Infinite elements allow the region of interest (the interior) to be modeled with a suitable mesh, while the far field is simulated with a set of infinite elements that are added to the perimeter of the interior mesh.

In each case considered (axisymmetric, plane strain, and three-dimensional), two meshes are used: (1) a small mesh defining the interior, surrounded by infinite elements, and (2) a larger model of ordinary finite elements, extended to a sufficient distance so that no waves are reflected back into the interior during the time of analysis. The purpose of having these two meshes is to verify the infinite elements. The larger model has exactly the same discretization in the interior region as the smaller mesh. If the infinite elements are performing properly, the solution should be nearly identical in the interior portion of both meshes.

The initial geostatic stress field is defined using initial conditions. One of the features of the infinite elements is that they will apply the proper tractions on the boundary to maintain an initial equilibrium stress field. The first step in this problem is of a duration of 5 milliseconds. Only the gravitational (self-weight) load corresponding to the geostatic field is applied. There should be no accelerations and no changes in the stresses during this step. The step is carried out to verify that the infinite elements do in fact maintain an equilibrium state of stress.

In the second step of the analysis a pressure is applied instantaneously over a portion of the top surface and held constant through the 8 milliseconds of response.

The granular material is simulated with the extended Drucker-Prager model. The frictional angle is 40°, while the material is nondilatational (the dilation angle is 0°). The yield surface in the deviatoric plane is assumed to be noncircular, with the parameter K, which defines the dependency on the third stress invariant, being 0.9. Perfect plasticity is assumed, with a yield stress in uniaxial compression of 5×10^3. Young's modulus is 1×10^9, and Poisson's ratio is 0.3.

Axisymmetric, plane strain, and three-dimensional models with the corresponding infinite elements are studied. The meshes are shown in *Figure 1* and *Figure 2*. In the plane strain and three-dimensional cases the model assumes symmetry about a center plane. The three-dimensional model has one layer of elements, with the displacement in the x-direction constrained to give plane strain response.

Results and discussion

Figure 3 through *Figure 5* show contours of pressure at the end of the first step. The pressure has a linear variation, corresponding to the equilibrium geostatic stress field. Immediately after the pressure pulse is applied at the beginning of the second step, elastic waves begin to traverse the medium. These are followed by plastic waves once the stress magnitude exceeds the yield strength. The deviatoric yield stress depends linearly on the magnitude of the pressure and, hence, is augmented in the region of high confinement.

Figure 6 shows the hydrostatic pressure in the axisymmetric case at the end of the second step. The two contour plots are very similar in the region under loading. Likewise, the plastic strain contours in *Figure 7* are almost identical. In this case the waves are spherical.

Figure 8 shows the hydrostatic pressure in the plane strain case at the end of the second step, and *Figure 9* shows the equivalent plastic strain. Almost identical patterns are again observed. Similar plots of pressure and plastic strain at the end of the second step for the three-dimensional model are shown in *Figure 10* and *Figure 11*. In these plane strain cases the waves are planar.

Figure 12 shows the pressure stress time history for elements 81 and 1361 in the plane strain case. The position of these elements is shown in *Figure 13*. These elements are at the same position in the small and large models. The results show the effect that the infinite element has in removing wave reflections from the boundaries. The dilatational wave speed for the material is 1160. In the mesh without infinite elements, the wave is expected to return to element 1361 0.033 sec after the pressure is applied, or at a total time of 0.041 sec. *Figure 12* shows the wave returning at this time. The mesh with infinite elements shows no wave as significant as this wave returning to element 81.

Input files

geostat_pe.inp	Input data for the plane strain model used in this analysis.
geostat_ax.inp	Input data for the axisymmetric case.
geostat_3d.inp	Input data for the three-dimensional case.

Figures

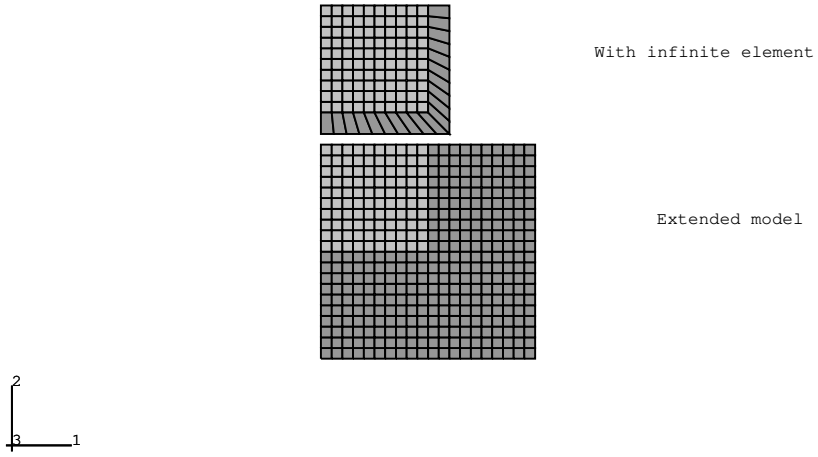


Figure 1: Undeformed axisymmetric and plane strain meshes.

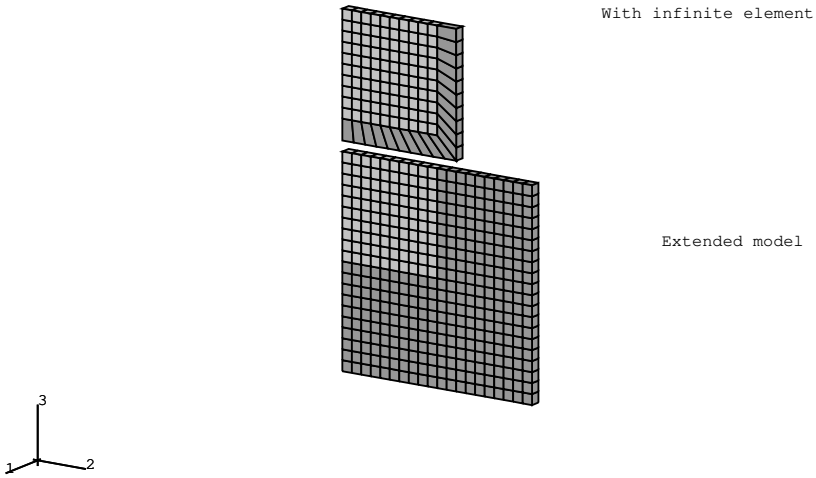


Figure 2: Undeformed three-dimensional mesh.

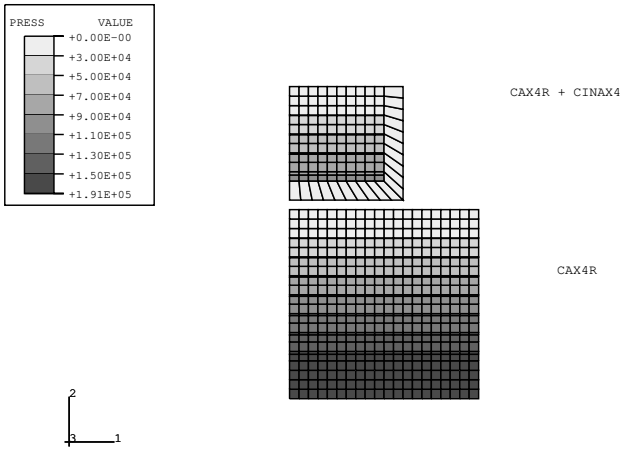


Figure 3: Pressure contours, axisymmetric case, end of Step 1.

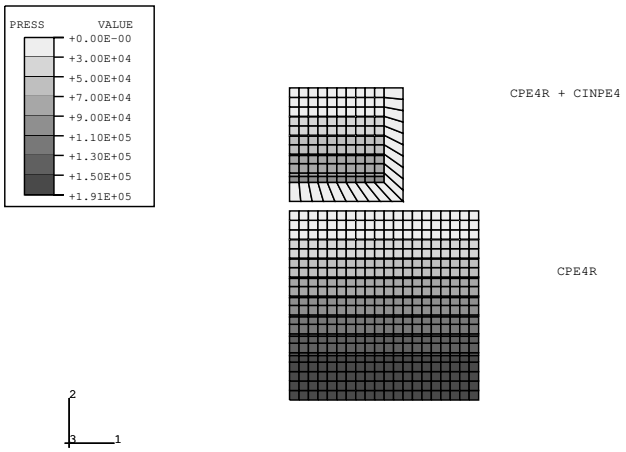


Figure 4: Pressure contours, plane strain case, end of Step 1.

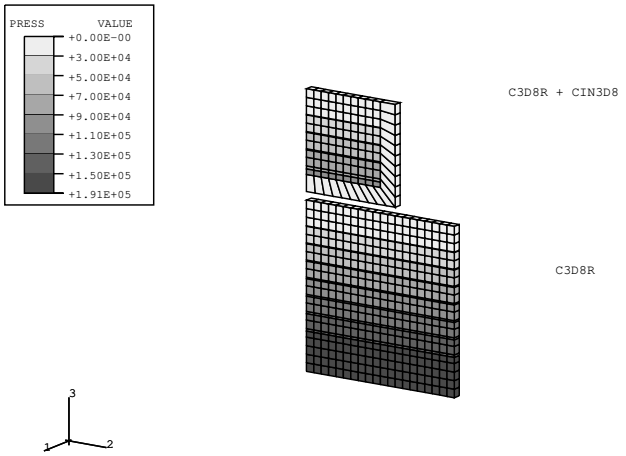


Figure 5: Pressure contours, three-dimensional case, end of Step 1.

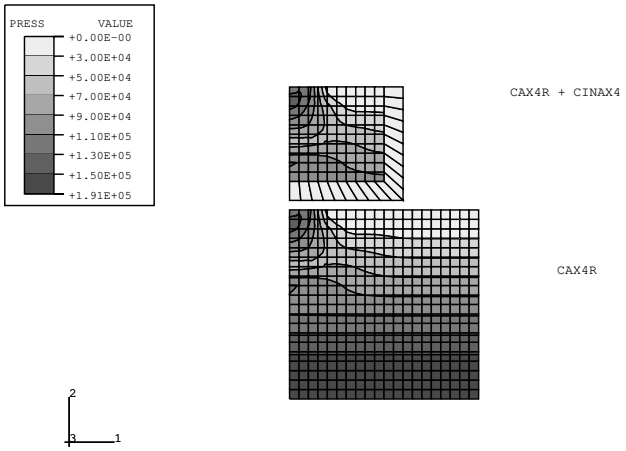


Figure 6: Hydrostatic pressure contours, axisymmetric case, end of Step 2.

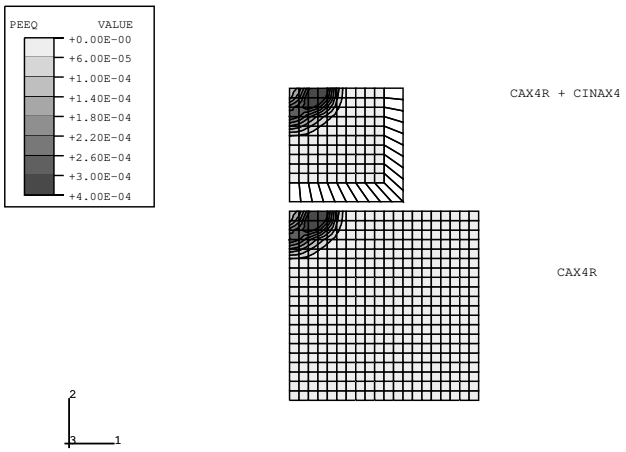


Figure 7: Equivalent plastic strain contours, axisymmetric case, end of Step 2.

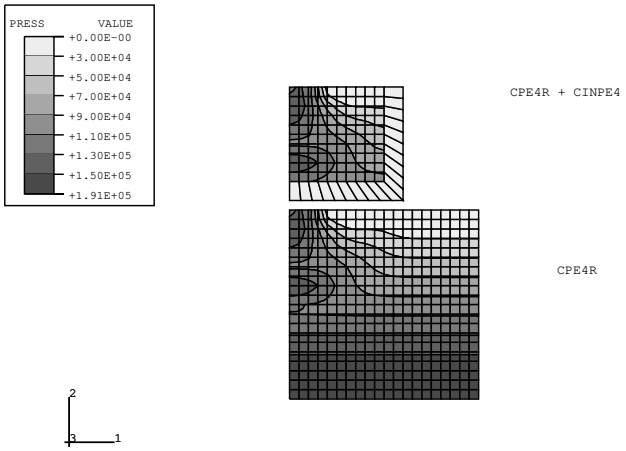


Figure 8: Hydrostatic pressure contours, plane strain case, end of Step 2.

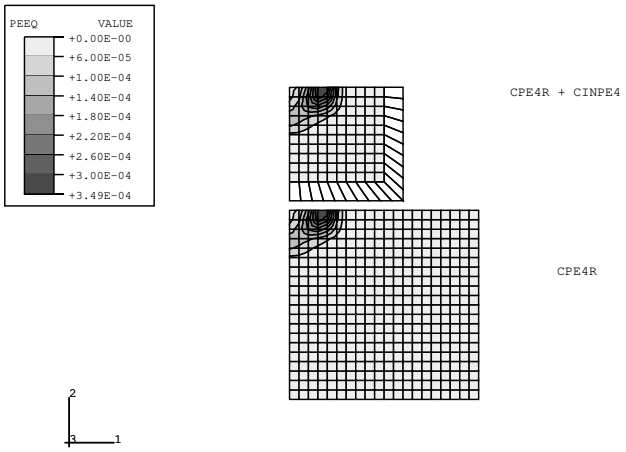


Figure 9: Equivalent plastic strain contours, plane strain case, end of Step 2.

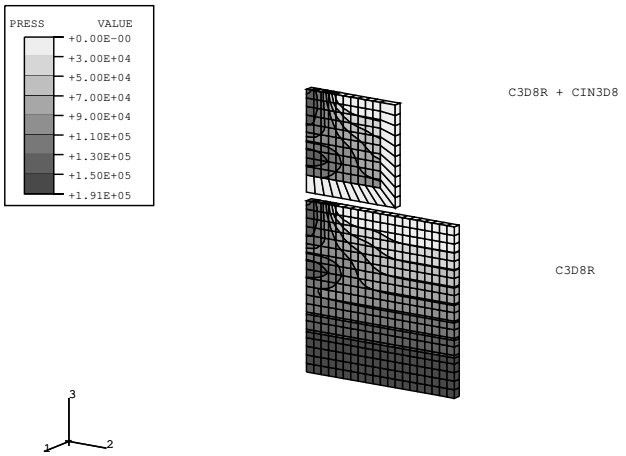


Figure 10: Hydrostatic pressure contours, three-dimensional case, end of Step 2.

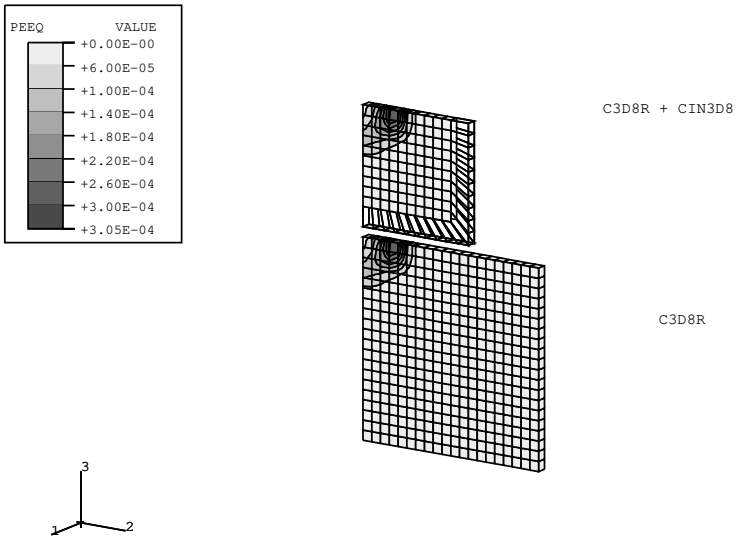


Figure 11: Equivalent plastic strain contours, three-dimensional case, end of Step 2.

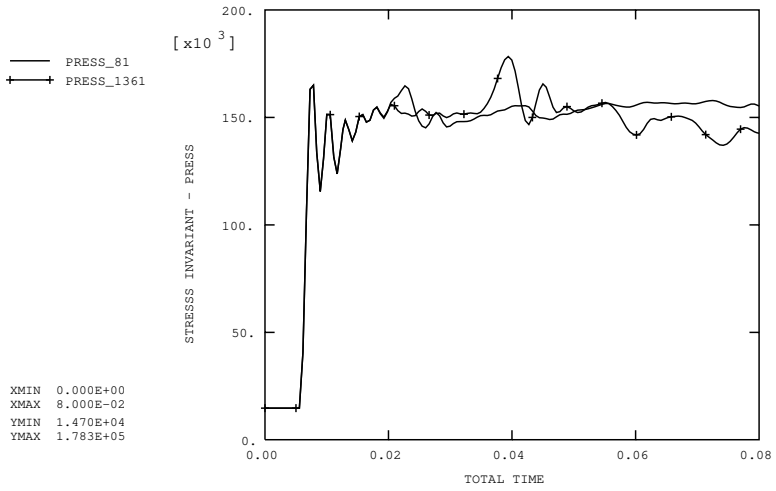


Figure 12: Pressure stress time history for same position in small and large models, plane strain case.

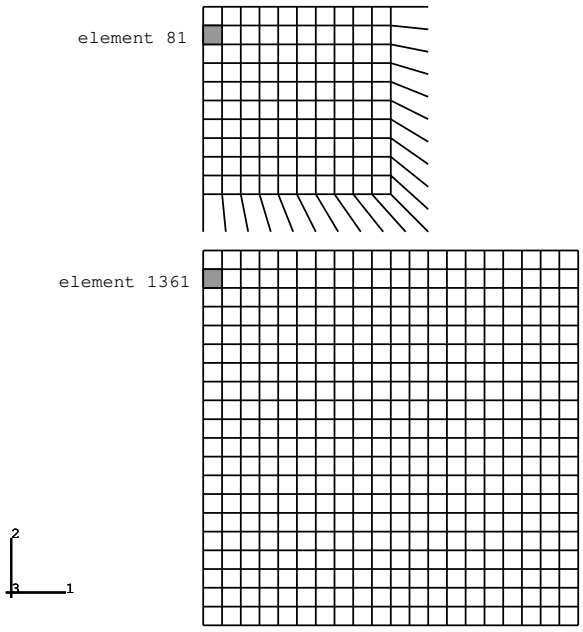


Figure 13: The position of elements 81 and 1361.

NAFEMS Benchmarks

This chapter defines some of the finite element benchmarks that are recommended by the National Agency for Finite Element Methods and Standards (UK). The first three categories of benchmarks documented here are the Linear Elastic, Temperature, and Free Vibration tests described in NAFEMS publication TNSB, Rev. 3, "The Standard NAFEMS Benchmarks," October 1990. Proposed benchmarks from the draft documents "Selected Benchmarks for Forced Vibration," R0016, March 1993, and "NAFEMS Non-linear Benchmarks," October 1989, are also included here. The benchmarks in this chapter are tested as part of the verification of Abaqus.

In this section:

- *Standard benchmarks: linear elastic tests*
- *Standard benchmarks: linear thermoelastic tests*
- *Standard benchmarks: free vibration tests*
- *Proposed forced vibration benchmarks*
- *Proposed nonlinear benchmarks*
- *Two-dimensional test cases in linear elastic fracture mechanics*
- *Fundamental tests of creep behavior*
- *Composite tests*
- *Geometric nonlinear tests*

Standard benchmarks: linear elastic tests

In this section:

- *LE1: Plane stress elements—elliptic membrane*
- *LE2: Cylindrical shell bending patch test*
- *LE3: Hemispherical shell with point loads*
- *LE4: Axisymmetric hyperbolic shell under uniform internal pressure*
- *LE5: Z-section cantilever*
- *LE6: Skew plate under normal pressure*
- *LE7: Axisymmetric cylinder/sphere under pressure*
- *LE8: Axisymmetric shell under pressure*
- *LE9: Axisymmetric branched shell under pressure*
- *LE10: Thick plate under pressure*
- *LE11: Solid cylinder/taper/sphere—temperature loading*

LE1: Plane stress elements—elliptic membrane

Products: Abaqus/Standard Abaqus/Explicit

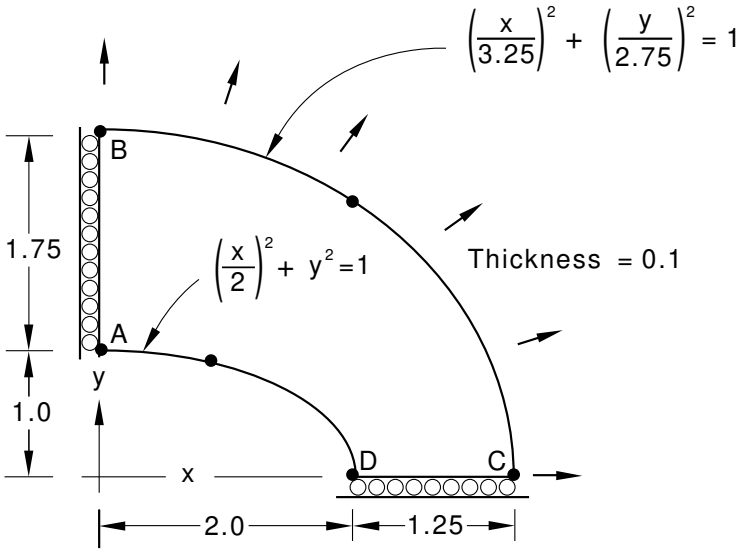
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CPS3 CPS4 CPS4I CPS4R CPS6 CPS6M CPS8 CPS8R

Problem description



Model: Plane stress problem with shape defined by ABCD. Functions defining the curves BC and AD are given above.

Mesh: A coarse and a fine mesh are tested for each element. In addition, a very fine mesh is tested for each element in the explicit dynamic analysis.

Material: Linear elastic, Young's modulus = 210 GPa, Poisson's ratio = 0.3, density = 7800 kg/m^3.

Boundary conditions: $u_x = 0$ along edge AB, $u_y = 0$ along edge CD.

Loading: Uniform outward pressure of 10 MPa at outer edge BC. In the explicit dynamic analysis the loading is applied such that a quasi-static solution is obtained.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test LE1 from NAFEMS publication TNSB, Rev. 3, "The Standard NAFEMS Benchmarks," October 1990.

Target solution: Tangential edge stress (σ_{yy}) at D is 92.7 MPa.

Results and discussion

The results are shown in *Table 1* and *Table 2*. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Table 1: Abaqus/Standard analysis.

Element	Coarse Mesh	Fine Mesh
CPS3	51.04 MPa (–45%)	71.26 MPa (–23%)
CPS4	66.73 MPa (–28%)	84.54 MPa (–9%)
CPS4I	58.82 MPa (–37%)	78.21 MPa (–16%)
CPS4R*	40.48 MPa (–56%)	56.18 MPa (–39%)
CPS6	89.10 MPa (–4%)	94.01 MPa (1%)
CPS6M	85.88 MPa (–7%)	93.71 MPa (1%)
CPS8	84.54 MPa (–9%)	92.81 MPa (0.12%)
CPS8R	85.80 MPa (–7%)	90.07 MPa (–3%)

*A comparison of the results for reduced-integration and full-integration lower-order elements indicates that the full-integration elements perform significantly better for problems with stress concentrations of this type.

Table 2: Abaqus/Explicit analysis.

Element	Coarse Mesh	Fine Mesh	Very Fine Mesh
CPS3	51.2 MPa (–45%)	71.5 MPa (–23%)	85.7 MPa (–8%)
CPS4R	39.6 MPa (–57%)	55.7 MPa (–40%)	87.3 MPa (–6%)
CPS6M	86.12 MPa (–7%)	92.93 MPa (–0.2%)	—

Input files

Abaqus/Standard input files

Coarse mesh tests:

nle1xf3c.inp	CPS3 elements.
nle1xf4c.inp	CPS4 elements.
nle1xi4c.inp	CPS4I elements.
nle1xr4c.inp	CPS4R elements.
nle1xf6c.inp	CPS6 elements.
nle1xm6c.inp	CPS6M elements.
nle1xf8c.inp	CPS8 elements.

nle1xr8c.inp	CPS8R elements.

Fine mesh tests:

nle1xf3f.inp	CPS3 elements.
nle1xf4f.inp	CPS4 elements.
nle1xi4f.inp	CPS4I elements.
nle1xr4f.inp	CPS4R elements.
nle1xf6f.inp	CPS6 elements.
nle1xm6f.inp	CPS6M elements.
nle1xf8f.inp	CPS8 elements.
nle1xr8f.inp	CPS8R elements.

Abaqus/Explicit input files

Coarse mesh tests:

le1_cps3_c.inp	CPS3 elements.
le1_cps4r_c.inp	CPS4R elements.
le1_cps6m_c.inp	CPS6M elements.

Fine mesh tests:

le1_cps3_f.inp	CPS3 elements.
le1_cps4r_f.inp	CPS4R elements.
le1_cps6m_f.inp	CPS6M elements.

Very fine mesh tests:

le1_cps3_vf.inp	CPS3 elements.
le1_cps4r_vf.inp	CPS4R elements.

LE2: Cylindrical shell bending patch test

Products: Abaqus/Standard Abaqus/Explicit

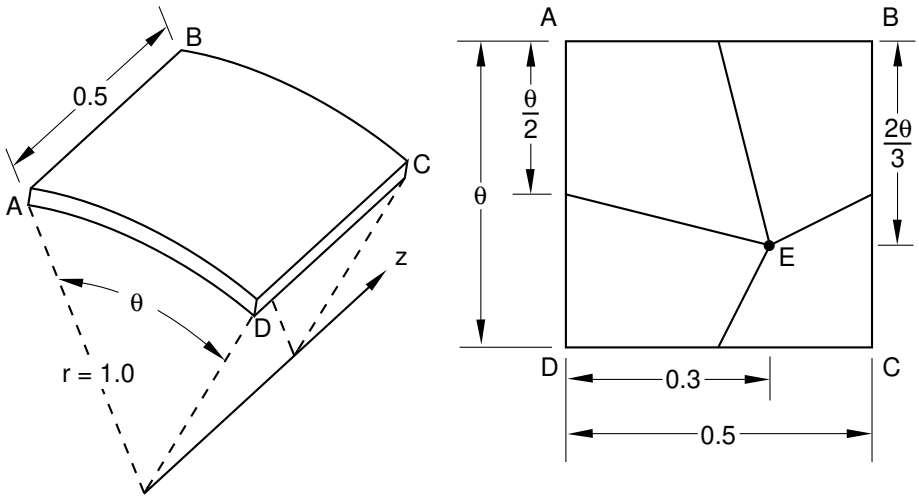
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

S3 S3R S3RS S4 S4R S4R5 S4RS S4RSW S8R S8R5 S9R5
STRI3 STRI65 SC6R SC8R

Problem description



Model: Sector of cylindrical shell with a thickness t= 0.01 m.

Material: Linear elastic, Young's modulus = 210 GPa, Poisson's ratio = 0.3, density = 7800 kg/m^3.

Boundary conditions: Edge AB is clamped. Axial displacements are constrained along edges AD and BC.

Loading: Uniform normal edge moment of 1000/unit length along edge DC. In the explicit dynamic analysis the loading is applied such that a quasi-static solution is obtained.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test LE2 from NAFEMS publication TNSB, Rev. 3, “The Standard NAFEMS Benchmarks,” October 1990.

Stress: Outer surface tangential stress at point E is 60 MPa.

Results and discussion

The results shown in *Table 1* through *Table 4* are interpolated from the integration points to the required nodal location. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Table 1: Abaqus/Standard analysis, $\theta = 30°$.

Element	Bottom Surface (MPa)	Top Surface (MPa)
S3/S3R	–44.3 (–26%)	40.6 (–32%)
S4	–63.2 (5%)	54.0 (–10%)
S4R	–58.0 (–3%)	58.0 (–3%)
S4R*	–55.0 (–8%)	55.2 (–8%)
S4R5	–58.6 (–2%)	58.6 (–2%)
S8R	–50.7 (–16%)	50.4 (–16%)
S8R5	–57.8 (–4%)	58.2 (–3%)
S9R5	–57.9 (–4%)	58.3 (–3%)
STRI3	–37.9 (–37%)	36.0 (–40%)
STRI65	–53.6 (–11%)	53.9 (–10%)
SC6R	–43.9 (–27%)	43.9 (–27%)
SC8R	–59.7 (–1%)	59.7 (–1%)
SC8R*	–54.8 (–9%)	–54.8 (–9%)
*Abaqus/Standard results with enhanced hourglass control.		

Table 2: Abaqus/Explicit analysis, $\theta = 30°$.

Element	Bottom Surface (MPa)	Top Surface (MPa)
S3R	–43.2 MPa (–28%)	39.7 MPa (–34%)
S3RS	–44.7 MPa (–26%)	42.2 MPa (–30%)
S4R	–58.2 MPa (–3%)	58.3 MPa (–2.8%)
S4RS	–57.0 MPa (–5%)	56.9 MPa (–5.2%)
S4RSW	–57.3 MPa (–4.5%)	57.4 MPa (–4.3%)

These results vary significantly from the target value since the mesh is too coarse to capture a curvature of $\theta = 30°$. The mesh can be refined easily by reducing the arc angle to $\theta = 10°$. The following results show that such mesh refinement greatly improves the accuracy of the results.

Table 3: Abaqus/Standard analysis, $\theta = 10°$.

Element	Bottom Surface (MPa)	Top Surface (MPa)
S3/S3R	–60.1 (0.2%)	59.9 (–0.2%)
S4	–60.5 (0.8%)	59.5 (–0.8%)
S4R	–60.0 (0%)	60.0 (0%)
S4R*	–60.0 (0%)	60.0 (0%)

Element	Bottom Surface (MPa)	Top Surface (MPa)
S4R5	−60.0 (0%)	60.0 (0%)
S8R	−59.6 (−0.7%)	59.7 (−0.5%)
S8R5	−59.9 (−0.2%)	60.0 (0%)
S9R5	−59.7 (−0.5%)	60.0 (0%)
STRI3	−60.8 (1.3%)	60.8 (1.3%)
STRI65	−59.6 (−0.7%)	59.7 (−0.5%)
SC6R	−60.2 (0.3%)	60.2 (0.3%)
SC8R	−60.2 (0.3%)	60.2 (0.3%)
SC8R*	−60.2 (0.3%)	60.2 (0.3%)
*Abaqus/Standard results with enhanced hourglass control.		

Table 4: Abaqus/Explicit analysis, $\theta = 10°$.

Element	Bottom Surface (MPa)	Top Surface (MPa)
S3R	−60.2 MPa (−0.3%)	59.9 MPa (−0.1%)
S3RS	−60.0 MPa (0%)	59.8 MPa (−0.3%)
S4R	−60.0 MPa (0%)	60.0 MPa (0%)
S4RS	−60.1 MPa (−0.1%)	60.1 MPa (−0.1%)
S4RSW	−60.0 MPa (0%)	59.9 MPa (−0.2%)

Input files

Abaqus/Standard input files

= 30°:

nle2xf3c.inp	S3/S3R elements.
nle2xe4c.inp	S4 elements.
nle2xf4c.inp	S4R elements.
nle2xf4c_eh.inp	S4R elements with enhanced hourglass control.
nle2x54c.inp	S4R5 elements.
nle2x68c.inp	S8R elements.
nle2x58c.inp	S8R5 elements.
nle2x59c.inp	S9R5 elements.
nle2x63c.inp	STRI3 elements.
nle2x56c.inp	STRI65 elements.
nle2_std_sc6r_30.inp	SC6R elements.
nle2_std_sc8r_30.inp	SC8R elements.
nle2_std_sc8r_30_eh.inp	SC8R elements with enhanced hourglass control.

= 10°:

nle2xf3f.inp	S3/S3R elements.
nle2xe4f.inp	S4 elements.

nle2xf4f.inp	S4R elements.
nle2xf4f_eh.inp	S4R elements with enhanced hourglass control.
nle2x54f.inp	S4R5 elements.
nle2x68f.inp	S8R elements.
nle2x58f.inp	S8R5 elements.
nle2x59f.inp	S9R5 elements.
nle2x63f.inp	STRI3 elements.
nle2x56f.inp	STRI65 elements.
nle2_std_sc6r_10.inp	SC6R elements.
nle2_std_sc8r_10.inp	SC8R elements.
nle2_std_sc8r_10_eh.inp	SC8R elements with enhanced hourglass control.

Abaqus/Explicit input files

= 30°:

le2_s3r_c.inp	S3R elements.
le2_s3rs_c.inp	S3RS elements.
le2_s4r_c.inp	S4R elements.
le2_s4rs_c.inp	S4RS elements.
le2_s4rsw_c.inp	S4RSW elements.

= 10°:

le2_s3r_f.inp	S3R elements.
le2_s3rs_f.inp	S3RS elements.
le2_s4r_f.inp	S4R elements.
le2_s4rs_f.inp	S4RS elements.
le2_s4rsw_f.inp	S4RSW elements.

LE3: Hemispherical shell with point loads

Products: Abaqus/Standard Abaqus/Explicit

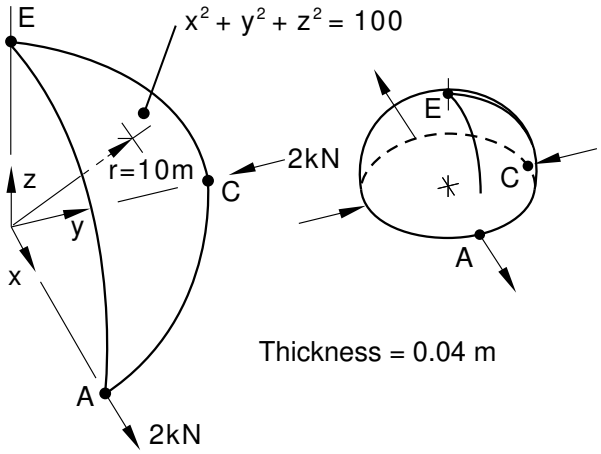
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

S3 S3R S4 S4R S4R5 S8R S8R5 S9R5 STRI3 STRI65 SC6R SC8R
SAXA12 SAXA22

Problem description



Model: The model is illustrated in the figure above. In addition, two input files are provided for the continuum shell element model to illustrate the use of the STACK DIRECTION=ORIENTATION parameter to define the element thickness (stacking) direction independent of the nodal connectivity using a spherical system.

Material: Linear elastic, Young's modulus = 68.25 GPa, Poisson's ratio = 0.3.

Boundary conditions: $u_x = u_y = u_z =$ 0 at point E. Along edge AE, symmetry about the *z*–*x* plane. Along edge CE, symmetry about the *y*–*z* plane.

Loading: Concentrated radial loads of 2 kN outward at A, inward at C.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test LE3 from NAFEMS publication TNSB, Rev. 3, "The Standard NAFEMS Benchmarks," October 1990.

Target solution: $u_x =$ 185 mm at point A.

Results and discussion

The values enclosed in parentheses are percentage differences with respect to the reference solution.

Element	u_x at A (Coarse)	u_x at A (Fine)
S3/S3R	0.080 (–57%)	0.161 (–13%)
S4	0.083 (–55%)	0.175 (–5%)
S4R	0.180 (–2.7%)	0.180 (–2.7%)
S4R*	0.072 (–61%)	0.170 (–8.1%)
S4R**	0.058 (–68%)	0.168 (–9.1%)
S4R5	0.190 (2.7%)	0.183 (–1.1%)
S8R	0.101 (–45%)	0.178 (–3.8%)
S8R5	0.179 (–3.2%)	0.185 (0.0%)
S9R5	0.179 (–3.2%)	0.185 (0.0%)
STRI3	0.173 (–1.2%)	0.185 (0.0%)
STRI65	0.169 (–8.6%)	0.182 (–1.6%)
SC6R	0.088 (–52.4%)	0.167 (–9.7%)
SC8R	0.210 (13.5%)	0.188 (1.6%)
SC8R***	0.194(4.9%)	0.185(0.0%)
SAXA12****	0.179 (–3.2%)	
SAXA22****	0.178 (–3.8%)	

* Abaqus/Explicit finite-strain element with enhanced hourglass control. **Abaqus/Standard finite-strain element with enhanced hourglass control. *** Abaqus/Explicit continuum shell element with the default “relax stiffness” hourglass control. **** Due to the loading position, only the Mode 2 and Mode 4 elements can be used.

Furthermore, due to the symmetries of the problem, only the Fourier interpolator $\cos 2\theta$ contributes to the solution. Thus, the Mode 4 elements produce identical results. Since Mode 4 is the highest-order Fourier term provided, no further circumferential mesh refinement is possible, and only coarse mesh results can be obtained.

The continuum shell element meshes using the STACK DIRECTION=ORIENTATION parameter yield identical results to the continuum shell element meshes in which the thickness direction is defined by the element nodal connectivity.

Input files

Abaqus/Standard input files

nle3xf3x.inp	S3/S3R elements.
nle3xe4x.inp	S4 elements.
nle3xf4x.inp	S4R elements.
nle3xf4x_eh.inp	S4R elements with enhanced hourglass control.
nle3x54x.inp	S4R5 elements.
nle3x68x.inp	S8R elements.
nle3x58x.inp	S8R5 elements.
nle3x59x.inp	S9R5 elements.

nle3x63x.inp	STRI3 elements.
nle3x56x.inp	STRI65 elements.
nle3xntx.inp	SAXA12 elements.
nle3xnxx.inp	SAXA22 elements.
nle3_std_sc6r.inp	SC6R elements.
nle3_std_sc8r.inp	SC8R elements.
nle3_std_sc6r_stackdir_sphori.inp	SC6R elements using the STACK DIRECTION=ORIENTATION parameter with a spherical orientation system to define the element thickness direction.
nle3_std_sc8r_stackdir_sphori.inp	SC8R elements using the STACK DIRECTION=ORIENTATION parameter with a spherical orientation system to define the element thickness direction.
nle3_std_sc8r_sgs.inp	SC8R elements using **SHELL GENERAL SECTION* to define section properties.

Abaqus/Explicit input files

le3_s4r.inp	S4R elements with enhanced hourglass control.
le3_sc8r.inp	SC8R elements with the default “relax stiffness” hourglass control.

LE4: Axisymmetric hyperbolic shell under uniform internal pressure

Products: Abaqus/Standard

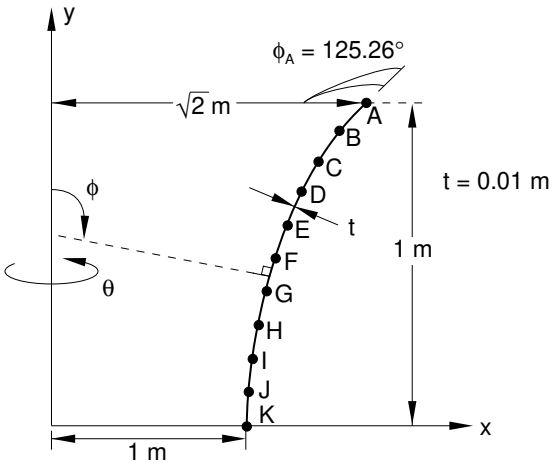
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

SAX1 SAX2 SAXA11 SAXA21

Problem description



Mesh: A coarse and a fine mesh are tested for each element.

Material: Linear elastic, Young's modulus = 210 GPa, Poisson's ratio = 0.3.

Boundary conditions: At point *B*, $u_z = 0$.

Loading: Uniform internal pressure of 1 MPa.

General: Gauss integration is used for the shell cross-section for the SAXA11 elements.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test LE4 from NAFEMS publication TNSB, Rev. 3, "The Standard NAFEMS Benchmarks," October 1990.

Target solution: On the midsurface at point *K* the meridional stress, $\sigma_{\phi\phi}$, is −50.0 MPa and the hoop stress, $\sigma_{\theta\theta}$, is 50.0 MPa.

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Element Type	$\sigma_{\phi\phi}$	$\sigma_{\theta\theta}$
SAX1 (Coarse)	−49.69 (−0.62%)	49.99 (−0.02%)
SAX1 (Fine)	−49.99 (−0.02%)	49.92 (−0.16%)
SAX2 (Coarse)	−50.09 (0.18%)	48.33 (−3.3%)
SAX2 (Fine)	−50.02 (0.04%)	48.34 (−3.3%)
SAXA11 (Coarse)	−49.69 (−0.62%)	49.92 (−0.16%)
SAXA11 (Fine)	−49.99 (−0.02%)	49.20 (−1.6%)
SAXA21 (Coarse)	−50.09 (0.18%)	48.33 (−3.3%)
SAXA21 (Fine)	−50.02 (0.04%)	48.34 (−3.3%)

Input files

Coarse mesh tests:

esa2smsf.inp	SAX1 elements.
esa3smsf.inp	SAX2 elements.
esnssmsf.inp	SAXA11 elements.
esnwsmsf.inp	SAXA21 elements.

Fine mesh tests:

esa2sfsf.inp	SAX1 elements.
esa3sfsf.inp	SAX2 elements.
esnssfsf.inp	SAXA11 elements.
esnwsfsf.inp	SAXA21 elements.

LE5: Z-section cantilever

Products: Abaqus/Standard Abaqus/Explicit

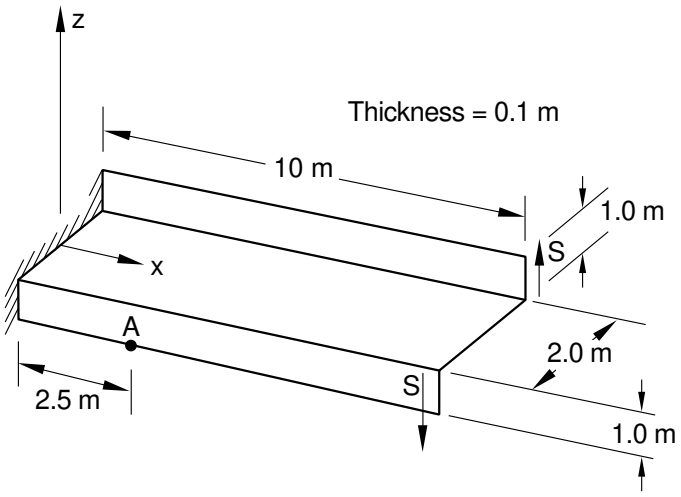
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

S3 S3R S4R S4R5 S4RS S4RSW S8R S8R5 S9R5 STRI3 STRI65
B31OS B32OS

Problem description



Model: Z-section cantilever under torsional loading.

Material: Linear elastic, Young's modulus = 210 GPa, Poisson's ratio = 0.3, density = 7800 kg/m^3.

Boundary conditions: All displacements are zero along the edge at $x = 0$.

Loading: Torque of 1.2 MN-m applied at $x = 10$. The torque is applied by two uniformly distributed edge shears of 0.6 MN at each flange when shell elements are used. In the explicit dynamic analysis the loading rate is applied such that a quasi-static solution is obtained.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test LE5 from NAFEMS publication TNSB, Rev. 3, "The Standard NAFEMS Benchmarks," October 1990.

Target solution: Axial stress, $\sigma_{xx} = -108$ MPa at midsurface, point A.

Results and discussion

The results are shown in *Table 1* and *Table 2*. The values enclosed in parentheses are percentage differences with respect to the reference solution. Slow convergence toward the target solution is seen as the mesh is refined.

Table 1: Abaqus/Standard analysis.

Element	σ_{xx}, Coarse Mesh	σ_{xx}, Refined Mesh
S3/S3R	−24.266 MPa (−78%)	−92.166 MPa (−15%)
S4	−110.36 MPa (2.2%)	−110.38 MPa (2.2%)
S4R	−50.480 MPa (−53%)	−96.732 MPa (−10%)
S4R5	−50.116 MPa (−54%)	−96.378 MPa (−11%)
S8R	−109.85 MPa (1.7%)	—
S8R5	−109.72 MPa (1.6%)	—
S9R5	−109.72 MPa (1.6%)	—
STRI3	−30.389 MPa (−72%)	−94.532 MPa (−12%)
STRI65	−107.32 MPa (−0.63%)	—
B31OS	−108.09 MPa (0.08%)	—
B32OS	−107.34 MPa (−0.61%)	—

Table 2: Abaqus/Explicit analysis.

Element	σ_{xx}, Coarse Mesh	σ_{xx}, Refined Mesh
S4R	−49.5 MPa (−54%)	−100.3 MPa (−7.1%)
S4RS	−87.5 MPa (−19%)	−100.3 MPa (−7.1%)
S4RSW	−87.7 MPa (−19%)	−100.3 MPa (−7.1%)

Input files

Abaqus/Standard input files

Coarse mesh tests:

nle5xf3c.inp	S3/S3R elements.
nle5xe4c.inp	S4 elements.
nle5xf4c.inp	S4R elements.
nle5x54c.inp	S4R5 elements.
nle5x68c.inp	S8R elements.
nle5x58c.inp	S8R5 elements.
nle5x59c.inp	S9R5 elements.
nle5x63c.inp	STRI3 elements.
nle5x56c.inp	STRI65 elements.
nle5xb2c.inp	B31OS elements.
nle5xb3c.inp	B32OS elements.

Fine mesh tests:

nle5xf3f.inp	S3/S3R elements.
nle5xe4f.inp	S4 elements.
nle5xf4f.inp	S4R elements.
nle5x54f.inp	S4R5 elements.
nle5x63f.inp	STRI3 elements.

Abaqus/Explicit input files

Coarse mesh tests:

le5_c.inp	S4R elements.
le5_c_s4rs.inp	S4RS elements.
le5_c_s4rsw.inp	S4RSW elements.

Fine mesh tests:

le5_f.inp	S4R elements.
le5_f_s4rs.inp	S4RS elements.
le5_f_s4rs_subcyc.inp	S4RS elements and subcycling.
le5_f_s4rsw.inp	S4RSW elements.

LE6: Skew plate under normal pressure

Products: Abaqus/Standard Abaqus/Explicit

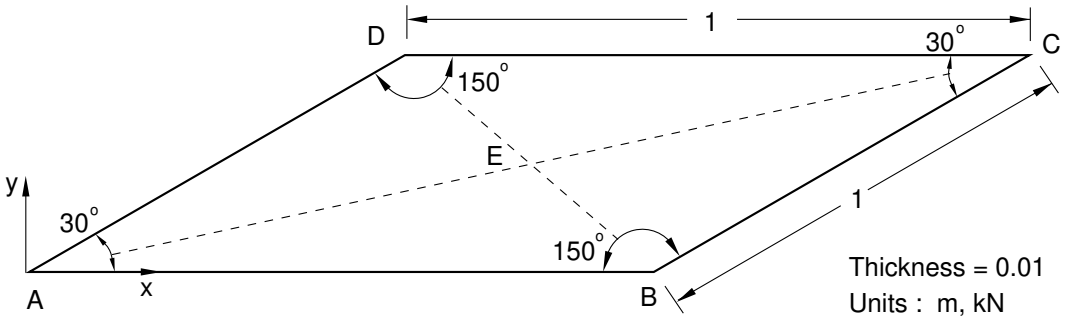
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

S4R S4RS S4RSW S8R5 S9R5

Problem description



Model: Skew plate under normal pressure.

Mesh: A coarse (2 × 2) and a fine (4 × 4) are tested for each element. In addition, a very fine (8 × 8) mesh is tested for each element in the explicit dynamic analysis.

Material: Linear elastic, Young's modulus = 210 GPa, Poisson's ratio = 0.3, density = 7800 kg/m^3.

Boundary conditions: $u_z = 0$ along edges AB, BC, CD, and AD. $u_x = u_y =$ 0 at point A and $u_y = 0$ at point B to prevent rigid body motion.

Loading: Uniform pressure of −0.7 kPa in the vertical z-direction. In the explicit dynamic analysis the loading is applied such that a quasi-static solution is obtained.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test LE6 from NAFEMS Publication TNSB, Rev. 3, “The Standard NAFEMS Benchmarks,” October 1990.

Target solution: Maximum principal stress = 0.802 MPa on the lower surface at point E.

Results and discussion

The results are shown in *Table 1* and *Table 2*. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Table 1: Abaqus/Standard analysis.

Element	Coarse Mesh	Fine Mesh
S8R5	1.156 MPa (+44.1%)	0.862 MPa (+7.5%)
S9R5	1.156 MPa (+44.1%)	0.862 MPa (+7.5%)

Table 2: Abaqus/Explicit analysis.

Element	Coarse Mesh	Fine Mesh	Very Fine Mesh
S4R	0.338 MPa (−58%)	0.703 MPa (−12.3%)	0.765 MPa (−4.61%)
S4RS	0.343 MPa (−57%)	0.745 MPa (−7.11%)	0.8021 MPa (+0.01%)
S4RSW	0.341 MPa (−57%)	0.674 MPa (−16.0%)	0.8034 MPa (+0.17%)

Remarks

The skew sensitivity of shell elements is discussed in *Skew sensitivity of shell elements*.

Input files

Abaqus/Standard input files

Coarse mesh tests:

nle6x58c.inp	S8R5 elements.
nle6x59c.inp	S9R5 elements.

Fine mesh tests:

nle6x58f.inp	S8R5 elements.
nle6x59f.inp	S9R5 elements.

Abaqus/Explicit input files

Coarse mesh tests:

le6_c.inp	S4R elements.
le6_c_s4rs.inp	S4RS elements.
le6_c_s4rsw.inp	S4RSW elements.

Refined mesh tests:

le6_f.inp	S4R elements.
le6_f_s4rs.inp	S4RS elements.
le6_f_s4rsw.inp	S4RSW elements.

Very refined mesh tests:

le6_vf.inp	S4R elements.

le6_vf_s4rs.inp	S4RS elements.
le6_vf_s4rsw.inp	S4RSW elements.

LE7: Axisymmetric cylinder/sphere under pressure

Products: Abaqus/Standard Abaqus/Explicit

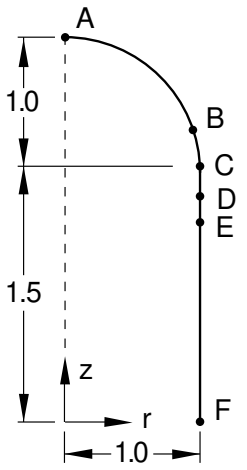
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

SAX1 SAX2

Problem description



Units: m, kN

Thickness = 0.025

Point	r	z
B	0.9814	1.6920
D	1.0	1.4034
E	1.0	1.1136

Model: Thin-walled pressure vessel.

Mesh: A coarse and a fine mesh are tested.

Material: Linear elastic, Young's modulus = 210 GPa, Poisson's ratio = 0.3, density = 7800 kg/m^3.

Boundary conditions: $u_r = \phi =$ 0 at point A. $u_z =$ 0 at point F.

Loading: Uniform internal pressure of 1.0 MPa. In the explicit dynamic analysis the loading is applied such that a quasi-static solution is obtained.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test LE7 from NAFEMS Publication TNSB, Rev. 3, “The Standard NAFEMS Benchmarks,” October 1990.

Target solution: Axial stress, $\sigma_{zz} =$ 25.9 MPa on the outer surface at point D.

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences from the target solution.

Element	σ_{zz}, Coarse Mesh	σ_{zz}, Fine Mesh
SAX1 (Abaqus/Explicit)	25.6 MPa (–1%)	25.7 MPa (–0.5%)
SAX2 (Abaqus/Standard)	26.034 MPa (+0.67%)	25.878 MPa (+0.07%)

Input files

Coarse mesh tests:

le7_c.inp	SAX1 elements.
nle7xa3c.inp	SAX2 elements.

Fine mesh tests:

le7_f.inp	SAX1 elements.
nle7xa3f.inp	SAX2 elements.

LE8: Axisymmetric shell under pressure

Products: Abaqus/Standard Abaqus/Explicit

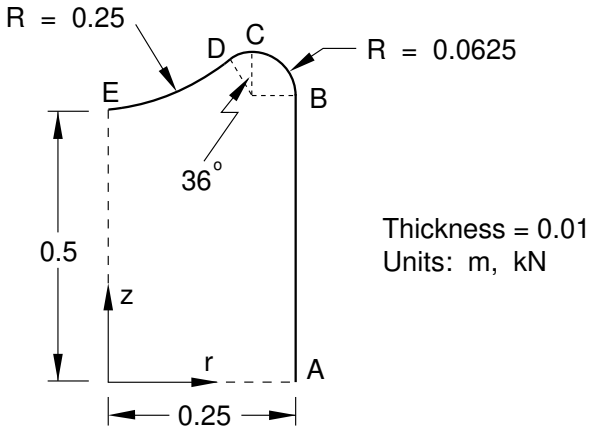
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

SAX1 SAX2

Problem description



Model: Axisymmetric shell under pressure.

Mesh: A coarse and a fine mesh are tested.

Material: Linear elastic, Young's modulus = 210 GPa, Poisson's ratio = 0.3, density = 7800 kg/m^3.

Boundary conditions: $u_z =$ 0 at point A. $u_r = \phi =$ 0 at point F.

Loading: Uniform internal pressure of 1.0 MPa. In the explicit dynamic analysis the loading is applied such that a quasi-static solution is obtained.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test LE8 from NAFEMS Publication TNSB, Rev. 3, "The Standard NAFEMS Benchmarks," October 1990.

Target solution: Hoop stress, $\sigma_{\theta\theta} =$ 94.5 MPa on the outer surface at point D.

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences from the target solution.

Element	$\sigma_{\theta\theta}$, Coarse Mesh	$\sigma_{\theta\theta}$, Fine Mesh
SAX1 (Abaqus/Explicit)	99.1 MPa (+5%)	89.3 MPa (−6%)
SAX2 (Abaqus/Standard)	90.12 MPa (−4.7%)	90.41 MPa (−4.4%)

Input files

Coarse mesh tests:

le8_c.inp	SAX1 elements.
nle8xa3c.inp	SAX2 elements.

Fine mesh tests:

le8_f.inp	SAX1 elements.
nle8xa3f.inp	SAX2 elements.

LE9: Axisymmetric branched shell under pressure

Products: Abaqus/Standard

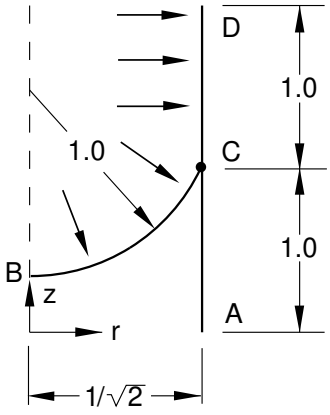
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

SAX2

Problem description



Units: m, kN

Thickness = 0.01

Mesh: A coarse and a fine mesh are tested.

Material: Linear elastic, Young's modulus = 210 GPa, Poisson's ratio = 0.3.

Boundary conditions: $u_r = u_z = \phi = 0$ at point A.

Loading: Uniform internal pressure of 1.0 MPa along edge BCD.

General: Gauss integration is used for the shell cross-section in input file *nle9xa3f.inp*.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test LE9 from NAFEMS Publication TNSB, Rev. 3, "The Standard NAFEMS Benchmarks," October 1990.

Target solution: Axial stress, σ_{zz} = −319.9 MPa on the outer surface of the upper cylinder at point C.

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Element	σ_{zz}, Coarse Mesh	σ_{zz}, Fine Mesh
SAX2	−307.24 MPa (−4.0%)	−314.81 MPa (−1.6%)

Input files

nle9xa3c.inp	Coarse mesh analysis.
nle9xa3f.inp	Fine mesh analysis.

LE10: Thick plate under pressure

Products: Abaqus/Standard Abaqus/Explicit

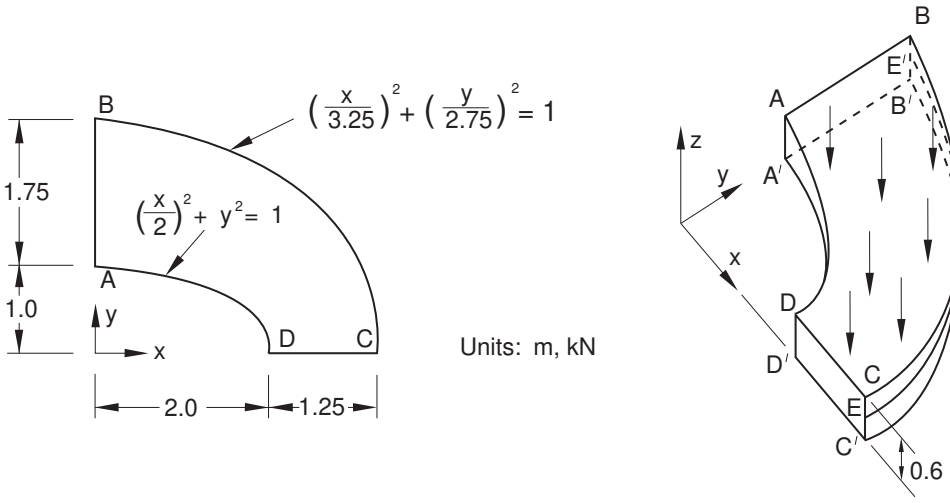
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

C3D20 C3D20R C3D10 C3D10HS C3D10M

Problem description



Model: Thick plate under uniform pressure.

Mesh: A coarse and a fine mesh are tested.

Material: Linear elastic, Young's modulus = 210 GPa, Poisson's ratio = 0.3, density = 7800 kg/m^3.

Boundary conditions: $u_y = 0$ on face DCD C . $u_x = 0$ on face ABA B . $u_x = u_y = 0$ on face BCB C . $u_z = 0$ on line EE (E is the midpoint of edge CC ; E is the midpoint of edge BB).

Loading: Uniform normal pressure of 1.0 MPa on the upper surface of the plate.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test LE10 from NAFEMS Publication TNSB, Rev. 3, “The Standard NAFEMS Benchmarks,” October 1990.

Target solution: Direct stress, $\sigma_{yy} = 5.38$ MPa at point D.

Results and discussion

The Abaqus/Standard results are shown in *Table 1*. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Table 1: Abaqus/Standard analysis.

Element	σ_{yy}, Coarse Mesh	σ_{yy}, Fine Mesh
C3D20	−6.72 MPa (+25.00%)	−5.64 MPa (+4.83%)
C3D20R	−7.93 MPa (+47.39%)	−5.53 MPa (+2.78%)
C3D10	−5.44 MPa (+1.15%)	−5.77 MPa (+7.24%)
C3D10HS	−5.08 MPa (−3.72%)	−5.51 MPa (+2.42%)
C3D10M	−5.57 MPa (+3.53%)	−5.89 MPa (+9.48%)

The C3D10 and C3D10M elements are more accurate with the coarse mesh than with the fine mesh: in the coarse meshes four elements come together at the point of interest, giving a more accurate result after averaging to the nodes. In the more refined mesh, only one element contains the point of interest; therefore, the extrapolation to the nodes is less accurate.

Unlike Abaqus/Standard, Abaqus/Explicit does not have the option for extrapolating integration point outputs (such as stresses) to the nodes. Consequently, the desired stress component at point D cannot be extracted except by rough interpretation of color contour plots. As an alternative, the value of σ_{yy} at an integration point near point D is compared between an Abaqus/Standard simulation and an Abaqus/Explicit simulation.

In the Abaqus/Explicit analyses the pressure is ramped up smoothly from zero to its final value of 1.0 MPa over a time period of 0.4 seconds, which is slow enough to be considered quasi-static (inertial effects play a minimal role).

Analysis Type	σ_{yy}, Coarse Mesh	σ_{yy}, Fine Mesh
Abaqus/Standard	−3.70 MPa	−4.61 MPa
Abaqus/Explicit	−3.79 MPa	−4.55 MPa

For the coarse mesh the point of comparison is at element 18, integration point 3. For the fine mesh the point of comparison is at element 199, integration point 1. Both are close neighbors of the physical corner point D.

Input files

Abaqus/Standard input files

Coarse mesh tests:

nle10fkc.inp	C3D20 elements.
nle10rkc.inp	C3D20R elements.
nle10c_c3d10.inp	C3D10 elements.
nle10c_c3d10hs.inp	C3D10HS elements.
nle10c_c3d10m.inp	C3D10M elements.

Fine mesh tests:

nle10fkf.inp	C3D20 elements.

nle10rkf.inp	C3D20R elements.
nle10f_c3d10.inp	C3D10 elements.
nle10f_c3d10hs.inp	C3D10HS elements.
nle10f_c3d10m.inp	C3D10M elements.

Abaqus/Explicit input files

exxle10_c.inp	C3D10M elements, coarse mesh.
exxle10_f.inp	C3D10M elements, fine mesh.

LE11: Solid cylinder/taper/sphere—temperature loading

Products: Abaqus/Standard

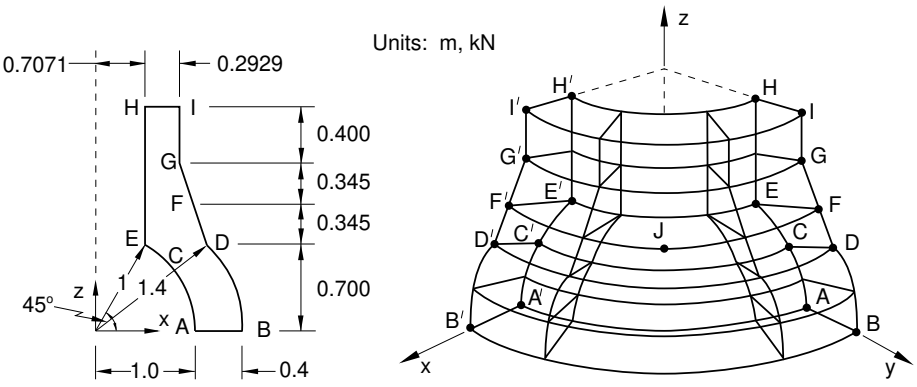
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

C3D20 C3D20R

Problem description



Mesh: A coarse and a fine mesh are tested.

Material: Linear elastic, Young's modulus = 210 GPa, Poisson's ratio = 0.3, coefficient of thermal expansion = 2.3E−4/°C.

Boundary conditions: $u_y = 0$ on the plane $y = 0$. $u_x = 0$ on the plane $x = 0$. $u_z = 0$ on the plane $z = 0$ and the face HIH I .

Loading: Linear temperature gradient in the radial and axial directions is given by

$$\Delta\theta = \sqrt{(x^2 + y^2)} + z.$$

This is applied using user subroutine *UTEMP*.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test LE11 from NAFEMS Publication TNSB, Rev. 3, “The Standard NAFEMS Benchmarks,” October 1990.

Target solution: Direct stress, σ_{zz} = −105 MPa at point A.

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Element	σ_{zz}, Coarse Mesh	σ_{zz}, Fine Mesh
C3D20	−96.71 MPa (−7.9%)	−103.26 MPa (−1.7%)
C3D20R	−93.04 MPa (−11.4%)	−99.60 MPa (−5.1%)

Input files

Coarse mesh tests:

nle11fkc.inp	C3D20 elements.
nle11fkc.f	User subroutine used in nle11fkc.inp.
nle11rkc.inp	C3D20R elements.
nle11rkc.f	User subroutine used in nle11rkc.inp.

Fine mesh tests:

nle11fkf.inp	C3D20 elements.
nle11fkf.f	User subroutine used in nle11fkf.inp.
nle11rkf.inp	C3D20R elements.
nle11rkf.f	User subroutine used in nle11rkf.inp.

Standard benchmarks: linear thermoelastic tests

In this section:

- *T1: Plane stress elements—membrane with hot-spot*
- *T2: One-dimensional heat transfer with radiation*
- *T3: One-dimensional transient heat transfer*
- *T4: Two-dimensional heat transfer with convection*

T1: Plane stress elements—membrane with hot-spot

Products: Abaqus/Standard

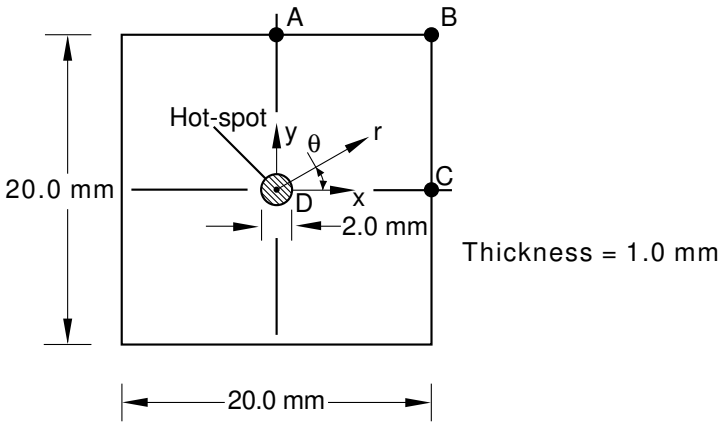
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CPS3 CPS4 CPS4I CPS4R CPS6 CPS6M CPS8 CPS8R

Problem description



Material: Linear elastic, Young's modulus = 100 GPa, Poisson's ratio = 0.3.

Boundary conditions: A quarter-section is modeled with the symmetry conditions $u_y = 0$ along $y = 0$ and $u_x = 0$ along $x = 0$.

Loading: Within the hot-spot, thermal strain (αT) = 1.0×10^{-3}. Outside the hot-spot, thermal strain (αT) = 0.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test T1 from NAFEMS publication TNSB, Rev. 3, "The Standard NAFEMS Benchmarks," October 1990.

Target solution: σ_{yy} at point D (outside the hot-spot) = 50.0 MPa.

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Element	σ_{yy} at D
CPS3	54.25 MPa (8.4%)
CPS4	50.94 MPa (1.9%)
CPS4I	46.29 MPa (−7.4%)
CPS4R	23.65 MPa (−52.7%)*
CPS6	54.46 MPa (8.9%)
CPS6M	54.25 MPa (8.5%)
CPS8	51.53 MPa (3.1%)
CPS8R	44.06 MPa (−11.9%)*

*A comparison of the results for reduced-integration and full-integration elements indicates that the full-integration elements perform significantly better for problems with stress concentrations of this type.

Input files

nt1xxf3x.inp	CPS3 elements.
nt1xxf4x.inp	CPS4 elements.
nt1xxf6x.inp	CPS6 elements.
nt1xxf8x.inp	CPS8 elements.
nt1xxi4x.inp	CPS4I elements.
nt1xxm6x.inp	CPS6M elements.
nt1xxr4x.inp	CPS4R elements.
nt1xxr8x.inp	CPS8R elements.

T2: One-dimensional heat transfer with radiation

Products: Abaqus/Standard Abaqus/Explicit

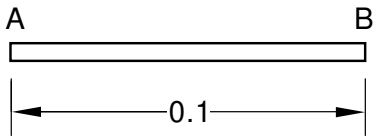
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

DC1D2 DC1D3
DC2D3 DC2D4 DC2D6 DC2D8
DCAX3 DCAX4 DCAX6 DCAX8
CAX3T CAX4RHT CAX4RT CAX4T
CPE3T CPE4RHT CPE4RT CPE4T CPE6MHT CPE6MT
CPS3T CPS4RT CPS4T CPS6MT
C3D4T C3D6PHT C3D6PT C3D6T C3D8RHT C3D8RT C3D8T SC6RT SC8RT
EC3D8RT

Problem description



Model: The geometry is shown above. A uniform mesh with 10 elements along the length of the bar is used. The thickness and width of the bar are each 0.01 m. In Abaqus/Standard a steady-state simulation is performed, while in Abaqus/Explicit a transient simulation is performed. The total simulation time in the latter case is 2500 seconds. This provides enough time for the transient solution to reach steady-state conditions in this problem.

Material: Conductivity = 55.6 W/m°C, specific heat = 460.0 J/kg°C, density = 7850 kg/m^3. At end B emissivity = 0.98, Stefan-Boltzmann constant = 5.67×10^{-8} W m^2/ K^4.

For the coupled temperature-displacement elements dummy mechanical properties are used to complete the material definition.

Boundary conditions: Prescribed temperature of 1000 K at end A. Radiation to ambient temperature of 300 K at end B. No heat flux perpendicular to AB.

Loading: Zero internal heat generation.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test T2 from NAFEMS publication TNSB, Rev. 3, “The Standard NAFEMS Benchmarks,” October 1990.

Target solution: Temperature at B=927 K (653.85°C).

Results and discussion

All elements yield the exact solution.

Input files

Abaqus/Standard input files

nt2xx12x.inp	DC1D2 elements.
nt2xx13x.inp	DC1D3 elements.
nt2xx23x.inp	DC2D3 elements.
nt2xx24x.inp	DC2D4 elements.
nt2xx26x.inp	DC2D6 elements.
nt2xx28x.inp	DC2D8 elements.
nt2xxa3x.inp	DCAX3 elements.
nt2xxa4x.inp	DCAX4 elements.
nt2xxa6x.inp	DCAX6 elements.
nt2xxa8x.inp	DCAX8 elements.
onedheattransrad_std_cpe3t.inp	CPE3T elements.
onedheattransrad_std_cpe4rht.inp	CPE4RHT elements.
onedheattransrad_std_cpe4rt.inp	CPE4RT elements.
onedheattransrad_std_cpe4t.inp	CPE4T elements.
onedheattransrad_std_cpe6mht.inp	CPE6MHT elements.
onedheattransrad_std_cpe6mt.inp	CPE6MT elements.
onedheattransrad_std_cps3t.inp	CPS3T elements.
onedheattransrad_std_cps4rt.inp	CPS4RT elements.
onedheattransrad_std_cps4t.inp	CPS4T elements.
onedheattransrad_std_cps6mt.inp	CPS6MT elements.
onedheattransrad_std_cax4rht.inp	CAX4RHT elements.
onedheattransrad_std_cax3t.inp	CAX3T elements.
onedheattransrad_std_cax4rt.inp	CAX4RT elements.
onedheattransrad_std_cax4t.inp	CAX4T elements.
onedheattransrad_std_cax6mht.inp	CAX6MHT elements.
onedheattransrad_std_cax6mt.inp	CAX6MT elements.
onedheattransrad_std_c3d4t.inp	C3D4T elements.
onedheattransrad_std_c3d6pht.inp	C3D6PHT elements.
onedheattransrad_std_c3d6pt.inp	C3D6PT elements.
onedheattransrad_std_c3d6t.inp	C3D6T elements.
onedheattransrad_std_c3d8rht.inp	C3D8RHT elements.
onedheattransrad_std_c3d8rt.inp	C3D8RT elements.
onedheattransrad_std_c3d8t.inp	C3D8T elements.

Abaqus/Explicit input files

onedheattransrad_xpl_cax3t.inp	CAX3T elements.
onedheattransrad_xpl_cax4rt.inp	CAX4RT elements.
onedheattransrad_xpl_cpe3t.inp	CPE3T elements.
onedheattransrad_xpl_cpe4rt.inp	CPE4RT elements.
onedheattransrad_xpl_cps3t.inp	CPS3T elements.
onedheattransrad_xpl_cps4rt.inp	CPS4RT elements.
onedheattransrad_xpl_c3d4t.inp	C3D4T elements.
onedheattransrad_xpl_c3d6t.inp	C3D6T elements.
onedheattransrad_xpl_c3d8rt.inp	C3D8RT elements.
onedheattransrad_xpl_c3d8t.inp	C3D8T elements.
onedheattransrad_xpl_ec3d8rt.inp	EC3D8RT elements.
onedheattransrad_xpl_sc6rt.inp	SC6RT elements.
onedheattransrad_xpl_sc8rt.inp	SC8RT elements.

T3: One-dimensional transient heat transfer

Products: Abaqus/Standard Abaqus/Explicit

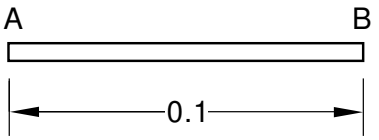
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

DC1D2 DC1D3
DC2D3 DC2D4 DC2D6 DC2D8
DCAX3 DCAX4 DCAX6 DCAX8
DS3 DS4 DS6 DS8
CAX3T CAX4T CAX4RT CAX4RHT CAX6MT CAX6MHT
CPE3T CPE4T CPE4RT CPE4RHT CPE6MT CPE6MHT
CPS3T CPS4T CPS4RT CPS6MT
S3RT S4RT

Problem description



Model: The geometry is shown above. A uniform mesh with 5 elements along the length of the bar is used. The thickness and width of the bar are each 0.01 m. A transient simulation is performed. The total simulation time is 32 seconds.

Material: Conductivity = 35.0 W/m°C, specific heat = 440.5 J/kg°C, density = 7200 kg/m^3.

For coupled temperature-displacement elements dummy mechanical properties are used to complete the material definition.

Boundary conditions: Temperature prescribed as 0°C at end A, and as 100 $\sin(\pi t/40)$°C at end B, where t is time in seconds. No heat flux perpendicular to AB.

Loading: Zero internal heat generation.

Initial conditions: All temperatures = 0°C.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test T3 from NAFEMS publication TNSB, Rev. 3, “The Standard NAFEMS Benchmarks,” October 1990.

Target solution: Temperature of 36.60°C at $x = 0.08$ m at 32 secs.

Results and discussion

The results from the Abaqus/Standard analysis are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Element	T, Coarse Mesh	T, Fine Mesh
DC1D2	34.54°C (–5.6%)	35.51°C (–3.0%)
DC1D3	35.88°C (–2.0%)	36.09°C (–1.4%)
DC2D3	34.54°C (–5.6%)	35.51°C (–3.0%)
DC2D4	34.54°C (–5.6%)	35.51°C (–3.0%)
DC2D6	36.27°C (–0.9%)	36.14°C (–1.3%)
DC2D8	35.88°C (–2.0%)	36.09°C (–1.4%)
DCAX3	34.54°C (–5.6%)	35.51°C (–3.0%)
DCAX4	34.54°C (–5.6%)	35.51°C (–3.0%)
DCAX6	35.91°C (–1.9%)	36.10°C (–1.4%)
DCAX8	35.88°C (–2.0%)	36.09°C (–1.4%)
DS3	34.54°C (–5.6%)	36.20°C (–1.1%)
DS4	34.54°C (–5.6%)	35.51°C (–3.0%)
DS6	35.37°C (–3.4%)	36.14°C (–1.3%)
DS8	35.88°C (–2.0%)	36.09°C (–1.4%)
CAX3T	34.54°C (–5.6%)	35.51°C (–3.0%)
CAX4T	34.54°C (–5.6%)	35.51°C (–3.0%)
CAX4RT	34.54°C (–5.6%)	35.51°C (–3.0%)
CAX4RHT	34.54°C (–5.6%)	35.51°C (–3.0%)
CAX6MHT	35.37°C (–3.4%)	35.85°C (–2.1%)
CAX6MT	35.37°C (–3.4%)	35.85°C (–2.1%)
CPE3T	34.54°C (–5.6%)	35.51°C (–3.0%)
CPE4T	34.54°C (–5.6%)	35.51°C (–3.0%)
CPE4RT	34.54°C (–5.6%)	35.51°C (–3.0%)
CPE4RHT	34.54°C (–5.6%)	35.51°C (–3.0%)
CPE6MT	35.81°C (–2.2%)	36.0°C (–1.6%)
CPE6MHT	35.81°C (–2.2%)	36.0°C (–1.6%)
CPS3T	34.54°C (–5.6%)	35.51°C (–3.0%)
CPS4T	34.54°C (–5.6%)	35.51°C (–3.0%)
CPS4RT	34.54°C (–5.6%)	35.51°C (–3.0%)
CPS6MT	35.81°C (–2.2%)	36.0°C (–1.6%)

The results from the Abaqus/Explicit analysis are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Element	T, Coarse Mesh	T, Fine Mesh
CAX3T	34.98°C (–4.4%)	36.08°C (–1.4%)
CAX4RT	34.99°C (–4.4%)	36.10°C (–1.4%)
CPE3T	35.08°C (–4.2%)	36.16°C (–1.2%)
CPE4RT	35.09°C (–4.1%)	36.27°C (–0.9%)
CPS3T	35.08°C (–4.2%)	36.16°C (–1.2%)
CPS4RT	35.09°C (–4.1%)	36.27°C (–0.9%)
S3RT	34.98°C (–4.4%)	36.08°C (–1.4%)
S4RT	35.09°C (–4.1%)	36.27°C (–0.9%)

Input files

Abaqus/Standard input files

Coarse mesh tests:

nt3xx12c.inp	DC1D2 elements.
nt3xx13c.inp	DC1D3 elements.
nt3xx23c.inp	DC2D3 elements.
nt3xx24c.inp	DC2D4 elements.
nt3xx26c.inp	DC2D6 elements.
nt3xx28c.inp	DC2D8 elements.
nt3xxa3c.inp	DCAX3 elements.
nt3xxa4c.inp	DCAX4 elements.
nt3xxa6c.inp	DCAX6 elements.
nt3xxa8c.inp	DCAX8 elements.
nt3xxs3c.inp	DS3 elements.
nt3xxs4c.inp	DS4 elements.
nt3xxs6c.inp	DS6 elements.
nt3xxs8c.inp	DS8 elements.
onedtransienthtc_std_cax3t.inp	CAX3T elements.
onedtransienthtc_std_cax4t.inp	CAX4T elements.
onedtransienthtc_std_cax4rt.inp	CAX4RT elements.
onedtransienthtc_std_cax4rht.inp	CAX4RHT elements.
onedtransienthtc_std_cax6mht.inp	CAX6MHT elements.
onedtransienthtc_std_cax6mt.inp	CAX6MT elements
onedtransienthtc_std_cpe3t.inp	CPE3T elements.
onedtransienthtc_std_cpe4t.inp	CPE4T elements.
onedtransienthtc_std_cpe4rt.inp	CPE4RT elements.
onedtransienthtc_std_cpe4rht.inp	CPE4RHT elements.
onedtransienthtc_std_cpe6mt.inp	CPE6MT elements.
onedtransienthtc_std_cpe6mht.inp	CPE6MHT elements.
onedtransienthtc_std_cps3t.inp	CPS3T elements.
onedtransienthtc_std_cps4t.inp	CPS4T elements.

onedtransienthtc_std_cps4rt.inp	CPS4RT elements.
onedtransienthtc_std_cps6mt.inp	CPS6MT elements.

Fine mesh tests:

nt3xx12f.inp	DC1D2 elements.
nt3xx13f.inp	DC1D3 elements.
nt3xx23f.inp	DC2D3 elements.
nt3xx24f.inp	DC2D4 elements.
nt3xx26f.inp	DC2D6 elements.
nt3xx28f.inp	DC2D8 elements.
nt3xxa3f.inp	DCAX3 elements.
nt3xxa4f.inp	DCAX4 elements.
nt3xxa6f.inp	DCAX6 elements.
nt3xxa8f.inp	DCAX8 elements.
nt3xxs3f.inp	DS3 elements.
nt3xxs4f.inp	DS4 elements.
nt3xxs6f.inp	DS6 elements.
nt3xxs8f.inp	DS8 elements.
onedtransienthtf_std_cax3t.inp	CAX3T elements.
onedtransienthtf_std_cax4rt.inp	CAX4RT elements.
onedtransienthtf_std_cax4rht.inp	CAX4RHT elements.
onedtransienthtf_std_cax6mht.inp	CAX6MHT elements.
onedtransienthtf_std_cax6mt.inp	CAX6MT elements.
onedtransienthtf_std_cpe3t.inp	CPE3T elements.
onedtransienthtf_std_cpe4t.inp	CPE4T elements.
onedtransienthtf_std_cpe4rt.inp	CPE4RT elements.
onedtransienthtf_std_cpe4rht.inp	CPE4RHT elements.
onedtransienthtf_std_cpe6mt.inp	CPE6MT elements.
onedtransienthtf_std_cpe6mht.inp	CPE6MHT elements.
onedtransienthtf_std_cps3t.inp	CPS3T elements.
onedtransienthtf_std_cps4t.inp	CPS4T elements.
onedtransienthtf_std_cps4rt.inp	CPS4RT elements.
onedtransienthtf_std_cps6mt.inp	CPS6MT elements.

Abaqus/Explicit input files

Coarse mesh tests:

onedtransienthtc_xpl_cax3t.inp	CAX3T elements.
onedtransienthtc_xpl_cax4rt.inp	CAX4RT elements.
onedtransienthtc_xpl_cpe3t.inp	CPE3T elements.
onedtransienthtc_xpl_cpe4rt.inp	CPE4RT elements.
onedtransienthtc_xpl_cps3t.inp	CPS3T elements.
onedtransienthtc_xpl_cps4rt.inp	CPS4RT elements.
onedtransienthtc_xpl_s4rt.inp	S4RT elements.

Fine mesh tests:

onedtransienthtf_xpl_cax3t.inp	CAX3T elements.
onedtransienthtf_xpl_cax4rt.inp	CAX4RT elements.
onedtransienthtf_xpl_cpe3t.inp	CPE3T elements.
onedtransienthtf_xpl_cpe4rt.inp	CPE4RT elements.
onedtransienthtf_xpl_cps3t.inp	CPS3T elements.
onedtransienthtf_xpl_cps4rt.inp	CPS4RT elements.
onedtransienthtf_xpl_s3rt.inp	S3RT elements.

T4: Two-dimensional heat transfer with convection

Products: Abaqus/Standard Abaqus/Explicit

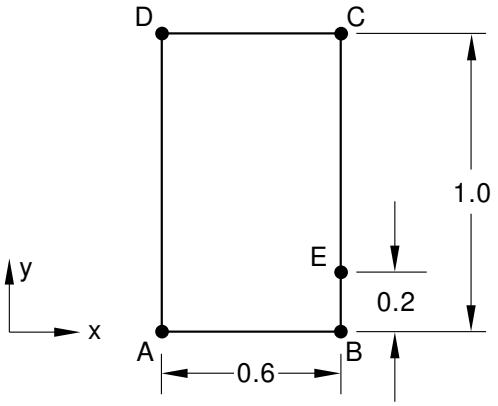
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

DC2D3 DC2D4 DC2D6 DC2D8
DC3D4 DC3D6 DC3D8 DC3D10 DC3D15 DC3D20
CPE3T CPE4RHT CPE4RT CPE6MHT CPE6MT CPS3T CPS4RT CPS6MT C3D4T C3D6PHT C3D6PT C3D6T C3D8RHT C3D8RT C3D8T C3D10MHT C3D10MT
EC3D8RT
SC6RT SC8RT

Problem description



Model: The two-dimensional geometry is shown above. Three-dimensional elements are tested with a thickness of 1.0 in the z-direction. In Abaqus/Standard a steady-state simulation is performed, while in Abaqus/Explicit a transient simulation is performed. The total simulation time in the latter case is 20000 seconds, which provides enough time for the transient solution to reach steady-state conditions in this problem.

Material: Conductivity = 52 W/m°C, surface convective coefficient = 750 W/m^2/°C.

For coupled temperature-displacement elements dummy mechanical properties are used to complete the material definition.

Boundary conditions: Temperature = 100°C along edge AB. Zero heat flux along edge DA. Convection to ambient temperature of 0°C along edges BC and CD.

Loading: Zero internal heat generation.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test T4 from NAFEMS publication TNSB, Rev. 3, "The Standard NAFEMS Benchmarks," October 1990.

Target solution: Temperature of 18.3°C at point E.

Results and discussion

The results for the Abaqus/Standard analysis are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Element	T, Coarse Mesh	T, Fine Mesh
DC2D3	22.40°C (22.4%)	19.22°C (5.0%)
DC2D4	19.34°C (5.7%)	18.91°C (3.3%)
DC2D6	17.99°C (–1.7%)	
DC2D8	17.90°C (–2.2%)	
DC3D4	18.81°C (2.8%)	
DC3D8	19.34°C (5.7%)	18.91°C (3.3%)
DC3D6	22.40°C (22.4%)	18.94°C (3.5%)
DC3D10	19.00°C (3.8%)	
DC3D15	17.99°C (–1.7%)	
DC3D20	17.89°C (–2.2%)	
CPE3T	22.40°C (22.4%)	18.94°C (3.5%)
CPE4RT	19.34°C (5.7%)	18.91°C (3.3%)
CPE4RHT	19.34°C (5.7%)	18.91°C (3.3%)
CPE6MT	17.22°C (5.8%)	
CPE6MHT	17.22°C (5.8%)	
CPS3T	22.40°C (22.4%)	18.94°C (3.5%)
CPS4RT	19.34°C (5.7%)	18.91°C (3.3%)
CPS6MT	17.22°C (5.8%)	
C3D4T	19.26°C (5.2%)	
C3D6T	22.40°C (22.4%)	18.94°C (3.5%)
C3D8T	19.34°C (5.7%)	18.91°C (3.3%)
C3D8RT	19.34°C (5.7%)	18.91°C (3.3%)
C3D8RHT	19.34°C (5.7%)	18.91°C (3.3%)
C3D10MT	19.79°C (8.1%)	
C3D10MHT	19.79°C (8.1%)	

The results for the Abaqus/Explicit analysis are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Element	T, Coarse Mesh	T, Fine Mesh
CPE3T	22.26°C (21.6%)	18.90°C (3.3%)

Element	T, Coarse Mesh	T, Fine Mesh
CPE4RT	19.29°C (5.4%)	18.87°C (3.1%)
CPS3T	22.36°C (22.2%)	18.90°C (3.3%)
CPS4RT	19.29°C (5.4%)	18.87°C (3.1%)
C3D4T	19.21°C (5.0%)	
C3D6T	22.36°C (22.2%)	18.90°C (3.3%)
C3D8RT	19.29°C (5.4%)	18.87°C (3.1%)
C3D8T	19.29°C (5.4%)	18.87°C (3.1%)
SC6RT	22.36°C (22.2%)	18.90°C (3.3%)
SC8RT	19.29°C (5.4%)	18.87°C (3.1%)
EC3D8RT	19.29°C (5.4%)	18.87°C (3.1%)

Input files

Abaqus/Standard input files

Coarse mesh tests:

nt4xx23c.inp	DC2D3 elements.
nt4xx24c.inp	DC2D4 elements.
nt4xx26c.inp	DC2D6 elements.
nt4xx28c.inp	DC2D8 elements.
nt4xx34c.inp	DC3D4 elements.
nt4xx36c.inp	DC3D6 elements.
nt4xx38c.inp	DC3D8 elements.
nt4xx3ac.inp	DC3D10 elements.
nt4xx3fc.inp	DC3D15 elements.
nt4xx3kc.inp	DC3D20 elements.
twodheattrconvecc_std_cpe3t.inp	CPE3T elements.
twodheattrconvecc_std_cpe4rt.inp	CPE4RT elements.
twodheattrconvecc_std_cpe4rht.inp	CPE4RHT elements.
twodheattrconvecc_std_cpe6mt.inp	CPE6MT elements.
twodheattrconvecc_std_cpe6mht.inp	CPE6MHT elements.
twodheattrconvecc_std_cps3t.inp	CPS3T elements.
twodheattrconvecc_std_cps4rt.inp	CPS4RT elements.
twodheattrconvecc_std_cps6mt.inp	CPS6MT elements.
twodheattrconvecc_std_c3d4t.inp	C3D4T elements.
twodheattrconvecc_std_c3d6pht.inp	C3D6PHT elements.
twodheattrconvecc_std_c3d6pt.inp	C3D6PT elements.
twodheattrconvecc_std_c3d6t.inp	C3D6T elements.
twodheattrconvecc_std_c3d8t.inp	C3D8T elements.
twodheattrconvecc_std_c3d8rt.inp	C3D8RT elements.
twodheattrconvecc_std_c3d8rht.inp	C3D8RHT elements.
twodheattrconvecc_std_c3d10mt.inp	C3D10MT elements.

twodheattrconvecc_std_c3d10mht.inp	C3D10MHT elements.

Fine mesh tests:

nt4xx23f.inp	DC2D3 elements.
nt4xx24f.inp	DC2D4 elements.
nt4xx36f.inp	DC3D6 elements.
nt4xx38f.inp	DC3D8 elements.
twodheattrconvecf_std_cpe3t.inp	CPE3T elements.
twodheattrconvecf_std_cpe4rt.inp	CPE4RT elements.
twodheattrconvecf_std_cpe4rht.inp	CPE4RHT elements.
twodheattrconvecf_std_cps3t.inp	CPS3T elements.
twodheattrconvecf_std_cps4rt.inp	CPS4RT elements.
twodheattrconvecf_std_c3d6t.inp	C3D6T elements.
twodheattrconvecf_std_c3d8t.inp	C3D8T elements.
twodheattrconvecf_std_c3d8rt.inp	C3D8RT elements.
twodheattrconvecf_std_c3d8rht.inp	C3D8RHT elements.

Abaqus/Explicit input files

Coarse mesh tests:

twodheattrconvecc_xpl_cpe3t.inp	CPE3T elements.
twodheattrconvecc_xpl_cpe4rt.inp	CPE4RT elements.
twodheattrconvecc_xpl_cps3t.inp	CPS3T elements.
twodheattrconvecc_xpl_cps4rt.inp	CPS4RT elements.
twodheattrconvecc_xpl_c3d4t.inp	C3D4T elements.
twodheattrconvecc_xpl_c3d6t.inp	C3D6T elements.
twodheattrconvecc_xpl_c3d8rt.inp	C3D8RT elements.
twodheattrconvecc_xpl_ec3d8rt.inp	EC3D8RT elements.
twodheattrconvecc_xpl_c3d8t.inp	C3D8T elements.
twodheattrconvecc_xpl_sc6rt.inp	SC6RT elements.
twodheattrconvecc_xpl_sc8rt.inp	SC8RT elements.

Fine mesh tests:

twodheattrconvecf_xpl_cpe3t.inp	CPE3T elements.
twodheattrconvecf_xpl_cpe4rt.inp	CPE4RT elements.
twodheattrconvecf_xpl_cps3t.inp	CPS3T elements.
twodheattrconvecf_xpl_cps4rt.inp	CPS4RT elements.
twodheattrconvecf_xpl_c3d6t.inp	C3D6T elements.
twodheattrconvecf_xpl_c3d8rt.inp	C3D8RT elements.
twodheattrconvecf_xpl_c3d8t.inp	C3D8T elements.
twodheattrconvecf_xpl_ec3d8rt.inp	EC3D8RT elements.
twodheattrconvecf_xpl_sc6rt.inp	SC6RT elements.
twodheattrconvecf_xpl_sc8rt.inp	SC8RT elements.

Standard benchmarks: free vibration tests

In this section:

- *FV2: Pin-ended double cross: in-plane vibration*
- *FV4: Cantilever with off-center point masses*
- *FV12: Free thin square plate*
- *FV15: Clamped thin rhombic plate*
- *FV16: Cantilevered thin square plate*
- *FV22: Clamped thick rhombic plate*
- *FV32: Cantilevered tapered membrane*
- *FV41: Free cylinder: axisymmetric vibration*
- *FV42: Thick hollow sphere: uniform radial vibration*
- *FV52: Simply supported “solid” square plate*

FV2: Pin-ended double cross: in-plane vibration

Products: Abaqus/Standard

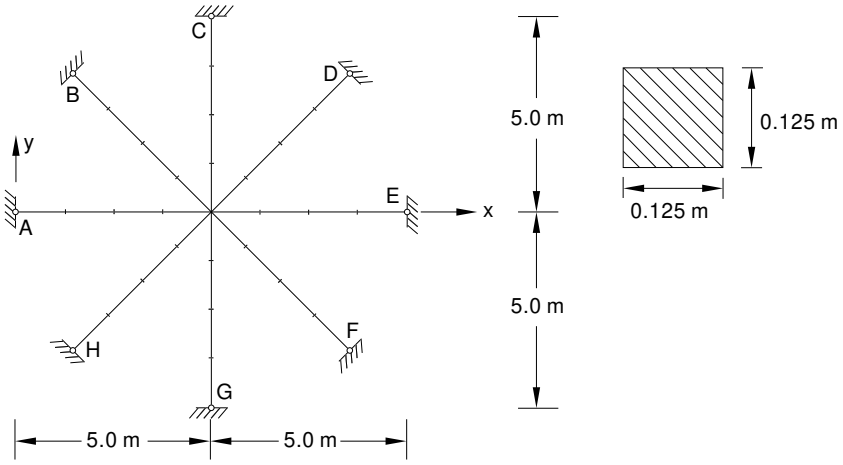
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

B22 B23

Problem description



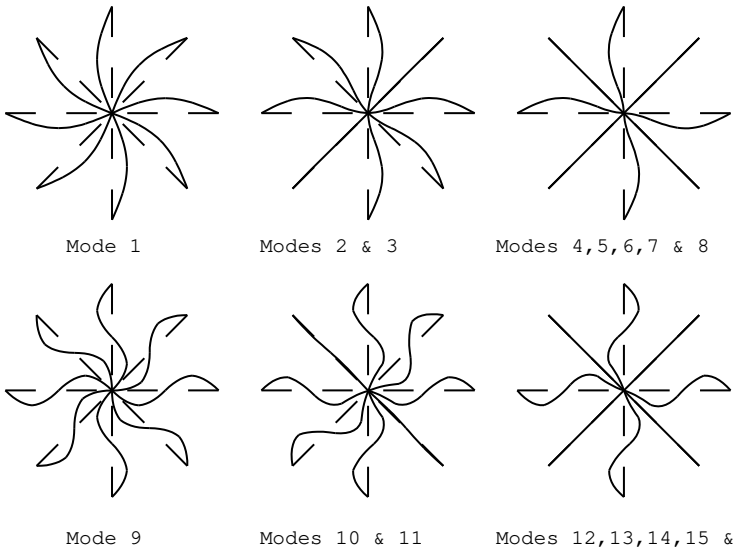
Material: Young's modulus = 200 GPa, Poisson's ratio = 0.3, density = 8000 kg/m^3.

Boundary conditions: Beams are pinned at A, B, C, D, E, F, G, H.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test FV2 from NAFEMS publication TNSB, Rev. 3, “The Standard NAFEMS Benchmarks,” October 1990.

Mode shapes predicted by Abaqus



Modes 2 through 8 are vibration modes with the same eigenvalues. Because the eigenvalues are identical, any linear combination of modes 2 through 8 is still a valid mode. Hence, the shapes of these modes are arbitrary linear combinations of the mode shapes shown in the figure, and in particular will vary from computer to computer. The same behavior is observed for modes 10 through 16.

Results and discussion

The results are shown in *Table 1* and *Table 2*. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Table 1: B22 and B23 Modes 1–8 element results.

	Mode							
	1	2	3	4	5	6	7	8
NAFEMS	11.336	17.709	17.709	17.709	17.709	17.709	17.709	17.709
B22	11.337 (0.01)	17.676 (–0.19)	17.676 (–0.19)	17.699 (–0.06)	17.706 (–0.02)	17.707 (–0.01)	17.707 (–0.01)	17.712 (0.02)
B23	11.343 (0.06)	17.697 (–0.07)	17.697 (–0.07)	17.721 (0.07)	17.721 (0.07)	17.728 (0.11)	17.728 (0.11)	17.733 (0.14)

Table 2: B22 and B23 Modes 9–16 element results.

	Mode							
	9	10	11	12	13	14	15	16
NAFEMS	45.345	57.390	57.390	57.390	57.390	57.390	57.390	57.390
B22	45.686 (0.75)	57.745 (0.62)	57.745 (0.62)	58.059 (1.17)	58.080 (1.20)	58.081 (1.20)	58.081 (1.20)	58.101 (1.24)
B23	45.544 (0.44)	57.457 (0.12)	57.457 (0.12)	57.759 (0.64)	58.780 (2.42)	58.781 (2.42)	58.781 (2.42)	58.801 (2.46)

Input files

nfv2x22x.inp B22 elements.

nfv2x23x.inp	B23 elements.

FV4: Cantilever with off-center point masses

Products: Abaqus/Standard

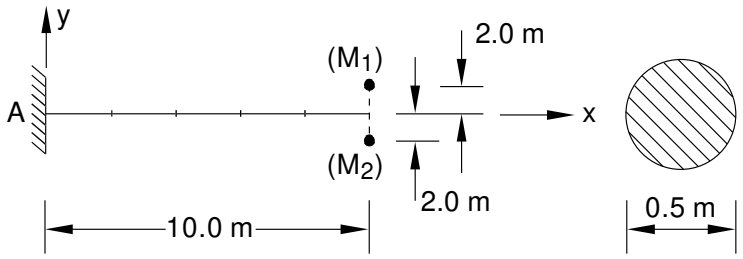
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

B32 B33

Problem description



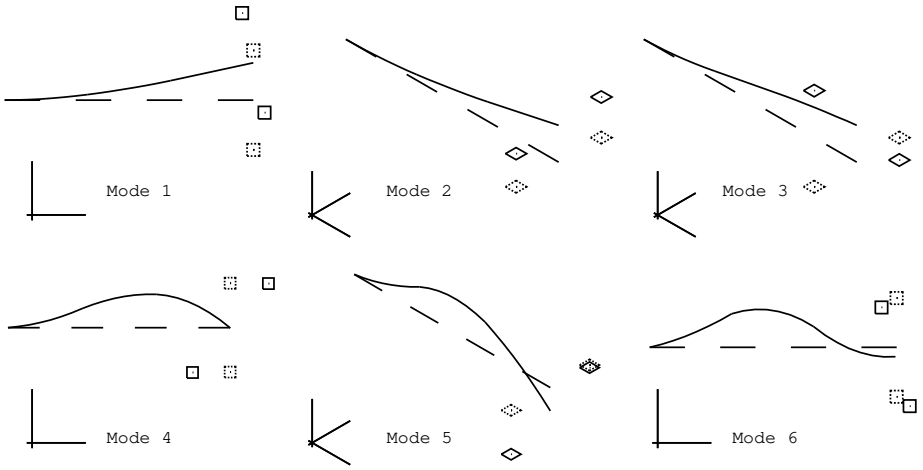
Material: Young's modulus = 200 GPa, Poisson's ratio = 0.3, density = 8000 kg/m^3.

Boundary conditions: All displacements are zero at A.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test FV4 from NAFEMS publication TNSB, Rev. 3, “The Standard NAFEMS Benchmarks,” October 1990.

Mode shapes predicted by Abaqus



(Dashed lines represent the undeformed configuration.)

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

	Mode					
	1	2	3	4	5	6
NAFEMS	1.723	1.727	7.413	9.972	18.155	26.957
B32	1.722 (–0.06)	1.726 (–0.06)	7.412 (–0.01)	9.955 (–0.17)	18.174 (0.10)	27.053 (0.36)
B33	1.723 (0.00)	1.727 (0.00)	7.414 (0.01)	9.975 (0.03)	18.187 (0.18)	27.001 (0.16)

Input files

nfv4x32x.inp	B32 elements.
nfv4x33x.inp	B33 elements.

FV12: Free thin square plate

Products: Abaqus/Standard

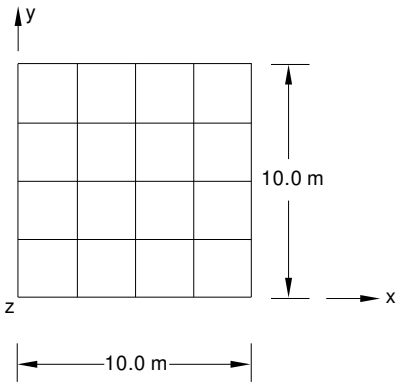
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CSS8 S3R S4 S4R S4R5 S8R STRI65 SC6R SC8R

Problem description



Model: Plate thickness = 0.05 m.

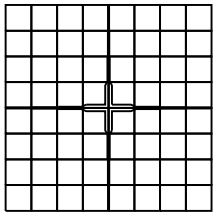
Material: Young's modulus = 200 GPa, Poisson's ratio = 0.3, density = 8000 kg/m^3.

Boundary conditions: $u_x = u_y = \phi_z = 0$ at all nodes. The continuum shell meshes use equation constraints to provide equivalent midsurface kinematic constraints.

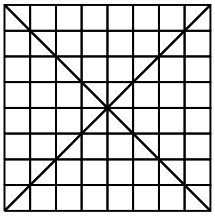
Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test FV12 from NAFEMS publication TNSB, Rev. 3, “The Standard NAFEMS Benchmarks,” October 1990.

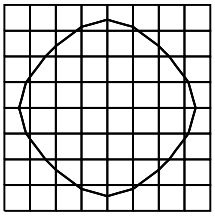
Mode shapes predicted by Abaqus (for element type S4R5)



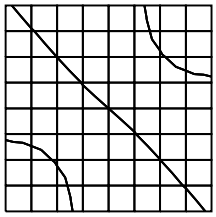
mode 4



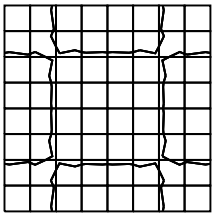
mode 5



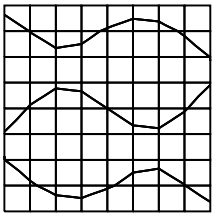
mode 6



mode 7



mode 9



mode 10

The contour plots were generated by setting the maximum and minimum contour levels close to zero. Where contour levels coincided with the element boundaries, the maximum contour level was increased and the minimum contour level was decreased appropriately.

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

	Mode							
	1, 2, 3	**4**	**5**	**6**	**7**	**8**	**9**	**10**
NAFEMS	RBM	1.622	2.360	2.922	4.233	4.233	7.416	N/A
S3R	RBM	1.64 (1.1)	2.40 (1.7)	3.00 (2.7)	4.38 (3.5)	4.47 (5.6)	7.98 (7.6)	7.98
S4	RBM	1.633 (0.68)	2.403 (1.82)	3.007 (2.91)	4.286 (1.25)	4.286 (1.25)	7.973 (7.55)	8.032
S4R	RBM	1.631 (0.55)	2.403 (1.82)	3.007 (2.91)	4.265 (0.76)	4.265 (0.76)	7.880 (6.26)	8.028
S4R5	RBM	1.622 (0.00)	2.362 (0.08)	2.931 (0.31)	4.131 (−2.41)	4.131 (−2.41)	7.875 (6.19)	7.875
S8R	RBM	1.626 (0.25)	2.369 (0.38)	2.935 (0.44)	4.185 (−1.13)	4.185 (−1.13)	7.559 (1.93)	7.559
STR165 (2×4×4 mesh)	RBM	1.628 (0.37)	2.364 (0.17)	2.931 (0.31)	4.229 (−0.09)	4.265 (−0.76)	7.490 (1.00)	7.533
SC6R	RBM	1.644 (1.4)	2.4 (1.7)	3.0 (2.7)	4.3 (1.6)	4.47 (5.6)	7.97 (7.7)	7.98
SC8R	RBM	1.63 (0.5)	2.4 (1.7)	3.0 (2.7)	4.26 (0.6)	4.26 (0.6)	7.8 (6.1)	8.02
SC8R*	RBM	1.635 (0.8)	2.403 (1.8)	3.008 (2.9)	4.308 (1.7)	4.308 (1.7)	8.037 (8.3)	8.037
CSS8	RBM	1.587 (2.2)	2.253 (4.5)	2.830 (3.1)	3.974 (6.1)	3.974 (6.1)	7.024 (5.3)	7.416
*Abaqus/Standard results with enhanced hourglass control.								

Input files

nfv12_std_s3r.inp	S3R elements.
nfv12e4x.inp	S4 elements.
nfv12f4x.inp	S4R elements.
nfv1254x.inp	S4R5 elements.
nfv1268x.inp	S8R elements.
nfv1256x.inp	STRI65 elements.
nfv12_std_sc6r.inp	SC6R elements.
nfv12_std_sc8r.inp	SC8R elements.
nfv12_std_sc8r_eh.inp	SC8R elements with enhanced hourglass control.
nfv12_std_css8.inp	CSS8 elements.

FV15: Clamped thin rhombic plate

Products: Abaqus/Standard

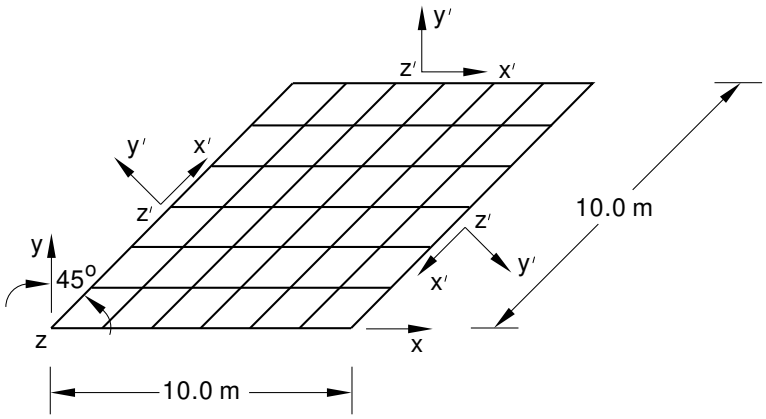
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

S8R5 STRI65

Problem description



Material: Young's modulus = 200 GPa, Poisson's ratio = 0.3, density = 8000 kg/m^3.

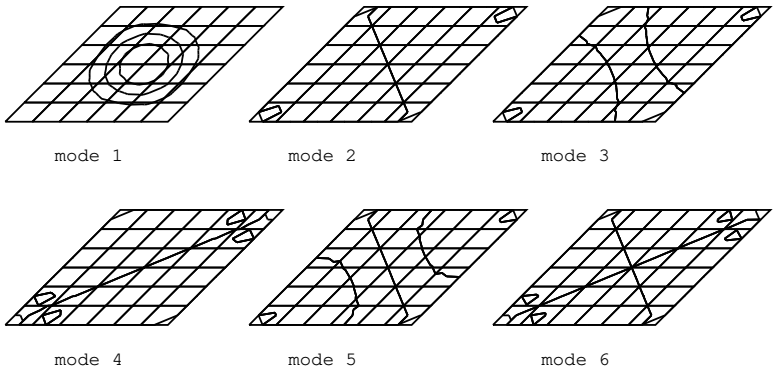
Boundary conditions: $u_x = u_y = \phi_z = 0$ at all nodes. $v_z = \phi_x = \phi_y = 0$ along all edges.

General: Gauss integration is used for the shell cross-section for element STRI65.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test FV15 from NAFEMS publication TNSB, Rev. 3, “The Standard NAFEMS Benchmarks,” October 1990.

Mode shapes predicted by Abaqus



The contour plots were generated by setting the maximum and minimum contour levels close to zero. Where contour levels coincided with the element boundaries, the maximum contour level was increased and the minimum contour level was decreased appropriately.

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

	Mode					
	1	2	3	4	5	6
NAFEMS	7.938	12.835	17.941	19.133	24.009	27.922
S8R5	7.902 (–0.45)	12.884 (0.38)	18.027 (0.48)	18.989 (–0.75)	24.119 (0.46)	28.234 (1.12)
STR165	7.935 (–0.04)	12.804 (–0.24)	18.095 (0.86)	19.267 (–0.70)	24.428 (1.75)	27.945 (0.08)

Input files

nfv1558x.inp	S8R5 elements.
nfv1556x.inp	STRI65 elements.

FV16: Cantilevered thin square plate

Products: Abaqus/Standard

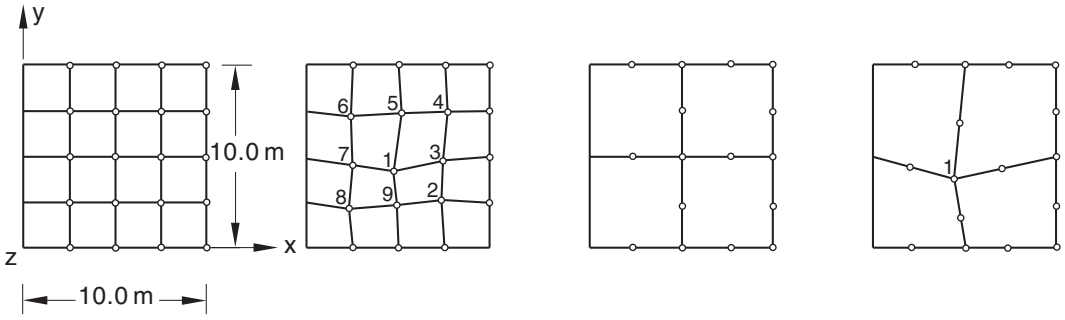
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

S8R S8R5 STRI65

Problem description



- Main degree of freedom (in Z-direction)

Node	Coordinates	
Numbers	x	y
1	4.00	4.00
2	2.25	2.25
3	4.75	2.50
4	7.25	2.75
5	7.50	4.75
6	7.75	7.25
7	5.25	7.25
8	2.25	7.25
9	2.50	4.75

Model: Plate thickness = 0.05 m.

Material: Young's modulus = 200 GPa, Poisson's ratio = 0.3, density = 8000 kg/m^3.

Boundary conditions: $u_x = u_y = u_z = \phi_y = 0$ along the *y*-axis.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test FV16 from NAFEMS publication TNSB, Rev. 3, "The Standard NAFEMS Benchmarks," October 1990.

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

	Mode					
	1	2	3	4	5	6
NAFEMS	0.421	1.029	2.582	3.306	3.753	6.555
STRI65 (Test 1)	0.418 (–0.71)	1.035 (0.58)	2.565 (–0.66)	3.325 (0.57)	3.870 (3.12)	6.966 (6.27)
STRI65 (Test 2)	0.420 (–0.24)	1.034 (0.49)	2.570 (–0.46)	3.326 (0.60)	3.850 (2.58)	6.843 (4.39)
S8R	0.491	1.025	2.583	3.330	3.742	7.278
(Test 1)	(–0.48)	(–0.39)	(0.04)	(0.73)	(–0.29)	(11.03)
S8R	0.423	1.044	2.634	3.345	4.105	7.048
(Test 2)	(0.48)	(1.46)	(2.01)	(1.18)	(9.38)	(7.52)
S8R	0.419	1.028	2.702	3.429	3.941	6.696
(Test 3)	(–0.48)	(–0.10)	(4.65)	(3.72)	(5.01)	(2.15)
S8R	0.420	1.049	2.707	3.723	4.128	7.008
(Test 4)	(–0.24)	(1.94)	(4.84)	(12.58)	(9.99)	(6.91)
S8R5 (Test 1)	0.418 (–0.71)	1.023 (–0.58)	2.569 (–0.50)	3.282 (–0.73)	3.721 (–0.85)	5.988 (–8.65)
S8R5 (Test 2)	0.418 (–0.71)	1.024 (–0.49)	2.569 (–0.50)	3.283 (–0.70)	3.722 (–0.83)	6.077 (–7.29)
S8R5 (Test 3)	0.418 (–0.71)	1.021 (–0.78)	2.661 (3.06)	3.382 (2.30)	3.946 (5.14)	6.814 (3.95)
S8R5 (Test 4)	0.419 (0.24)	1.022 (–0.68)	2.630 (1.86)	3.560 (1.63)	3.920 (4.45)	6.602 (0.72)

Input files

nfv16561.inp	STRI65 elements, Test 1.
nfv16562.inp	STRI65 elements, Test 2.
nfv16681.inp	S8R elements, Test 1.
nfv16682.inp	S8R elements, Test 2.
nfv16683.inp	S8R elements, Test 3.
nfv16684.inp	S8R elements, Test 4.
nfv16581.inp	S8R5 elements, Test 1.
nfv16582.inp	S8R5 elements, Test 2.
nfv16583.inp	S8R5 elements, Test 3.
nfv16584.inp	S8R5 elements, Test 4.

FV22: Clamped thick rhombic plate

Products: Abaqus/Standard

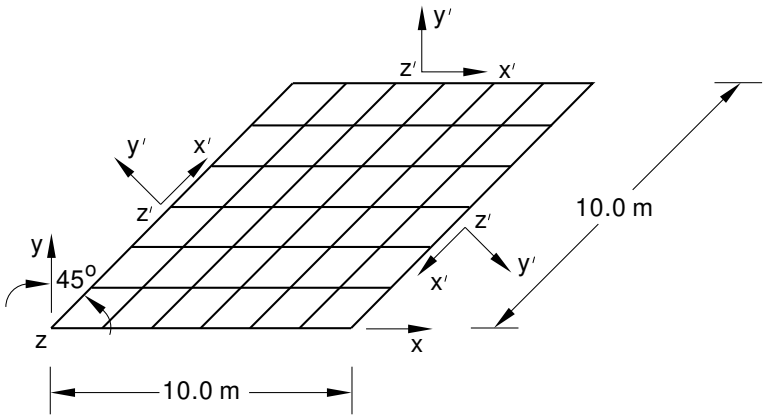
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

S3R S4 S4R S8R SC6R SC8R

Problem description



Model: Plate thickness = 1.0 m.

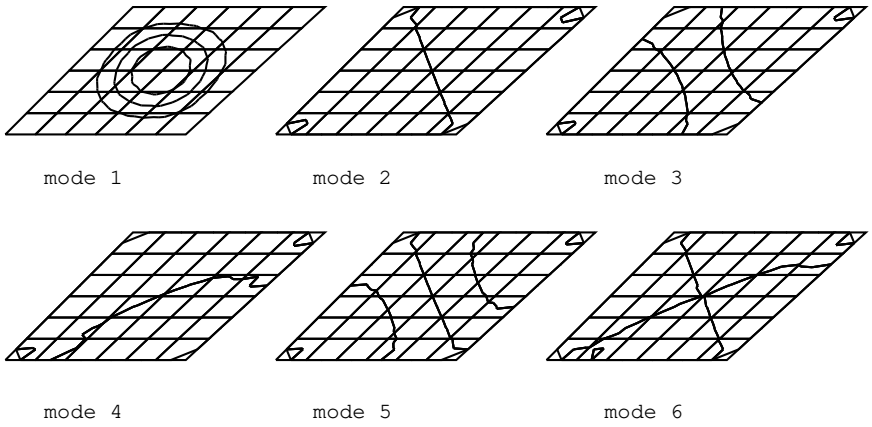
Material: Young's modulus = 200 GPa, Poisson's ration = 0.3, density = 8000 kg/m^3.

Boundary conditions: $u_x = u_y = \phi_z = 0$ at all nodes. $v_z = \phi_x = \phi_y = 0$ along all edges. The continuum shell meshes use equation constraints to provide equivalent midsurface kinematic constraints.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test FV22 from NAFEMS publication TNSB, Rev. 3, "The Standard NAFEMS Benchmarks," October 1990.

Mode shapes predicted by Abaqus



The contour plots were generated by setting the maximum and minimum contour levels close to zero. Where contour levels coincided with the element boundaries, the maximum contour level was increased and the minimum contour level was decreased appropriately.

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

	Mode					
	1	2	3	4	5	6
NAFEMS	133.95	201.41	265.81	282.74	334.45	N.A.
S3R	143.11 (6.8)	232.33 (15.4)	307.61 (15.7)	333.37 (17.9)	446.63 (33.6)	456.72
S4	136.50 (1.9)	215.60 (7.04)	292.01 (9.86)	295.96 (4.68)	380.96 (13.91)	423.86
S4R	135.50 (1.16)	213.61 (6.06)	287.25 (8.07)	293.99 (3.98)	372.50 (11.38)	421.20
S8R	132.28 (1.25)	199.67 (0.86)	265.78 (0.01)	281.67 (0.38)	338.72 (1.28)	382.09
SC6R	144.18 (7.6)	235.09 (16.7)	312.44 (17.5)	338.54 (19.7)	455.64 (36.2)	465.54
SC8R	136.44 (1.9)	216.00 (7.2)	291.27 (9.58)	298.43 (5.5)	378.66 (13.2)	429.43

Input files

nfv22_std_s3r.inp	S3R elements.
nfv22e4f.inp	S4 elements.
nfv22f4f.inp	S4R elements.
nfv2268c.inp	S8R elements.
nfv22_std_sc6r.inp	SC6R elements.
nfv22_std_sc8r.inp	SC8R elements.

FV32: Cantilevered tapered membrane

Products: Abaqus/Standard

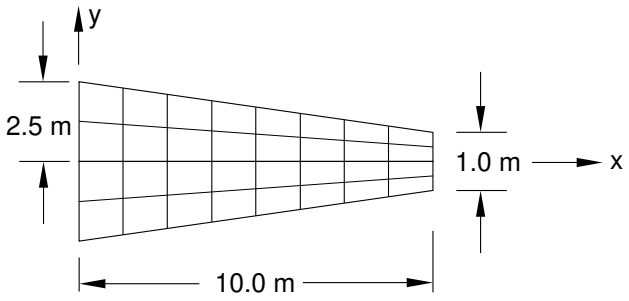
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CPS4 CPS4I CPS8 CPS8R CPS6 CPS6M

Problem description



Model: Plate thickness = 0.05 m.

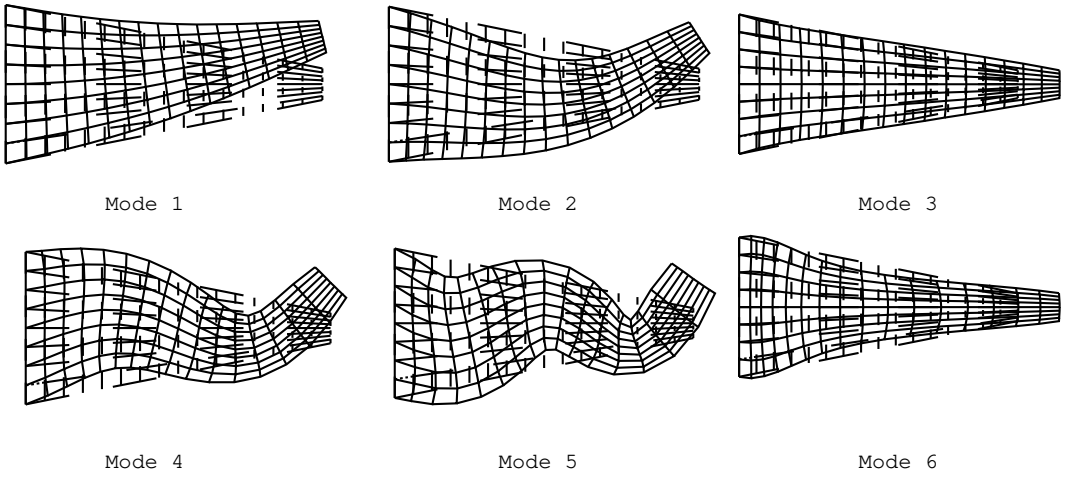
Material: Young's modulus = 200 GPa, Poisson's ratio = 0.3, density = 8000 kg/m^3.

Boundary conditions: $u_x = u_y = 0$ along the *y*-axis. $u_z = 0$ at all nodes.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test FV32 from NAFEMS publication TNSB, Rev. 3, “The Standard NAFEMS Benchmarks,” October 1990.

Mode shapes predicted by Abaqus (for element type CPS4)



Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

	Mode					
	1	2	3	4	5	6
NAFEMS	44.623	130.03	162.70	246.05	379.90	391.44
CPS4	44.782 (0.36)	130.63 (0.46)	162.59 (–0.07)	246.79 (0.30)	379.14 (–0.20)	389.83 (–0.41)
CPS4I	44.524 (–0.23)	129.55 (–0.09)	162.55 (–0.09)	244.13 (–0.78)	374.46 (–1.43)	389.60 (–0.47)
CPS8	44.636 (0.04)	130.14 (0.08)	162.72 (0.01)	246.63 (0.24)	382.02 (0.56)	391.55 (0.03)
CPS8R	44.629 (0.02)	130.11 (0.06)	162.70 (0.00)	246.42 (0.15)	381.32 (0.37)	391.51 (0.02)
CPS6	44.624 (0.00)	130.04 (0.00)	162.70 (0.00)	246.09 (0.02)	379.99 (0.02)	391.45 (0.02)
CPS6M	44.637 (0.03)	129.88 (–0.12)	162.67 (–0.02)	245.29 (–0.31)	377.64 (–0.59)	390.98 (–0.12)

Input files

nfv32f4f.inp	CPS4 elements.
nfv32i4f.inp	CPS4I elements.
nfv32f8c.inp	CPS8 elements.
nfv32r8c.inp	CPS8R elements.
nfv32f6c.inp	CPS6 elements.
nfv32m6c.inp	CPS6M elements.

FV41: Free cylinder: axisymmetric vibration

Products: Abaqus/Standard

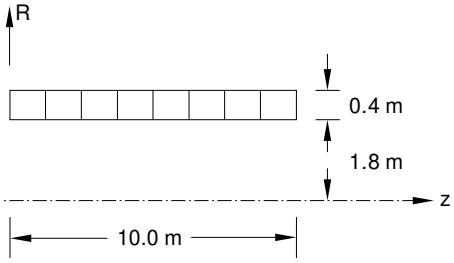
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CAX4 CAX4I CAX6 CAX6M CAX8

Problem description



Model: Wall thickness = 0.4 m.

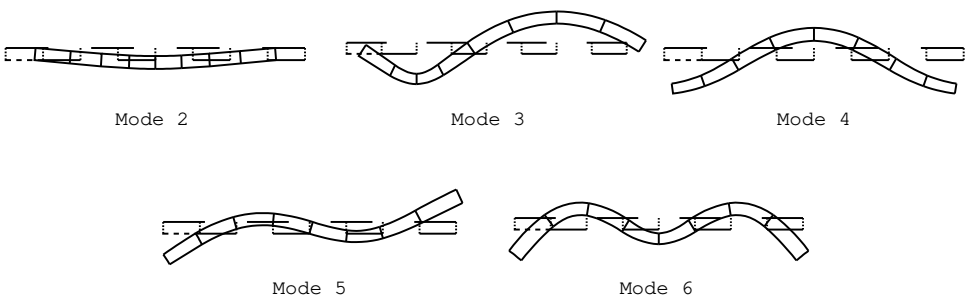
Material: Young's modulus = 200 GPa, Poisson's ration = 0.3, density = 8000 kg/m^3.

Boundary conditions: Unsupported.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test FV41 from NAFEMS publication TNSB, Rev. 3, “The Standard NAFEMS Benchmarks,” October 1990.

Mode shapes predicted by Abaqus



Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

	Mode					
	1	2	3	4	5	6
NAFEMS	RBM	243.53	377.41	394.11	397.72	405.28
CAX4	RBM	243.05 (–0.20)	368.41 (–2.38)	378.05 (–4.08)	384.00 (–3.45)	389.02 (–4.01)
CAX4I	RBM	243.17 (–0.15)	370.80 (–1.75))	379.28 (–3.76)	385.92 (–2.06)	389.54 (–3.88)
CAX6	RBM	243.50 (–0.01)	377.41 (0.00)	394.26 (0.04)	397.90 (0.05)	406.42 (0.28)
CAX6M	RBM	243.37 (–0.07)	376.19 (–0.32)	392.80 (–0.33)	394.48 (–0.81)	399.76 (–1.36)
CAX8	RBM	243.50 (–0.01)	377.46 (0.01)	394.30 (0.05)	397.97 (0.06)	406.44 (0.29)

Remarks

In comparison to element types CAX6 and CAX8, element types CAX4, CAX4I, and CAX6M require a greater mesh refinement to capture the higher modes accurately.

Input files

nfv41f4f.inp	CAX4 elements.
nfv41i4f.inp	CAX4I elements.
nfv41f6f.inp	CAX6 elements.
nfv41m6f.inp	CAX6M elements.
nfv41f8c.inp	CAX8 elements.

FV42: Thick hollow sphere: uniform radial vibration

Products: Abaqus/Standard

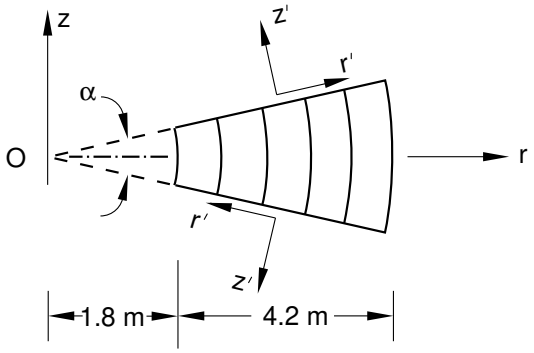
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CAX4 CAX8 CAX6 CAX6M

Problem description



Model: Shell thickness = 4.2 m.

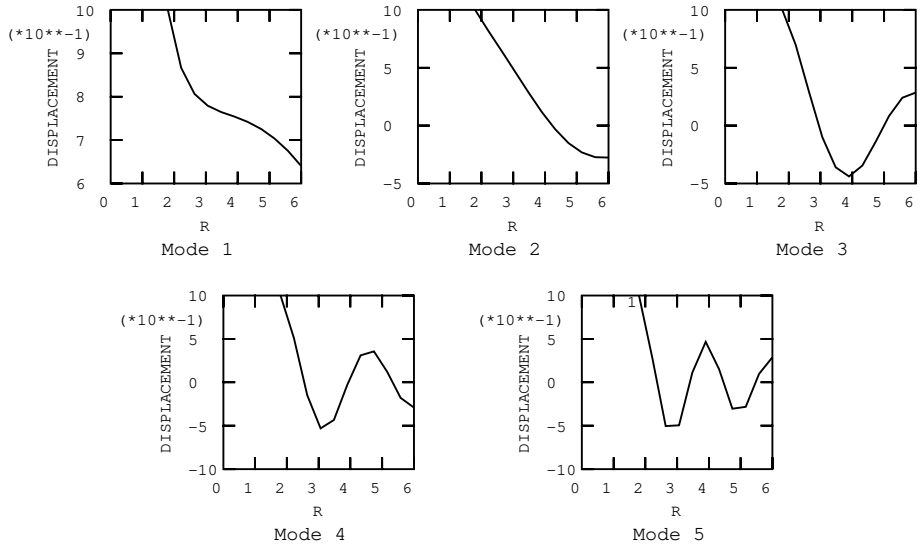
Material: Young's modulus = 200 GPa, Poisson's ratio = 0.3, density = 8000 kg/m^3.

Boundary conditions: Unsupported.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test FV42 from NAFEMS publication TNSB, Rev. 3, “The Standard NAFEMS Benchmarks,” October 1990.

Radial displacement; Mode shapes predicted by Abaqus (for element type CAX8)



Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

	Mode				
	1	2	3	4	5
NAFEMS	369.91	838.03	1451.2	2111.7	2795.8
CAX4	367.79 (–0.03)	829.11 (–1.06)	1417.6 (–2.32)	2025.8 (–4.07)	2598.4 (7.60)
CAX8	370.97 (0.29)	840.49 (0.29)	1457.3 (0.42)	2137.9 (0.99)	2861.1 (2.33)
CAX6	369.95 (0.01)	838.04 (0.00)	1451.3 (0.00)	2118.0 (0.30)	2800.2 (0.16)
CAX6M	370.06 (0.04)	836.50 (–0.18)	1443.0 (–0.57)	2092.1 (–0.93)	2738.8 (–2.04)

Input files

nfv4264f.inp	CAX4 elements.
nfv4268c.inp	CAX8 elements.
nfv4266c.inp	CAX6 elements.
nfv4266m.inp	CAX6M elements.

FV52: Simply supported “solid” square plate

Products: Abaqus/Standard

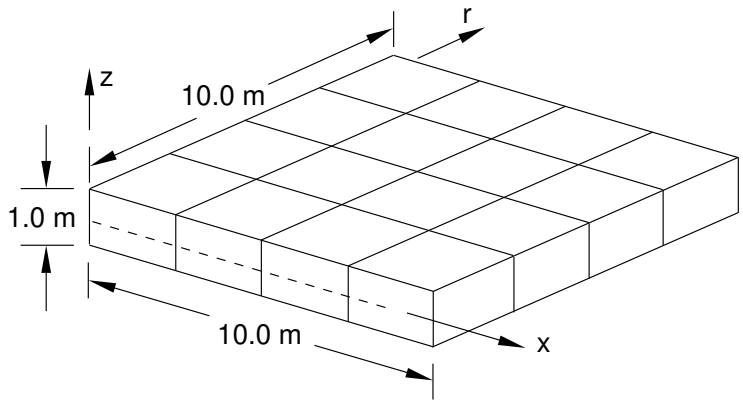
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

C3D8I C3D10 C3D10HS C3D10M C3D20

Problem description



Model: Plate thickness = 1.0 m.

Material: Young's modulus = 200 GPa, Poisson's ratio = 0.3, density = 8000 kg/m^3.

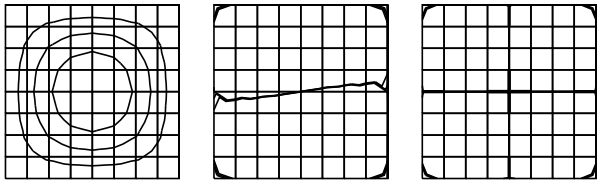
Boundary conditions: $u_z = 0$ along all four edges on the plane z =−0.5.

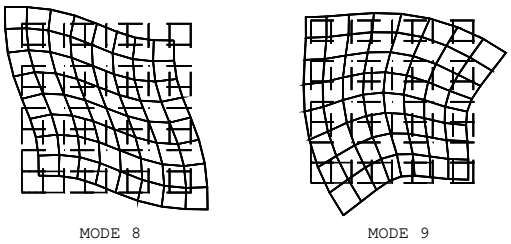
Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test FV52 from NAFEMS publication TNSB, Rev. 3, “The Standard NAFEMS Benchmarks,” October 1990.

Mode shapes predicted by Abaqus (for element type C3D8I)

The following contour plots were generated by setting the maximum and minimum contour levels close to zero. Where contour levels coincided with the element boundaries, the maximum contour level was increased and the minimum contour level was decreased appropriately.





Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

	Mode							
	1, 2, 3	**4**	**5**	**6**	**7**	**8**	**9**	**10**
NAFEMS	RBM	44.092	106.66	106.66	156.23	193.58	200.13	200.13
C3D8I	RBM	44.092 (0.0)	106.66 (0.0)	106.66 (0.0)	156.23 (0.0)	193.58 (0.0)	200.13 (0.0)	200.13 (0.0)
C3D10	RBM	44.348 (0.58)	107.73 (1.00)	107.73 (1.00)	163.58 (4.70)	193.63 (0.02)	204.74 (2.30)	205.10 (2.48)
C3D10HS	RBM	44.348 (0.58)	107.73 (1.00)	107.73 (1.00)	163.58 (4.70)	193.63 (0.02)	204.74 (2.30)	205.10 (2.48)
C3D10M	RBM	42.687 (–3.19)	101.57 (–4.77)	101.57 (–4.77)	151.22 (–3.21)	192.89 (–0.35) (Mode 10)	203.76 (1.81) (Mode 11)	203.76 (1.81) (Mode 12)
C3D20	RBM	44.796 (1.60)	110.54 (3.64)	110.54 (3.64)	169.10 (8.24)	193.92 (0.18)	206.64 (3.25)	206.64 (3.25)

Remarks

Element types C3D10, C3D10M, and C3D20 capture the same eigenmodes, but the order of eigenmodes 8 through 12 is different. For example, the same mode is captured as mode 12 by C3D10, as mode 11 by C3D10M, and as mode 9 by C3D20. Element type C3D8I captures the eigenmodes in the same order as C3D20.

Input files

nfv52i8f.inp	C3D8I elements.
nfv52f10.inp	C3D10 elements.
nfv52i10.inp	C3D10HS elements.
nfv52m10.inp	C3D10M elements.
nfv52fkc.inp	C3D20 elements.

Proposed forced vibration benchmarks

In this section:

- *Test 5: Deep simply supported beam: frequency extraction*
- *Test 5H: Deep simply supported beam: harmonic forced vibration*
- *Test 5T: Deep simply supported beam: transient forced vibration*
- *Test 5R: Deep simply supported beam: random forced vibration*
- *Test 13: Simply supported thin square plate: frequency extraction*
- *Test 13H: Simply supported thin square plate: harmonic forced vibration*
- *Test 13T: Simply supported thin square plate: transient forced vibration*
- *Test 13R: Simply supported thin square plate: random forced vibration*
- *Test 21: Simply supported thick square plate: frequency extraction*
- *Test 21H: Simply supported thick square plate: harmonic forced vibration*
- *Test 21T: Simply supported thick square plate: transient forced vibration*
- *Test 21R: Simply supported thick square plate: random forced vibration*

Test 5: Deep simply supported beam: frequency extraction

Products: Abaqus/Standard

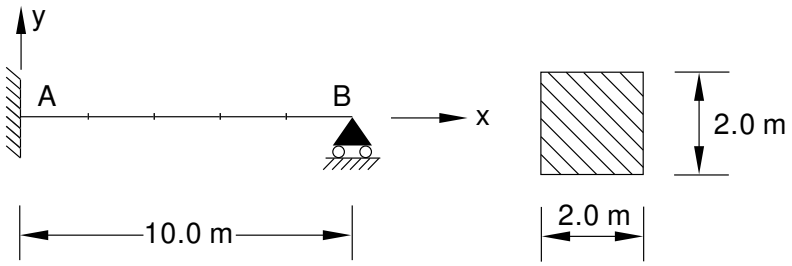
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

B32

Problem description



Material: Young's modulus = 200 GPa, Poisson's ratio = 0.3, density = 8000 kg/m^3.

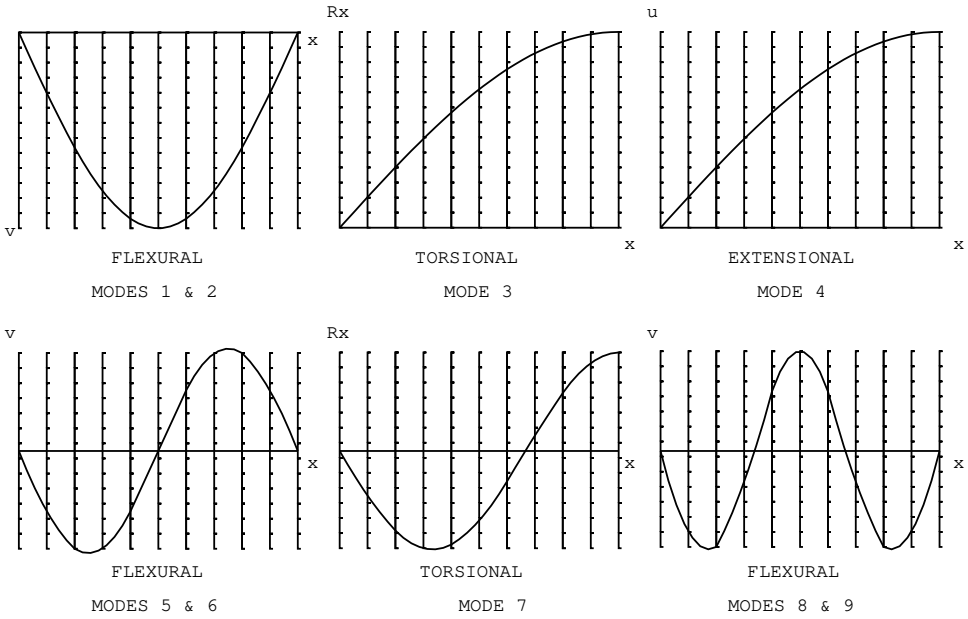
Boundary conditions: $u_x = u_y = u_z = \phi_x = 0$ at A, $u_y = u_z = 0$ at B.

Frequency extraction is performed in Step 1.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 5 from NAFEMS “Selected Benchmarks for Forced Vibration,” R0016, March 1993.

Mode shapes predicted by Abaqus



Results and discussion

The results are shown in the following table.

Mode	Abaqus result	NAFEMS	% Difference
		reference result	
1	42.658	42.650	0.02
2	42.658	42.650	0.02
3	71.261	71.200	0.09
4	125.00	125.00	0.00
5	148.72	148.15	0.38
6	148.72	148.15	0.38
7	213.89	213.61	0.13
8	287.84	283.47	1.52
9	287.84	283.47	1.52

Input files

nfm5x32x.inp B32 elements.

Test 5H: Deep simply supported beam: harmonic forced vibration

Products: Abaqus/Standard

Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

B21 B22 B23 B31 B32 B33
C3D4 C3D4H C3D6 C3D6H C3D8 C3D8H C3D8I C3D8R C3D8RH
C3D10 C3D10H C3D10HS C3D10M C3D10MH C3D15 C3D15H C3D15V C3D15VH
C3D20 C3D20H C3D20R C3D20RH C3D27 C3D27H C3D27R C3D27RH

Problem description

Material and geometry specifications are as given in *Test 5: Deep simply supported beam: frequency extraction*.

Forcing function: Steady-state harmonic.

$F = F_0 \sin(\omega t)$

$F_0 =$ 1 MN/m over whole length of beam.

$\omega = 2\pi f$

$f =$ 40 to 45 Hz

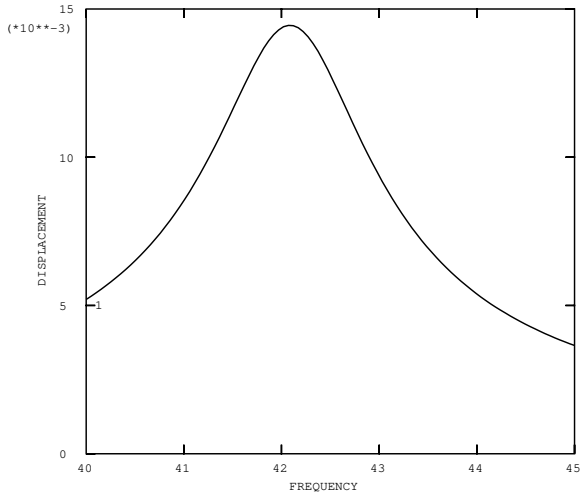
Damping: $\zeta =$ 2%

Response: u_y and extreme fiber bending stress.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 5H from NAFEMS "Selected Benchmarks for Forced Vibration," R0016, March 1993.

Response predicted by Abaqus



Results and discussion

The results are shown in *Table 1* and *Table 2*. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Table 1: Modal solution.

	Peak displacement (mm)	Peak stress (N/mm^2)	Frequency (Hz)
Reference solution	13.45	241.9	42.65
B32	13.48 (0.22%)	238.6 (–1.36%)	42.70 (0.12%)

Table 2: Direct solution.

	Peak displacement (mm)	Peak stress (N/mm^2)	Frequency (Hz)
Reference solution	13.45	241.9	42.65
B21	13.87 (3.12%)	247.83 (2.45%)	42.63 (–0.05%)
B22	13.97 (3.87%)	249.25 (3.04%)	42.63 (–0.05%)
B23	12.19 (–9.37%)	243.46 (0.64%)	45.33 (6.28%)
B31	13.99 (4.01%)	248.08 (2.55%)	42.68 (0.07%)
B32	13.98 (3.94%)	249.25 (3.04%)	42.63 (–0.05%)
B33	12.19 (–9.37%)	243.46 (0.64%)	45.33 (6.28%)
C3D4	12.89 (–4.16%)	235.83 (–2.51%)	43.06 (0.96%)
C3D4H	12.89 (–4.16%)	235.23 (–2.76%)	43.06 (0.96%)
C3D6	11.43 (–13.93%)	204.56 (–15.44%)	45.76 (7.29%)
C3D6H	11.43 (–13.93%)	204.08 (–15.63%)	45.76 (7.29%)
C3D8	13.54 (0.68%)	214.94 (–11.15%)	42.55 (–0.23%)
C3D8H	13.54 (0.68%)	214.93 (–11.15%)	42.55 (–0.23%)
C3D8I	13.00 (–3.34%)	235.82 (–2.51%)	42.65 (0%)
C3D8IH	13.00 (–3.34%)	235.80 (–2.52%)	42.65 (0%)

	Peak displacement (mm)	Peak stress (N/mm^2)	Frequency (Hz)
C3D8R	14.62 (8.70%)	209.11 (–13.56%)	41.02 (–3.82%)
C3D8RH	14.62 (8.70%)	209.01 (–13.60%)	41.02 (–3.82%)
C3D10	13.13 (–2.38%)	243.66 (0.73%)	42.75 (0.23%)
C3D10H	14.12 (4.98%)	238.68 (–1.33%)	41.43 (2.86%)
C3D10HS	13.13 (–2.38%)	242.29 (0.16%)	42.76 (0.26%)
C3D10M	13.66 (1.56%)	259.82 (7.41%)	41.94 (–1.66%)
C3D10MH	13.66 (1.56%)	259.86 (7.42%)	41.94 (–1.66%)
C3D15	13.31 (–1.04%)	242.08 (0.07%)	42.76 (0.26%)
C3D15V	13.31 (–1.04%)	242.08 (0.07%)	42.76 (0.26%)
C3D15VH	13.30 (–1.12%)	244.53 (1.09%)	42.76 (0.26%)
C3D15H	13.31 (–1.04%)	241.35 (–0.23%)	42.76 (0.26%)
C3D20	13.38 (–0.52%)	242.62 (0.30%)	42.65 (0%)
C3D20H	13.43 (–0.15%)	238.13 (–1.56%)	42.55 (–0.23%)
C3D20R	13.51 (0.45%)	237.53 (–1.81%)	42.45 (–0.46%)
C3D20RH	13.59 (1.04%)	236.71 (–2.15%)	42.35 (–0.70%)
C3D27	13.37 (–0.59%)	242.37 (0.19%)	42.65 (0%)
C3D27H	13.53 (0.59%)	241.44 (–0.19%)	42.24 (–0.96%)
C3D27R	13.48 (0.22%)	237.58 (–1.79%)	42.55 (–0.23%)
C3D27RH	13.77 (2.37%)	241.11 (–0.33%)	42.04 (–1.43%)

Input files

nfh5x21x.inp	B21 elements.
nfh5x22x.inp	B22 elements.
nfh5x23x.inp	B23 elements.
nfh5x31x.inp	B31 elements.
nfh5x32x.inp	B32 elements.
nfh5x33x.inp	B33 elements.
nfh5xf4x.inp	C3D4 elements.
nfh5xh4x.inp	C3D4H elements.
nfh5xf6x.inp	C3D6 elements.
nfh5xh6x.inp	C3D6H elements.
nfh5xf8x.inp	C3D8 elements.
nfh5xi8x.inp	C3D8I elements.
nfh5xj8x.inp	C3D8IH elements.
nfh5xr8x.inp	C3D8R elements.
nfh5xy8x.inp	C3D8RH elements.
nfh5xfax.inp	C3D10 elements.
nfh5xhax.inp	C3D10H elements.
nfh5xiax.inp	C3D10HS elements.

nfh5xkax.inp	C3D10M elements.
nfh5xlax.inp	C3D10MH elements.
nfh5xffx.inp	C3D15 elements.
nfh5xffv.inp	C3D15V elements.
nfh5xhfv.inp	C3D15VH elements.
nfh5xhfx.inp	C3D15H elements.
nfh5xfkx.inp	C3D20 elements.
nfh5xhkx.inp	C3D20H elements.
nfh5xrkx.inp	C3D20R elements.
nfh5xykx.inp	C3D20RH elements.
nfh5xfrv.inp	C3D27 elements.
nfh5xhrv.inp	C3D27H elements.
nfh5xrrv.inp	C3D27R elements.
nfh5xyrv.inp	C3D27RH elements.

Test 5T: Deep simply supported beam: transient forced vibration

Products: Abaqus/Standard Abaqus/Explicit

Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

B21 B22 B31 B32

Problem description

Material and geometry specifications are as given in *Test 5: Deep simply supported beam: frequency extraction*.

Mesh: A coarse mesh and a fine mesh are tested in the Abaqus/Explicit analyses.

Forcing function: Suddenly applied step load, transverse to the beam.

$F_0 =$ 1 MN/m over whole length of beam.

Damping: $\zeta =$ 2% [2% of critical damping in the dominant first mode with analytical frequency value $f =$ 42.650 (Hz) or $\omega = 2\pi f =$ 267.98 (sec^{-1})].

The damping factors are chosen as $\alpha_R =$ 5.36 (sec^{-1}) and $\beta_R =$ 7.46×10^{-5} (sec) so that

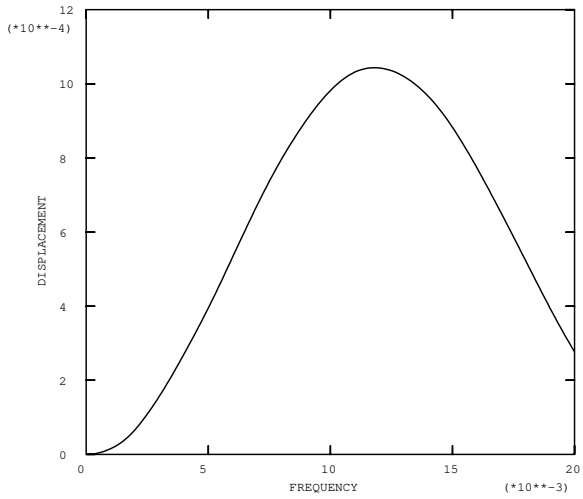
$$\zeta = \frac{\alpha_R}{2\omega} + \frac{\beta_R \omega}{2} = 0.01 + 0.01 = 0.02.$$

Response: u_y and extreme fiber bending stress.

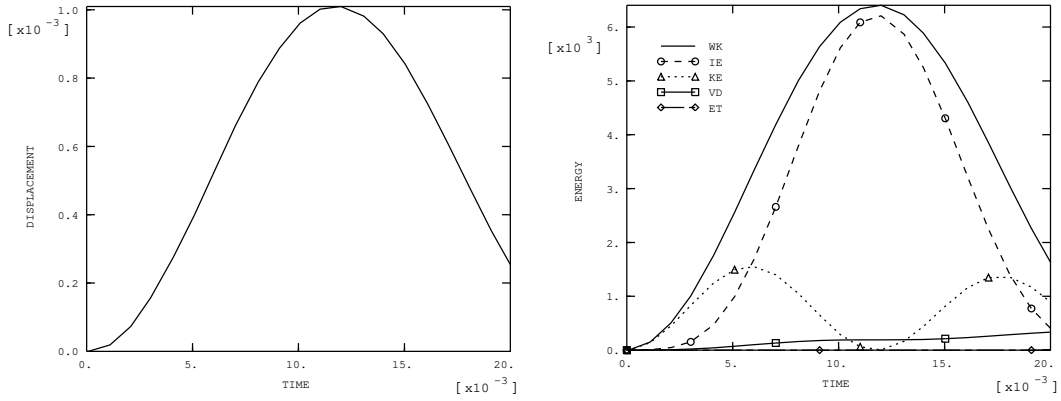
Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 5T from NAFEMS "Selected Benchmarks for Forced Vibration," R0016, March 1993.

Response predicted by Abaqus/Standard



Response predicted by Abaqus/Explicit



Results and discussion

The results are given in *Table 1* through *Table 5*. The values enclosed in parentheses are percentage differences with respect to the reference solution. The static displacement was obtained by running the analysis for a second step with a time period of ten seconds.

Table 1: Element type: B21, Abaqus/Explicit analysis.

	Peak displacement		Peak stress	Static disp.
	δ_p (mm)	t_p (sec)	σ (N/mm^2)	δ_s (mm)
Reference solution	1.043	0.0117	18.76	0.538
Coarse mesh	0.991	0.0120	17.35	0.510
	(–4.99%)	(2.56%)	(–7.52%)	(–5.20%)
Fine mesh	1.009	0.0120	18.04	0.520
	(–3.24%)	(2.56%)	(–3.84%)	(–3.35%)

Table 2: Element type: B22, Abaqus/Explicit analysis.

	Peak displacement		Peak stress	Static disp.
	δ_p (mm)	t_p (sec)	σ (N/mm^2)	δ_s (mm)
Reference solution	1.043	0.0117	18.76	0.538
Coarse mesh	1.042	0.0119	17.80	0.532
	(–0.09%)	(1.70%)	(–5.12%)	(–1.12%)
Fine mesh	1.043	0.0116	18.17	0.537
	(0.00%)	(–0.85%)	(–3.14%)	(–0.19%)

Table 3: Element type: B31, Abaqus/Explicit analysis.

	Peak displacement		Peak stress	Static disp.
	δ_p (mm)	t_p (sec)	σ (N/mm^2)	δ_s (mm)
Reference solution	1.043	0.0117	18.76	0.538
Coarse mesh	0.991	0.0120	17.37	0.510
	(–4.99%)	(2.56%)	(–7.41%)	(–5.20%)
Fine mesh	1.010	0.0120	18.06	0.518
	(–3.16%)	(2.56%)	(–3.73%)	(–3.72%)

Table 4: Element type: B32, Abaqus/Explicit analysis.

	Peak displacement		Peak stress	Static disp.
	δ_p (mm)	t_p (sec)	σ (N/mm^2)	δ_s (mm)
Reference solution	1.043	0.0117	18.76	0.538
Coarse mesh	1.042	0.0117	17.80	0.535
	(–0.09%)	(0.00%)	(–5.11%)	(–0.56%)
Fine mesh	1.043	0.0116	18.17	0.539
	(0.00%)	(–0.85%)	(–3.14%)	(0.19%)

Table 5: Element type: B32, Abaqus/Standard analysis.

	Peak displacement		Peak stress	Static disp.
	δ_p (mm)	t_p (sec)	σ (N/mm^2)	δ_s (mm)
Reference solution	1.043	0.0117	18.76	0.538
Direct solution	1.043	0.0118	18.29	0.536
		(0.85%)	(–2.50%)	(–0.37%)
Modal solution	1.041	0.0116	18.09	0.507
	(–0.19%)	(–0.85%)	(–5.57%)	(–5.76%)

Input files

Abaqus/Standard input files

nft5x32x.inp	B32 elements.

The modal solution in Abaqus/Standard is obtained from Steps 3 and 4 in *nfm5x32x.inp*.

Abaqus/Explicit input files

fv5t_b21_c.inp	B21 elements, coarse mesh.
fv5t_b21_f.inp	B21 elements, fine mesh.
fv5t_b22_c.inp	B22 elements, coarse mesh.
fv5t_b22_f.inp	B22 elements, fine mesh.
fv5t_b31_c.inp	B31 elements, coarse mesh.
fv5t_b31_f.inp	B31 elements, fine mesh.
fv5t_b32_c.inp	B32 elements, coarse mesh.
fv5t_b32_f.inp	B32 elements, fine mesh.

Test 5R: Deep simply supported beam: random forced vibration

Products: Abaqus/Standard

Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

B32

Problem description

Material and geometry specifications are as given in *Test 5: Deep simply supported beam: frequency extraction*.

Loading: Uniform force of 1.E6 N/m applied to all elements with unit white noise power spectral density in the frequency region between 0 Hz and 1000 Hz, using the fully correlated type of random loading.

Damping: 16 modes with direct damping. $\zeta = 2\%$.

Response: Peak PSD of displacement and stress.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 5R from NAFEMS "Selected Benchmarks for Forced Vibration," R0016, September 1993.

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

	Peak δ PSD	Peak σ PSD	Frequency
	(mm^2/Hz)	[(N/mm^2)^2/Hz]	(Hz)
Reference solution	180.90	58516	42.65
Abaqus solution	181.75 (0.47%)	56926 (–2.72%)	42.66 (0.12%)

Input files

nfr5x32x.inp B32 elements.

Test 13: Simply supported thin square plate: frequency extraction

Products: Abaqus/Standard

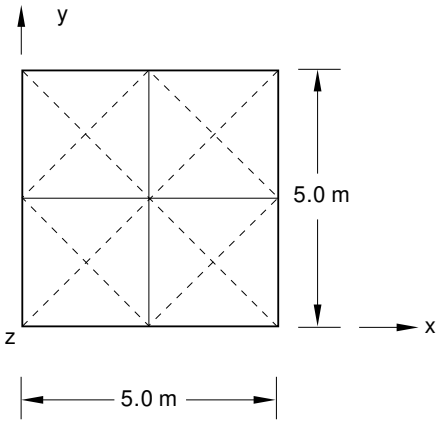
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

S8R5

Problem description



Model: Plate thickness = 0.05 m.

Material: Young's modulus = 200 GPa, Poisson's ratio = 0.3, density = 8000 kg/m^3.

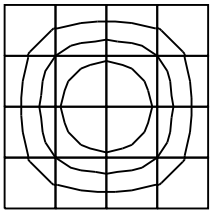
Boundary conditions: $u_x = u_y = \phi_z = 0$ at all nodes. $u_z = 0$ along all four edges. $\phi_x = 0$ along edges $x = 0$ and $x = 10$, $\phi_y = 0$ along edges $y = 0$ and $y = 10$.

Frequency extraction is performed in Step 1.

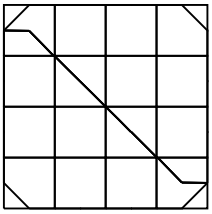
Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 13 from NAFEMS “Selected Benchmarks for Forced Vibration,” R0016, March 1993.

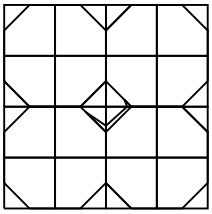
Mode shapes predicted by Abaqus



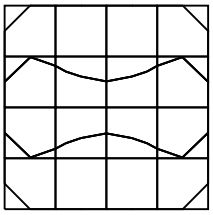
mode 1



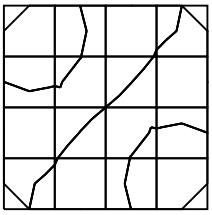
mode 2



mode 4



mode 5



mode 7

The contour plots were generated by setting the maximum and minimum contour levels close to zero. Where contour levels coincided with the element boundaries, the maximum contour level was increased and the minimum contour level was decreased appropriately.

Results and discussion

The results are shown in the following table.

Mode	Abaqus result	NAFEMS reference result	% Difference
1	2.377	2.377	0.00
2, 3	5.961	5.942	0.32
4	9.483	9.507	–0.25
5, 6	12.133	11.884	2.10
7, 8	15.468	15.449	0.12

Input files

nfm1358x.inp S8R5 elements.

Test 13H: Simply supported thin square plate: harmonic forced vibration

Products: Abaqus/Standard

Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

S3 S3R S4 S4R S4R5 S8R S8R5 S9R5 STRI3 STRI65 SC6R SC8R

Problem description

Material and geometry specifications are as given in *Test 13: Simply supported thin square plate: frequency extraction*.

Forcing function: Steady-state harmonic.

$F = F_0 \sin(\omega t)$

$F_0 =$ 100 N/m^2 over whole plate.

$\omega = 2\pi f$

$f =$ 0 to 4.16 Hz

Damping: $\zeta =$ 2%

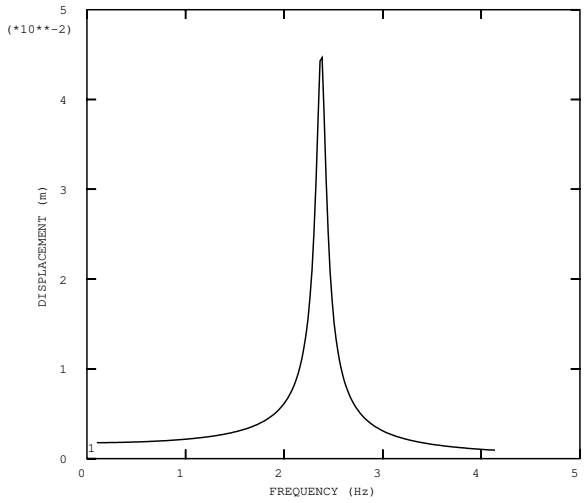
Response: u_z and σ_{11} at center of plate.

General: Gauss integration is used for the shell cross-section in input file *nfh1368x.inp*.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 13H from NAFEMS "Selected Benchmarks for Forced Vibration," R0016, March 1993.

Response predicted by Abaqus



Results and discussion

The results are given in *Table 1* and *Table 2*. The values enclosed in parentheses are percentage differences with respect to the reference solution. The modal solutions are obtained from Step 2 in files whose names begin with nfm13.

Table 1: Modal solution.

	Peak displacement (mm)	Peak stress (N/mm^2)	Frequency (Hz)
Reference solution	45.42	30.03	2.377
S4	45.03 (–0.85%)	31.33 (4.3%)	2.418 (–1.72%)
S4R	45.33 (–0.20%)	30.34 (1.03%)	2.410 (1.39%)
S4R5	45.42 (0.00%)	30.41 (1.27%)	2.407 (1.26%)
S8R	43.12 (–5.06%)	33.73 (12.32%)	2.441 (2.69%)
S8R5	45.50 (0.18%)	35.12 (16.95%)	2.377 (0.00%)

Table 2: Direct solution.

	Peak displacement (mm)	Peak stress (N/mm^2)	Frequency (Hz)
Reference solution	45.42	30.03	2.377
S3/S3R	41.56 (–8.50%)	27.82 (–7.36%)	2.49 (4.75%)
S4	44.93 (–1.08%)	31.26 (4.10%)	2.420 (1.81%)
S4 (collapsed)	41.56 (–8.50%)	27.82 (–7.36%)	2.49 (4.75%)
S4R	45.38 (–0.09%)	30.37 (1.13%)	2.405 (1.18%)
S4R (collapsed)	41.56 (–8.50%)	27.82 (–7.36%)	2.49 (4.75%)
S4R5	45.41 (–0.02%)	30.39 (1.20%)	2.405 (1.18%)
S8R	43.86 (–3.43%)	34.31 (14.25%)	2.446 (2.90%)
S8R5	44.66 (–1.34%)	34.49 (14.85%)	2.385 (0.34%)
S9R5	44.59 (–1.83%)	32.24 (7.36%)	2.39 (0.55%)
STRI3	44.74 (–1.49%)	32.81 (9.26%)	2.36 (–0.71%)

	Peak displacement (mm)	Peak stress (N/mm^2)	Frequency (Hz)
STRI65	44.96 (–1.01%)	33.19 (10.5%)	2.36 (–0.71%)
SC6R	41.56 (–8.5%)	27.82 (–7.36)	2.49 (4.75%)
SC8R	45.38 (–0.09%)	30.37 (1.13%)	2.405 (1.18%)

Input files

nfh13f3x.inp	S3/S3R elements.
nfh13e4x.inp	S4 elements.
nfh13e41.inp	Collapsed S4 elements.
nfh13f4x.inp	S4R elements.
nfh13641.inp	Collapsed S4R elements.
nfh1354x.inp	S4R5 elements.
nfh1368x.inp	S8R elements.
nfh1358x.inp	S8R5 elements.
nfh1359x.inp	S9R5 elements.
nfh1363x.inp	STRI3 elements.
nfh1356x.inp	STRI65 elements.
nfh13_std_sc6r.inp	SC6R elements.
nfh13_std_sc8r.inp	SC8R elements.

Test 13T: Simply supported thin square plate: transient forced vibration

Products: Abaqus/Standard Abaqus/Explicit

Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

S3R S3RS S4R S4RS S4RSW S8R5

Problem description

Material and geometry specifications are as given in *Test 13: Simply supported thin square plate: frequency extraction*.

Mesh: A coarse mesh, a fine mesh, and a very fine mesh of a quarter of the plate are tested for elements S3R, S3RS, S4R, S4RS, and S4RSW in Abaqus/Explicit. For the quadrilateral element types the mesh densities of the coarse, fine, and very fine meshes are 2×2, 3×3, and 4×4, respectively; for the triangular element types the mesh densities are $2 \times 2 \times 4$, $3 \times 3 \times 4$, and $4 \times 4 \times 4$, respectively.

Forcing function: Suddenly applied pressure.

$F_0 =$ 100 N/m^2 over whole plate.

Damping: $\zeta =$ 2% [2% of critical damping in the dominant first mode with analytical frequency value $f =$ 2.377 (Hz) or $\omega = 2\pi f =$ 14.935 (sec^{-1})].

The damping factors are chosen as $\alpha_R =$ 0.299 (sec^{-1}) and $\beta_R =$ 1.339×10^{-3} (sec) so that

$$\zeta = \frac{\alpha_R}{2\omega} + \frac{\beta_R \omega}{2} = 0.01 + 0.01 = 0.02.$$

For the sole purpose of validation and comparison, the models with a fine mesh are also used to test band-limited damping in Abaqus/Explicit. The damping parameters are chosen as damping factor $\alpha_R = 0.299$ (sec^{-1}), damping ratio $\eta_d = 0.01$, high-frequency cutoff $f_h = 10$ (Hz), and low-frequency cutoff $f_l = 1.5$ (Hz).

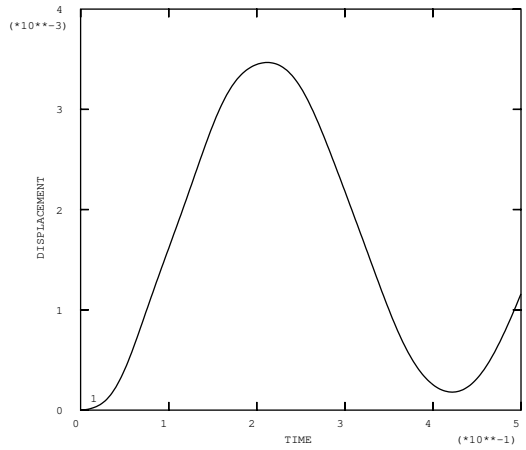
Response location: u_z and σ_{11} at center of plate.

General: Gauss integration is used for the shell cross-section in input file *nft1358x.inp*.

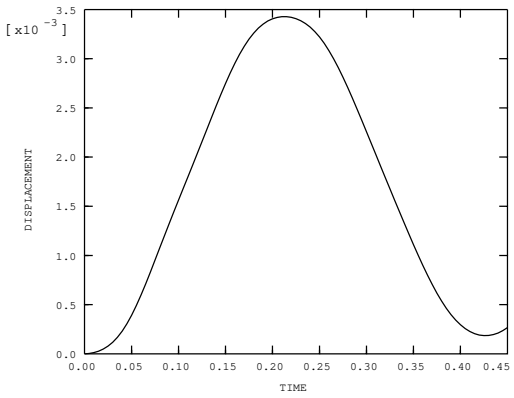
Reference solution

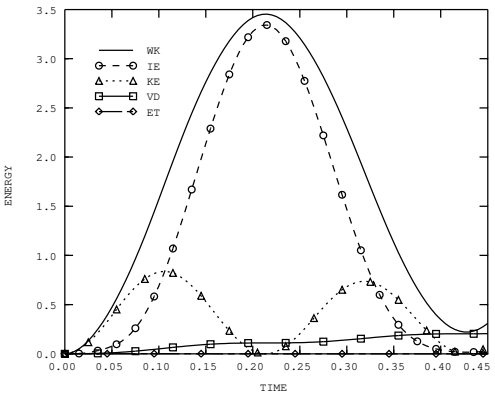
This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 13T from NAFEMS "Selected Benchmarks for Forced Vibration," R0016, March 1993.

Response predicted by Abaqus/Standard



Response predicted by Abaqus/Explicit





Results and discussion

The results are given in *Table 1* through *Table 7*. The values enclosed in parentheses are percentage differences with respect to the reference solution. The static displacement was obtained by creating a second step with a time period of 23 seconds.

Table 1: Element type: S3R, Abaqus/Explicit analysis.

	Peak displacement		Peak stress	Static displacement
	δ_p (mm)	t_p (sec)	σ (N/mm^2)	δ_s (mm)
Reference solution	3.523	0.210	2.484	1.817
Coarse mesh	3.147	0.210	2.079	1.616
	(−10.67%)	(0.0%)	(−16.30%)	(−11.06%)
Fine mesh	3.313	0.210	2.207	1.701

	Peak displacement		Peak stress	Static displacement
	δ_p (mm)	t_p (sec)	σ (N/mm^2)	δ_s (mm)
	(–5.96%)	(0.0%)	(–11.15%)	(–6.38%)
Very fine mesh	3.370	0.210	2.239	1.732
	(–4.34%)	(0.0%)	(–9.86%)	(–4.68%)

Table 2: Element type: S3RS, Abaqus/Explicit analysis.

	Peak displacement		Peak stress	Static displacement
	δ_p (mm)	t_p (sec)	σ (N/mm^2)	δ_s (mm)
Reference solution	3.523	0.210	2.484	1.817
Coarse mesh	3.263	0.210	2.028	1.679
	(–7.38%)	(0.0%)	(–18.36%)	(–7.59%)
Fine mesh	3.391	0.210	2.188	1.742
	(–3.75%)	(0.0%	(–11.92%)	(–4.13%)
Very fine mesh	3.429	0.210	2.244	1.762
	(–2.67%)	(0.0%)	(–9.66%)	(–3.01%)

Table 3: Element type: S4R, Abaqus/Explicit analysis.

	Peak displacement		Peak stress	Static displacement
	δ_p (mm)	t_p (sec)	σ (N/mm^2)	δ_s (mm)
Reference solution	3.523	0.210	2.484	1.817
Coarse mesh	3.366	0.225	1.889	1.760
	(–4.46%)	(7.14%)	(–23.95%)	(–3.14%)
Fine mesh	3.414	0.215	2.105	1.756
	(–3.09%)	(2.38%)	(–15.26%)	(–3.36%)
Very fine mesh	3.427	0.215	2.186	1.762
	(–2.72%)	(2.38%)	(–12.00%)	(–3.03%)

Table 4: Element type: S4RS, Abaqus/Explicit analysis.

	Peak displacement		Peak stress	Static displacement
	δ_p (mm)	t_p (sec)	σ (N/mm^2)	δ_s (mm)
Reference solution	3.523	0.210	2.484	1.817
Coarse mesh	3.495	0.240	1.974	1.797
	(–0.79%)	(14.3%)	(–20.53%)	(–1.1%)
Fine mesh	3.477	0.220	2.185	1.783
	(–1.31%)	(4.8%)	(–12.04%)	(–1.87%)
Very fine mesh	3.461	0.215	2.236	1.785

	Peak displacement		Peak stress	Static displacement
	δ_p (mm)	t_p (sec)	σ (N/mm^2)	δ_s (mm)
	(−1.76%)	(2.4%)	(−9.98%)	(−1.76%)

Table 5: Element type: S4RSW, Abaqus/Explicit analysis.

	Peak displacement		Peak stress	Static displacement
	δ_p (mm)	t_p (sec)	σ (N/mm^2)	δ_s (mm)
Reference solution	3.523	0.210	2.484	1.817
Coarse mesh	2.486	0.205	1.400	1.759
	(−29.44%)	(−2.4%)	(−43.64%)	(−3.19%)
Fine mesh	3.254	0.215	2.015	1.759
	(−7.64%)	(2.4%)	(−18.88%)	(−3.19%)
Very fine mesh	3.395	0.215	2.181	1.774
	(−3.63%)	(2.4%)	(−12.20%)	(−2.24%)

Table 6: Band-limited damping with fine mesh, Abaqus/Explicit analysis.

	Peak displacement		Peak stress
	δ_p (mm)	t_p (sec)	σ (N/mm^2)
Reference solution	3.523	0.210	2.484
S3R	3.264	0.205	2.202
	(−7.35%)	(−2.4%)	(−11.35%)
S3RS	3.293	0.210	2.121
	(−6.53%)	(0.0%)	(−14.61%)
S4R	3.411	0.215	2.171
	(−3.18%)	(2.38%)	(−12.60%)
S4RS	3.472	0.225	2.239
	(−1.45%)	(7.14%)	(−9.86%)
S4RSW	3.251	0.220	2.074
	(−7.72%)	(4.76%)	(−16.50%)

Table 7: Element type: S8R5, Abaqus/Standard analysis.

	Peak displacement		Peak stress	Static displacement
	δ_p (mm)	t_p (sec)	σ (N/mm^2)	δ_s (mm)
Reference solution	3.523	0.210	2.484	1.817
Direct solution	3.467 (−1.59%)	0.212 (0.95%)	2.476 (−2.50%)	1.780 (−2.04%)
Modal solution	3.456 (−1.93%)	0.214 (1.90%)	2.426 (−2.33%)	1.775 (2.37%)

Input files

fv13t_s3r_c.inp	S3R elements, coarse mesh.
fv13t_s3r_f.inp	S3R elements, fine mesh.
fv13t_s3r_vf.inp	S3R elements, very fine mesh.
fv13t_s3r_f_band.inp	S3R elements, fine mesh, band-limited damping.
fv13t_s3rs_c.inp	S3RS elements, coarse mesh.
fv13t_s3rs_f.inp	S3RS elements, fine mesh.
fv13t_s3rs_vf.inp	S3RS elements, very fine mesh.
fv13t_s3rs_f_band.inp	S3RS elements, fine mesh, band-limited damping.
fv13t_s4r_c.inp	S4R elements, coarse mesh.
fv13t_s4r_f.inp	S4R elements, fine mesh.
fv13t_s4r_vf.inp	S4R elements, very fine mesh.
fv13t_s4r_f_band.inp	S4R elements, fine mesh, band-limited damping.
fv13t_s4rs_c.inp	S4RS elements, coarse mesh.
fv13t_s4rs_f.inp	S4RS elements, fine mesh.
fv13t_s4rs_vf.inp	S4RS elements, very fine mesh.
fv13t_s4rs_f_band.inp	S4RS elements, fine mesh, band-limited damping.
fv13t_s4rsw_c.inp	S4RSW elements, coarse mesh.
fv13t_s4rsw_f.inp	S4RSW elements, fine mesh.
fv13t_s4rsw_vf.inp	S4RSW elements, very fine mesh.
fv13t_s4rsw_f_band.inp	S4RSW elements, fine mesh, band-limited damping.
nft1358x.inp	S8R5 elements.

The modal solution in Abaqus/Standard is obtained from Steps 3 and 4 in *nfm1358x.inp*.

Test 13R: Simply supported thin square plate: random forced vibration

Products: Abaqus/Standard

Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

S4 S4R S8R5 STRI65

Problem description

Material and geometry specifications are as given in *Test 13: Simply supported thin square plate: frequency extraction*.

Loading: Uniform pressure of 1.0 applied to all elements, with white noise power spectral density of 10000 Pa^2/Hz in the frequency region between 0 Hz and 1000 Hz, using the fully correlated type of random loading.

Damping: 16 modes with direct damping. $\zeta = 2\%$.

Response location: Peak PSD of displacement and stress at center of plate.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 13R from NAFEMS "Selected Benchmarks for Forced Vibration," R0016, September 1993.

	Peak δ PSD	Peak σ PSD	Frequency
	(mm^2/Hz)	[$(N/mm^2)^2/Hz$]	(Hz)
Reference solution	2063.20	1025.44	2.377

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Element type	Peak δ PSD	Peak σ PSD	Frequency
(Mesh Size)	(mm^2/Hz)	[$(N/mm^2)^2/Hz$]	(Hz)
S4 (4 × 4)	1901.8 (–7.8%)	844.25 (–17.67%)	2.549 (7.24%)
S4 (8 × 8)	2028.6 (–1.68%)	981.39 (–4.30%)	2.417 (1.68%)
S4R (4 × 4)	2040.2 (–1.11%)	665.98 (–35.05%)	2.5043 (5.36%)
S4R (8 × 8)	2062.2 (–0.05%)	923.83 (–9.91%)	2.4079 (1.30%)
S8R5 (4 × 4)	2071.5 (–0.40%)	1201.83 (17.2%)	2.3759 (–0.05%)

Element type	Peak δ PSD	Peak σ PSD	Frequency
(Mesh Size)	(mm^2/Hz)	[(N/mm^2)^2/Hz]	(Hz)
S8R5 (8 × 8)	2064.9 (0.08%)	1011.33 (−1.38%)	2.3767 (−0.01%)
STRI65 (2 × 4 × 4)	1989.4 (−3.58%)	942.13 (−8.14%)	2.4028 (1.09%)

Input files

nfr13e4c.inp	S4 elements, coarse mesh.
nfr13e4f.inp	S4 elements, fine mesh.
nfr13f4c.inp	S4R elements, coarse mesh.
nfr13f4f.inp	S4R elements, fine mesh.
nfr1358c.inp	S8R5 elements, coarse mesh.
nfr1358f.inp	S8R5 elements, fine mesh.
nfr1356c.inp	STRI65 elements, coarse mesh.

Test 21: Simply supported thick square plate: frequency extraction

Products: Abaqus/Standard

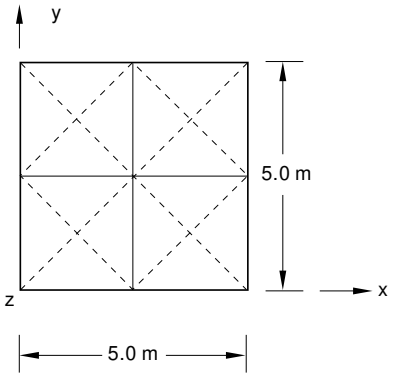
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

S4 S4R S4R5 S8R

Problem description



Model: Plate thickness = 0.05 m.

Material: Young's modulus = 200 GPa, Poisson's ratio = 0.3, density = 8000 kg/m^3.

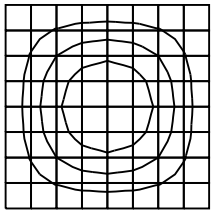
Boundary conditions: $u_x = u_y = \phi_z = 0$ at all nodes. $u_z = 0$ along all four edges. $\phi_x = 0$ along edges $x = 0$ and $x = 10$, $\phi_y = 0$ along edges $y = 0$ and $y = 10$.

Frequency extraction is performed in Step 1.

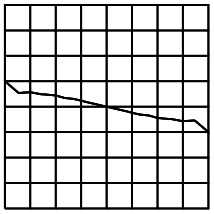
Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 21 from NAFEMS "Selected Benchmarks for Forced Vibration," R0016, March 1993.

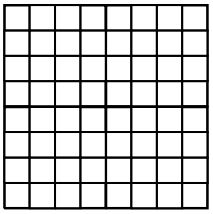
Mode shapes predicted by Abaqus (for element type S4R5)



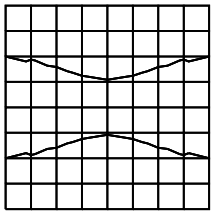
mode 1



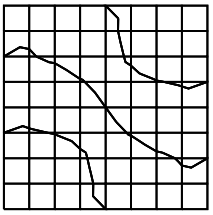
mode 2



mode 4



mode 5



mode 7

The contour plots were generated by setting the maximum and minimum contour levels close to zero. Where contour levels coincided with the element boundaries, the maximum contour level was increased and the minimum contour level was decreased appropriately.

Results and discussion

The results are given in *Table 1* through *Table 4*.

Table 1: Element type: S4.

Mode	Abaqus result	NAFEMS reference result	% Difference
1	46.667	45.897	1.68
2, 3	115.92	109.41	5.95
4	178.00	167.89	6.02
5, 6	233.73	204.51	14.28
7, 8	285.20	256.50	11.19

Table 2: Element type: S4R.

Mode	Abaqus result	NAFEMS reference result	% Difference
1	46.485	45.897	1.28
2, 3	115.24	109.41	5.33
4	175.47	167.89	4.51
5, 6	232.39	204.51	13.63
7, 8	280.08	256.50	9.19

Table 3: Element type: S4R5.

Mode	Abaqus result	NAFEMS reference result	% Difference
1	46.514	45.897	1.34
2, 3	115.55	109.41	5.61
4	177.26	167.89	−5.29
5, 6	233.66	204.51	14.25
7, 8	285.99	256.50	11.48

Table 4: Element type: S8R.

Mode	Abaqus result	NAFEMS reference result	% Difference
1	45.936	45.897	0.08
2, 3	110.41	109.41	0.91
4	170.38	167.89	1.48
5, 6	212.81	204.51	4.06
7, 8	269.96	256.50	5.25

Input files

nfm21e4x.inp	S4 elements.
nfm21f4x.inp	S4R elements.
nfm2154x.inp	S4R5 elements.
nfm2168x.inp	S8R elements.

Test 21H: Simply supported thick square plate: harmonic forced vibration

Products: Abaqus/Standard

Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

S3 S3R S4 S4R S4R5 S8R S8R5 S9R5 STRI3 STRI65

Problem description

Material and geometry specifications are as given in *Test 21: Simply supported thick square plate: frequency extraction*.

Forcing function: Steady-state harmonic.

$F = F_0 \sin(\omega t)$

$F_0 =$ 1 MN/m^2 over whole plate.

$\omega = 2\pi f$

$f =$ 0 to 78.17 Hz

Damping: $\zeta =$ 2%

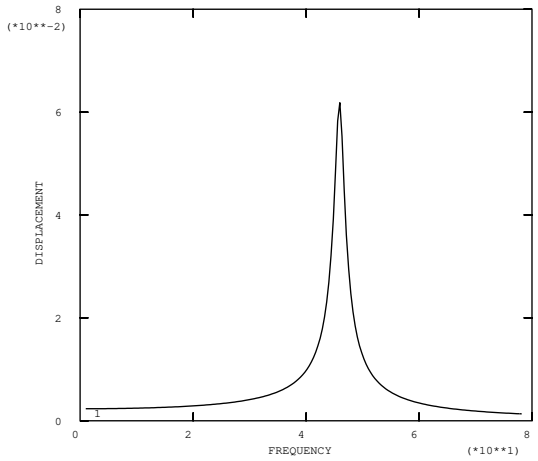
Response: u_z and σ_{11} at center of plate.

The modal solution is obtained from Step 2 in files whose names begin with nfm21.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 21H from NAFEMS "Selected Benchmarks for Forced Vibration," R0016, March 1993.

Response predicted by Abaqus



Results and discussion

The results are given in *Table 1* and *Table 2*. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Table 1: Modal solution.

	Peak displacement (mm)	Peak stress (N/mm^2)	Frequency (Hz)
Reference solution	58.33	800.8	45.90
S4	59.54 (2.03%)	784.01 (–2.09%)	46.67(1.68%)
S4R	60.01 (2.88%)	760.4 (–5.04%)	46.52 (1.35%)
S4R5	59.93 (2.74%)	760.8 (–5.00%)	46.51 (1.33%)
S8R	59.94 (2.76%)	880.1 (9.90%)	45.94 (0.09%)

Table 2: Direct solution.

	Peak displacement (mm)	Peak stress (N/mm^2)	Frequency (Hz)
Reference solution	58.33	800.8	45.90
S3/S3R	57.52 (–1.39%)	745.7 (–6.88%)	47.92 (4.40%)
S4R (collapsed)	57.52 (–1.39%)	745.7 (–6.88%)	47.92 (4.40%)
S4 (collapsed)	57.52 (–1.39%)	745.7 (–6.88%)	47.92 (4.40%)
S4	60.77 (4.18%)	800.3 (–0.06%)	46.76(1.87%)
S4R	61.33 (5.14%)	776.9 (–2.98%)	46.39 (1.07%)
S4R5	61.09 (4.73%)	775.7 (–3.13%)	45.98 (0.17%)
S8R	61.87 (6.07%)	908.4 (13.44%)	45.98 (0.17%)
S8R5	60.81 (4.25%)	887.8 (10.8%)	46.0 (0.22%)
S9R5	60.73 (4.11%)	830.3 (3.68%)	46.0 (0.22%)
STRI3	55.88 (–4.20%)	818.4 (2.19%)	46.8 (1.96%)
STRI65	62.47 (7.09%)	860.7 (7.48%)	46.0 (0.22%)

Input files

nfh21f3x.inp	S3/S3R elements.
nfh21641.inp	Collapsed S4R elements.
nfh21e41.inp	Collapsed S4 elements.
nfh21e4x.inp	S4 elements.
nfh21f4x.inp	S4R elements.
nfh2154x.inp	S4R5 elements.
nfh2168x.inp	S8R elements.
nfh2158x.inp	S8R5 elements.
nfh2159x.inp	S9R5 elements.
nfh2163x.inp	STRI3 elements.
nfh2156x.inp	STRI65 elements.

Test 21T: Simply supported thick square plate: transient forced vibration

Products: Abaqus/Standard Abaqus/Explicit

Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

S3R S3RS S4 S4R S4RS S4RSW S4R5 S8R SC8R

Problem description

Material and geometry specifications are as given in *Test 21: Simply supported thick square plate: frequency extraction*. The plate thickness for the Abaqus/Explicit analyses is 1.0 in.

Mesh: A coarse mesh, a fine mesh, and a very fine mesh of a quarter of the plate are tested for elements S3R, S3RS, S4R, S4RS, and S4RSW in Abaqus/Explicit. For the quadrilateral element types the mesh densities of the coarse, fine, and very fine meshes are 2×2, 3×3, and 4×4, respectively; for the triangular element types the meshes are $2 \times 2 \times 4$, $3 \times 3 \times 4$, and $4 \times 4 \times 4$, respectively.

Forcing function: Suddenly applied pressure.

$F_0 =$ 1 MN/m^2 over whole plate.

Damping: $\zeta =$ 2% [2% of critical damping in the dominant first mode with analytical frequency value $f =$ 45.897 (Hz) or $\omega = 2\pi f =$ 288.379 (sec^{-1})].

The damping factors are chosen as $\alpha_R =$ 5.772 (sec^{-1}) and $\beta_R =$ 6.929×10^{-5} (sec) so that

$$\zeta = \frac{\alpha_R}{2\omega} + \frac{\beta_R \omega}{2} = 0.01 + 0.01 = 0.02.$$

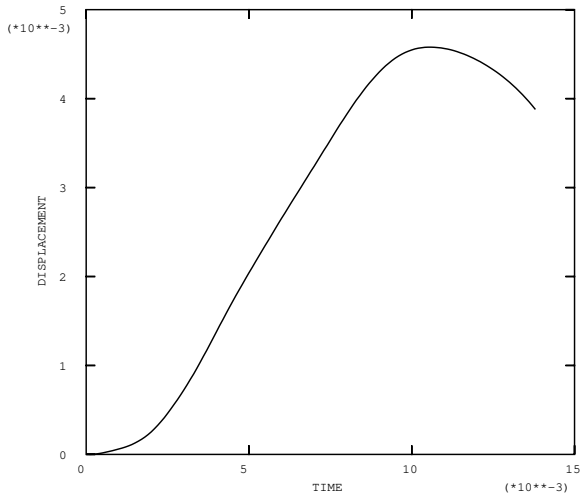
For the sole purpose of validation and comparison, the models with a fine mesh are also used to test band-limited damping in Abaqus/Explicit. The damping parameters are chosen as damping factor $\alpha_R = 5.772$ (sec^{-1}), damping ratio $\eta_d = 0.01$, high-frequency cutoff $f_h = 50$ (Hz), and low-frequency cutoff $f_l = 1.5$ (Hz).

Response location: u_z and σ_{11} at center of plate.

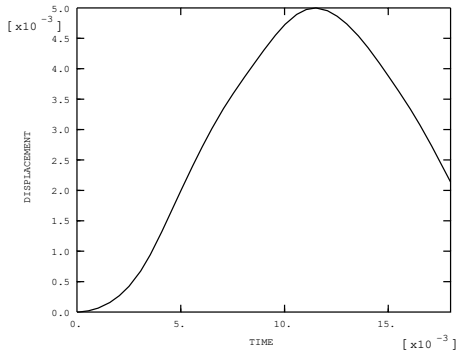
Reference solution

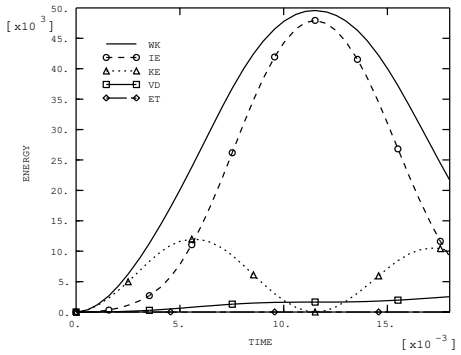
This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 21T from NAFEMS "Selected Benchmarks for Forced Vibration," R0016, March 1993.

Response predicted by Abaqus/Standard



Response predicted by Abaqus/Explicit





Results and discussion

The results are given in *Table 1* through *Table 10*. The values enclosed in parentheses are percentage differences with respect to the reference solution. The static displacement was obtained by creating a second step with a time period of about 2 seconds.

Table 1: Element type: S4, Abaqus/Standard analysis.

	Peak displacement		Peak stress	Static disp.
	δ_p (mm)	t_p (sec)	σ (N/mm^2)	δ_s (mm)
Reference solution	4.524	0.0108	62.11	2.333
Direct solution	4.569 (0.99%)	0.0108 (0.00%)	58.61 (–5.63%)	2.338 (0.21%)
Modal solution	4.564 (0.88%)	0.0107 (–0.93%)	58.57 (–5.69%)	2.334 (0.04%)

Table 2: Element type: S4R, Abaqus/Standard analysis.

	Peak displacement		Peak stress	Static disp.
	δ_p (mm)	t_p (sec)	σ (N/mm^2)	δ_s (mm)
Reference solution	4.524	0.0108	62.11	2.333
Direct solution	4.603 (1.75%)	0.0108 (0.00%)	56.91 (–8.37%)	2.338 (0.21%)
Modal solution	4.534 (0.22%)	0.0107 (–0.93%)	53.82 (–13.35%)	2.334 (0.04%)

Table 3: Element type: S4R5, Abaqus/Standard analysis.

	Peak displacement		Peak stress	Static disp.
	δ_p (mm)	t_p (sec)	σ (N/mm^2)	δ_s (mm)
Reference solution	4.524	0.0108	62.11	2.333
Direct solution	4.599 (1.66%)	0.0108	56.90 (–8.68%)	2.339 (0.26%)
Modal solution	4.536 (0.27%)	0.0109 (0.93%)	54.04 (–12.99%)	2.335 (0.09%)

Table 4: Element type: S8R, Abaqus/Standard analysis.

	Peak displacement		Peak stress	Static disp.
	δ_p (mm)	t_p (sec)	σ (N/mm^2)	δ_s (mm)
Reference solution	4.524	0.0108	62.11	2.333
Direct solution	4.571 (1.04%)	0.0104 (–3.70%)	64.03 (3.09%)	2.331 (–0.09%)
Modal solution	4.578 (1.19%)	0.0106 (–1.85%)	63.88 (2.85%)	2.331 (–0.09%)

Table 5: Element type: SC8R, Abaqus/Standard analysis.

	Peak displacement		Peak stress	Static disp.
	δ_p (mm)	t_p (sec)	σ (N/mm^2)	δ_s (mm)
Reference solution	4.524	0.0108	62.11	2.333
Direct solution	4.627 (2.27%)	0.0110 (1.85%)	57.9 (–6.77%)	2.337 (–0.17%)
Modal solution	4.544 (0.33%)	0.0107 (–0.93%)	54.0 (13.1%)	2.339 (0.26%)

Table 6: Element type: S3R, Abaqus/Explicit analysis.

	Peak displacement		Peak stress	Static disp.
	δ_p (mm)	t_p (sec)	σ (N/mm^2)	δ_s (mm)
Reference solution	4.524	0.0108	62.11	2.333
Coarse mesh	4.223	0.0110	52.78	2.170
	(–6.65%)	(1.85%)	(–15.02%)	(–7.00%)
Fine mesh	4.441	0.0107	57.02	2.265
	(–1.83%)	(–0.92%)	(–8.20%)	(–2.91%)
Very fine mesh	4.517	0.0107	58.67	2.296

	Peak displacement		Peak stress	Static disp.
	δ_p (mm)	t_p (sec)	σ (N/mm^2)	δ_s (mm)
	(–0.15%)	(–0.92%)	(–5.54%)	(–1.58%)

Table 7: Element type: S3RS, Abaqus/Explicit analysis.

	Peak displacement		Peak stress	Static disp.
	δ_p (mm)	t_p (sec)	σ (N/mm^2)	δ_s (mm)
Reference solution	4.524	0.0108	62.11	2.333
Coarse mesh	4.208	0.0109	53.00	2.148
	(–6.98%)	(0.93%)	(–19.50%)	(–7.93%)
Fine mesh	4.425	0.0106	56.99	2.252
	(–2.19%)	(–1.85%)	(–8.24%)	(–3.47%)
Very fine mesh	4.507	0.0107	58.67	2.288
	(–0.38%)	(–0.93%)	(–5.54%)	(–1.93%)

Table 8: Element type: S4R, Abaqus/Explicit analysis.

	Peak displacement		Peak stress	Static disp.
	δ_p (mm)	t_p (sec)	σ (N/mm^2)	δ_s (mm)
Reference solution	4.524	0.0108	62.11	2.333
Coarse mesh	4.466	0.0122	48.21	2.368
	(–1.28%)	(12.96%)	(–22.38%)	(1.50%)
Fine mesh	4.556	0.0114	55.02	2.348
	(0.71%)	(5.56%)	(–11.42%)	(0.64%)
Very fine mesh	4.577	0.0110	57.33	2.351
	(1.17%)	(1.85%)	(–7.70%)	(0.77%)

Table 9: Element type: S4RS, Abaqus/Explicit analysis.

	Peak displacement		Peak stress	Static disp.
	δ_p (mm)	t_p (sec)	σ (N/mm^2)	δ_s (mm)
Reference solution	4.524	0.0108	62.11	2.333
Coarse mesh	4.624	0.0125	49.84	2.381
	(2.21%)	(15.74%)	(–19.75%)	(2.06%)
Fine mesh	4.625	0.0114	55.75	2.353
	(2.23%)	(5.56%)	(–10.24%)	(0.86%)
Very fine mesh	4.612	0.0110	57.67	2.344
	(1.95%)	(1.85%)	(–7.15%)	(0.47%)

Table 10: Element type: S4RSW, Abaqus/Explicit analysis.

	Peak displacement		Peak stress	Static disp.
	δ_p (mm)	t_p (sec)	σ (N/mm^2)	δ_s (mm)
Reference solution	4.524	0.0108	62.11	2.333
Coarse mesh	4.618	0.0121	49.77	2.381
	(2.08%)	(12.04%)	(−19.87%)	(2.06%)
Fine mesh	4.623	0.0114	55.74	2.353
	(2.19%)	(5.56%)	(−10.26%)	(0.86%)
Very fine mesh	4.612	0.0109	57.67	2.344
	(1.95%)	(0.93%)	(−7.15%)	(0.47%)

Table 11: Band-limited damping with fine mesh, Abaqus/Explicit analysis.

	Peak displacement		Peak stress	Static disp.
	δ_p (mm)	t_p (sec)	σ (N/mm^2)	δ_s (mm)
Reference solution	4.524	0.0108	62.11	2.333
S3R	4.258	0.0105	53.87	2.265
	(−5.88%)	(−2.78%)	(−13.27%)	(−2.91%)
S3RS	4.246	0.0106	54.21	2.252
	(−6.15%)	(-1.89%)	(−12.72%)	(−3.47%)
S4R	4.381	0.0110	53.04	2.348
	(−3.16%)	(1.85%)	(−14.60%)	(0.64%)
S4RS	4.449	0.0115	54.23	2.353
	(−1.66%)	(6.48%)	(−12.69%)	(0.86%)
S4RSW	4.448	0.0115	54.22	2.353
	(−1.68%)	(6.48%)	(−12.70%)	(0.86%)

Input files

Abaqus/Standard input files

nft21e4x.inp	S4 elements.
nft21f4x.inp	S4R elements.
nft2154x.inp	S4R5 elements.
nft2168x.inp	S8R elements.
nft21_std_sc8r.inp	SC8R elements.

The modal solution in Abaqus/Standard is obtained from Steps 3 and 4 in files whose names begin with nfm21.

Abaqus/Explicit input files

fv21t_s3r_c.inp	S3R elements, coarse mesh.

fv21t_s3r_f.inp	S3R elements, fine mesh.
fv21t_s3r_vf.inp	S3R elements, very fine mesh.
fv21t_s3r_f_band.inp	S3R elements, fine mesh, band-limited damping.
fv21t_s3rs_c.inp	S3RS elements, coarse mesh.
fv21t_s3rs_f.inp	S3RS elements, fine mesh.
fv21t_s3rs_vf.inp	S3RS elements, very fine mesh.
fv21t_s3rs_f_band.inp	S3RS elements, fine mesh, band-limited damping.
fv21t_s4r_c.inp	S4R elements, coarse mesh.
fv21t_s4r_f.inp	S4R elements, fine mesh.
fv21t_s4r_vf.inp	S4R elements, very fine mesh.
fv21t_s4r_f_band.inp	S4R elements, fine mesh, band-limited damping.
fv21t_s4rs_c.inp	S4RS elements, coarse mesh.
fv21t_s4rs_f.inp	S4RS elements, fine mesh.
fv21t_s4rs_vf.inp	S4RS elements, very fine mesh.
fv21t_s4rs_f_band.inp	S4RS elements, fine mesh, band-limited damping.
fv21t_s4rsw_c.inp	S4RSW elements, coarse mesh.
fv21t_s4rsw_f.inp	S4RSW elements, fine mesh.
fv21t_s4rsw_vf.inp	S4RSW elements, very fine mesh.
fv21t_s4rsw_f_band.inp	S4RSW elements, fine mesh, band-limited damping.

Test 21R: Simply supported thick square plate: random forced vibration

Products: Abaqus/Standard

Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

S4 S4R S4R5 S8R STRI65

Problem description

Material and geometry specifications are as given in *Test 21: Simply supported thick square plate: frequency extraction*.

Loading: Uniform pressure of 1.0 applied to all elements, with white noise power spectral density of 1.E12 Pa^2/Hz in the frequency region between 0 Hz and 1000 Hz, using the fully correlated type of random loading.

Damping: 16 modes with direct damping. $\zeta = 2\%$.

Response location: Peak PSD of displacement and stress at center of plate.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 21R from NAFEMS "Selected Benchmarks for Forced Vibration," R0016, September 1993.

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

	Peak δ PSD(mm^2/Hz)	Peak σ PSD[$(N/mm^2)^2/Hz$]	Frequency(Hz)
Reference solution	3401.81	641200	45.90
S4	3546.78 (4.26%)	614460 (–4.17%)	46.65 (1.63%)
S4R	3599.92 (5.82%)	—	46.51 (1.33%)
S4R5	3593.22 (5.63%)	—	46.50 (1.31%)
S8R	3594.31 (5.66%)	—	45.92 (0.04%)
STRI65	3712.97 (9.15%)	—	45.868 (–0.07%)

Input files

nfr21e4x.inp	S4 elements.
nfr21f4x.inp	S4R elements.

nfr2154x.inp	S4R5 elements.
nfr2168x.inp	S8R elements.
nfr2156x.inp	STRI65 elements.

Proposed nonlinear benchmarks

In this section:

- *NL1: Prescribed biaxial strain history, plane strain*
- *NL2: Axisymmetric thick cylinder*
- *NL3: Hardening with two variables under load control*
- *NL4: Snap-back under displacement control*
- *NL5: Straight cantilever with end moment*
- *NL6: Straight cantilever with axial end point load*
- *NL7: Lee's frame buckling problem*

NL1: Prescribed biaxial strain history, plane strain

Products: Abaqus/Standard

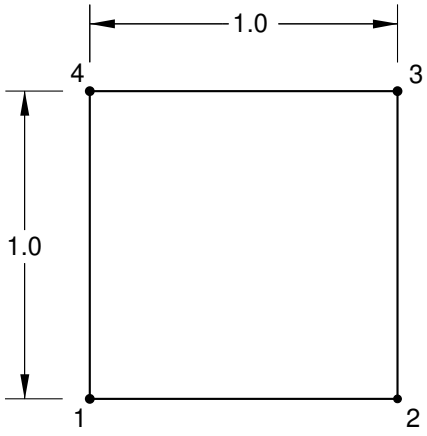
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CPE4R

Problem description



Units: m

Material: Linear elastic, Young's modulus = 250 GPa, Poisson's ratio = 0.25, yield stress = 5 MPa, strain at first yield = 0.25×10^{-4}, hardening modulus = 0 or 62.5 GPa.

Boundary conditions:

Step 1: $u_x = 0.25 \times 10^{-4}$ at nodes 2 and 3

Step 2: $u_x = 0.50 \times 10^{-4}$ at nodes 2 and 3

Step 3: $u_x = 0.50 \times 10^{-4}$ at nodes 2 and 3, $u_y = 0.25 \times 10^{-4}$ at nodes 3 and 4

Step 4: $u_x = 0.50 \times 10^{-4}$ at nodes 2 and 3, $u_y = 0.50 \times 10^{-4}$ at nodes 3 and 4

Step 5: $u_x = 0.25 \times 10^{-4}$ at nodes 2 and 3, $u_y = 0.50 \times 10^{-4}$ at nodes 3 and 4

Step 6: $u_y = 0.50 \times 10^{-4}$ at nodes 3 and 4

Step 7: $u_y = 0.25 \times 10^{-4}$ at nodes 2 and 3

Step 8: all degrees of freedom constrained with zero displacement

All degrees of freedom are constrained with zero displacement unless stated otherwise.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test NL1 from NAFEMS Publication NNB, Rev. 1, "NAFEMS Non-Linear Benchmarks," October 1989.

Strain ($\times 10^{-4}$)			Target effective stress (MPa)	
ε_x	ε_y	ε_z	Perfect plasticity	Isotropic hardening
			(H = 0 GPa)	(H = 62.5 GPa)
0.25	0.0	0.0	5.000	5.000
0.50	0.0	0.0	5.000	5.862
0.50	0.25	0.0	5.000	5.482
0.50	0.50	0.0	5.000	6.362
0.25	0.50	0.0	5.000	6.640
0.0	0.50	0.0	5.000	7.322
0.0	0.25	0.0	3.917	4.230
0.0	0.0	0.0	5.000	5.673

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Strain ($\times 10^{-4}$)			Target effective stress (MPa)	
ε_x	ε_y	ε_z	Perfect plasticity	Isotropic hardening
			(H = 0 GPa)	(H = 62.5 GPa)
0.25	0.0	0.0	5.000 (0%)	5.000 (0%)
0.50	0.0	0.0	5.000 (0%)	5.862 (0%)
0.50	0.25	0.0	5.000 (0%)	5.482 (0%)
0.50	0.50	0.0	5.000 (0%)	6.359 (–0.05%)
0.25	0.50	0.0	5.000 (0%)	6.626 (–0.21%)
0.0	0.50	0.0	5.000 (0%)	7.297 (–0.34%)
0.0	0.25	0.0	3.824 (–2.4%)	4.114 (–2.70%)
0.0	0.0	0.0	5.000 (0%)	5.532 (–2.50%)

Remarks

The loading and constraints on the model ensure that the force residuals at the nodes are always zero, regardless of the state of stress in the element. Abaqus, therefore, does not iterate. The results tabulated above are obtained using integration with 10 constant increments, as recommended in the test description. More accurate results are obtained if a larger number of increments is specified.

Input files

nnl1xr4x.inp	CPE4R elements.

NL2: Axisymmetric thick cylinder

Products: Abaqus/Standard

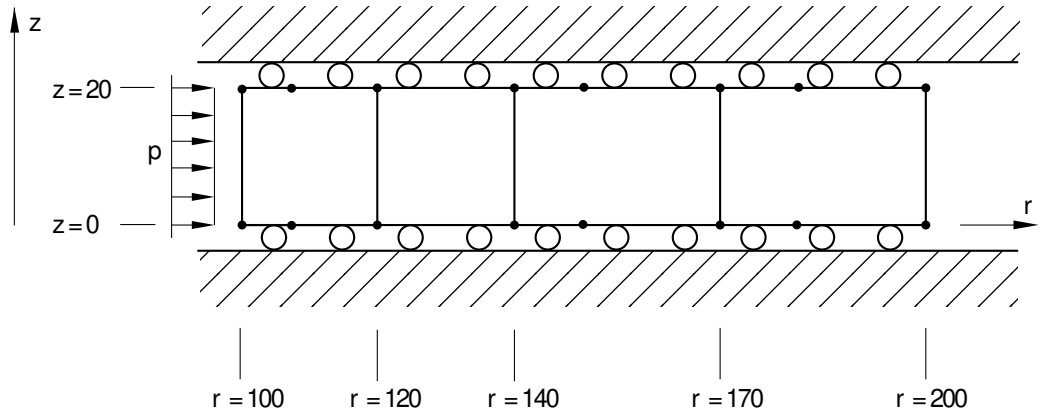
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CAX8R CCL24R

Problem description



Material: Linear elastic, Young's modulus = 207 GPa, Poisson's ratio = 0.3, yield stress = 207.9 MPa.

Boundary conditions: $u_z = 0$ for all nodes.

Loading: An initial internal pressure of 80 MPa is increased in steps of 20 MPa to 160 MPa.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test NL2 from NAFEMS Publication NNB, Rev. 1, "NAFEMS Non-Linear Benchmarks," October 1989.

The target radial stress and the target circumferential (hoop) stress are given by σ_{rr} and $\sigma_{\theta\theta}$, respectively.

r (mm)	Internal pressure (MPa)									
	80		100		120		140		160	
	σ_{rr}	$\sigma_{\theta\theta}$	σ_{rr}	$\sigma_{\theta\theta}$	σ_{rr}	$\sigma_{\theta\theta}$	σ_{rr}	$\sigma_{\theta\theta}$	σ_{rr}	$\sigma_{\theta\theta}$
104.2	−71.55	124.9	−89.74	149.9	−110.1	130.0	−130.1	110.0	−150.1	89.92
115.8	−52.89	106.2	−66.65	133.7	−84.87	154.7	−104.9	135.2	−124.9	115.2

r (mm)	Internal pressure (MPa)									
	80		100		120		140		160	
	σ_{rr}	$\sigma_{\theta\theta}$	σ_{rr}	$\sigma_{\theta\theta}$	σ_{rr}	$\sigma_{\theta\theta}$	σ_{rr}	$\sigma_{\theta\theta}$	σ_{rr}	$\sigma_{\theta\theta}$
124.2	−42.46	98.50	−53.47	120.6	−68.32	154.1	−87.92	151.9	−107.9	132.1
135.8	−31.19	84.52	−39.27	106.4	−50.17	136.0	−66.66	172.3	−86.61	153.3
146.3	−23.16	76.49	−29.16	96.32	−37.26	123.1	−49.51	163.5	−68.61	170.9
163.7	−13.14	66.48	−16.55	83.71	−21.15	107.0	−28.10	142.1	−41.94	195.9
176.3	−7.643	60.98	−9.624	76.78	−12.30	98.10	−16.34	130.4	−24.55	195.9
193.7	−1.769	55.10	−2.227	69.38	−2.846	88.65	−3.781	117.8	−5.682	177.0

Results and discussion

All results agree exactly with the reference solution, except for the result of $\sigma_{\theta\theta}$ at a load of 80 MPa and a distance of 124.2 mm. The value obtained here for both element types is 95.8 MPa, a difference of −2.74%.

Input files

nnl2xr8x.inp	CAX8R elements.
nnl2xrccl24.inp	CCL24R elements.

NL3: Hardening with two variables under load control

Products: Abaqus/Standard

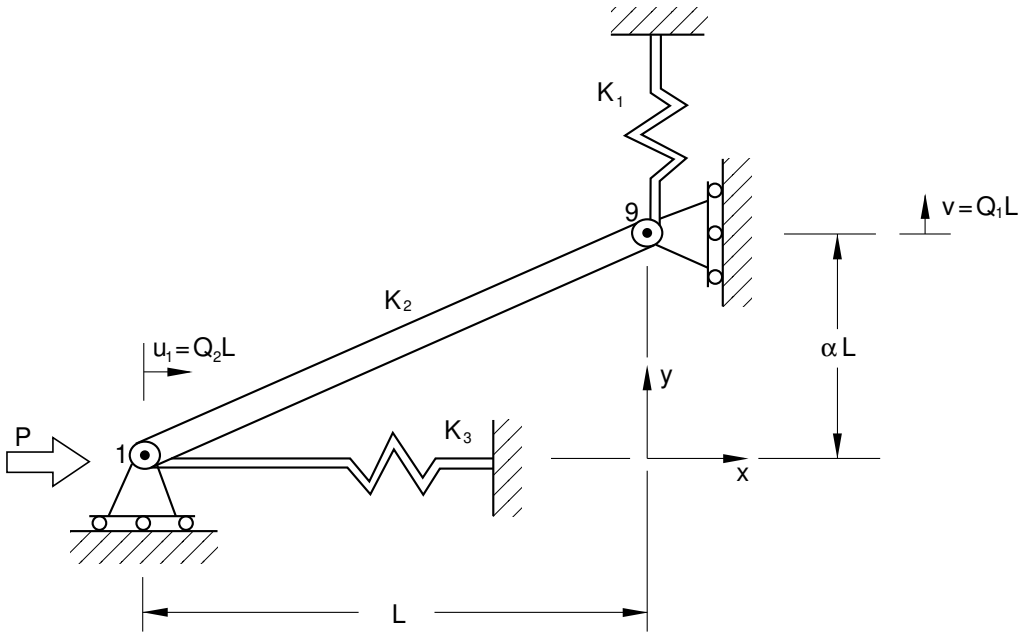
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

T2D2

Problem description



Model: AE = 5.0×10^7, L = 2500, αL = 25, K_1= 1.5, $K_2 = AE/L(1 + \alpha^2)^{1/2}$ = 19999.0, K_3 = 2.0.

Boundary conditions: $u_y = 0$ at node 1, $u_x = 0$ at node 9.

Loading: A load is applied to node 1 in the *x*-direction.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test NL3 from NAFEMS Publication NNB, Rev. 1, "NAFEMS Non-Linear Benchmarks," October 1989.

Step	P	u_1	v
1	1100	0.1803	10.37
2	2200	0.7160	35.45
3	3300	8.515	180.7
4	4400	1339	2189
5	5500	3468	2280
6	6600	4986	236.3

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Step	P	u_1	v
1	1100	0.1802 (–0.06%)	10.37 (0.0%)
2	2200	0.7157 (–0.0%)	35.44 (–0.03%)
3	3300	8.564 (+0.58%)	181.3 (+0.33%)
4	4400	1339 (0.0%)	2189 (0.0%)
5	5500	3469 (+0.03%)	2280 (0.0%)
6	6600	4985 (–0.02%)	244.7 (+3.55%)

Input files

nnl3xf2x.inp T2D2 elements.

NL4: Snap-back under displacement control

Products: Abaqus/Standard

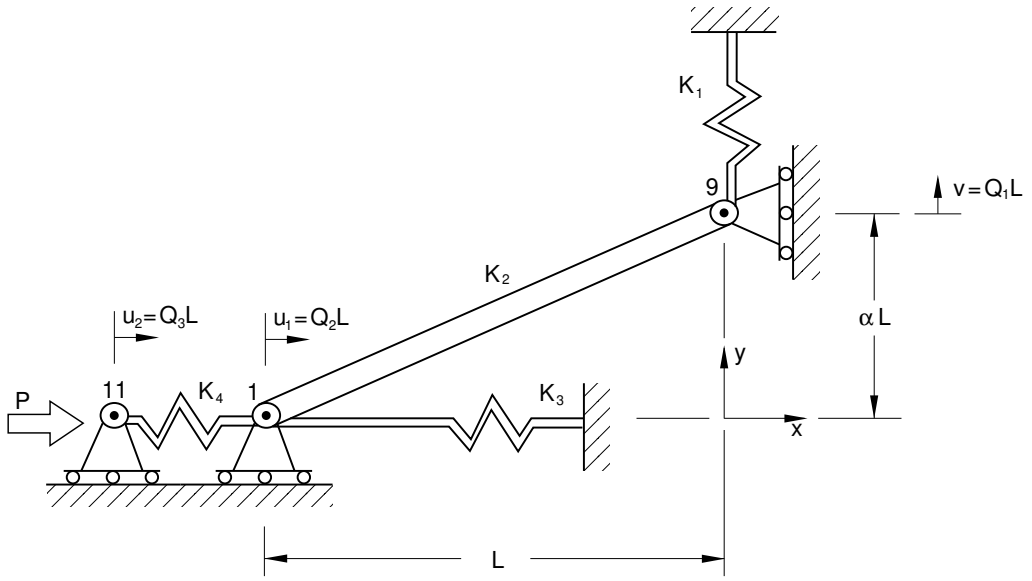
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

T2D2

Problem description



Model: AE = 5.0×10^7, L = 2500, αL = 25, K_1 = 1.5, K_2 = AE/L$(1 + \alpha^2)^{1/2}$ = 19999.0, K_3 = 0.25, K_4 = 1.0

Boundary conditions: $u_y =$ 0 at node 1 and node 11, $u_x = 0$ at node 9.

Loading: A load is applied to node 11 in the *x*-direction.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test NL4 from NAFEMS Publication NNB, Rev. 1, "NAFEMS Non-Linear Benchmarks," October 1989.

P	u_1	u_2	v
649.9	0.0904	650	5.241

P	u_1	u_2	v
1300	0.2328	1300	13.26
1949	0.5149	1950	27.08
2599	1.334	2600	56.50
3243	7.089	3250	162.6
–1099	4999	3900	41.95

Results and discussion

The RIKS algorithm was used for this problem. In this case it is not possible to obtain the solution at particular force values. The results below were obtained from the Abaqus results by linear interpolation between the two nearest increments. Therefore, there is some error associated with this interpolation procedure. The values enclosed in parentheses are percentage differences with respect to the reference solution.

P	u_1	u_2	v
649.9	0.0906 (+0.22%)	650 (0.0%)	5.254 (+0.25%)
1300	0.2333 (+0.21%)	1300 (0.0%)	13.29 (+0.23%)
1949	0.5152 (+0.06%)	1949 (–0.05%)	27.08 (0.0%)
2599	1.337 (+0.22%)	2600 (0.0%)	56.57 (+0.12%)
3243	7.241 (+2.14%)	3250 (0.0%)	163.9 (+0.08%)
–1099	4999 (0.0%)	3900 (0.0%)	43.49 (+3.67%)

Input files

nnl4xf2x.inp T2D2 elements.

NL5: Straight cantilever with end moment

Products: Abaqus/Standard

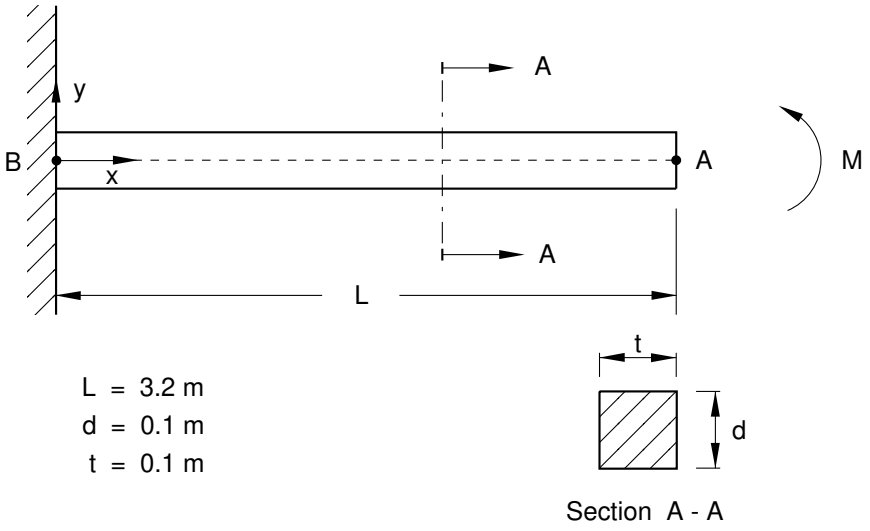
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

B22

Problem description



Material: Linear elastic, Young's modulus = 210 GPa, Poisson's ratio = 0.0.

Boundary conditions: $u_x = u_y = \phi_z = 0$ at point B.

Loading: A concentrated moment at point A applied in increments up to a maximum value of

$$\frac{ML}{2\pi EI} = 1.0, \text{ where } I = \frac{1}{12}td^3.$$

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test NL5 from NAFEMS Publication NNB, Rev. 1, “NAFEMS Non-Linear Benchmarks,” October 1989.

$\frac{ML}{2\pi EI}$	Deformation at A		
	$\frac{u_x}{L}$	$\frac{u_y}{L}$	$\frac{\phi_z}{2\pi}$
0.5	1.000	0.637	0.500
1.0	1.000	0.000	1.000

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

$\frac{ML}{2\pi EI}$	Deformation at A		
	$\frac{u_x}{L}$	$\frac{u_y}{L}$	$\frac{\phi_z}{2\pi}$
0.5	1.000 (0.0%)	0.637 (0.0%)	0.500 (0.0%)
1.0	1.000 (0.0%)	2.0×10^{-7}	1.000 (0.0%)

Input files

nnl5x22x.inp B22 elements.

NL6: Straight cantilever with axial end point load

Products: Abaqus/Standard

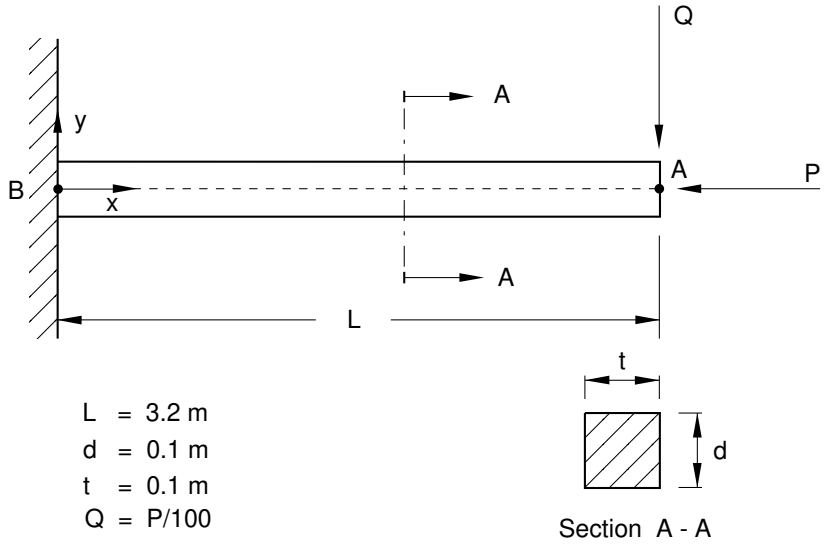
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

B22

Problem description



Material: Linear elastic, Young's modulus = 210 GPa, Poisson's ratio = 0.0.

Boundary conditions: $u_x = u_y = \phi_z = 0$ at point B.

Loading: A concentrated load at point A applied in increments up to a maximum value of

$$\frac{PL^2}{EI} = 22.493, \text{ where } I = \frac{1}{12}td^3.$$

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test NL6 from NAFEMS Publication NNB, Rev. 1, “NAFEMS Non-Linear Benchmarks,” October 1989.

$\frac{PL^2}{EI}$	Deformation at A		
	$\frac{u_x}{L}$	$\frac{u_y}{L}$	$\frac{\phi_z}{\pi}$
3.190	−0.440	0.719	0.444
22.493	−1.577	0.421	0.978

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

$\frac{PL^2}{EI}$	Deformation at A		
	$\frac{u_x}{L}$	$\frac{u_y}{L}$	$\frac{\phi_z}{\pi}$
3.190	−0.447 (+1.59%)	0.723 (+0.56%)	0.448 (+0.90%)
22.493	−1.577 (0.0%)	0.427 (+1.43%)	0.975 (−0.31%)

Input files

nnl6x22x.inp B22 elements.

NL7: Lee's frame buckling problem

Products: Abaqus/Standard

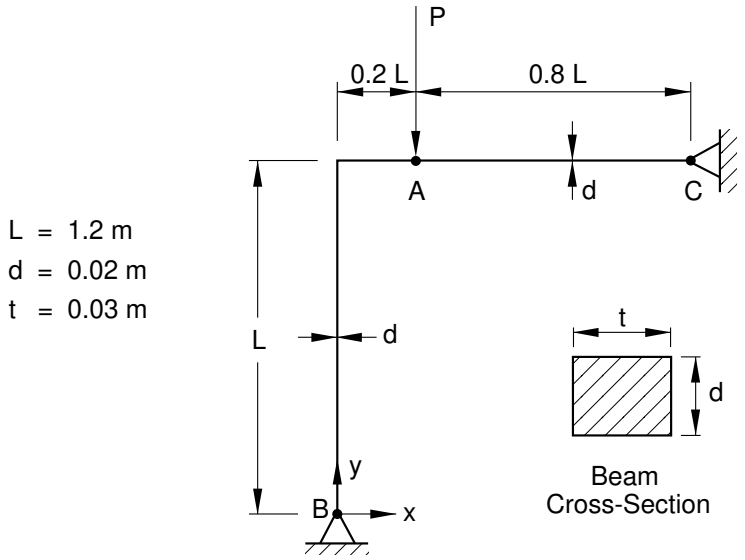
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

B22

Problem description



Material: Linear elastic, Young's modulus = 71.74 GPa, Poisson's ratio = 0.0.

Boundary conditions: $u_x = u_y =$ 0 at points B and C.

Loading: A concentrated load is applied incrementally at point A using the RIKS algorithm.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test NL7 from NAFEMS Publication NNB, Rev. 1, “NAFEMS Non-Linear Benchmarks,” October 1989.

$\frac{PL^2}{EI}$	Deformation at A
	$\frac{u_y}{L}$
18.552	0.407
31.887	0.784

Results and discussion

The RIKS algorithm was used for this problem. In this case it is not possible to obtain the solution at particular values of load or displacement. The values shown below were interpolated from the two nearest points available. There may be some error associated with the interpolation procedure. The values enclosed in parentheses are percentage differences with respect to the reference solution.

$\frac{PL^2}{EI}$	Deformation at A
	$\frac{u_y}{L}$
18.542 (–0.05%)	0.407 (0.0%)
31.887	0.781 (–0.38%)

Input files

nnl7x22x.inp	B22 elements.

Two-dimensional test cases in linear elastic fracture mechanics

In this section:

- *Test 1.1: Center cracked plate in tension*
- *Test 1.2: Center cracked plate with thermal load*
- *Test 2.1: Single edge cracked plate in tension*
- *Test 3: Angle crack embedded in a plate*
- *Test 4: Cracks at a hole in a plate*
- *Test 5: Axisymmetric crack in a bar*
- *Test 6: Compact tension specimen*
- *Test 7.1: T-joint weld attachment*
- *Test 8.1: V-notch specimen in tension*

Test 1.1: Center cracked plate in tension

Products: Abaqus/Standard

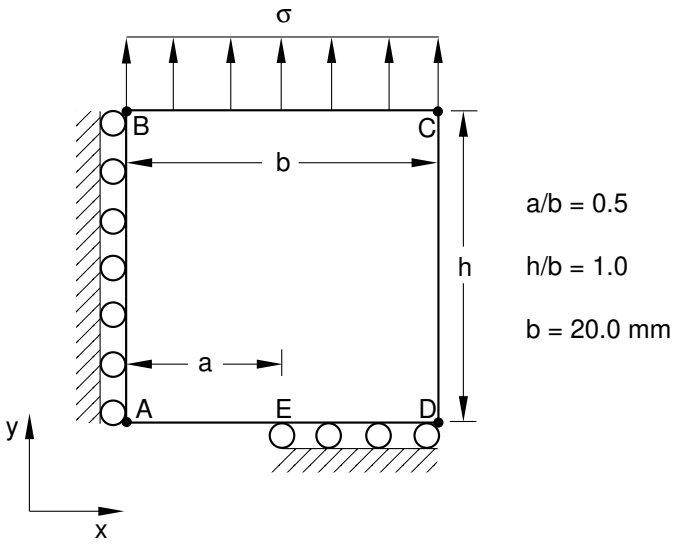
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CPE8 CPE8R

Problem description



Mesh: Collapsed elements with 1/4 point midside nodes are used at the crack tip. One-quarter of the test geometry is modeled. A coarse and a fine mesh are tested.

Material: Young's modulus = 207 GPa, Poisson's ratio = 0.3.

Boundary conditions: $u_x = 0$ along edge AB, $u_y = 0$ along edge DE.

Loading: Uniform stress, $\sigma_{yy} =$ 100 N/mm^2.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 1.1 from NAFEMS publication "2D Test Cases in Linear Elastic Fracture Mechanics," R0020.

Target solution: $K\boldsymbol{I}/K_0 = 1.325$, $K_0 = \sigma\sqrt{\pi a}$

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Element Type	$K\boldsymbol{I}/K_0$ (Coarse)	$K\boldsymbol{I}/K_0$ (Fine)
CPE8	1.325 (0.0%)	1.333 (+0.63%)
CPE8R	1.329 (+0.3%)	1.334 (+0.63%)

Remarks

$K\boldsymbol{I} = \sqrt{J\boldsymbol{E}/(1-\nu^2)}$. An average of the J values calculated by Abaqus, excluding the first contour, is used in reporting the results. Experience has shown that the crack-tip elements do not give sufficiently accurate results to give good estimates of the J-integral for the first contour.

Input files

nlf11f8c.inp	CPE8 elements, coarse mesh.
nlf11f8f.inp	CPE8 elements, fine mesh.
nlf11r8c.inp	CPE8R elements, coarse mesh.
nlf11r8f.inp	CPE8R elements, fine mesh.

Test 1.2: Center cracked plate with thermal load

Products: Abaqus/Standard

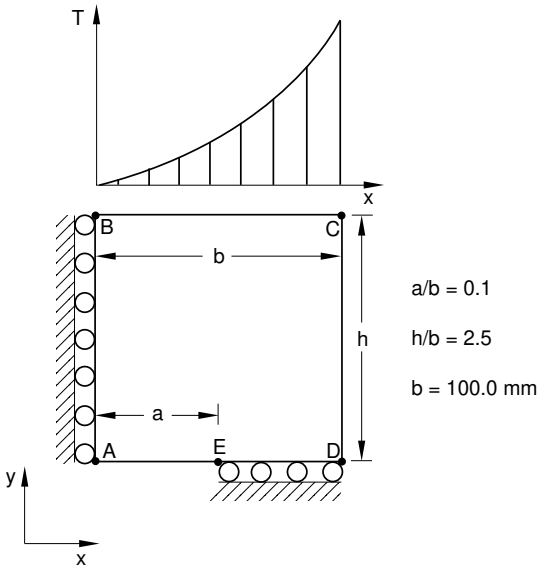
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CPS8 CPS8R

Problem description



Mesh: Collapsed elements with 1/4 point midside nodes are used at the crack tip. One-quarter of the test geometry is modeled.

Material: Young's modulus = 207 GPa, Poisson's ratio = 0.3, thermal expansion coefficent = 1.35E−5.

Boundary conditions: $u_x = 0$ along edge AB, $u_y = 0$ along edge DE.

Loading: Quadratic thermal distribution $T = T_0 \frac{x^2}{c^2} = 0.01x^2$, where $T_0 = 100$, c = 100.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 1.2 from NAFEMS publication “2D Test Cases in Linear Elastic Fracture Mechanics,” R0020.

Target solution: $K\boldsymbol{I}/K_0 = 1.000$, $K_0 = \frac{E\alpha T}{c^2}\left(\frac{b^2}{3} - \frac{a^2}{2}\right)\sqrt{\pi a}$

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Element Type	$K\boldsymbol{I}/K_0$
CPS8	1.003 (+0.3%)
CPS8R	1.005 (+0.5%)

Remarks

$K\boldsymbol{I} = \sqrt{J\boldsymbol{E}}$. An average of the *J* values calculated by Abaqus, excluding the first contour, is used in reporting the results. Experience has shown that the crack-tip elements do not give sufficiently accurate results to give good estimates of the *J*-integral for the first contour. The thermal loading is applied with user subroutine *UTEMP*.

Input files

nlf12f8x.inp	CPS8 elements.
nlf12f8x.f	User subroutine *UTEMP* used in nlf12f8x.inp.
nlf12r8x.inp	CPS8R elements.
nlf12r8x.f	User subroutine *UTEMP* used in nlf12r8x.inp.

Test 2.1: Single edge cracked plate in tension

Products: Abaqus/Standard

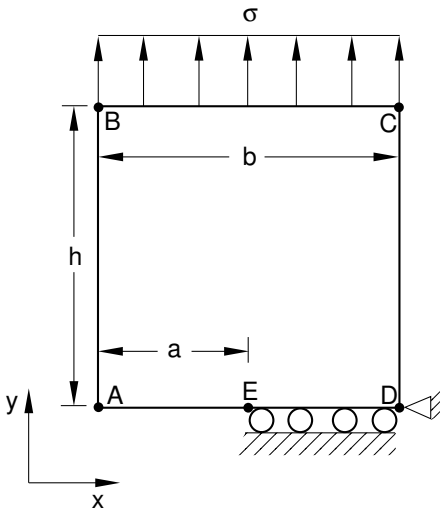
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CPE8 CPE8R

Problem description



a/b = 0.5

h/b = 0.5

b = 20.0 mm

Mesh: Collapsed elements with 1/4 point midside nodes are used at the crack tip. Half of the test geometry is modeled. A coarse and a fine mesh are tested.

Material: Young's modulus = 207 GPa, Poisson's ratio = 0.3.

Boundary conditions: $u_y = 0$ along edge ED, $u_x = 0$ at point D.

Loading: Uniform stress, σ_{yy} = 100 N/mm^2.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 2.1 from NAFEMS publication "2D Test Cases in Linear Elastic Fracture Mechanics," R0020.

Target solution: $K_I/K_0 = 3.0$, $K_0 = \sigma\sqrt{\pi a}$

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Element Type	Coarse Mesh	Fine Mesh
CPE8	2.911 (−2.95%)	3.007 (+0.25%)
CPE8R	2.938 (−2.08%)	3.008 (+0.26%)

Remarks

K$I = \sqrt{JE/(1-\nu^2)}$. An average of the J values calculated by Abaqus, excluding the first contour, is used in reporting the results. Experience has shown that the crack-tip elements do not give sufficiently accurate results to give good estimates of the J-integral for the first contour.

Input files

nlf21f8c.inp	CPE8 elements, coarse mesh.
nlf21f8f.inp	CPE8 elements, fine mesh.
nlf21r8c.inp	CPE8R elements, coarse mesh.
nlf21r8f.inp	CPE8R elements, fine mesh.

Test 3: Angle crack embedded in a plate

Products: Abaqus/Standard

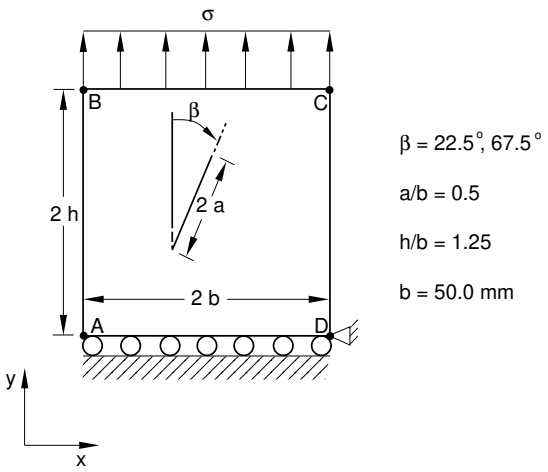
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CPE8 CPE8R

Problem description



Mesh: Collapsed elements with 1/4 point midside nodes are used at the crack tip. The complete test geometry is modeled.

Material: Young's modulus = 207 GPa, Poisson's ratio = 0.3.

Boundary conditions: $u_y = 0$ along edge AD, $u_x = 0$ at point D.

Loading: Uniform stress, σ_{yy} = 100 N/mm^2.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 3.1 and 3.2 from NAFEMS publication "2D Test Cases in Linear Elastic Fracture Mechanics," R0020.

Target solution (β = 22.5°): K_I/K_0 = 0.190, K_{II}/K_0 = 0.405, $K_0 = \sigma\sqrt{\pi a}$

Target solution (β = 67.5°): K_I/K_0 = 1.030, K_{II}/K_0 = 0.370, $K_0 = \sigma\sqrt{\pi a}$

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

β	Element Type	K_I/K_0	K_{II}/K_0
22.5°	CPE8	0.185 (−2.9%)	0.405 (+0.1%)
22.5°	CPE8R	0.184 (−2.9%)	0.407 (+0.4%)
67.5°	CPE8	1.035 (+0.2%)	0.364 (−1.7%)
67.5°	CPE8R	1.038 (+0.8%)	0.368 (−0.5%)

Remarks

$K_I = \sqrt{\frac{JE}{(1+R^2)(1-\nu^2)}}$, $K_{II} = RK_I$, $R = \frac{\mathbf{t}\cdot(\mathbf{u}^+ - \mathbf{u}^-)}{\mathbf{n}\cdot(\mathbf{u}^+ - \mathbf{u}^-)}$.

An average of the J values calculated by Abaqus, excluding the first contour, is used in reporting the results. Experience has shown that the crack-tip elements do not give sufficiently accurate results to give good estimates of the J-integral for the first contour. $\mathbf{u}^+$ and $\mathbf{u}^-$ are the displacements of nodes on the positive and negative sides of the crack, respectively, that are initially located at the same position in the undeformed state. An average value for R based on the first five nodal locations behind (not including) the crack tip was used in the calculations. $\mathbf{t}$ and $\mathbf{n}$ are the tangent and normal, respectively, to the direction of crack propagation.

Input files

nlf31f8x.inp	CPE8 elements, Test 3.1.
nlf31r8x.inp	CPE8R elements, Test 3.1.
nlf32f8x.inp	CPE8 elements, Test 3.2.
nlf32r8x.inp	CPE8R elements, Test 3.2.

Test 4: Cracks at a hole in a plate

Products: Abaqus/Standard

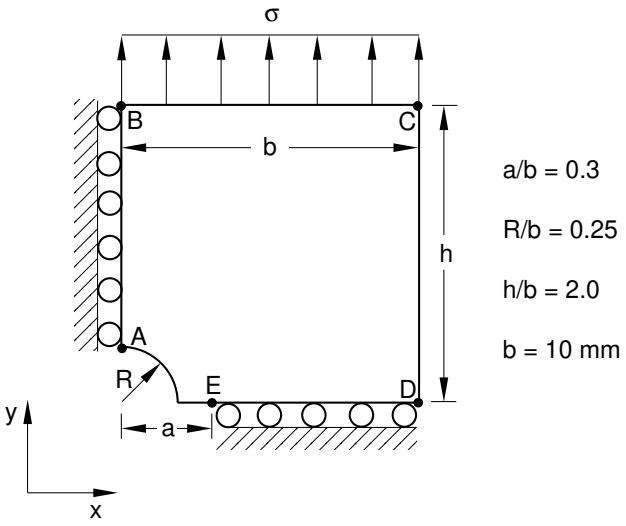
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CPE8 CPE8R CPS8 CPS8R

Problem description



Mesh: Collapsed elements with 1/4 point midside nodes are used at the crack tip. One-quarter of the test geometry is modeled.

Material: Young's modulus = 207 GPa, Poisson's ratio = 0.3

Boundary conditions: $u_x = 0$ along edge AB, $u_y = 0$ along edge ED.

Loading: Uniform stress, σ_{yy} = 100 N/mm^2.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 4.1 and 4.2 from NAFEMS publication "2D Test Cases in Linear Elastic Fracture Mechanics," R0020.

Target solution: KI/K$_0$ = 1.050, K$_0$ = $\sigma\sqrt{\pi a}$

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Element Type	K_I/K_0
CPE8	1.060 (+0.9%)
CPE8R	1.059 (+0.8%)
CPS8	1.060 (+1.0%)
CPS8R	1.059 (+0.8%)

Remarks

$K_I = \sqrt{JE'}$. $E' = E$ for plane stress, $E' = E/\left(1-\nu^2\right)$ for plane strain. An average of the J values calculated by Abaqus, excluding the first contour, is used in reporting the results. Experience has shown that the crack-tip elements do not give sufficiently accurate results to give good estimates of the J-integral for the first contour.

Input files

nlf41f8x.inp	CPE8 elements.
nlf41r8x.inp	CPE8R elements.
nlf42f8x.inp	CPS8 elements.
nlf42r8x.inp	CPS8R elements.

Test 5: Axisymmetric crack in a bar

Products: Abaqus/Standard

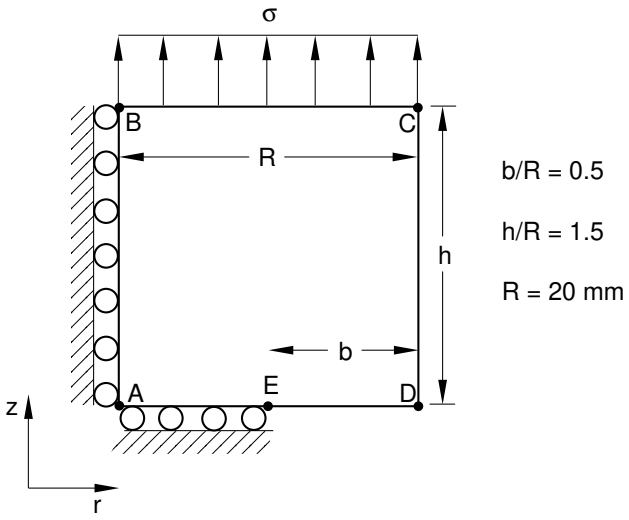
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CAX8 CAX8R

Problem description



Mesh: Collapsed elements with 1/4 point midside nodes are used at the crack tip. Half of the axisymmetric test geometry is modeled.

Material: Young's modulus = 207 GPa, Poisson's ratio = 0.3.

Boundary conditions: $u_r = 0$ along edge AB, $u_z = 0$ along edge AE.

Loading: Uniform stress, σ_{yy} = 100 N/mm^2.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 5 from NAFEMS publication "2D Test Cases in Linear Elastic Fracture Mechanics," R0020.

Target solution: $K\boldsymbol{I}/K_0 = 0.475$, $K_0 = \frac{\sigma R^2\sqrt{\pi b}}{b^2}$

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Element Type	$K\boldsymbol{I}/K_0$
CAX8	0.484 (+1.87%)
CAX8R	0.485 (+2.06%)

Remarks

$K\boldsymbol{I} = \sqrt{J\boldsymbol{E}/(1-\nu^2)}$. An average of the J values calculated by Abaqus, excluding the first contour, is used in reporting the results. Experience has shown that the crack-tip elements do not give sufficiently accurate results to give good estimates of the J-integral for the first contour.

Input files

nlf5xf8x.inp	CAX8 elements.
nlf5xr8x.inp	CAX8R elements.

Test 6: Compact tension specimen

Products: Abaqus/Standard

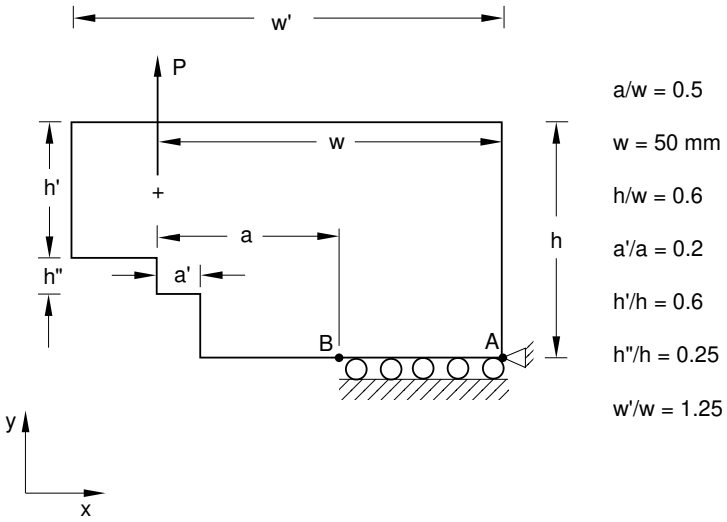
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CPE8 CPE8R

Problem description



Mesh: Collapsed elements with 1/4 point midside nodes are used at the crack tip. Half of the test geometry is modeled.

Material: Young's modulus = 207 GPa, Poisson's ratio = 0.3.

Boundary conditions: $u_x = 0$ at point A, $u_y = 0$ along edge BA.

Loading: Concentrated force, P = 1000 N.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 6 from NAFEMS publication "2D Test Cases in Linear Elastic Fracture Mechanics," R0020.

Target solution: $KI/K_0 = 9.659$, $K_0 = \frac{P}{t\sqrt{w}}$

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Element Type	K_I/K_0
CPE8	9.572 (–0.9%)
CPE8R	9.639 (–0.2%)

Remarks

$K_I = \sqrt{JE/(1-\nu^2)}$. An average of the J values calculated by Abaqus, excluding the first contour, is used in reporting the results. Experience has shown that the crack-tip elements do not give sufficiently accurate results to give good estimates of the J-integral for the first contour.

Input files

nlf6xf8x.inp	CPE8 elements.
nlf6xr8x.inp	CPE8R elements.

Test 7.1: T-joint weld attachment

Products: Abaqus/Standard

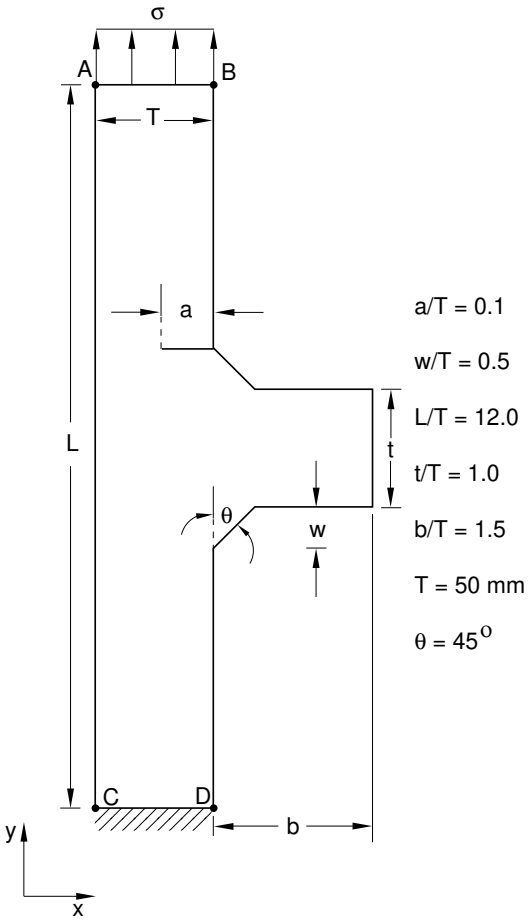
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CPE8 CPE8R

Problem description



Mesh: Collapsed elements with 1/4 point midside nodes are used at the crack tip. The full test geometry is modeled.

Material: Young's modulus = 207 GPa, Poisson's ratio = 0.3.

Boundary conditions: $u_x = 0$ along edge AB and CD, $u_y = 0$ along edge CD.

Loading: Uniform stress, σ_{yy} = 100 N/mm^2.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 7.1 from NAFEMS publication "2D Test Cases in Linear Elastic Fracture Mechanics," R0020.

Target solution: KI/K$_0$ = 1.317, K$_0$ = $\sigma\sqrt{\pi a}$

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Element Type	KI/K$_0$
CPE8	1.327 (+0.78%)
CPE8R	1.329 (+0.88%)

Remarks

KI = $\sqrt{JE/(1-\nu^2)}$. An average of the J values calculated by Abaqus, excluding the first contour, is used in reporting the results. Experience has shown that the crack-tip elements do not give sufficiently accurate results to give good estimates of the J-integral for the first contour.

Input files

nlf71f8x.inp	CPE8 elements.
nlf71r8x.inp	CPE8R elements.

Test 8.1: V-notch specimen in tension

Products: Abaqus/Standard

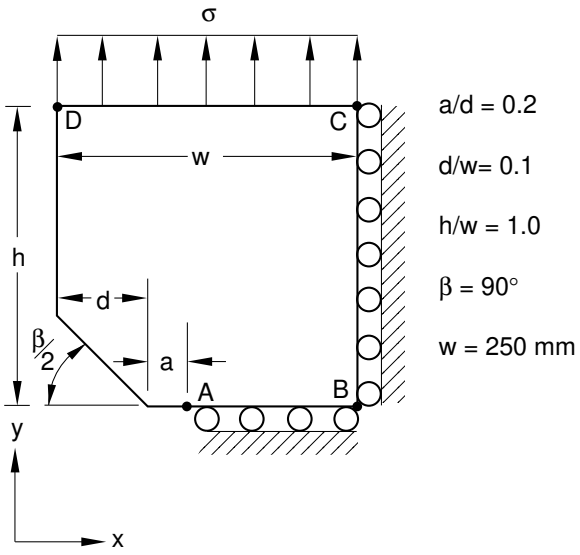
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CPE8 CPE8R

Problem description



a/d = 0.2

d/w= 0.1

h/w = 1.0

β = 90°

w = 250 mm

Mesh: Collapsed elements with 1/4 point midside nodes are used at the crack tip. One-quarter of the test geometry is modeled.

Material: Young's modulus = 207 GPa, Poisson's ratio = 0.3.

Boundary conditions: $u_x = 0$ along edge CB, $u_y = 0$ along edge AB.

Loading: Uniform stress, σ_{yy} = 100 N/mm^2.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 8.1 from NAFEMS publication "2D Test Cases in Linear Elastic Fracture Mechanics," R0020.

Target solution: $K\boldsymbol{I}/K_0 = 2.74$, $K_0 = \sigma\sqrt{\pi a}$

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Element Type	$K\boldsymbol{I}/K_0$
CPE8	2.793 (+1.94%)
CPE8R	2.794 (+1.97%)

Remarks

$K\boldsymbol{I} = \sqrt{J\boldsymbol{E}/(1-\nu^2)}$. An average of the J values calculated by Abaqus, excluding the first contour, is used in reporting the results. Experience has shown that the crack-tip elements do not give sufficiently accurate results to give good estimates of the J-integral for the first contour.

Input files

nlf81f8x.inp	CPE8 elements.
nlf81r8x.inp	CPE8R elements.

Fundamental tests of creep behavior

In this section:

- *Test 1A: 2D plane stress – uniaxial load, secondary creep*
- *Test 1B: 2D plane stress – uniaxial displacement, secondary creep*
- *Test 2A: 2D plane stress – biaxial load, secondary creep*
- *Test 2B: 2D plane stress – biaxial displacement, secondary creep*
- *Test 3A: 2D plane stress – biaxial (negative) load, secondary creep*
- *Test 3B: 2D plane stress – biaxial (negative) displacement, secondary creep*
- *Test 4A: 2D plane stress – biaxial (double) load, secondary creep*
- *Test 4B: 2D plane stress – biaxial (double) displacement, secondary creep*
- *Test 4C: 2D plane stress – shear loading, secondary creep*
- *Test 5A: 2D plane strain – biaxial load, secondary creep*
- *Test 5B: 2D plane strain – biaxial displacement, secondary creep*
- *Test 6A: 3D – triaxial load, secondary creep*
- *Test 6B: 3D – triaxial displacement, secondary creep*
- *Test 7: Axisymmetric – pressurized cylinder, secondary creep*
- *Test 8A: 2D plane stress – uniaxial load, primary creep*
- *Test 8B: 2D plane stress – uniaxial displacement, primary creep*
- *Test 8C: 2D plane stress – stepped load, primary creep*
- *Test 9A: 2D plane stress – biaxial load, primary creep*
- *Test 9B: 2D plane stress – biaxial displacement, primary creep*
- *Test 9C: 2D plane stress – biaxial stepped load, primary creep*
- *Test 10A: 2D plane stress – biaxial (negative) load, primary creep*
- *Test 10B: 2D plane stress – biaxial (negative) displacement, primary creep*
- *Test 10C: 2D plane stress – biaxial (negative) stepped load, primary creep*
- *Test 11: 3D – triaxial load, primary creep*
- *Test 12A: 2D plane stress – uniaxial load, primary-secondary creep*
- *Test 12B: 2D plane stress – uniaxial displacement, primary-secondary creep*
- *Test 12C: 2D plane stress – stepped load, primary-secondary creep*

Test 1A: 2D plane stress – uniaxial load, secondary creep

Products: Abaqus/Standard

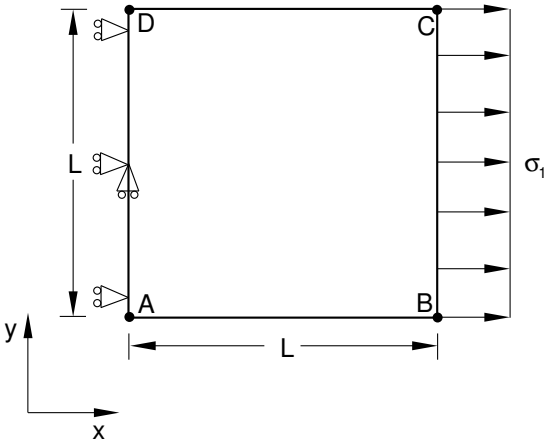
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CPS8R CPS6 CPS6M

Problem description



Plane stress

L = 100 mm

σ_1 = 200 N/mm^2

Material: Young's modulus = 200 × 10^3 N/mm^2, Poisson's ratio = 0.3, Creep law: $\varepsilon = A\sigma^n t$, A = 3.125 × 10^{-14} per hour (σ in N/mm^2), n = 5.

Boundary conditions: $u_x = 0$ on line AD and $u_y = 0$ at midpoint of line AD.

Loading: Prescribed tensile stress σ_1 = 200 N/mm^2 on line BC.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 1(a) from NAFEMS Publication Ref: R0027, "NAFEMS Fundamental Tests of Creep Behaviour," June 1993.

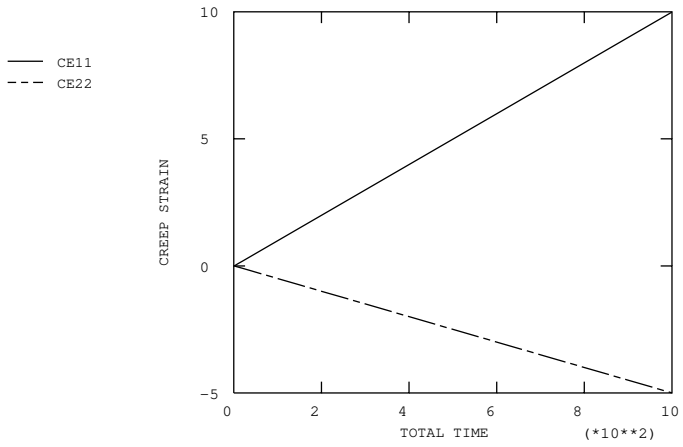
$$\varepsilon^c_{xx} = \varepsilon^c_{eff} = 0.01\,t$$

$$\varepsilon^c_{yy} = \varepsilon^c_{zz} = -0.005\,t$$

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Abaqus Results		
t	ε^c_{xx}	ε^c_{yy}
0.00	0.000 (0.00%)	0.000 (0.00%)
0.52	0.005 (0.00%)	−0.003 (0.00%)
8.39	0.084 (0.00%)	−0.042 (0.00%)
33.55	0.336 (0.00%)	−0.168 (0.00%)
134.22	1.342 (0.00%)	−0.671 (0.00%)
536.87	5.369 (0.00%)	−2.684 (0.00%)
1000.00	10.000 (0.00%)	−5.000 (0.00%)



Remarks

The total creep time for this test is 1000 hours. The times listed in the above table are the times calculated by the Abaqus automatic time stepping algorithm with CETOL = $5. \times 10^{-3}$.

Input files

ncr1ar8x.inp	CPS8R elements.
ncr1ar6x.inp	CPS6 elements.
ncr1ar6m.inp	CPS6M elements.

Test 1B: 2D plane stress – uniaxial displacement, secondary creep

Products: Abaqus/Standard

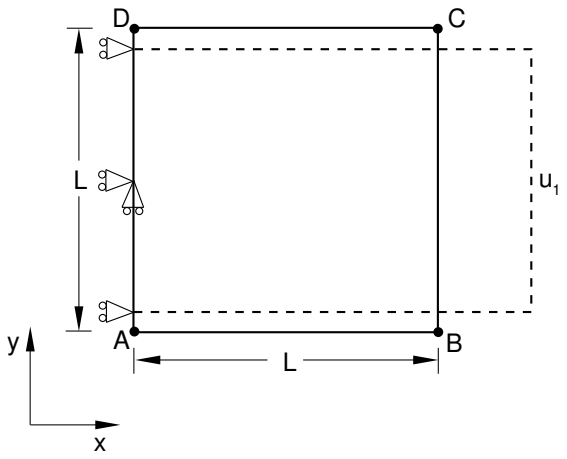
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CPS8R

Problem description



Plane stress

L = 100 mm

u_1 = 0.1 mm

Material: Young's modulus = 200×10^3 N/mm^2, Poisson's ratio = 0.3, Creep law: $\varepsilon = A\sigma^n t$, A = 3.125×10^{-14} per hour (σ in N/mm^2), n = 5.

Boundary conditions: $u_x = 0$ on line AD, $u_y = 0$ at mid-point of line AD and $u_x = 0.1$ on line BC.

Reference solution

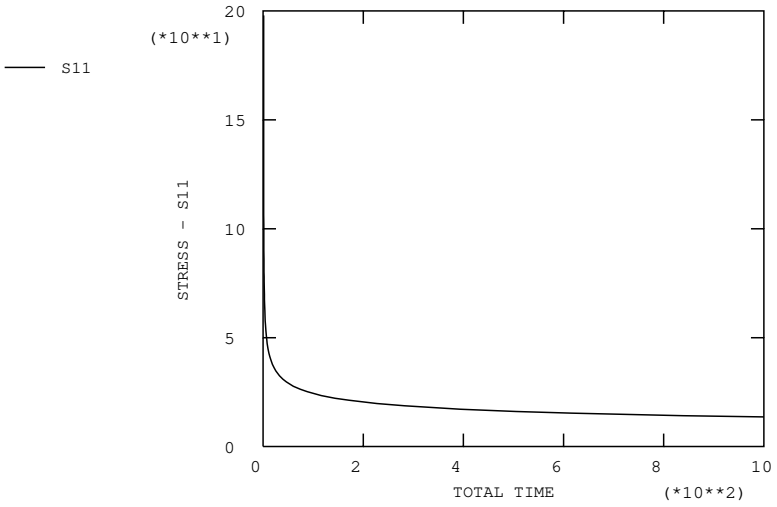
This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 1(b) from NAFEMS Publication Ref: R0027, "NAFEMS Fundamental Tests of Creep Behaviour," June 1993.

$$\sigma_{xx} = \sigma_{eff} = \left[\left(6.25 \times 10^{-10}\right) + \left(2.5 \times 10^{-8}\right)\, t\,\right]^{-\frac{1}{4}}$$

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Abaqus Results	
t	σ_{xx}
0.00	200.00 (0.00%)
0.1359	127.35 (1.41%)
1.2918	75.71(1.98%)
6.3455	51.27 (2.42%)
48.736	30.94 (2.80%)
372.67	18.71 (3.38%)
1000.00	14.67 (3.75%)



Remarks

The total creep time for this test is 1000 hours. The times listed in the above table are the times calculated by the Abaqus automatic time stepping algorithm with CETOL = $5. \times 10^{-6}$.

Input files

ncr1br8x.inp	CPS8R elements.

Test 2A: 2D plane stress – biaxial load, secondary creep

Products: Abaqus/Standard

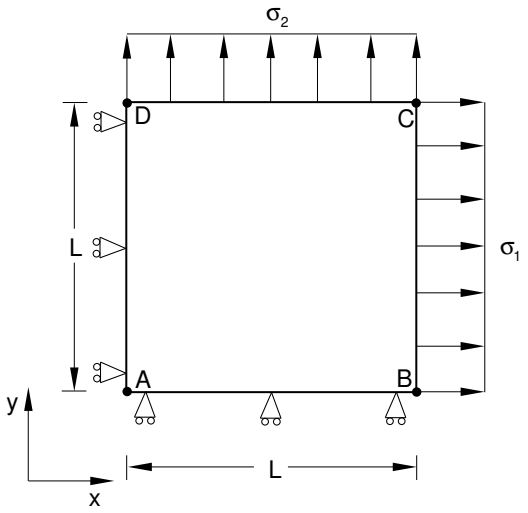
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CPS8R

Problem description



Plane stress

L = 100 mm

$\sigma_1 = \sigma_2 = 200$ N/mm^2

Material: Young's modulus = 200×10^3 N/mm^2, Poisson's ratio = 0.3, Creep law: $\boldsymbol{\varepsilon} = A\boldsymbol{\sigma}^n \boldsymbol{t}$, A = 3.125×10^{-14} per hour ($\boldsymbol{\sigma}$ in N/mm^2), $n = 5$.

Boundary conditions: $u_x = 0$ on line AD and $u_y = 0$ on line AB.

Loading: Prescribed tensile stress $\boldsymbol{\sigma_1}$ = 200 N/mm^2 on line BC. Prescribed tensile stress $\boldsymbol{\sigma_2}$ = 200 N/mm^2 on line DC.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 2(a) from NAFEMS Publication Ref: R0027, "NAFEMS Fundamental Tests of Creep Behaviour," June 1993.

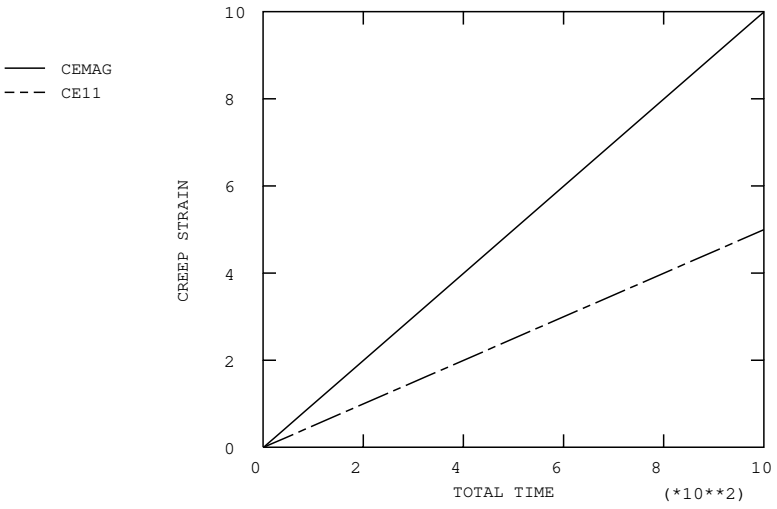
$$\varepsilon^c_{xx} = \varepsilon^c_{yy} = 0.005\,t$$

$$\varepsilon^c_{eff} = -\varepsilon^c_{zz} = 0.01\,t$$

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Abaqus Results		
t	ε^c_{xx}	ε^c_{eff}
0.00	0.000 (0.00%)	0.000 (0.00%)
1.05	0.005 (0.00%)	0.010 (0.00%)
8.39	0.042 (0.00%)	0.084 (0.00%)
33.55	0.168 (0.00%)	0.336 (0.00%)
134.22	0.671 (0.00%)	1.342 (0.00%)
536.87	2.684 (0.00%)	5.369 (0.00%)
1000.00	5.000 (0.00%)	10.000 (0.00%)



Remarks

The total creep time for this test is 1000 hours. The times listed in the above table are the times calculated by the Abaqus automatic time stepping algorithm with CETOL = $5. \times 10^{-3}$.

Input files

ncr2ar8x.inp CPS8R elements.

Test 2B: 2D plane stress – biaxial displacement, secondary creep

Products: Abaqus/Standard

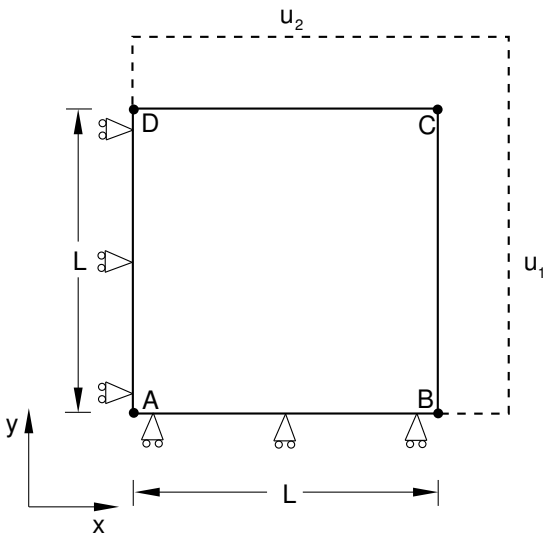
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CPS8R

Problem description



Plane stress

L = 100 mm

$u_1 = u_2 = 0.1$ mm

Material: Young's modulus = 200×10^3 N/mm^2, Poisson's ratio = 0.3, Creep law: $\varepsilon = A\sigma^n t$, A = 3.125×10^{-14} per hour (σ in N/mm^2), $n = 5$.

Boundary conditions: $u_x = 0$ on line AD, $u_y = 0$ on line AD, $u_x = 0.1$ on line BC and $u_y = 0.1$ on line CD.

Reference solution

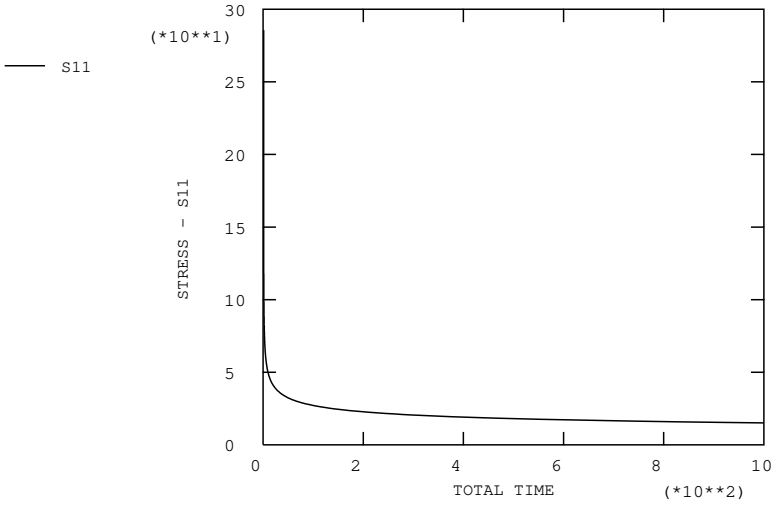
This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 2(b) from NAFEMS Publication Ref: R0027, “NAFEMS Fundamental Tests of Creep Behaviour,” June 1993.

The time marching program provided in Appendix B of the NAFEMS Publication can be used to obtain the stress variation with time.

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Abaqus Results	
t	σ_{xx}
0.00	285.71 (0.00%)
0.13	138.46 (0.74%)
5.31	55.40 (1.34%)
15.80	42.15 (0.52%)
74.52	28.42 (–0.04%)
544.28	17.05 (–1.69%)
1000.00	14.68 (–2.56%)



Remarks

The total creep time for this test is 1000 hours. The times listed in the above table are the times calculated by the Abaqus automatic time stepping algorithm with CETOL = $1. \times 10^{-5}$.

Input files

ncr2br8x.inp CPS8R elements.

Test 3A: 2D plane stress – biaxial (negative) load, secondary creep

Products: Abaqus/Standard

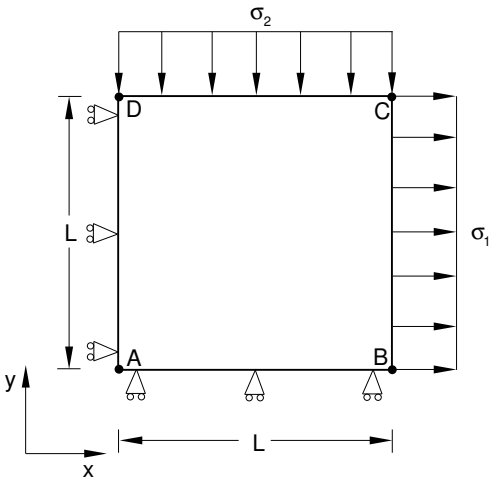
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CPS8R

Problem description



Plane stress

L = 100 mm

σ_1 = 200 N/mm^2

σ_2 = -200 N/mm^2

Material: Young's modulus = 200 × 10^3 N/mm^2, Poisson's ratio = 0.3, Creep law: $\boldsymbol{\varepsilon} = A\boldsymbol{\sigma}^n\boldsymbol{t}$, A = 3.125 × 10^{-14} per hour ($\boldsymbol{\sigma}$ in N/mm^2), n = 5.

Boundary conditions: $\boldsymbol{u_x = 0}$ on line AD and $\boldsymbol{u_y = 0}$ on line AB.

Loading: Prescribed tensile stress $\boldsymbol{\sigma_1}$ = 200 N/mm^2 on line BC and $\boldsymbol{\sigma_2}$ = –200 N/mm^2 on line CD.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 3(a) from NAFEMS Publication Ref: R0027, "NAFEMS Fundamental Tests of Creep Behaviour," June 1993.

$$\varepsilon^c_{xx} = -\varepsilon^c_{yy} = 0.135\,t$$

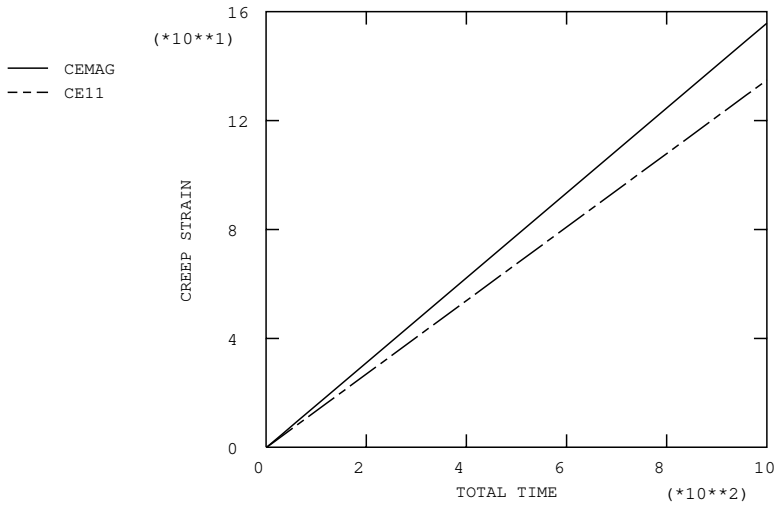
$$\varepsilon^c_{eff} = 0.1559\,t$$

$$\varepsilon^c_{yy} = 0.0$$

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Abaqus Results		
t	ε^c_{xx}	ε^c_{eff}
0.00	0.000 (0.00%)	0.000 (0.00%)
1.05	0.142 (0.00%)	0.163 (–0.01%)
16.78	2.265 (0.00%)	2.615 (–0.01%)
67.11	9.060 (0.00%)	10.461 (–0.01%)
134.22	18.120 (0.00%)	20.92 (–0.01%)
536.87	72.478 (0.00%)	83.690 (–0.01%)
1000.00	135.000 (0.00%)	155.880 (–0.01%)



Remarks

The total creep time for this test is 1000 hours. The times listed in the above table are the times calculated by the Abaqus automatic time stepping algorithm with CETOL = $5. \times 10^{-4}$.

Input files

ncr3ar8x.inp CPS8R elements.

Test 3B: 2D plane stress – biaxial (negative) displacement, secondary creep

Products: Abaqus/Standard

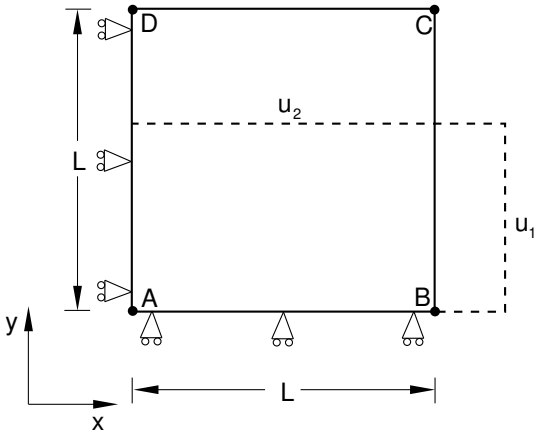
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CPS8R

Problem description



Plane stress

L = 100 mm

u_1 = 0.1 mm

u_2 = -0.1 mm

Material: Young's modulus = 200×10^3 N/mm^2, Poisson's ratio = 0.3, Creep law: $\varepsilon = A\sigma^n t$, A = 3.125×10^{-14} per hour (σ in N/mm^2), n = 5.

Boundary conditions: $u_x = 0$ on line AD, $u_y = 0$ on line AB, $u_x = 0.1$ on line BC and $u_y = -0.1$ on line CD.

Reference solution

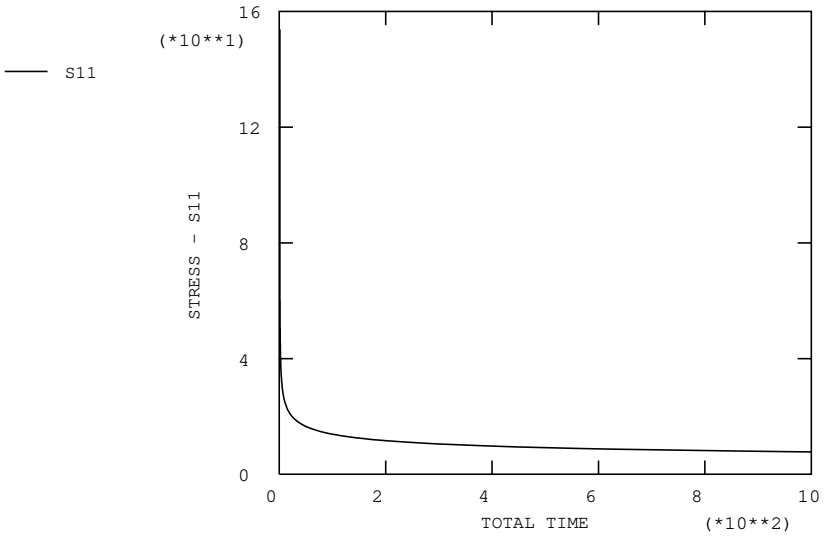
This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 3(b) from NAFEMS Publication Ref: R0027, "NAFEMS Fundamental Tests of Creep Behaviour," June 1993.

The time marching program provided in Appendix B of the NAFEMS Publication can be used to obtain the stress variation with time.

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Abaqus Results	
t	σ_{xx}
0.00	153.84 (–0.01%)
0.11	74.42 (–2.12%)
5.49	27.85 (–2.87%)
52.85	15.53 (–4.00%)
126.79	12.54 (–6.99%)
496.47	8.88 (–4.00%)
1000.00	7.43 (–3.88%)



Remarks

The total creep time for this test is 1000 hours. The times listed in the above table are the times calculated by the Abaqus automatic time stepping algorithm with CETOL = $1. \times 10^{-5}$.

Input files

ncr3br8x.inp CPS8R elements.

Test 4A: 2D plane stress – biaxial (double) load, secondary creep

Products: Abaqus/Standard

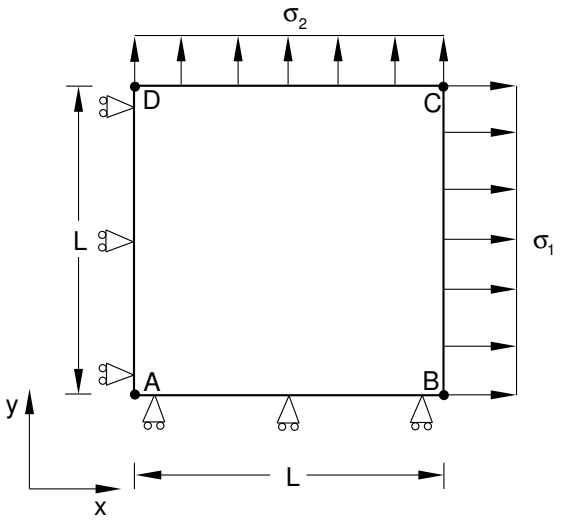
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CPS8R

Problem description



Plane stress

L = 100 mm

σ_1 = 200 N/mm^2

σ_2 = 100 N/mm^2

Material: Young's modulus = 200 × 10^3 N/mm^2, Poisson's ratio = 0.3, Creep law: $\boldsymbol{\varepsilon} = A\boldsymbol{\sigma}^n\boldsymbol{t}$, A = 3.125 × 10^{-14} per hour ($\boldsymbol{\sigma}$ in N/mm^2), n = 5.

Boundary conditions: $\boldsymbol{u_x = 0}$ on line AD and $\boldsymbol{u_y = 0}$ on line AB.

Loading: Prescribed tensile stress $\boldsymbol{\sigma_1}$ = 200 N/mm^2 on line BC and $\boldsymbol{\sigma_2}$ = 100 N/mm^2 on line CD.

Reference solution

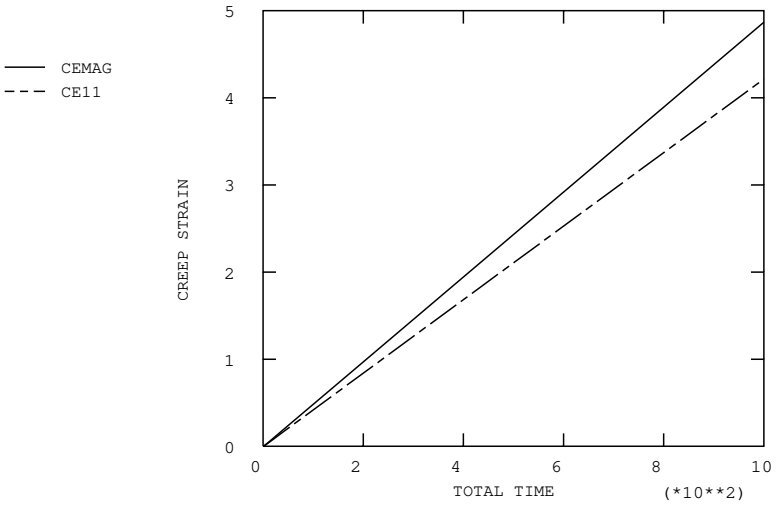
This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 4(a) from NAFEMS Publication Ref: R0027, “NAFEMS Fundamental Tests of Creep Behaviour,” June 1993.

$$\varepsilon^c_{xx} = -\varepsilon^c_{zz} = 0.004218\,t$$
$$\varepsilon^c_{eff} = 0.004871\,t$$
$$\varepsilon^c_{yy} = 0.0$$

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Abaqus Results		
t	ε^c_{xx}	ε^c_{eff}
0.00	0.000 (0.00%)	0.000 (0.00%)
4.19	0.018 (0.02%)	0.020 (0.01%)
16.78	0.071 (0.02%)	0.082 (0.01%)
67.11	0.283 (0.02%)	0.327 (0.01%)
134.22	0.566 (0.02%)	0.654 (0.01%)
536.87	2.265 (0.02%)	2.615 (0.01%)
1000.00	4.219 (0.02%)	4.871 (0.00%)



Remarks

The total creep time for this test is 1000 hours. The times listed in the above table are the times calculated by the Abaqus automatic time stepping algorithm with CETOL = $5. \times 10^{-4}$.

Input files

ncr4ar8x.inp CPS8R elements.

Test 4B: 2D plane stress – biaxial (double) displacement, secondary creep

Products: Abaqus/Standard

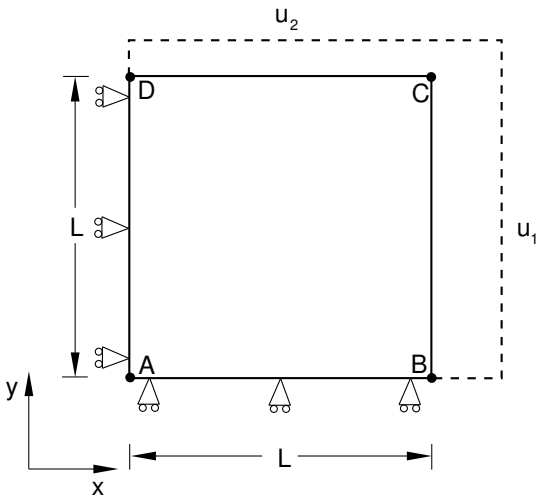
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CPS8R

Problem description



Plane stress

L = 100 mm

u_1 = 0.1 mm

u_2 = 0.05 mm

Material: Young's modulus = 200×10^3 N/mm^2, Poisson's ratio = 0.3, Creep law: $\varepsilon = A\sigma^n t$, A = 3.125×10^{-14} per hour (σ in N/mm^2), $n = 5$.

Boundary conditions: $u_x = 0$ on line AD, $u_y = 0$ on line AB, $u_x = 0.1$ on line BC and $u_y = 0.05$ on line CD.

Reference solution

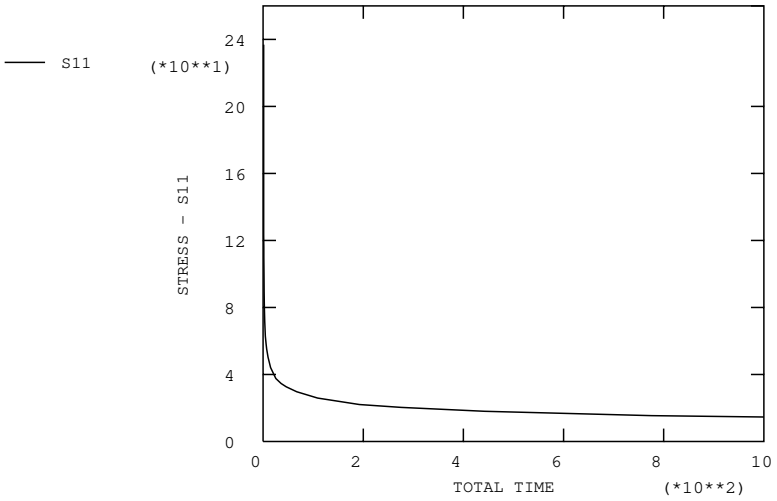
This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 4(b) from NAFEMS Publication Ref: R0027, "NAFEMS Fundamental Tests of Creep Behaviour," June 1993.

The time marching program provided in Appendix B of the NAFEMS Publication can be used to obtain the stress variation with time.

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Abaqus Results	
t	σ_{xx}
0.00	252.75 (0.00%)
0.10	159.07 (0.48%)
1.03	91.37 (0.66%)
5.00	60.98 (0.94%)
51.89	33.34 (0.24%)
524.11	18.38 (−0.20%)
1000.00	15.58 (0.19%)



Remarks

The total creep time for this test is 1000 hours. The times listed in the above table are the times calculated by the Abaqus automatic time stepping algorithm with CETOL = $1. \times 10^{-6}$.

Input files

ncr4br8x.inp	CPS8R elements.

Test 4C: 2D plane stress – shear loading, secondary creep

Products: Abaqus/Standard

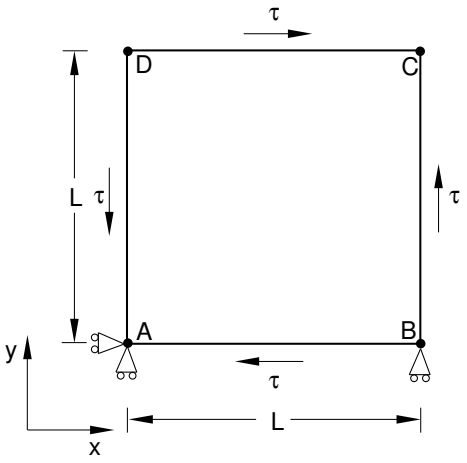
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CPS8R

Problem description



Plane stress

L = 100 mm

τ = 100 N/mm^2

Material: Young's modulus = 200×10^3 N/mm^2, Poisson's ratio = 0.3, Creep law: ε=A$\sigma^n t$, A = 3.125×10^{-14} per hour (σ in N/mm^2), n=5.

Boundary conditions: $u_x = u_y = 0$ at point A and $u_y = 0$ along line AB.

Loading: Prescribed shear forces: $\tau/6, 2\tau/3, \tau/6$ on line AB, $\tau/6, 2\tau/3, \tau/6$ on line BC, $\tau/6, 2\tau/3, \tau/6$ on line CD, and $\tau/6, 2\tau/3, \tau/6$ on line AD.

Reference solution

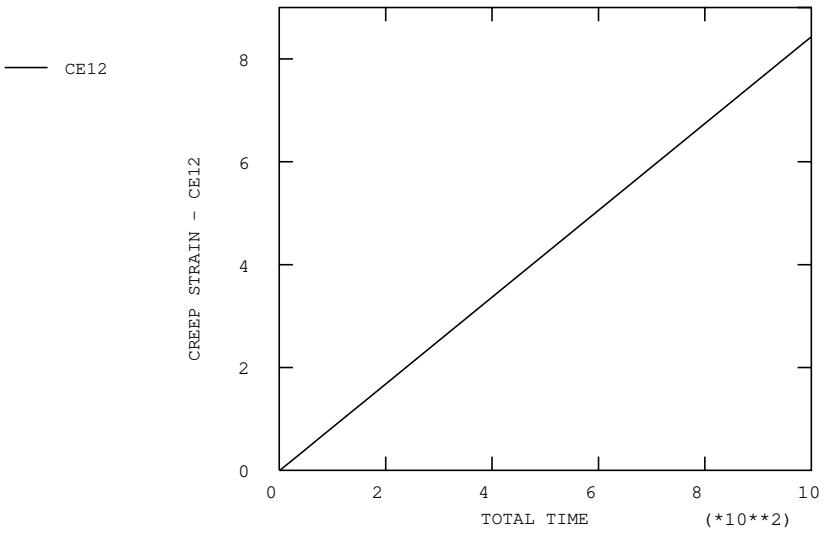
This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 4(c) from NAFEMS Publication Ref: R0027, "NAFEMS Fundamental Tests of Creep Behaviour," June 1993.

$$\gamma_{xy}^c = 0.008438\, t$$

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Abaqus Results	
t	γ_{xy}^{c}
0.00	0.000 (0.00%)
4.19	0.035 (–0.29%)
33.55	0.283 (0.00%)
134.22	1.133 (0.00%)
268.44	2.265 (–0.01%)
536.87	4.530 (0.00%)
1000.00	8.438 (0.00%)



Remarks

The total creep time for this test is 1000 hours. The times listed in the above table are the times calculated by the Abaqus automatic time stepping algorithm with CETOL=5. × 10^{-4}.

Input files

ncr4cr8x.inp CPS8R elements.

Test 5A: 2D plane strain – biaxial load, secondary creep

Products: Abaqus/Standard

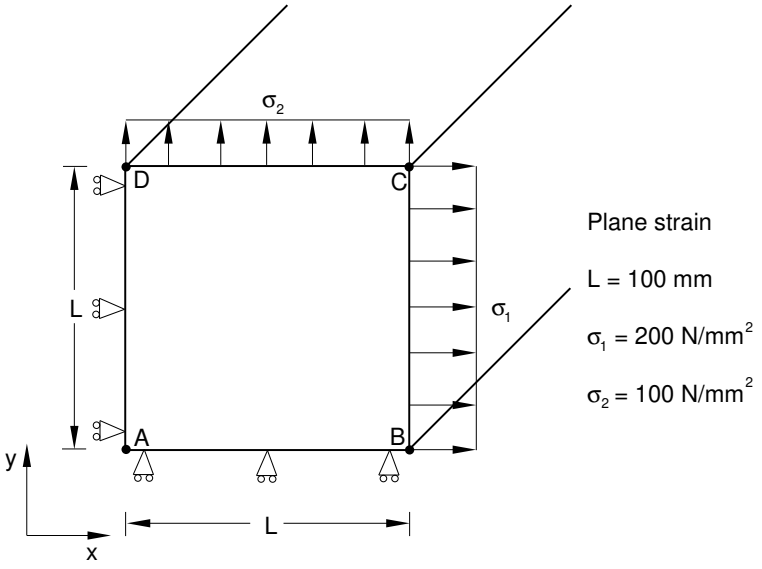
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CPE8R

Problem description



Material: Young's modulus = 200×10^3 N/mm^2, Poisson's ratio = 0.3, Creep law: $\varepsilon = A\sigma^n t$, A = 3.125×10^{-14} per hour (σ in N/mm^2), $n = 5$.

Boundary conditions: $u_x = 0$ on line AD and $u_y = 0$ on line AB.

Loading: Prescribed tensile stress σ_1 = 200 N/mm^2 on line BC and σ_2 = 100 N/mm^2 on line CD.

Reference solution

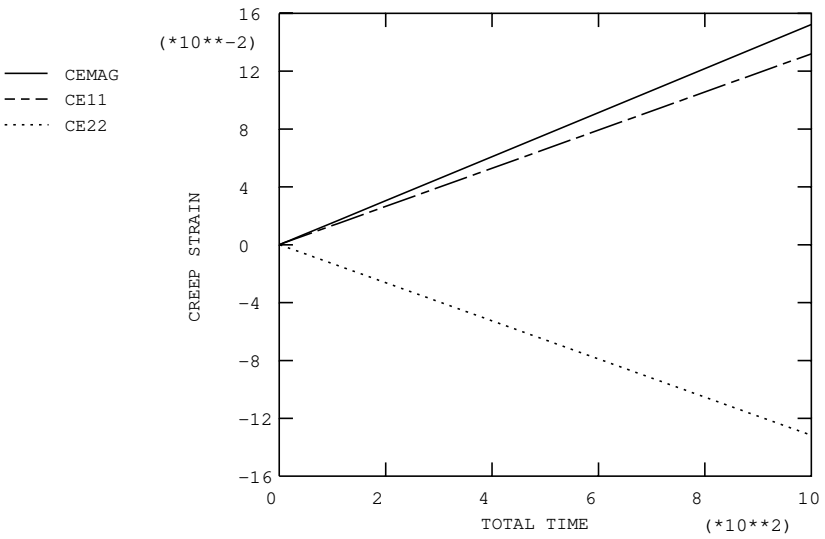
This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 5(a) from NAFEMS Publication Ref: R0027, "NAFEMS Fundamental Tests of Creep Behaviour," June 1993.

The time marching program provided in Appendix A of the NAFEMS Publication can be used to obtain the stress variation with time.

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Abaqus Results		
t	ε^c_{xx}	ε^c_{yy}
0.00	0.0000 (0.00%)	0.0000 (0.00%)
8.46	0.0014 (–1.04%)	–0.0010 (0.60%)
43.01	0.0059 (–0.17%)	–0.0056 (–0.19%)
107.58	0.0145 (–0.08%)	–0.0142 (–0.09%)
365.86	0.0485 (–0.02%)	–0.0482 (–0.02%)
710.23	0.0939 (0.00%)	–0.0936 (0.00%)
1000.00	0.1321 (0.00%)	–0.1318 (0.00%)



Remarks

The total creep time for this test is 1000 hours. The times listed in the above table are the times calculated by the Abaqus automatic time stepping algorithm with CETOL = $5. \times 10^{-4}$.

Input files

ncr5ar8x.inp CPE8R elements.

Test 5B: 2D plane strain – biaxial displacement, secondary creep

Products: Abaqus/Standard

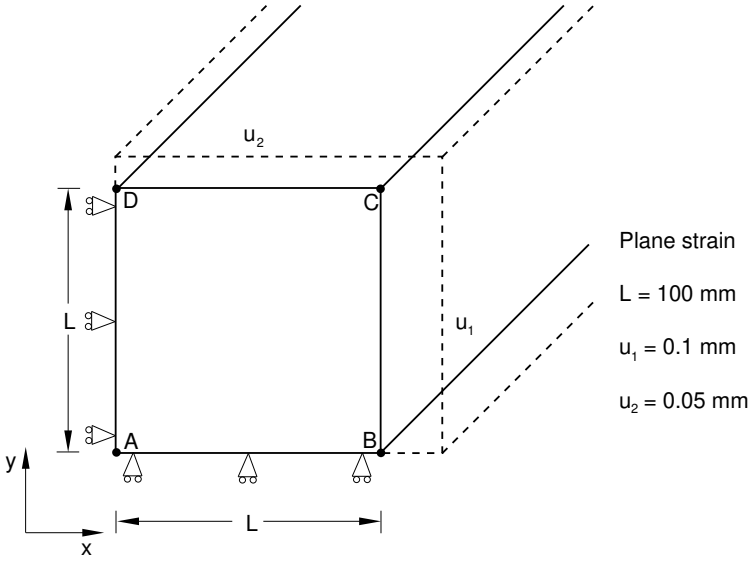
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CPE8R

Problem description



Material: Young's modulus = 200×10^3 N/mm^2, Poisson's ratio = 0.3, Creep law: $\varepsilon = A\sigma^n t$, A = 3.125×10^{-14} per hour (σ in N/mm^2), $n = 5$.

Boundary conditions: $u_x = 0$ on line AD, $u_y = 0$ on line AB, $u_x = 0.1$ on line BC and $u_y = 0.05$ on line CD.

Reference solution

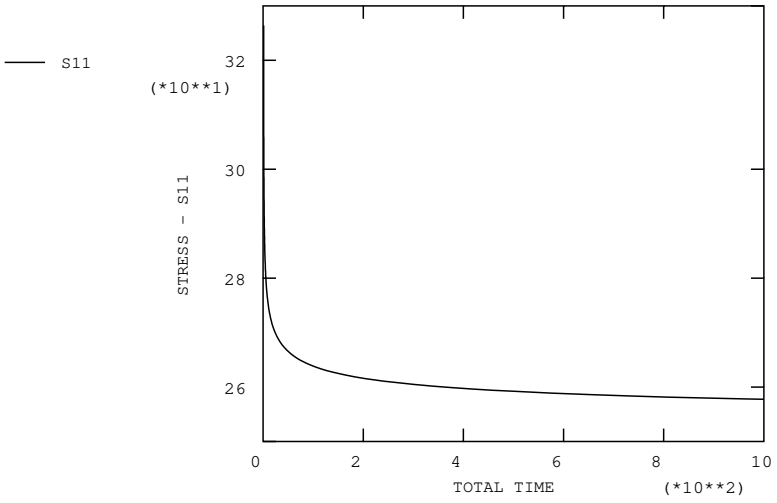
This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 5(b) from NAFEMS Publication Ref: R0027, "NAFEMS Fundamental Tests of Creep Behaviour," June 1993.

The time marching program provided in Appendix B of the NAFEMS Publication can be used to obtain the stress variation with time.

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Abaqus Results	
t	σ_{xx}
0.00	326.90 (0.00%)
0.13	313.96 (0.44%)
5.93	279.08 (0.47%)
35.29	268.89 (0.54%)
169.51	262.80 (0.39%)
572.16	259.48 (0.28%)
1000.00	258.24 (0.19%)



Remarks

The total creep time for this test is 1000 hours. The times listed in the above table are the times calculated by the Abaqus automatic time stepping algorithm with CETOL = $1. \times 10^{-5}$.

Input files

ncr5br8x.inp	CPE8R elements.

Test 6A: 3D – triaxial load, secondary creep

Products: Abaqus/Standard

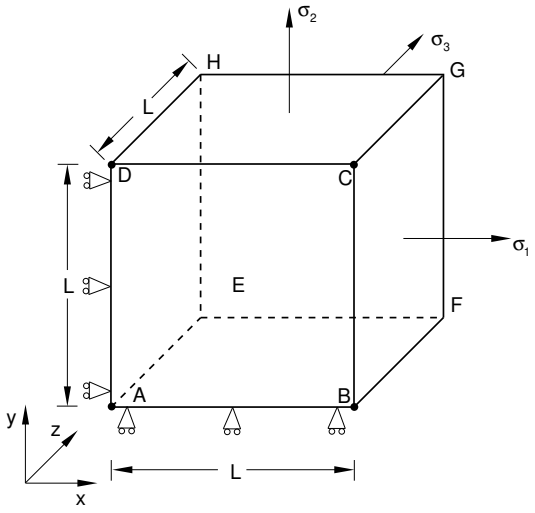
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

C3D20R

Problem description



Three-dimensional

L = 100 mm

σ_1 = 300 N/mm^2

σ_2 = 200 N/mm^2

σ_3 = 100 N/mm^2

Material: Young's modulus = 200×10^3 N/mm^2, Poisson's ratio = 0.3, Creep law: $\boldsymbol{\varepsilon} = A\boldsymbol{\sigma}^n\boldsymbol{t}$, A = 3.125×10^{-14} per hour ($\boldsymbol{\sigma}$ in N/mm^2), n = 5.

Boundary conditions: $\boldsymbol{u_x = 0}$ on face ADEH, $\boldsymbol{u_y = 0}$ on face ABFE and $\boldsymbol{u_z = 0}$ on face ABCD.

Loading: Prescribed tensile stress $\boldsymbol{\sigma_1}$ = 300 N/mm^2 on face BCGF, $\boldsymbol{\sigma_2}$ = 200 N/mm^2 on face CDHG, and $\boldsymbol{\sigma_3}$ = 100 N/mm^2 on face EFGH.

Reference solution

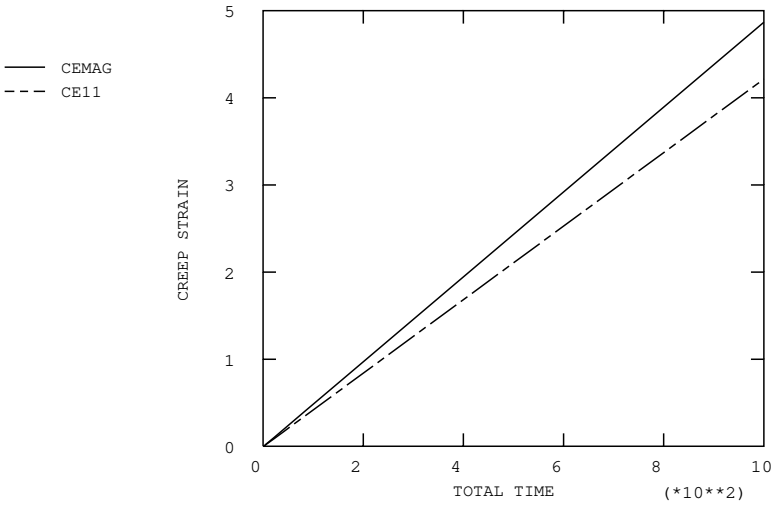
This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 6(a) from NAFEMS Publication Ref: R0027, “NAFEMS Fundamental Tests of Creep Behaviour,” June 1993.

$$\varepsilon^c_{xx} = -\varepsilon^c_{zz} = 0.004218\,t$$
$$\varepsilon^c_{eff} = 0.004871\,t$$
$$\varepsilon^c_{yy} = 0.0$$

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Abaqus Results		
t	ε^c_{xx}	ε^c_{eff}
0.00	0.0000 (0.00%)	0.0000 (0.00%)
8.39	0.0354 (0.02%)	0.0409 (0.01%)
33.55	0.1416 (0.02%)	0.1635 (0.01%)
134.22	0.5662 (0.02%)	0.6538 (0.01%)
536.87	2.2649 (0.03%)	2.6153 (0.01%)
805.31	3.3974 (0.02%)	3.9230 (0.01%)
1000.00	4.2188 (0.02%)	4.8714 (0.01%)



Remarks

The total creep time for this test is 1000 hours. The times listed in the above table are the times calculated by the Abaqus automatic time stepping algorithm with CETOL = $5. \times 10^{-5}$.

Input files

ncr6arkx.inp C3D20R elements.

Test 6B: 3D – triaxial displacement, secondary creep

Products: Abaqus/Standard

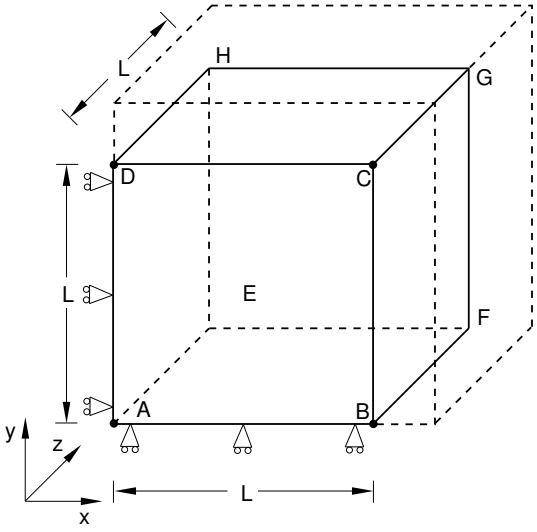
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

C3D20R

Problem description



Three-dimensional

L = 100 mm

u_1 = 0.3 mm

u_2 = 0.2 mm

u_3 = 0.1 mm

Material: Young's modulus = 200×10^3 N/mm^2, Poisson's ratio = 0.3, Creep law: $\varepsilon = A\sigma^n t$, A = 3.125×10^{-14} per hour (σ in N/mm^2), $n = 5$.

Boundary conditions: $u_x = 0$ on face ADEH, $u_y = 0$ on face ABFE, $u_z = 0$ on face ABCD, $u_x = 0.3$ on face BCGF, $u_y = 0.2$ on face CDHG, and $u_z = 0.1$ on face EFGH.

Reference solution

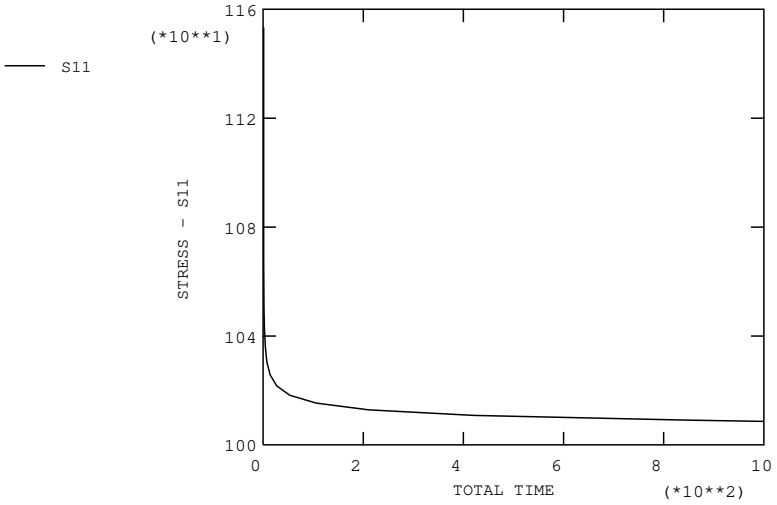
This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 6(b) from NAFEMS Publication Ref: R0027, "NAFEMS Fundamental Tests of Creep Behaviour," June 1993.

The time marching program provided in Appendix B of the NAFEMS Publication can be used to obtain the stress variation with time.

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Abaqus Results	
t	σ_{xx}
0.00	1153.85 (0.00%)
1.00	1046.70 (0.31%)
9.91	1026.80 (0.26%)
77.02	1016.20 (0.18%)
144.13	1013.90 (0.19%)
681.00	1009.40 (0.10%)
1000.00	1008.40 (0.07%)



Remarks

The total creep time for this test is 1000 hours. The times listed in the above table are the times calculated by the Abaqus automatic time stepping algorithm with CETOL = $1. \times 10^{-4}$.

Input files

ncr6brkx.inp C3D20R elements.

Test 7: Axisymmetric – pressurized cylinder, secondary creep

Products: Abaqus/Standard

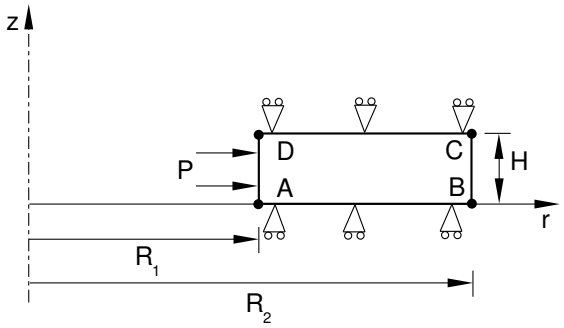
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CAX8R CCL24R

Problem description



Axisymmetric

R_1 = 100 mm

R_2 = 200 mm

H = 25 mm

P = 200 N/mm^2

Material: Young's modulus = 200 × 10^3 N/mm^2, Poisson's ratio = 0.3, Creep law: $\varepsilon = A\sigma^n t$, A = 3.125 × 10^{-14} per hour (σ in N/mm^2), n = 5.

Boundary conditions: $u_z = 0$ on line AB and $u_z = 0$ on line CD.

Loading: Prescribed pressure P = 200 N/mm^2 on line AD.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 7 from NAFEMS Publication Ref: R0027, "NAFEMS Fundamental Tests of Creep Behaviour," June 1993.

$$\sigma_{rr} = 625.96\left[1-\left(\frac{200}{r}\right)^{\frac{2}{5}}\right]$$

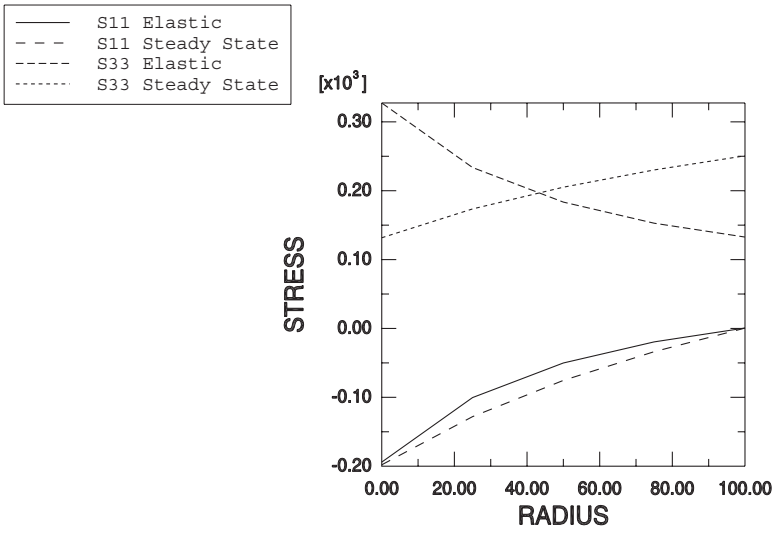
$$\sigma_{\theta\theta} = 625.96\left[1-0.6\left(\frac{200}{r}\right)^{\frac{2}{5}}\right]$$

$$\sigma_{zz} = 625.96\left[1-0.8\left(\frac{200}{r}\right)^{\frac{2}{5}}\right]$$

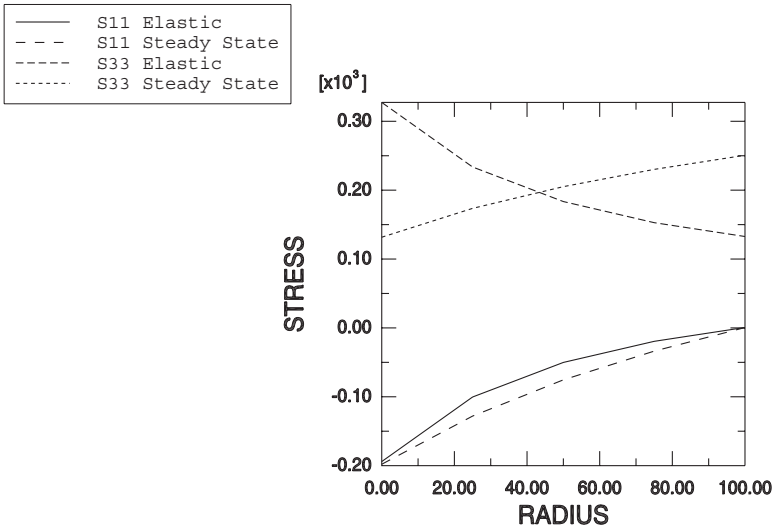
Results and discussion

The results are shown in the following tables. The values enclosed in parentheses are percentage differences with respect to the reference solution. A graphic following each table gives a representation of the different stresses through the thickness of the cylinder.

Abaqus results for CAX8R elements		
Radius	σ_{rr}	$\sigma_{\theta\theta}$
	(steady state)	(steady state)
100.0	−197.95 (−1.02%)	131.68 (1.00%)
125.0	−128.03 (−1.11%)	173.55 (0.49%)
150.0	−75.416 (−1.21%)	205.12 (0.27%)
175.0	−33.708 (−1.84%)	230.15 (0.16%)
200.0	0.499	250.67 (0.11%)



Abaqus results for CCL24R elements		
Radius	σ_{rr}	$\sigma_{\theta\theta}$
	(steady state)	(steady state)
100.0	−197.95 (−1.02%)	130.38 (1.00%)
125.0	−128.03 (−1.11%)	172.70 (0.49%)
150.0	−75.417 (−1.21%)	204.12 (0.26%)
175.0	−33.707 (−1.85%)	230.15 (0.16%)
200.0	0.505	250.67 (0.11%)



Remarks

The total creep time for this test is 1000 hours. In the case where CAX8R elements have been used, four elements were used to model the cylinder. In the case where CCL24R elements were used, four elements were used in the radial direction and eight elements were used to model the full cylinder in the circumferential direction; that is, a total of 32 elements.

Input files

ncr7xr8x.inp	CAX8R elements.
ncr7xrccl24.inp	CCL24R elements.

Test 8A: 2D plane stress – uniaxial load, primary creep

Products: Abaqus/Standard

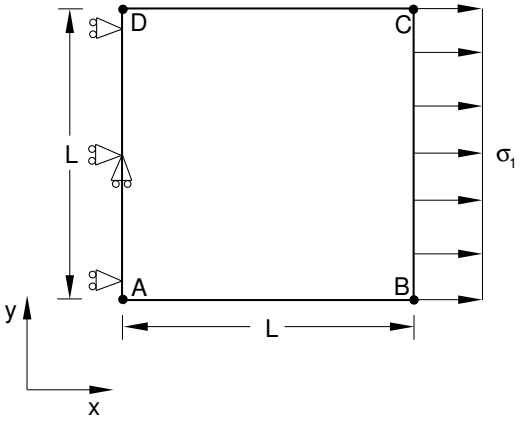
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CPS8R

Problem description



Plane stress

L = 100 mm

σ_1 = 200 N/mm^2

Material: Young's modulus = 200 × 10^3 N/mm^2, Poisson's ratio = 0.3, Creep law: $\varepsilon = A\sigma^n t^m$, A = 3.125 × 10^{-14} per hour (σ in N/mm^2), n = 5, m = 0.5.

Boundary conditions: $u_x = 0$ on line AD and $u_y = 0$ at mid-point of line AD.

Loading: Prescribed tensile stress σ_1 = 200 N/mm^2 on line BC.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 8(a) from NAFEMS Publication Ref: R0027, "NAFEMS Fundamental Tests of Creep Behaviour," June 1993.

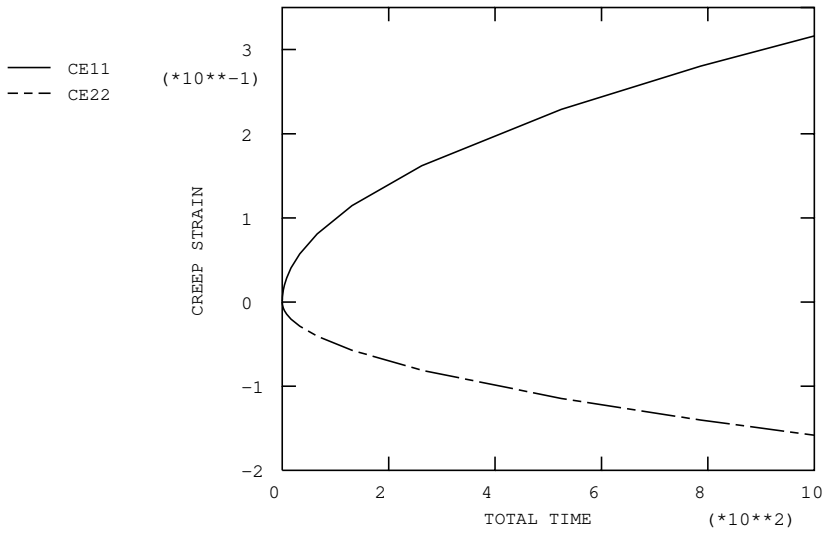
$$\varepsilon^c_{xx} = \varepsilon^c_{eff} = 0.01\sqrt{t}$$

$$\varepsilon^c_{yy} = \varepsilon^c_{zz} = -0.005\sqrt{t}$$

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Abaqus Results		
t	ε^c_{xx}	ε^c_{yy}
0.00	0.000 (0.00%)	0.000 (0.00%)
0.54	0.0074 (–0.14%)	–0.0037 (–0.16%)
8.22	0.0287 (–0.03%)	–0.0143 (–0.04%)
65.57	0.0810 (–0.01%)	–0.0405 (–0.01%)
262.17	0.1619 (0.00%)	–0.0810 (–0.01%)
786.46	0.2804 (0.00%)	–.1402 (–0.01%)
1000.00	0.3162 (0.00%)	–0.1581 (0.00%)



Remarks

The total creep time for this test is 1000 hours. The times listed in the above table are the times calculated by the Abaqus automatic time stepping algorithm with CETOL = $5. \times 10^{-3}$.

Input files

ncr8ar8x.inp CPS8R elements.

Test 8B: 2D plane stress – uniaxial displacement, primary creep

Products: Abaqus/Standard

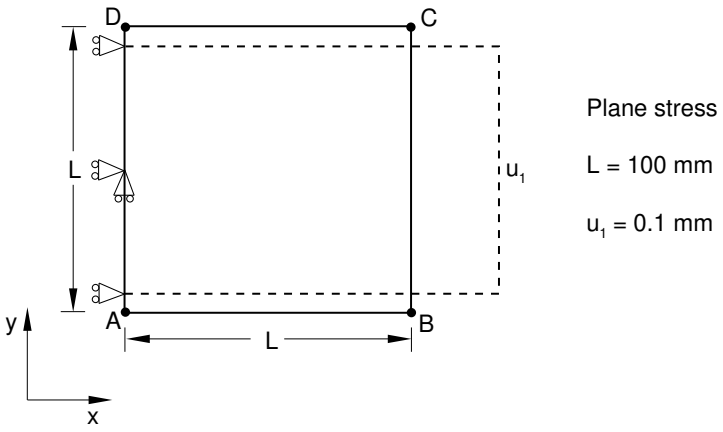
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CPS8R

Problem description



Material: Young's modulus = 200×10^3 N/mm^2, Poisson's ratio = 0.3, Creep law: $\varepsilon = A\sigma^n t^m$, A = 3.125×10^{-14} per hour (σ in N/mm^2), $n = 5$, $m = 0.5$.

Boundary conditions: $u_x = 0$ on line AD, $u_y = 0$ at mid-point of line AD and $u_1 = 0.1$ on line BC.

Reference solution

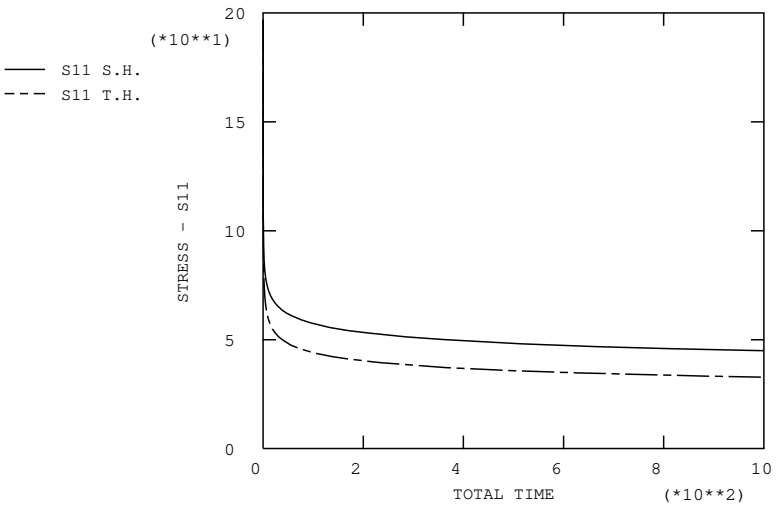
This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 8(b) from NAFEMS Publication Ref: R0027, "NAFEMS Fundamental Tests of Creep Behaviour," June 1993.

The time marching program provided in Appendix B of the NAFEMS Publication can be used to obtain the stress variation with time.

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Abaqus Results			
Time Hardening		Strain Hardening	
t	σ^c_{xx}	t	σ^c_{xx}
0.00	200.00 (0.00%)	0.00	200.00 (0.00%)
0.56	95.09 (4.41%)	0.57	99.60 (–4.46%)
6.38	63.78 (0.77%)	6.29	76.80 (–0.71%)
30.96	51.34 (–4.55%)	57.42	60.92 (–1.39%)
235.76	39.42 (–5.78%)	285.71	51.27 (0.49%)
628.98	34.75 (–2.61%)	666.19	46.80 (–1.99%)
1000.00	32.79 (–1.80%)	1000.00	44.91 (–0.92%)



Remarks

The total creep time for this test is 1000 hours. The times listed in the above table are the times calculated by the Abaqus automatic time stepping algorithm with CETOL = $5. \times 10^{-6}$.

Input files

ncr8br8t.inp	Time hardening.
ncr8br8s.inp	Strain hardening.

Test 8C: 2D plane stress – stepped load, primary creep

Products: Abaqus/Standard

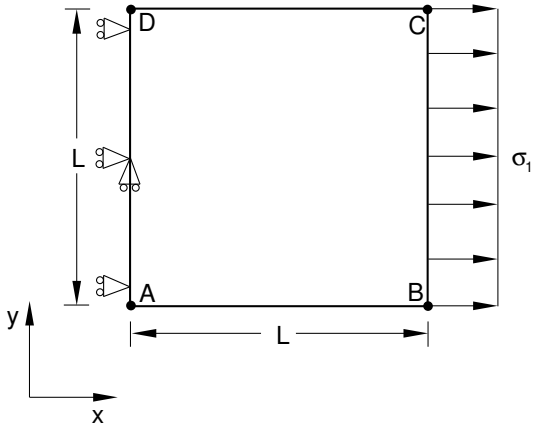
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CPS8R

Problem description



Plane stress

L = 100 mm

σ_1 = 200 N/mm^2 (t = 100 hours)
and 250 N/mm (t > 100 hours)

Material: Young's modulus = 200 × 10^3 N/mm^2, Poisson's ratio = 0.3, Creep law: $\varepsilon = A\sigma^n t^m$, A = 3.125 × 10^{-14} per hour (σ in N/mm^2), $n = 5$, $m = 0.5$.

Boundary conditions: $u_x = 0$ on line AD and $u_y = 0$ at mid-point of line AD.

Loading: σ_1 = 200 N/mm^2 for $t \leq$ 100 hours and σ_1 = 250 N/mm^2 for $t >$ 100 hours.

Reference solution

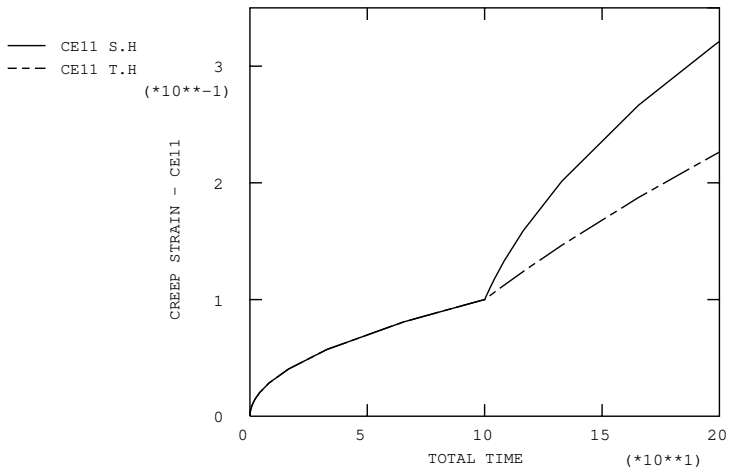
This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 8(c) from NAFEMS Publication Ref: R0027, "NAFEMS Fundamental Tests of Creep Behaviour," June 1993.

	$\varepsilon^c_{xx} = 0.01\sqrt{t}$	for $t \leq$ 100 hours
Time hardening:	$\varepsilon^c_{xx} = 0.03052\sqrt{t} - 0.2052$	for $t >$ 100 hours
Strain hardening:	$\varepsilon^c_{xx} = 0.03052\sqrt{t - 89.262}$	for $t >$ 100 hours

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Abaqus Results			
Time Hardening		Strain Hardening	
t	ε^c_{xx}	t	ε^c_{xx}
0.00	0.0000 (0.00%)	0.00	0.0000 (0.00%)
0.54	0.0073 (–0.14%)	0.54	0.0073 (–0.01%)
8.22	0.0287 (–0.03%)	8.22	0.0287 (0.00%)
65.57	0.0810 (–0.01%)	65.56	0.0810 (0.00%)
116.39	0.1240 (–0.02%)	116.45	0.1591 (–0.01%)
165.54	0.1875 (–0.01%)	165.60	0.2666 (–0.01%)
200.00	0.2264 (0.00%)	200.00	0.3211 (–0.01%)



Remarks

The total creep time for this test is 200 hours. The times listed in the above table are the times calculated by the Abaqus automatic time stepping algorithm with CETOL = $5. \times 10^{-3}$.

Input files

ncr8cr8t.inp	Time hardening.
ncr8cr8s.inp	Strain hardening.

Test 9A: 2D plane stress – biaxial load, primary creep

Products: Abaqus/Standard

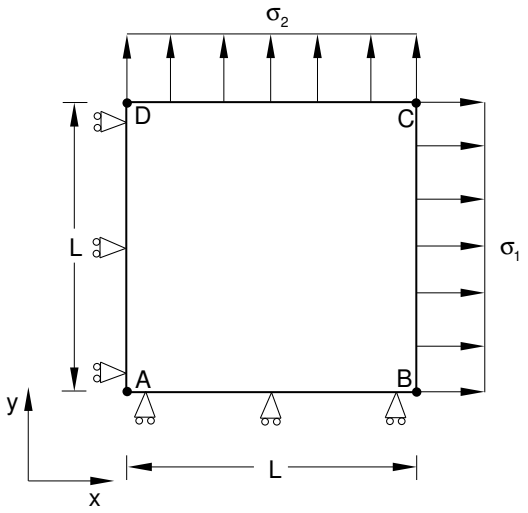
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CPS8R

Problem description



Plane stress

L = 100 mm

$\sigma_1 = \sigma_2 = 200$ N/mm^2

Material: Young's modulus = 200×10^3 N/mm^2, Poisson's ratio = 0.3, Creep law: $\varepsilon = A\sigma^n t^m$, A = 3.125×10^{-14} per hour (σ in N/mm^2), $n = 5$, $m = 0.5$.

Boundary conditions: $u_x = 0$ on line AD and $u_y = 0$ on line AB.

Loading: Prescribed tensile stress σ_1 = 200 N/mm^2 on line BC.

Prescribed tensile stress σ_2 = 200 N/mm^2 on line DC.

Reference solution

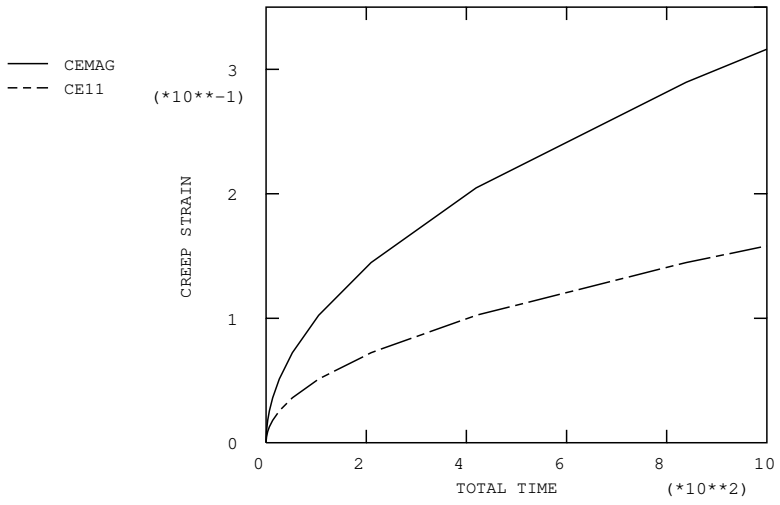
This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 9(a) from NAFEMS Publication Ref: R0027, “NAFEMS Fundamental Tests of Creep Behaviour,” June 1993.

$$\varepsilon^c_{xx} = \varepsilon^c_{yy} = 0.005\sqrt{t}$$
$$\varepsilon^c_{eff} = -\varepsilon^c_{zz} = 0.01\sqrt{t}$$

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Abaqus Results		
t	ε^c_{xx}	ε^c_{eff}
0.00	0.0000 (0.00%)	0.0000 (0.00%)
0.83	0.0045 (–0.11%)	0.0091 (–0.11%)
6.56	0.0128 (–0.04%)	0.0256 (–0.04%)
52.44	0.0362 (–0.01%)	0.0724 (–0.01%)
104.86	0.0512 (–0.01%)	0.1024 (–0.01%)
419.44	0.1024 (0.00%)	0.2048 (–0.01%)
1000.00	0.1581 (0.00%)	0.3162 (0.00%)



Remarks

The total creep time for this test is 1000 hours. The times listed in the above table are the times calculated by the Abaqus automatic time stepping algorithm with CETOL = 5. × 10^{-3}.

Input files

ncr9ar8x.inp CPS8R elements.

Test 9B: 2D plane stress – biaxial displacement, primary creep

Products: Abaqus/Standard

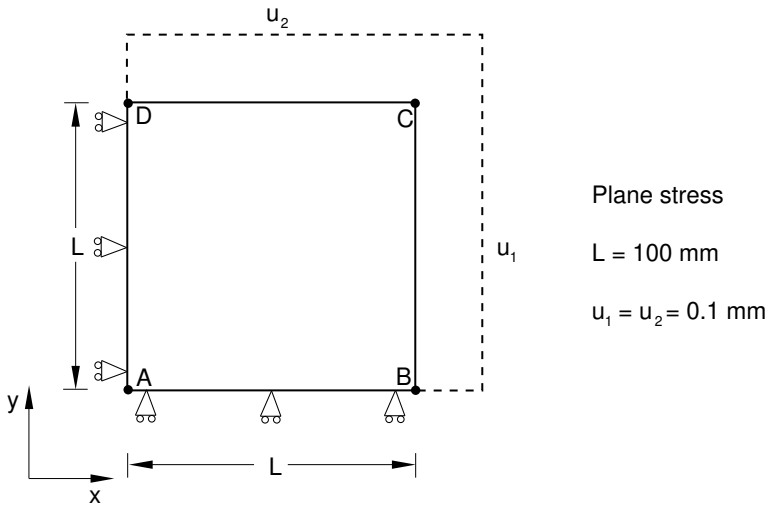
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CPS8R

Problem description



Material: Young's modulus = 200×10^3 N/mm^2, Poisson's ratio = 0.3, Creep law: $\varepsilon = A\sigma^n t^m$, A = 3.125×10^{-14} per hour (σ in N/mm^2), $n = 5$, $m = 0.5$.

Boundary conditions: $u_x = 0$ on line AD, $u_y = 0$ on line AB, $u_x = 0.1$ on line BC and $u_y = 0.1$ on line CD.

Reference solution

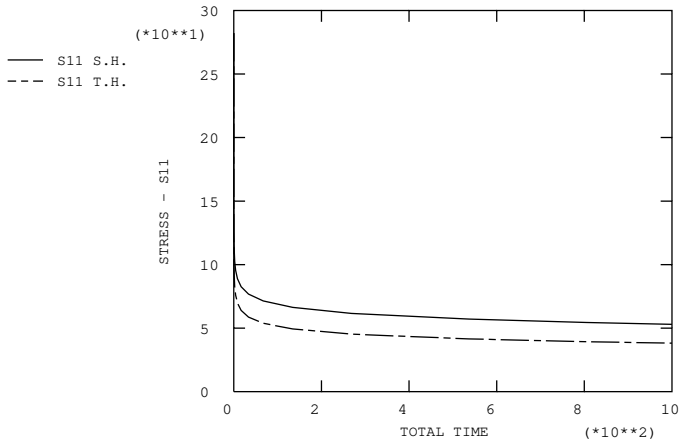
This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 9(b) from NAFEMS Publication Ref: R0027, "NAFEMS Fundamental Tests of Creep Behaviour," June 1993.

The time marching program provided in Appendix B of the NAFEMS Publication can be used to obtain the stress variation with time.

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Abaqus Results			
Time Hardening		Strain Hardening	
t	σ_{xx}	t	σ_{xx}
0.00	285.70 (0.00%)	0.00	285.70 (0.00%)
0.52	98.00 (−1.45%)	0.52	118.12 (−1.53%)
8.39	69.72 (6.52%)	8.39	88.88 (4.83%)
33.55	58.70 (3.40%)	33.55	76.84 (2.27%)
268.44	45.31 (2.79%)	268.44	61.59 (1.06%)
536.87	41.55 (7.12%)	536.87	57.18 (5.41%)
1000.00	38.18 (6.92%)	1000.00	53.02 (5.08%)



Remarks

The total creep time for this test is 1000 hours. The times listed in the above table are the times calculated by the Abaqus automatic time stepping algorithm with CETOL = $5. \times 10^{-3}$.

Input files

ncr9br8t.inp	Time hardening.
ncr9br8s.inp	Strain hardening.

Test 9C: 2D plane stress – biaxial stepped load, primary creep

Products: Abaqus/Standard

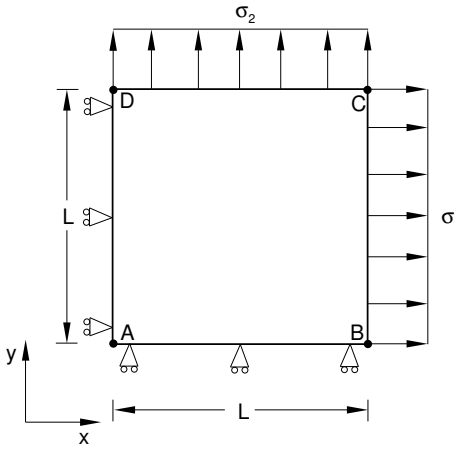
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CPS8R

Problem description



Plane stress

L = 100 mm

σ_1 = 200 N/mm^2 (t = 100 hours)
and 250 N/mm (t > 100 hours)

σ_2= 200 N/mm^2 (t = 100 hours)
and 250 N/mm (t > 100 hours)

Material: Young's modulus = 200 × 10^3 N/mm^2, Poisson's ratio = 0.3, Creep law: $\varepsilon = A\sigma^n t^m$, A = 3.125 × 10^{-14} per hour (σ in N/mm^2), $n = 5$, $m = 0.5$.

Boundary conditions: $u_x = 0$ on line AD and $u_y = 0$ on line AB.

Loading: σ_1 = 200 N/mm^2 for $t \leq$ 100 hours and σ_1 = 250 N/mm^2 for $t >$ 100 hours;

σ_2 = 200 N/mm^2 for $t \leq$ 100 hours and σ_2 = 250 N/mm^2 for $t >$ 100 hours.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 9(c) from NAFEMS Publication Ref: R0027, "NAFEMS Fundamental Tests of Creep Behaviour," June 1993.

Time hardening:

$$\varepsilon^c_{xx} = \varepsilon^c_{yy} = 0.005\sqrt{t} \quad \text{for } t \leq 100 \text{ hours}$$

$$\varepsilon^c_{xx} = \varepsilon^c_{yy} = 0.01529\sqrt{t} - 0.1026 \quad \text{for } t > 100 \text{ hours}$$

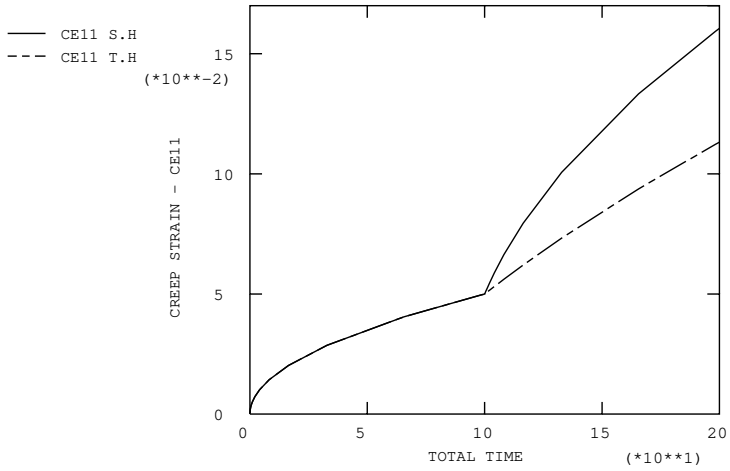
Strain hardening:

$$\varepsilon^c_{xx} = \varepsilon^c_{yy} = 0.01529\sqrt{t - 89.262} \quad \text{for } t > 100 \text{ hours}$$

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Abaqus Results			
Time Hardening		Strain Hardening	
t	ε^c_{xx}	t	ε^c_{xx}
0.00	0.0000 (0.00%)	0.00	0.0000 (0.00%)
0.55	0.0037 (–0.13%)	0.55	0.0037 (–0.01%)
8.23	0.0143 (–0.03%)	8.23	0.0143 (0.00%)
65.58	0.0405 (–0.01%)	65.58	0.0405 (0.00%)
116.39	0.0620 (–0.53%)	116.39	0.0795 (–0.21%)
165.54	0.0937 (–0.42%)	165.54	0.1333 (–0.21%)
200.00	0.1132 (0.00%)	200.00	0.1606 (0.00%)



Remarks

The total creep time for this test is 200 hours. The times listed in the above table are the times calculated by the Abaqus automatic time stepping algorithm with CETOL = $5. \times 10^{-3}$.

Input files

ncr9cr8t.inp	Time hardening.
ncr9cr8s.inp	Strain hardening.

Test 10A: 2D plane stress – biaxial (negative) load, primary creep

Products: Abaqus/Standard

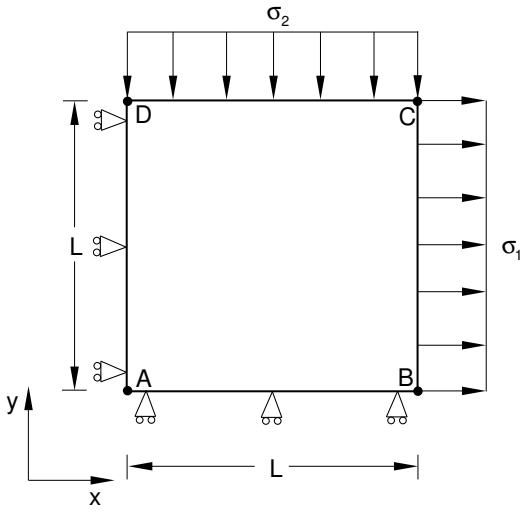
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CPS8R

Problem description



Plane stress

L = 100 mm

σ_1 = 200 N/mm^2

σ_2 = -200 N/mm^2

Material: Young's modulus = 200 × 10^3 N/mm^2, Poisson's ratio = 0.3, Creep law: $\varepsilon = A\sigma^n t$, A = 3.125 × 10^{-14} per hour (σ in N/mm^2), n = 5, m = 0.5.

Boundary conditions: $u_x = 0$ on line AD and $u_y = 0$ on line AB.

Loading: Prescribed tensile stress σ_1 = 200 N/mm^2 on line BC and σ_2 −200 N/mm^2 on line CD.

Reference solution

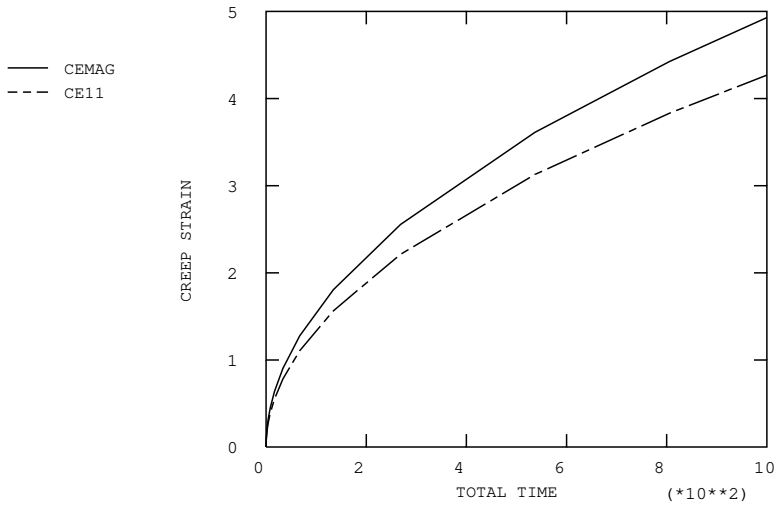
This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 10(a) from NAFEMS Publication Ref: R0027, “NAFEMS Fundamental Tests of Creep Behaviour,” June 1993.

$$\varepsilon^c_{xx} = -\varepsilon^c_{yy} = 0.135\sqrt{t}$$
$$\varepsilon^c_{eff} = 0.1559\sqrt{t}$$
$$\varepsilon^c_{yy} = 0.0$$

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Abaqus Results		
t	ε^c_{xx}	ε^c_{eff}
0.00	0.0000 (0.00%)	0.0000 (0.00%)
1.05	0.1381 (–0.07%)	0.1595 (–0.08%)
16.78	0.5528(–0.03%)	0.6384 (–0.04%)
134.22	1.5639 (–0.01%)	1.8058 (–0.02%)
268.44	2.2117 (–0.01%)	2.5539 (–0.02%)
536.87	3.1279 (0.00%)	3.612 (–0.01%)
1000.00	4.2689 (0.00%)	4.9293 (–0.01%)



Remarks

The total creep time for this test is 1000 hours. The times listed in the above table are the times calculated by the Abaqus automatic time stepping algorithm with CETOL = 5. × 10^{-3}.

Input files

ncraar8x.inp CPS8R elements.

Test 10B: 2D plane stress – biaxial (negative) displacement, primary creep

Products: Abaqus/Standard

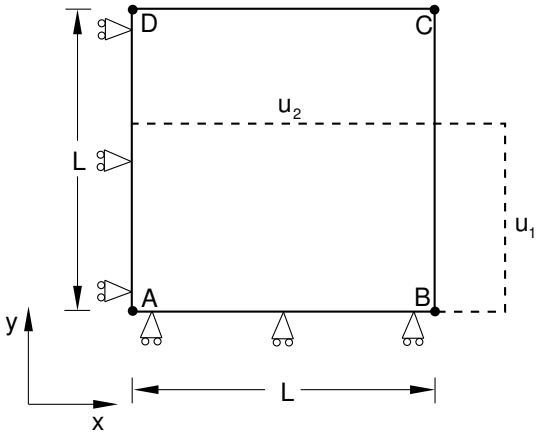
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CPS8R

Problem description



Plane stress

L = 100 mm

u_1 = 0.1 mm

u_2 = -0.1 mm

Material: Young's modulus = 200×10^3 N/mm^2, Poisson's ratio = 0.3, Creep law: $\varepsilon = A\sigma^n t^m$, A = 3.125×10^{-14} per hour (σ in N/mm^2), $n = 5$, $m = 0.5$.

Boundary conditions: $u_x = 0$ on line AD, $u_y = 0$ on line AB, $u_x = 0.1$ on line BC and $u_y = -0.1$ on line CD.

Reference solution

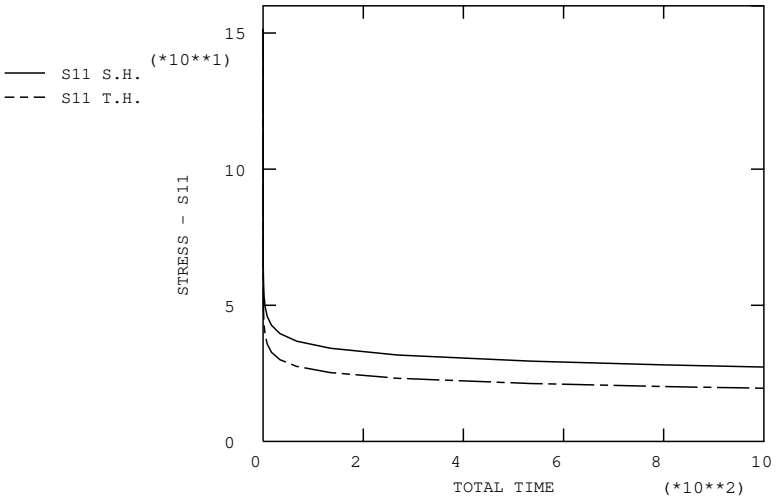
This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 10(b) from NAFEMS Publication Ref: R0027, "NAFEMS Fundamental Tests of Creep Behaviour," June 1993.

The time marching program provided in Appendix B of the NAFEMS Publication can be used to obtain the stress variation with time.

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Abaqus Results			
Time Hardening		Strain Hardening	
t	σ_{xx}	t	σ_{xx}
0.00	153.80 (0.00%)	0.00	153.80 (0.00%)
0.52	50.23 (–1.34%)	0.52	61.08 (–1.60%)
8.39	35.71 (6.45%)	8.39	45.89 (4.80%)
33.55	30.07 (3.21%)	33.55	39.65 (2.30%)
268.44	23.20 (2.80%)	268.44	31.77 (1.85%)
536.87	21.28 (6.99%)	536.87	28.10 (5.61%)
1000.00	19.55 (6.60%)	1000.00	27.34 (4.75%)



Remarks

The total creep time for this test is 1000 hours. The times listed in the above table are the times calculated by the Abaqus automatic time stepping algorithm with CETOL = $5. \times 10^{-3}$.

Input files

ncrabr8t.inp	Time hardening.
ncrabr8s.inp	Strain hardening.

Test 10C: 2D plane stress – biaxial (negative) stepped load, primary creep

Products: Abaqus/Standard

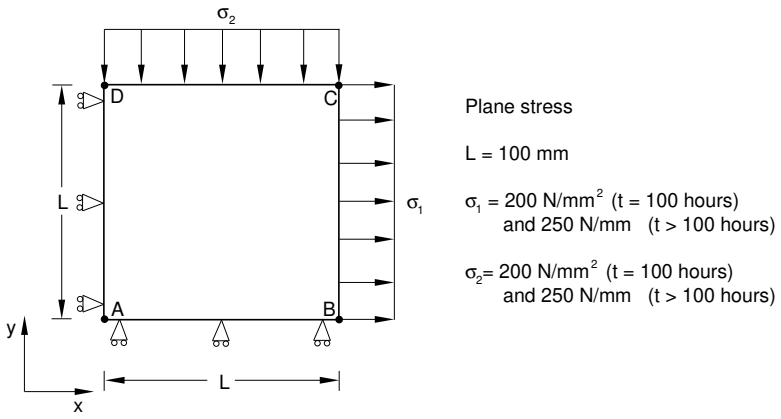
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CPS8R

Problem description



Material: Young's modulus = 200×10^3 N/mm^2, Poisson's ratio = 0.3, Creep law: $\varepsilon = A\sigma^n t^m$, A = 3.125×10^{-14} per hour (σ in N/mm^2), $n = 5$, $m = 0.5$.

Boundary conditions: $u_x = 0$ on line AD and $u_y = 0$ on line AB.

Loading: σ_1 = 200 N/mm^2 for $t \leq$ 100 hours and σ_1 = 250 N/mm^2 for $t >$ 100 hours;

σ_2 = −200 N/mm^2 for $t \leq$ 100 hours and σ_2 = −250 N/mm^2 for $t >$ 100 hours.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 10(c) from NAFEMS Publication Ref: R0027, "NAFEMS Fundamental Tests of Creep Behaviour," June 1993.

Time hardening:

$$\varepsilon^c_{xx} = -\varepsilon^c_{yy} = 0.135\sqrt{t} \quad \text{for } t \leq 100 \text{ hours}$$

$$\varepsilon^c_{xx} = -\varepsilon^c_{yy} = 0.4120\sqrt{t} - 2.7699 \quad \text{for } t > 100 \text{ hours}$$

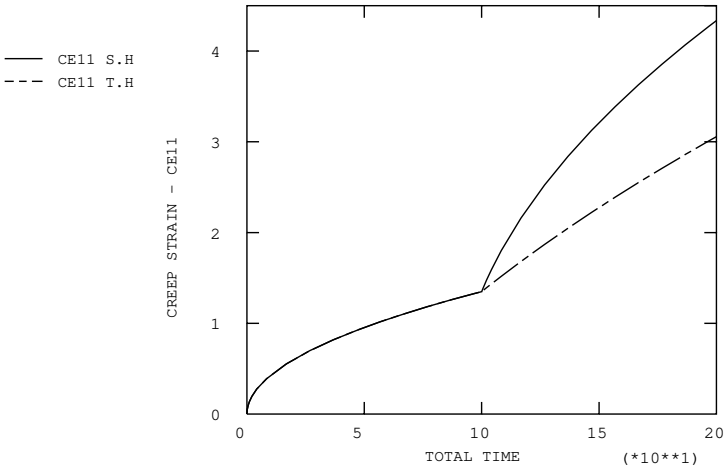
Strain hardening:

$$\varepsilon^c_{xx} = -\varepsilon^c_{yy} = 0.4120\sqrt{t - 89.2605} \qquad \text{for } t > 100 \text{ hours}$$

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Abaqus Results			
Time Hardening		Strain Hardening	
t	ε^c_{xx}	t	ε^c_{xx}
0.00	0.0000 (0.00%)	0.00	0.0000 (0.00%)
0.52	0.0976 (–0.19%)	0.52	0.0978 (–0.05%)
8.39	0.3909 (–0.03%)	8.39	0.3910 (0.00%)
76.78	1.1828 (–0.01%)	76.78	1.1829 (0.00%)
136.78	2.0483 (–0.01%)	136.78	2.8399 (–0.22%)
176.78	2.7077 (–0.01%)	176.78	3.8541 (–0.01%)
200.00	3.0564 (0.00%)	200.00	4.3354 (0.00%)



Remarks

The total creep time for this test is 200 hours. The times listed in the above table are the times calculated by the Abaqus automatic time stepping algorithm with CETOL = $5. \times 10^{-3}$.

Input files

ncracr8t.inp	Time hardening.
ncracr8s.inp	Strain hardening.

Test 11: 3D – triaxial load, primary creep

Products: Abaqus/Standard

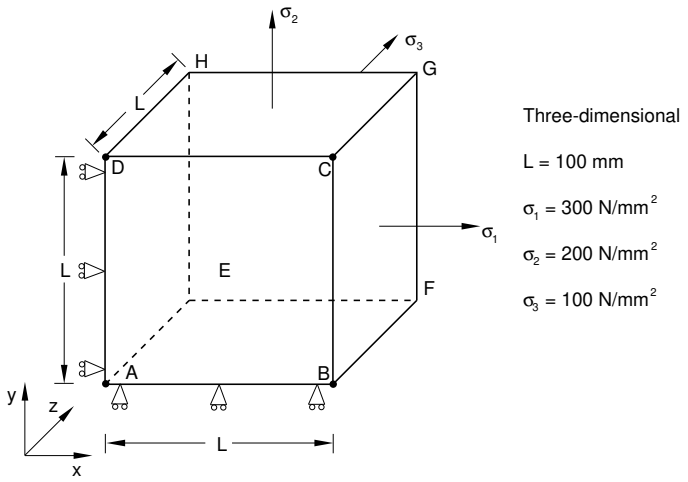
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

C3D20R

Problem description



Three-dimensional

L = 100 mm

σ_1 = 300 N/mm^2

σ_2 = 200 N/mm^2

σ_3 = 100 N/mm^2

Material: Young's modulus = 200 × 10^3 N/mm^2, Poisson's ratio = 0.3, Creep law: $\varepsilon = A\sigma^n t^m$, A = 3.125 × 10^{-14} per hour (σ in N/mm^2), n = 5, m = 0.5.

Boundary conditions: $u_x = 0$ on face ADEH, $u_y = 0$ on face ABFE and $u_z = 0$ on face ABCD.

Loading: Prescribed tensile stress σ_1 = 300 N/mm^2 on face BCGF, σ_2 = 200 N/mm^2 on face CDHG, and σ_3 = 100 N/mm^2 on face EFGH.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 11 from NAFEMS Publication Ref: R0027, "NAFEMS Fundamental Tests of Creep Behaviour," June 1993.

$$\varepsilon^c_{xx} = -\varepsilon^c_{zz} = 0.004218\sqrt{t}$$

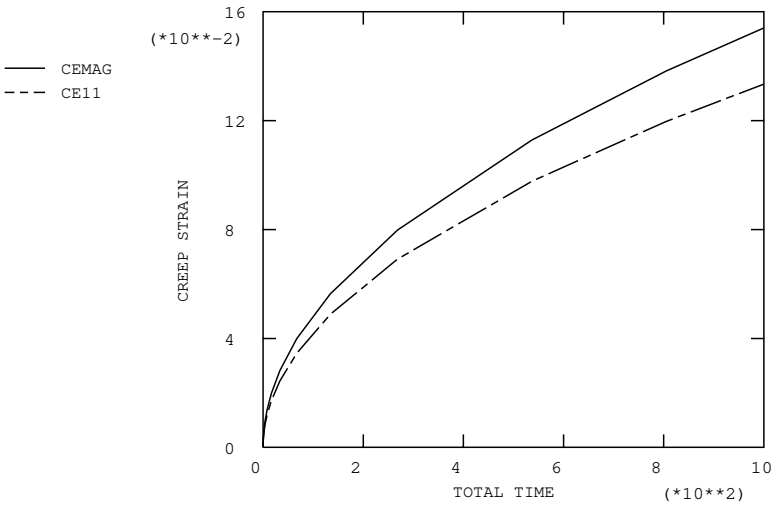
$$\varepsilon^c_{eff} = 0.004871\sqrt{t}$$

$$\varepsilon^c_{yy} = 0.0$$

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Abaqus Results		
t	ε^c_{xx}	ε^c_{eff}
0.00	0.0000 (0.00%)	0.0000 (0.00%)
8.39	0.0122 (–0.02%)	0.0141 (–0.03%)
67.11	0.0346 (0.01%)	0.0399 (0.00%)
134.22	0.0489 (0.01%)	0.0564 (0.00%)
536.87	0.0977 (0.01%)	0.1129 (0.01%)
805.31	0.1197 (0.02%)	0.1382 (0.01%)



Remarks

The total creep time for this test is 1000 hours. The times listed in the above table are the times calculated by the Abaqus automatic time stepping algorithm with CETOL = $5. \times 10^{-7}$.

Input files

ncrbxrkx.inp C3D20R elements.

Test 12A: 2D plane stress – uniaxial load, primary-secondary creep

Products: Abaqus/Standard

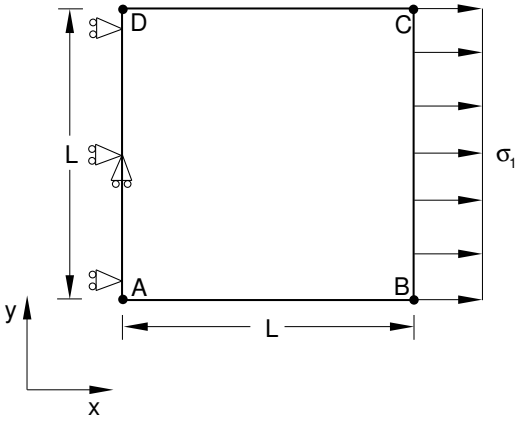
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CPS8R

Problem description



Plane stress

L = 100 mm

σ_1 = 200 N/mm^2

Material: Young's modulus = 200 × 10^3 N/mm^2, Poisson's ratio = 0.3, Creep law: $\varepsilon = A_1\sigma^{n1}t + A_2\sigma^{n2}t^m$, $A_1 = 10^{-16}$, $A_2 = 10^{-14}$, $n1 = n2 = 5.0$, and $m = 0.5$.

Boundary conditions: $u_x = 0$ on line AD and $u_y = 0$ at mid-point of line AD.

Loading: Prescribed tensile stress σ_1 = 200 N/mm^2 on line BC.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 12(a) from NAFEMS Publication Ref: R0027, "NAFEMS Fundamental Tests of Creep Behaviour," June 1993.

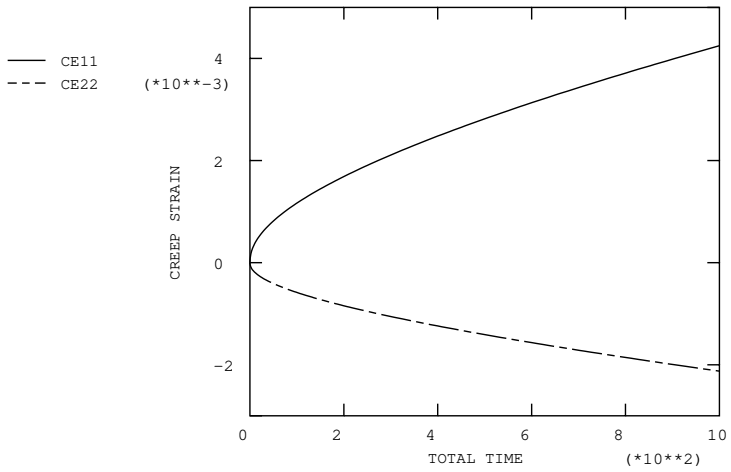
$$\varepsilon^c_{xx} = 0.0001\left(0.01t + \sqrt{t}\right)$$

$$\varepsilon^c_{yy} = -\frac{1}{2}\varepsilon^c_{xx}$$

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Abaqus Results		
t	ε^c_{xx}	ε^c_{yy}
0.00	0.000 (0.00%)	0.000 (0.00%)
6.41	0.000222 (–14.35%)	–0.000111 (–14.35%)
70.41	0.000845 (–7.09%)	–0.000423 (–7.09%)
382.73	0.002260 (–3.78%)	–0.001130 (–3.78%)
536.33	0.002770 (–2.87%)	–0.001385 (–2.87%)
823.05	0.003606 (–2.32%)	–0.001803 (–2.32%)
1000.00	0.00408 (–2.02%)	–0.002038 (–2.02%)



Remarks

The total creep time for this test is 1000 hours. The times listed in the above table are the times calculated by the Abaqus automatic time stepping algorithm with CETOL = $1. \times 10^{-5}$. The creep law is defined via user subroutine *CREEP*.

Input files

ncrcar8x.inp	CPS8R elements.
ncrcar8x.f	User subroutine *CREEP* used in ncrcar8x.inp.

Test 12B: 2D plane stress – uniaxial displacement, primary-secondary creep

Products: Abaqus/Standard

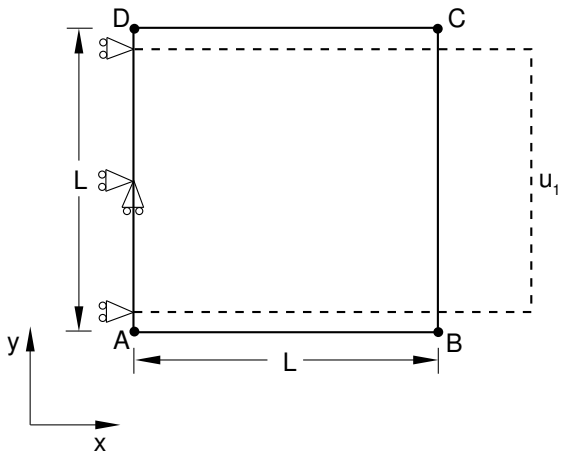
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CPS8R

Problem description



Plane stress

L = 100 mm

u_1 = 0.1 mm

Material: Young's modulus = 200×10^3 N/mm^2, Poisson's ratio = 0.3, Creep law: $\varepsilon = A_1 \sigma^{n1} t + A_2 \sigma^{n2} t^m$, $A_1 = 10^{-16}$, $A_2 = 10^{-14}$, $n1 = n2 = 5.0$, and $m = 0.5$.

Boundary conditions: $u_x = 0$ on line AD, $u_y = 0$ at mid-point of line AD and $u_x = 0.1$ on line BC.

Reference solution

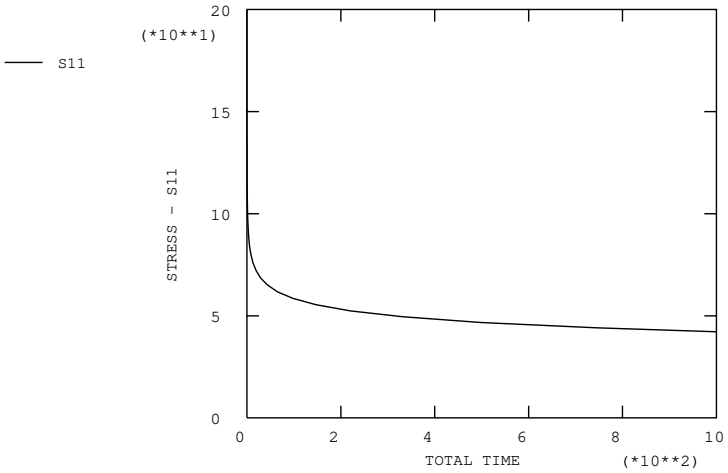
This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 12(b) from NAFEMS Publication Ref: R0027, "NAFEMS Fundamental Tests of Creep Behaviour," June 1993.

$$\sigma_{xx} = \left(6.25 \times 10^{-10} + 8 \times 10^{-9} \left[0.01\, t + \sqrt{t}\right]\right)^{-\frac{1}{4}}$$

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Abaqus Results	
t	σ_{xx}
0.00	200.00 (0.00%)
1.64	97.44 (–0.18%)
14.23	74.39 (–0.55%)
68.75	60.47 (–0.79%)
270.08	49.94 (–1.12%)
538.51	45.08 (–1.35%)



Remarks

The total creep time for this test is 1000 hours. The times listed in the above table are the times calculated by the Abaqus automatic time stepping algorithm with CETOL = $1. \times 10^{-5}$. The creep law is defined via user subroutine *CREEP*.

Input files

ncrcbr8x.inp	CPS8R elements.
ncrcbr8x.f	User subroutine *CREEP* used in ncrcbr8x.inp.

Test 12C: 2D plane stress – stepped load, primary-secondary creep

Products: Abaqus/Standard

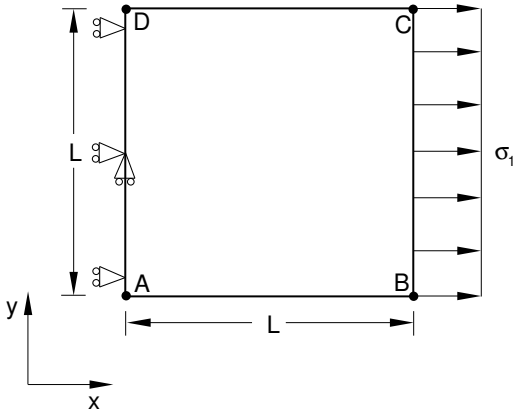
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CPS8R

Problem description



Plane stress

L = 100 mm

σ_1 = 100 N/mm^2 (t = 10000 hours)
and 110 N/mm^2 (t > 10000 hours)

Material: Young's modulus = 200 × 10^3 N/mm^2, Poisson's ratio = 0.3, Creep law: $\boldsymbol{\varepsilon} = \mathrm{A}_1\boldsymbol{\sigma}^{n1}\boldsymbol{t} + \mathrm{A}_2\boldsymbol{\sigma}^{n2}\boldsymbol{t}^{m}$, $\mathrm{A}_1 = 10^{-16}$, $\mathrm{A}_2 = 10^{-14}$, $\boldsymbol{n1} = \boldsymbol{n2} = 5.0$, and $m = 0.5$.

Boundary conditions: $\boldsymbol{u_x} = \mathbf{0}$ on line AD and $\boldsymbol{u_y} = \mathbf{0}$ at mid-point of line AD.

Loading: $\boldsymbol{\sigma_1}$ = 100 N/mm^2 for $\boldsymbol{t} \leq$ 10000 hours and $\boldsymbol{\sigma_1}$ = 110 N/mm^2 for $\boldsymbol{t} >$ 10000 hours.

Reference solution

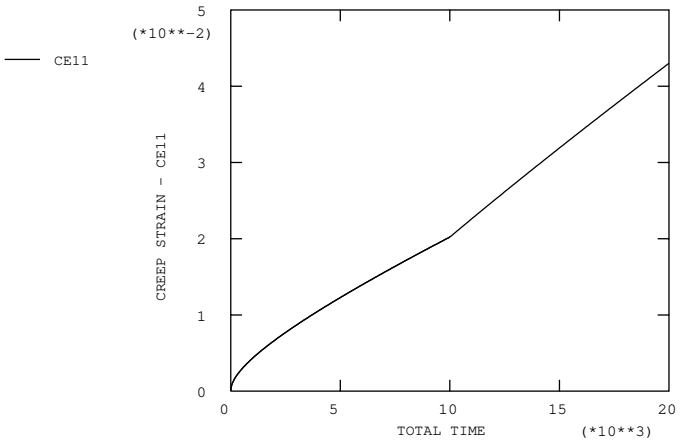
This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 12(c) from NAFEMS Publication Ref: R0027, "NAFEMS Fundamental Tests of Creep Behaviour," June 1993.

$$\varepsilon^c_{xx} = \begin{cases} 0.0001\left(0.01\,t + \sqrt{t}\right) & \text{for } t \leq 10000 \text{ hours} \\ 0.020 + 1.6105 \times 10^{-4}\left(0.01\,t + \sqrt{t} - 200.0\right) & \text{for } t > 10000 \text{ hours} \end{cases}$$

Results and discussion

The results are shown in the following table. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Abaqus Results	
t	ε^c_{xx}
0.00	0.0000 (0.00%)
147.21	0.0013 (–5.29%)
1580.80	0.0055 (–1.62%)
8800.00	0.0181 (–0.54%)
15028.00	0.0316 (–0.32%)
18438.00	0.0393 (–0.26%)
20000.00	0.0427 (–0.23%)



Remarks

The total creep time for this test is 20000 hours. The times listed in the above table are the times calculated by the Abaqus automatic time stepping algorithm with CETOL = 0.1. The creep law is defined via user subroutine *CREEP*.

Input files

ncrccr8x.inp	CPS8R elements.
ncrccr8x.f	User subroutine *CREEP* used in ncrccr8x.inp.

Composite tests

In this section:

- *R0031(1): Laminated strip under three-point bending*
- *R0031(2): Wrapped thick cylinder under pressure and thermal loading*
- *R0031(3): Three-layer sandwich shell under normal pressure loading*

R0031(1): Laminated strip under three-point bending

Products: Abaqus/Standard Abaqus/Explicit

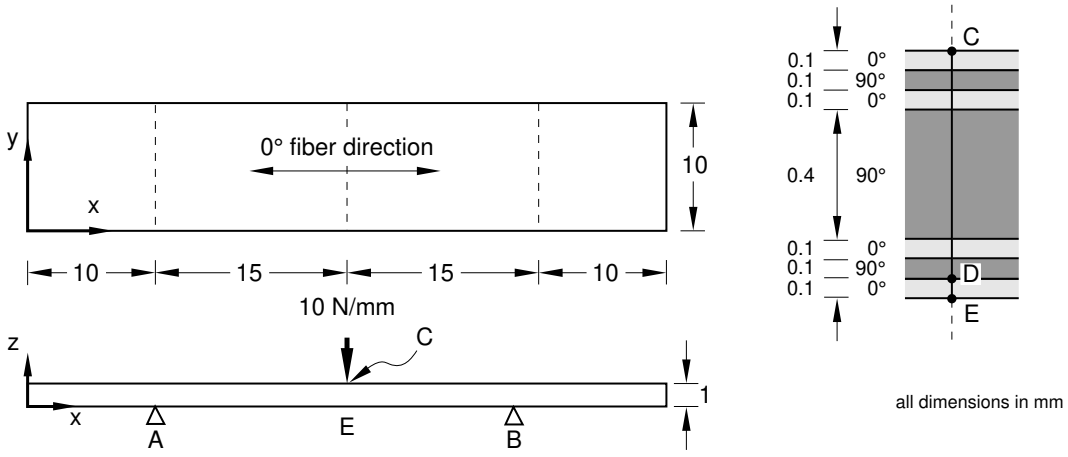
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

C3D8R C3D20 CSS8 S4R S8R SC6R SC8R

Problem description



Mesh: One-quarter of the laminated strip is modeled. The same problem is analyzed with different meshes. Meshes using linear solids and shells consist of ten elements along the length and two elements along the breadth. Meshes using quadratic solids and general-purpose shells consist of five elements along the length and one element along the breadth.

Various modeling options are used to model the laminated strip through the thickness. In Abaqus/Standard the laminated solid model (using C3D20 elements) consists of two four-layer elements through the thickness, and the stacked solid model consists of seven single-layer elements through the thickness. In Abaqus/Explicit the solid model (using C3D8R elements) consists of fourteen elements through the thickness. The models using S4R and S8R shells use a composite section definition. The models using SC6R and SC8R continuum shells employ three different techniques: (1) a single element using a composite section, (2) seven single-layer elements stacked through the thickness, and (3) two composite elements representing the skin and one single-layer element representing the core stacked through the thickness.

Material: E_1 = 100 GPa, E_2 = 5 GPa, E_3 = 5 GPa, ν_{12} = 0.4, ν_{13} = 0.3, ν_{23} = 0.3, G_{12} = 3 GPa, G_{13} = 2 GPa, G_{23} = 2 GPa

Boundary conditions: Simply supported at A and B. The continuum shell meshes use an equation constraint to provide an equivalent kinematic constraint at the midsurface along A and B.

Loading: Line load of 10 N/mm at C ($x = 25$, $z = 1$).

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test R0031/1 from NAFEMS publication R0031, "Composites Benchmarks," February 1995.

Results and discussion

The results are given in *Table 1* and *Table 2*. The values enclosed in parentheses are percentage differences with respect to the reference solution. Two values for transverse shear stress (σ_{13} at point D) are reported for the layered and stacked Abaqus/Standard solid models and for the Abaqus/Explicit solid model. The values are for stresses at the two coincident section points in the layers adjacent to point D.

Table 1: Abaqus/Standard analysis.

Model	σ_{11} at E	σ_{13} at D	u_z at E
NAFEMS	684	−4.1	−1.06
Composite S8R	681 (−0.4%)	−4.08 (+0.5%)	−1.06 (0%)
Composite C3D20	708 (+3.5%)	−7.10 (−73%)	−1.157 (+9.2%)
		at sect. pt. 3	
		0.44 (+111%)	
		at sect. pt. 4	
Stacked C3D20	707 (+3.4%)	−4.42 (−7.8%)	−1.10 (+3.7%)
		at elem. 9	
		−0.56 (+86%)	
		at elem. 2009	
Composite SC6R	630 (−7.9%)	−4.28 (4.4%)	−1.05 (−0.9%)
Stacked SC6R	630 (−7.9%)	not available	−1.04 (−1.9%)
Stacked-composite SC6R	628 (−8.2%)	−5.58 (36.1%)	−1.07 (+0.9%)
Composite SC8R	627 (−8.3%)	−4.33 (5.5%)	−1.05 (−0.9%)
Stacked SC8R	635 (−7.2%)	not available	−1.04 (−1.9%)
Stacked-composite SC8R	632 (−7.6%)	−5.15 (25.6%)	−1.07 (+0.9%)
Composite CSS8	631 (−7.7%)	−5.28 (−29%)	−1.05 (−0.9%)
		at sect. pt. 3	
		0.40 (+110%)	
		at sect. pt. 4	
Stacked CSS8	634 (−7.3%)	−2.67 (+35%)	−1.05 (−0.9%)
		at elem. 110	
		0.01 (+100%)	

Model	σ_{11} at E	σ_{13} at D	u_z at E
		at elem. 10110	
Stacked-composite CSS8	630 (−7.9%)	–4.23 (–3.2%)	–1.05 (–0.9%)
		at sect. pt. 3	
		0.08 (+102%)	
		at sect. pt. 4	

Table 2: Abaqus/Explicit analysis.

Model	σ_{11} at E	σ_{13} at D	u_z at E
NAFEMS	684	–4.1	–1.06
Composite S4R	623.5 (−8.8%)	−4.02 (−2.0%)	−1.11 (4.7%)
Stacked C3D8R	596.7 (−12.7%)	−2.39 (+41.7%)	−1.04 (+1.6%)
		at elem. 1019	
		−.57 (+86.2%)	
		at elem. 2019	
Composite SC8R	637.5 (−6.8%)	not available	–1.05 (−1.0%)

Remarks

The results show that the transverse shear stress obtained with the three solid element models is discontinuous at point D. In addition, the transverse shear stresses for the solid elements do not vanish at the free surfaces of the structure because they are obtained directly from the displacement field; in the shell element models the transverse shear stresses are obtained from an equilibrium calculation (see *Transverse shear stiffness in composite shells and offsets from the midsurface*). As the number of solid elements used in the discretization through the section thickness is increased, the transverse shear stresses become more accurate.

The displacement field and components of stress in the plane of the layer are in good agreement with the reference result.

Input files

Abaqus/Standard input files

nco1s8rx.inp	Composite model using S8R elements.
nco1clay.inp	Composite model using C3D20 elements.
nco1csta.inp	Stacked model using C3D20 elements.
r311_std_sc6r_composite.inp	Composite model using SC6R elements.
r311_std_sc6r_stacked.inp	Stacked model using SC6R elements.
r311_std_sc6r_stacked_composite.inp	Stacked-composite model using SC6R elements.
r311_std_sc8r_composite.inp	Composite model using SC8R elements.
r311_std_sc8r_stacked.inp	Stacked model using SC8R elements.
r311_std_sc8r_stacked_composite.inp	Stacked-composite model using SC8R elements.
r311_std_css8_composite.inp	Composite model using CSS8 elements.
r311_std_css8_stacked.inp	Stacked model using CSS8 elements.

r311_std_css8_stacked_composite.inp	Stacked-composite model using CSS8 elements.

Abaqus/Explicit input files

r311shl.inp	Composite model using S4R elements.
r311sol.inp	Composite model using C3D8R elements.
r311_xpl_sc8r_stacked.inp	Stacked model using SC8R elements.

R0031(2): Wrapped thick cylinder under pressure and thermal loading

Products: Abaqus/Standard Abaqus/Explicit

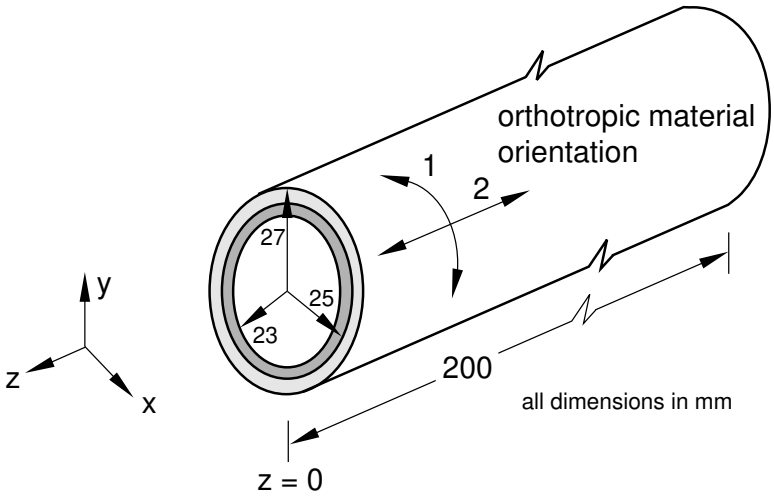
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

C3D8R C3D20 CSS8 S4R S8R SC8R

Problem description



Mesh: One-quarter of the cylinder cross-section and half of the length is modeled. The same problem is analyzed with different meshes. Meshes using linear solids and shells consist of eight elements along the radial and axial directions. Meshes using quadratic solids and general-purpose shells consist of four elements along the radial and axial directions. The models using continuum elements are stacked with two single-layer elements through the thickness in Abaqus/Standard and with four single-layer elements in Abaqus/Explicit. The models using the S4R and S8R elements use a composite section definition. The continuum shell models use two modeling techniques to model the cylinder in the thickness direction: (1) a single element with a composite section and (2) four single-layer elements stacked through the thickness.

Material: Inner isotropic cylinder: E = 2.1E5 MPa, $\nu = 0.3$, $\alpha = 2.0 \times 10^{-5}$/°C.

Outer circumferentially wound cylinder: E_1 = 130 GPa, E_2 = 5 GPa, E_3 = 5 GPa, $\nu_{12} = 0.25$, $\nu_{13} = 0.25$, $\nu_{23} = 0$, G_{12} = 10 GPa, G_{13} = 10 GPa, G_{23} = 5 GPa, $\alpha_{11} = 3.0 \times 10^{-6}$/°C, $\alpha_{22} = 2.0 \times 10^{-5}$/°C, $\alpha_{33} = 2.0 \times 10^{-5}$/°C.

Boundary conditions: $u_z = 0$ at $z = 0$.

Loading: Case 1: Internal pressure of 200 MPa.

Case 2 (Abaqus/Standard only): Internal pressure of 200 MPa + temperature rise of 130°C.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test R0031/2 from NAFEMS publication R0031, "Composites Benchmarks," Issue 2, February 5, 2001.

Results and discussion

The results are given in *Table 1* and *Table 2*. All results at $z = 0$. The values enclosed in parentheses are percentage differences with respect to the reference solution.

Table 1: Abaqus/Standard analysis.

Analysis	Case 1				Case 2			
	inner cylinder		outer cylinder		inner cylinder		outer cylinder	
	σ_θ at r = 23	σ_θ at r = 25	σ_θ at r = 25	σ_θ at r = 27	σ_θ at r = 23	σ_θ at r = 25	σ_θ at r = 25	σ_θ at r = 27
Composite S8R	1421 (–9.2%)	1421 (–0.6%)	878 (0.4%)	878 (15.7%)	1241 (–10.1%)	1241 (–1.5%)	1056 (–3.7%)	1056 (12.8%)
Stacked C3D20	1567 (0.1%)	1432 (0.2%)	880 (0.6%)	756 (–0.4%)	1382 (0.1%)	1262 (0.2%)	1062 (–3.1%)	933 (–0.3%)
Composite SC8R	1477 (–5.6%)	1477 (3.3%)	900 (2.9%)	900 (18.6%)	1299 (–5.9%)	1299 (3.1%)	1078 (-1.6%)	1078 (15.2%)
Stacked SC8R	1552 (–0.9 %)	1470 (2.8%)	849 (–2.9%)	787 (3.7%)	1450 (5.0%)	1319 (4.7%)	1011 (–7.8%)	935 (–0.1%)
Composite SC8R*	1477 (–5.6%)	1477 (3.3%)	900 (2.9%)	900 (18.6%)	1299 (–5.9%)	1299 (3.1%)	1078 (–1.6%)	1078 (15.2%)
Stacked SC8R*	1587 (1.4%)	1491 (4.3%)	848 (–3.1%)	785 (3.4%)	1452 (5.1%)	1318 (4.6%)	1011 (–7.8%)	935 (–0.1%)
Stacked CSS8	1600 (2.2%)	1406 (–1.7%)	876 (0.1%)	732 (–3.6%)	1419 (2.8%)	1245 (–1.2%)	1060 (–3.3%)	904 (–3.4%)
Reference solution	1565	1430	875	759	1381	1260	1096	936

* Abaqus/Standard results with enhanced hourglass control.

Table 2: Abaqus/Explicit analysis.

Analysis	Case 1			
	inner cylinder		outer cylinder	
	σ_θ at r = 23	σ_θ at r = 25	σ_θ at r = 25	σ_θ at r = 27
Composite S4R	1417 (–9.5%)	1427 (–0.2%)	879 (0.5%)	879 (15.8%)
Stacked C3D8R	1548 (–1.1%)	1479 (3.5%)	850 (–2.8%)	788 (3.8%)
Reference solution	1565	1430	875	759

Input files

Abaqus/Standard input files

nco2s8rx.inp	Composite S8R model.
nco2c3d2.inp	Stacked C3D20 model.
r312_std_sc8r_composite.inp	Composite SC8R model.
r312_std_sc8r_stacked.inp	Stacked SC8R model.
r312_std_sc8r_composite_eh.inp	Composite SC8R model with enhanced hourglass control.
r312_std_sc8r_stacked_eh.inp	Stacked SC8R model with enhanced hourglass control.
nco2c3d2_css8_exp.inp	Stacked CSS8 model.

Abaqus/Explicit input files

r312shl.inp	Composite S4R model.
r312sol.inp	Stacked C3D8R model.

R0031(3): Three-layer sandwich shell under normal pressure loading

Products: Abaqus/Standard Abaqus/Explicit

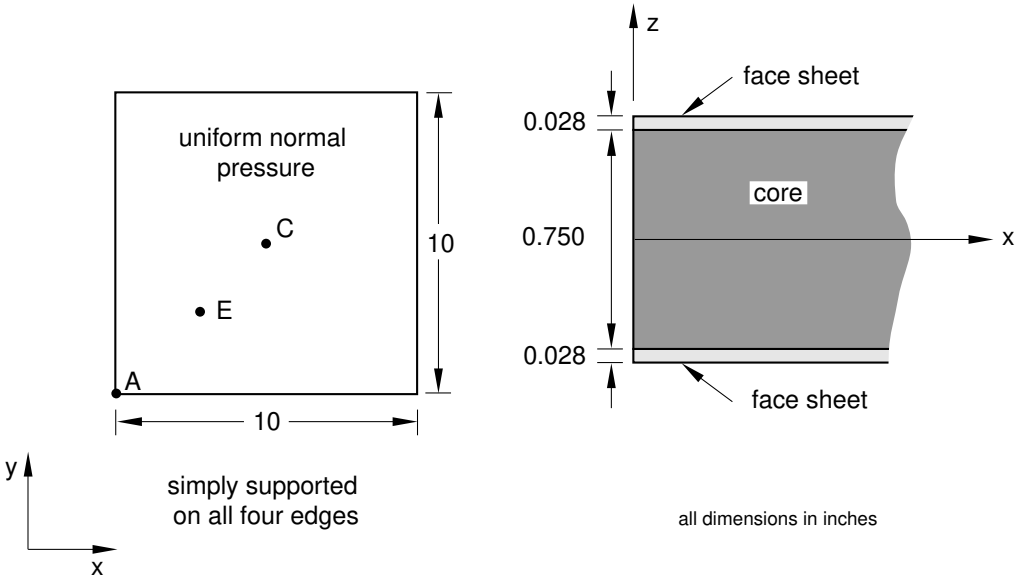
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

S4R S8R SC6R SC8R

Problem description



Mesh: One-quarter model with a 4 × 4 mesh of composite S8R elements in Abaqus/Standard, an 8 × 8 mesh of composite S4R elements in Abaqus/Explicit, and an 8 × 8 mesh of continuum shell elements in Abaqus/Standard. Two types of continuum shell models are provided: (1) a single composite element and (2) three single-layer elements stacked in the thickness direction.

Material: Face sheets: $E_1 = 1.0 \times 10^7$ psi, $E_2 = 4.0 \times 10^6$ psi, $\nu_{12} = 0.3$, $G_{12} = 1.875 \times 10^6$ psi, $G_{13} = 1.875 \times 10^6$ psi, $G_{23} = 1.875 \times 10^6$ psi

Core: $E_1 = 10.0$ psi, $E_2 = 10.0$ psi, $\nu_{12} = 0$, $G_{12} = 10.0$ psi, $G_{13} = 3.0\text{E}4$ psi, $G_{23} = 1.2\text{E}4$ psi. The thickness moduli in continuum shell models are chosen sufficiently high to avoid pinching effects, which are neglected in the analytical solution.

Boundary conditions: Simply supported on all four edges. The continuum shell models use an equation constraint to provide an equivalent midsurface constraint.

Loading: Uniform normal pressure of 100 psi.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test R0031/3 from NAFEMS publication R0031, "Composites Benchmarks," February 1995.

Results and discussion

The results are given in *Table 1* and *Table 2*. The values enclosed in parentheses are percentage differences with respect to the reference solution. The displacements reported for the stacked continuum shell model are the average displacements of the bottom and top skins.

Table 1: Abaqus/Standard analysis.

Model	u_z at C	σ_{11} at C	σ_{22} at C	σ_{12} at E
NAFEMS	–0.123	34449	13350	–5068
S8R	–0.122 (0.6%)	35307 (2.5%)	13802 (3.4%)	–5236 (3.3%)
Composite SC6R	–0.120 (–2.4%)	34687 (0.7%)	13675 (–1.8%)	–5132 (2.5%)
Stacked SC6R	–0.129 (4.8%)	35382 (2.7%)	13745 (–1.3%)	–5219 (3.0%)
Composite SC8R	–0.122 (–0.8%)	35312 (2.5%)	13805 (–0.9%)	–5237 (3.3%)
Stacked SC8R	–0.131 (6.5%)	36118 (5.7%)	13900 (4.1%)	–5312 (4.8%)

Table 2: Abaqus/Explicit analysis.

Model	u_z at C	σ_{11} at C	σ_{22} at C	σ_{12} at E
NAFEMS	–0.123	34449	13932	–5068
S4R	–0.135 (9.8%)	36272 (5.3%)	13287 (5.8%)	–5696 (12.3%)*

*Nodal stress value obtained by averaging integration point stress values of all adjoining elements.

Input files

Abaqus/Standard input files

nco3s8rx.inp	S8R elements.
r313_std_sc6r_composite.inp	Composite SC6R analysis.
r313_std_sc6r_stacked.inp	Stacked SC6R analysis.
r313_std_sc8r_composite.inp	Composite SC8R analysis.
r313_std_sc8r_stacked.inp	Stacked SC8R analysis.

Abaqus/Explicit input file

r313shl.inp	S4R elements.

Geometric nonlinear tests

In this section:

- *3DNLG-1: Elastic large deflection response of a Z-shaped cantilever under an end load*
- *3DNLG-2: Elastic large deflection response of a pear-shaped cylinder under end shortening*
- *3DNLG-3: Elastic lateral buckling of a right angle frame under in-plane end moments*
- *3DNLG-4: Lateral torsional buckling of an elastic cantilever subjected to a transverse end load*
- *3DNLG-5: Large deflection of a curved elastic cantilever under transverse end load*
- *3DNLG-6: Buckling of a flat plate when subjected to in-plane shear*
- *3DNLG-7: Elastic large deflection response of a hinged spherical shell under pressure loading*
- *3DNLG-8: Collapse of a straight pipe segment under pure bending*
- *3DNLG-9: Large elastic deflection of a pinched hemispherical shell*
- *3DNLG-10: Elastic-plastic behavior of a stiffened cylindrical panel under compressive end load*

3DNLG-1: Elastic large deflection response of a Z-shaped cantilever under an end load

Products: Abaqus/Standard

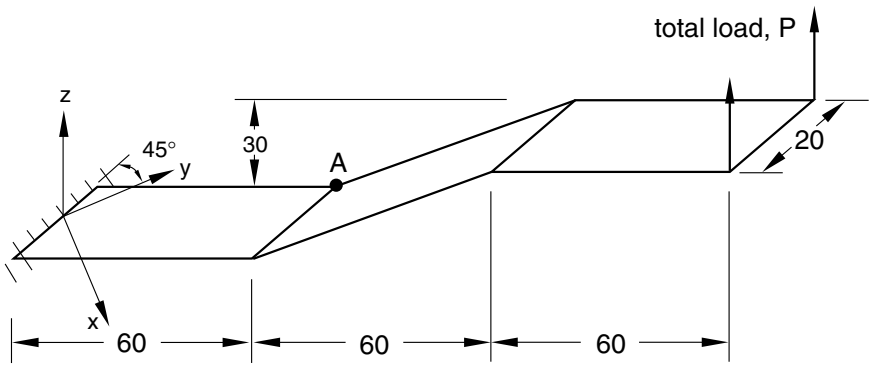
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

B31 B31H B32 B32H B33 B33H
CSS8
S4 S4R
SC6R SC8R

Problem description



Model: Uniform thickness (t = 1.7).

Material: Linear elastic, Young's modulus = 2.0×10^5, Poisson's ratio = 0.3.

Boundary conditions: All degrees of freedom restrained at built-in end.

Loading: Concentrated end load (P = 4000).

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 3DNLG-1 from NAFEMS Publication R0024 “A Review of Benchmark Problems for Geometric Non-linear Behaviour of 3D Beams and Shells (SUMMARY).”

The published results for this problem were obtained with Abaqus. Thus, a comparison of Abaqus and NAFEMS results is not an independent verification of Abaqus. The NAFEMS study includes results from other sources for comparison that may provide a basis for verification of this problem.

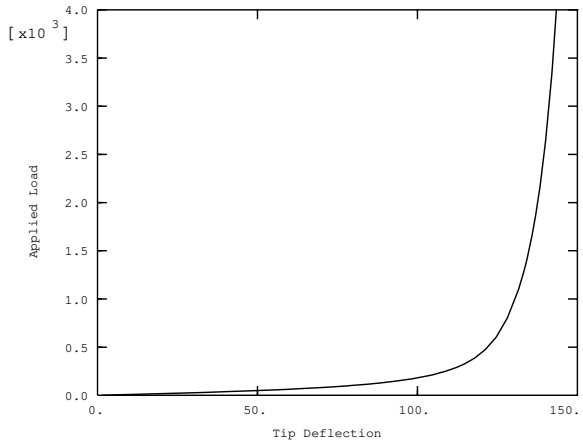
Results and discussion

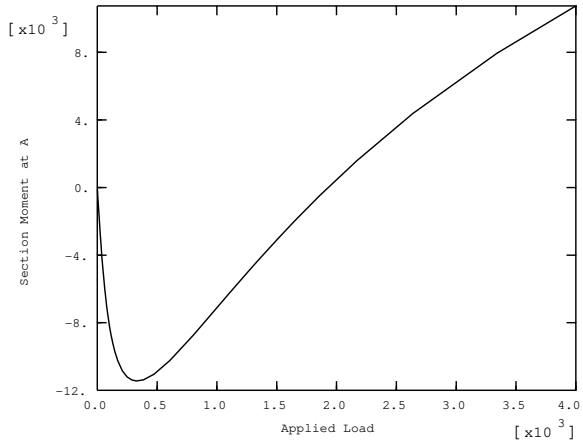
Displacements converge faster than stresses. Even though the displacements seem to have converged, the lower-order elements need more refined meshes (compared to the higher-order elements) before the stresses are observed to converge. Stresses are most accurate at the integration points within the element. When stress values are extrapolated from the integration points to the nodes and then averaged, the stress values calculated may not capture the peak values if a stress gradient is present. Since the higher-order elements (B32, B32H, B33, and B33H) use linear extrapolation within an element, a stress gradient in an element may be captured adequately when extrapolating stresses to the nodes. However, constant extrapolation is used for linear elements (B31, B31H, S4, S4R, and SC8R), which results in slow convergence of nodal stress values. Higher mesh refinement near stress gradients is needed for such elements.

Tip Displacement				
Element Type	Number of Elements	Applied Load		
		104.5	1263.0	4000.0
B31	72	80.42	133.1	143.5
B31H	72	80.42	133.1	143.5
B32	9	80.42	133.1	143.4
B32H	9	80.42	133.1	143.4
B33	9	80.42	133.1	143.4
B33H	9	80.42	133.1	143.4
S4	1 × 72	80.42	133.1	143.5
S4R	1 × 72	80.42	133.1	143.5
SC6R	2 × 72 × 1	80.61	133.1	143.5
SC8R	1 × 72 × 1	80.61	133.1	143.5
CSS8	1 × 72 × 1	80.46	133.1	143.5

Moment at A				
Element Type	Number of Elements	Applied Load		
		104.5	1263.0	4000.0
B31	72	–8333	–5508	9926
B31H	72	–8333	–5508	9925
B32	9	–8308	–4962	10747
B32H	9	–8308	–4961	10746
B33	9	–8316	–4982	10659
B33H	9	–8317	–4979	10661
S4	1 × 72	–8334	–5506	9940
S4R	1 × 72	–8334	–5506	9940
SC6R	2 × 72 × 1	–8358	–5608	9720
SC8R	1 × 72 × 1	–8324	–5466	9832

Response predicted by Abaqus (element B32)





Input files

n3g1x33x_b31.inp	B31 elements.
n3g1x33x_b31h.inp	B31H elements.
n3g1x33x_b32.inp	B32 elements.
n3g1x33x_b32h.inp	B32H elements.
n3g1x33x_b33.inp	B33 elements.
n3g1x33x_b33h.inp	B33H elements.
n3g1x33x_s4.inp	S4 elements.
n3g1x33x_s4r.inp	S4R elements.
nlg1_std_sc6r.inp	SC6R elements.
nlg1_std_sc8r.inp	SC8R elements.
nlg1_std_css8.inp	CSS8 elements.

3DNLG-2: Elastic large deflection response of a pear-shaped cylinder under end shortening

Products: Abaqus/Standard

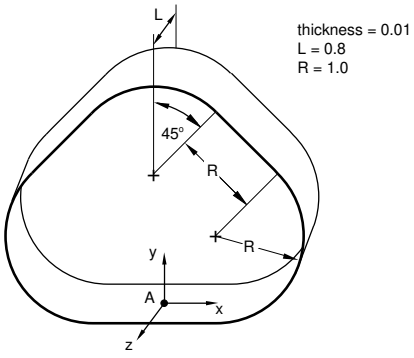
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

S3R S4 S4R S4R5 S8R S8R5 S9R5
STRI3 STRI65
SC6R SC8R

Problem description



Model: A quarter model is used because of symmetry.

Material: Young's modulus = 1.0×10^7, Poisson's ratio = 0.3.

Boundary conditions: Symmetry on plane $x = 0$ ($\boldsymbol{U_x = \theta_y = \theta_z = 0}$). Symmetry on plane $z = 0$ ($\boldsymbol{U_z = \theta_x = \theta_y = 0}$). Simply supported on plane $z = \boldsymbol{L/2} = 0.4$ ($\boldsymbol{U_x = U_y = 0}$).

Loading: Uniform end shortening. A displacement of –0.0016 is applied incrementally using the RIKS algorithm to all nodes lying on one end of the cylinder with the plane $z = 0.4$.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 3DNLG-2 from NAFEMS Publication R0024 “A Review of Benchmark Problems for Geometric Non-linear Behaviour of 3D Beams and Shells (SUMMARY).”

The published results of this problem were obtained with Abaqus. Thus, a comparison of Abaqus and NAFEMS results is not an independent verification of Abaqus. The NAFEMS study includes results from other sources for comparison that may provide a basis for verification of this problem.

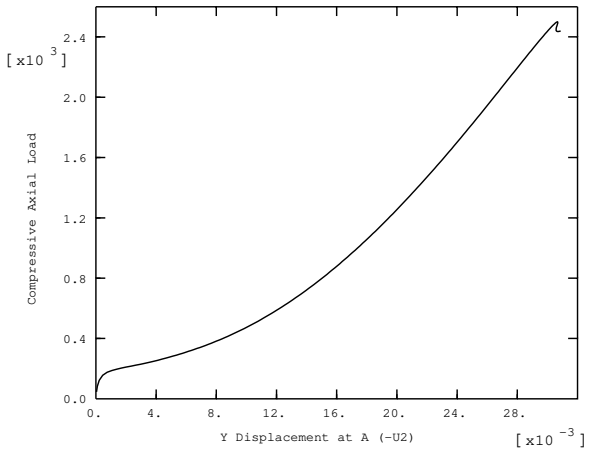
Results and discussion

All elements are tested with meshes using the same nodal spacing.

The collapse load is defined as the loading point in the deformation path at which further deformation occurs without an increase in the applied load. The path after this point is termed the postbuckling path. The postbuckling path shows a decrease in the load with further deformation; the postbuckled structure can attain equilibrium only at a load level lower than the buckling load once it has buckled.

Collapse Loads		
Element	P_{max}	U_2
S3R	2708	–0.03168
S4R	2626	–0.03124
S4	2619	–0.03116
S4R5	2604	–0.03127
S8R	2490	–0.03072
S8R5	2472	–0.03063
S9R5	2474	–0.03034
STRI3	2596	–0.03096
STRI65	2460	–0.03051
SC6R	2653	–0.03166
SC8R	2601	–0.03155

Response predicted by Abaqus (element S8R5)



Input files

n3g2x58x_s3r.inp S3R elements.

n3g2x58x_s4.inp	S4 elements.
n3g2x58x_s4r.inp	S4R elements.
n3g2x58x_s4r5.inp	S4R5 elements.
n3g2x58x_s8r.inp	S8R elements.
n3g2x58x_s8r5.inp	S8R5 elements.
n3g2x58x_s9r5.inp	S9R5 elements.
n3g2x58x_stri3.inp	STRI3 elements.
n3g2x58x_stri65.inp	STRI65 elements.
nlg2_std_sc6r.inp	SC6R elements.
nlg2_std_sc8r.inp	SC8R elements.

3DNLG-3: Elastic lateral buckling of a right angle frame under in-plane end moments

Products: Abaqus/Standard

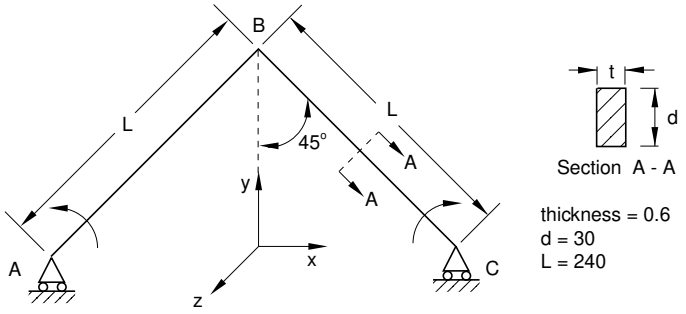
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

B31 B31H B32 B32H

Problem description



Material: Young's modulus = 7.1240×10^4, Poisson's ratio = 0.31.

Boundary conditions: Symmetry on plane $x = 0$ ($\boldsymbol{U_x = \theta_y = \theta_z = 0}$). Only x-translation and z-rotation at end A ($\boldsymbol{U_z = U_y = \theta_x = \theta_y = 0}$).

Loading: i) Prescribe z-rotation at end A from 0 to $2\boldsymbol{\pi}$ using an arc length solution procedure. ii) Unload the applied rotation. iii) To overcome the initial singularity, apply a perturbation load $\boldsymbol{P_z}$ ($\boldsymbol{P_z}$ = 0.01). This load is subsequently reduced to a negligible value ($\boldsymbol{P_z}$ = 1.0×10^{-7}) or zero once the singularity is overcome. The perturbation load is an order of magnitude larger than the load used by NAFEMS, which ensures that the structure initially buckles into the desired mode, regardless of mesh size or element used.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 3DNLG-3 from NAFEMS Publication R0024 “A Review of Benchmark Problems for Geometric Non-linear Behaviour of 3D Beams and Shells (SUMMARY).”

The published results of this problem were obtained with Abaqus. Thus, a comparison of Abaqus and NAFEMS results is not an independent verification of Abaqus. The NAFEMS study includes results from other sources for comparison that may provide a basis for verification of this problem.

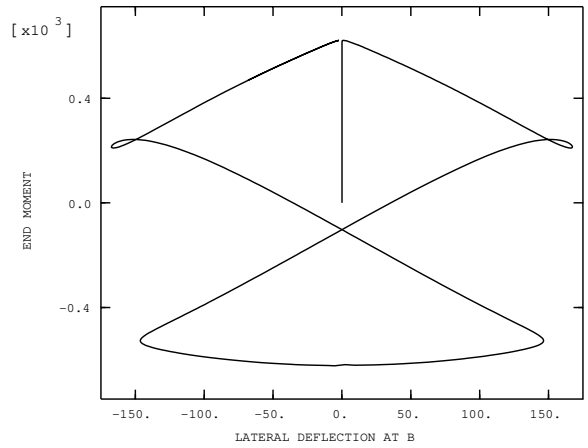
Results and discussion

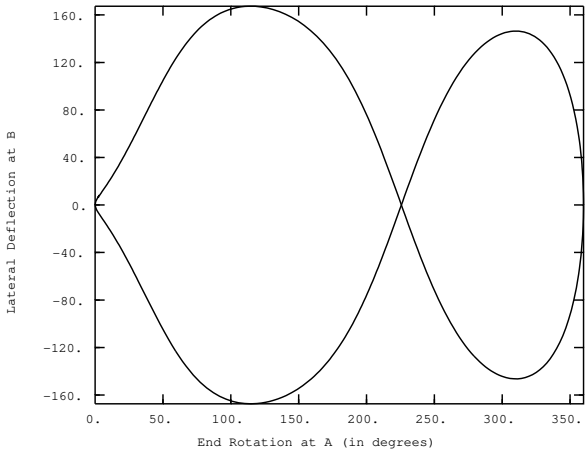
Elements B32 and B32H use a mesh with 21 nodes (10 elements) to model half of the frame using symmetry, and elements B31 and B31H use a mesh of 41 nodes (40 elements).

The reaction moment at A and the tip lateral deflection at B are given at different rotations of end A, making comparison of the response of different elements difficult. This problem tests the automatic arc length (RIKS) algorithm as well as the large-rotation formulation of the beam elements. The "increment size" (more appropriately, arc length) varies throughout an analysis and will be different for each mesh or element used. In addition, the RIKS algorithm allows the analysis to be terminated only after a user-defined load or nodal displacement (or rotation) value is exceeded. Thus, the results from an analysis with different elements or meshes usually cannot be compared at arbitrary points in the deformation history. If results are desired at specific points along the deformation path, a limit can be set on the arc length to give output at closely spaced loading increments, or interpolation of the analysis data can be used to obtain approximate analysis results at intermediate loading values. In this problem, tight arc length tolerances are not needed to resolve the buckling behavior of the structure accurately.

Abaqus Solution			
Element	**Applied**	**End A**	**Tip B Lateral**
Type	**Rotation (deg.) at A**	**Moment**	**Deflection**
B31	123.8	221.8	166.6
	311.3	−529.8	−146.3
	359.9	−633.6	−7.49
B31H	118.2	216.8	167.3
	315.0	−534.8	−145.8
	359.8	−632.8	−7.97
B32	122.3	220.4	166.8
	311.4	−529.0	−146.4
	359.9	−619.5	−3.19
B32H	128.7	226.5	165.3
	310.4	−527.8	−146.4
	359.9	−623.8	−3.28

Response predicted by Abaqus (element B32)





Input files

There are two analysis files for each element tested. Two files are needed since the removal of the perturbation load requires a general static step following the initial RIKS step, and a RIKS analysis must be the last step in an analysis.

n3g3x331_b31.inp	B31 elements, analysis 1.
n3g3x331_b31h.inp	B31H elements, analysis 1.
n3g3x331_b32.inp	B32 elements, analysis 1.
n3g3x331_b32h.inp	B32H elements, analysis 1.
n3g3x332_b31.inp	B31 elements, analysis 2.
n3g3x332_b31h.inp	B31H elements, analysis 2.
n3g3x332_b32.inp	B32 elements, analysis 2.
n3g3x332_b32h.inp	B32H elements, analysis 2.

3DNLG-4: Lateral torsional buckling of an elastic cantilever subjected to a transverse end load

Products: Abaqus/Standard

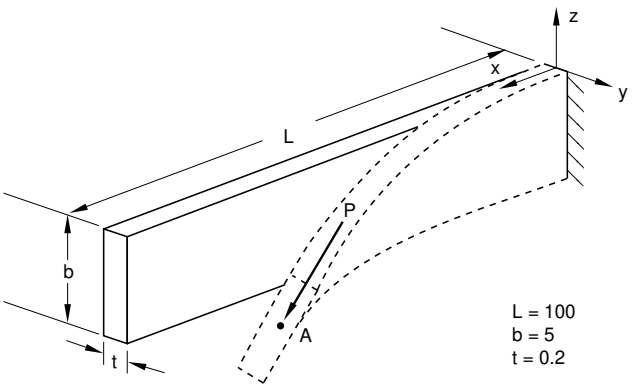
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

S3R S4 S4R S4R5 S8R S8R5 S9R5
STRI3 STRI65
SC6R SC8R

Problem description



Material: Young's modulus = 1.0×10^4, Poisson's ratio = 0.0.

Boundary conditions: All degrees of freedom restrained at built-in end.

Loading: a) Conservative load: apply concentrated nodal force P using an arc length procedure. In one test the conservative load is applied with a distributed edge traction instead of a nodal force. $P_{max} = 0.017$. Terminate when y-displacement at A exceeds 0.06. b) Nonconservative load: apply concentrated nodal follower force P using an arc length procedure. In one test the nonconservative load is applied with a distributed edge traction instead of a nodal force. The continuum shell models for the nonconservative load case include two conventional shell elements (with a reduced stiffness) overlaid on the continuum shell element mesh to activate the rotational degrees of freedom for the follower force. $P_{max} = 0.032$. There is an initial out-of-plane imperfection (y-coordinate) in the mesh.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 3DNLG-4 from NAFEMS Publication R0024 “A Review of Benchmark Problems for Geometric Non-linear Behaviour of 3D Beams and Shells (SUMMARY).”

The published results of this problem were obtained with Abaqus. Thus, a comparison of Abaqus and NAFEMS results is not an independent verification of Abaqus. The NAFEMS study includes results from other sources for comparison that may provide a basis for verification of this problem.

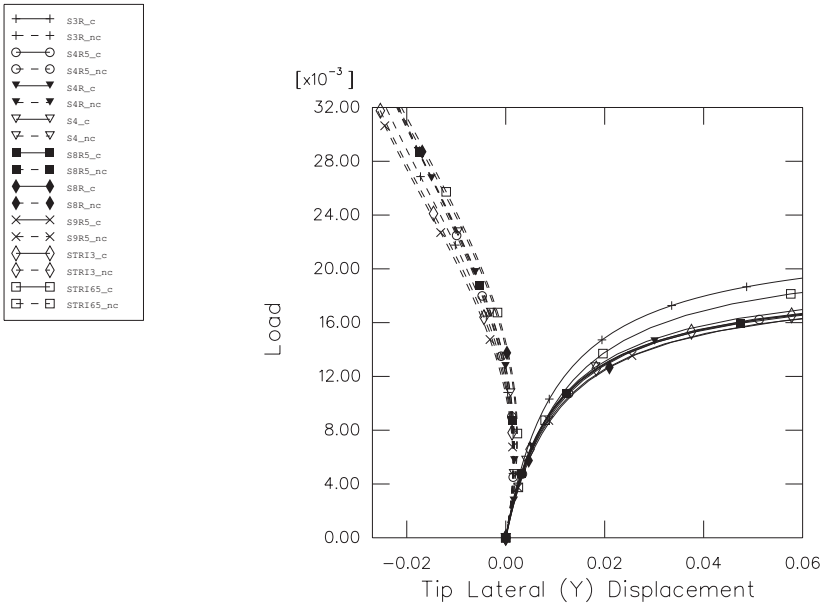
Results and discussion

Mesh refinement is needed to evaluate the accuracy of some of the elements; however, the imperfection is specified at a number of discrete points that serve as the nodal positions of the original mesh used by NAFEMS. Refining the mesh introduces a change in the geometry since the imperfection is not known at points other than the nodal positions of the original mesh. Therefore, results are reported using the same nodal spacing for all element types (the same nodal spacing with the omission of one center node per element is used in the S8R and S8R5 meshes).

The results for the continuum shell models, not depicted in the figure, show similar load-displacement results as for the conventional shell element models.

The results for the edge traction loading, not depicted in the figure, show nearly identical results as for the nodal loads.

Response predicted by Abaqus



Input files

Conservative loading case:

n3g4x541_s3r.inp S3R elements.

n3g4x541_s4.inp	S4 elements.
n3g4x541_s4r.inp	S4R elements.
n3g4x541_s4r_edld.inp	S4R elements, loaded with a distributed edge traction.
n3g4x541_s4r5.inp	S4R5 elements.
n3g4x541_s8r.inp	S8R elements.
n3g4x541_s8r5.inp	S8R5 elements.
n3g4x541_s9r5.inp	S9R5 elements.
n3g4x541_stri3.inp	STRI3 elements.
n3g4x541_stri65.inp	STRI65 elements.
nlg4_std_sc6r_1.inp	SC6R elements.
nlg4_std_sc8r_1.inp	SC8R elements.

Nonconservative loading case:

n3g4x542_s3r.inp	S3R elements.
n3g4x542_s4.inp	S4 elements.
n3g4x542_s4r.inp	S4R elements.
n3g4x542_s4r_edld.inp	S4R elements, loaded with a distributed edge traction.
n3g4x542_s4r5.inp	S4R5 elements.
n3g4x542_s8r.inp	S8R elements.
n3g4x542_s8r5.inp	S8R5 elements.
n3g4x542_s9r5.inp	S9R5 elements.
n3g4x542_stri3.inp	STRI3 elements.
n3g4x542_stri65.inp	STRI65 elements.
nlg4_std_sc6r_2.inp	SC6R elements.
nlg4_std_sc8r_2.inp	SC8R elements.

3DNLG-5: Large deflection of a curved elastic cantilever under transverse end load

Products: Abaqus/Standard

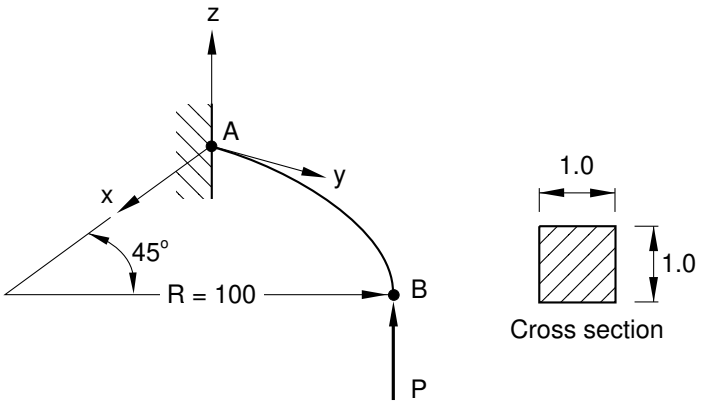
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

B31 B31H B32 B32H B33 B33H

Problem description



Material: Young's modulus = 1.0×10^7, Poisson's ratio = 0.0.

Boundary conditions: All degrees of freedom restrained at *A*.

Loading: Concentrated load at *B* ($\boldsymbol{P_z}$ = 3000).

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 3DNLG-5 from NAFEMS Publication R0024 “A Review of Benchmark Problems for Geometric Non-linear Behaviour of 3D Beams and Shells (SUMMARY).”

The published results of this problem were obtained with Abaqus. Thus, a comparison of Abaqus and NAFEMS results is not an independent verification of Abaqus. The NAFEMS study includes results from other sources for comparison that may provide a basis for verification of this problem.

Results and discussion

All elements are tested with meshes using the same nodal spacing.

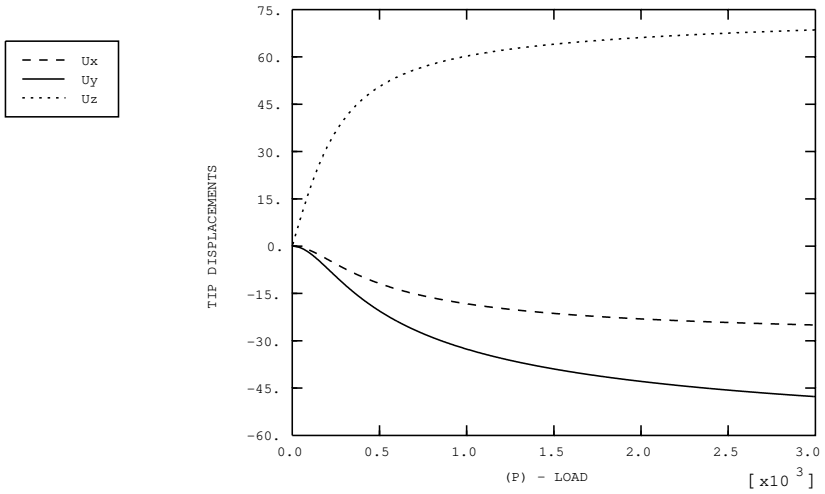
The results vary slightly between different element types. The maximum difference in the reported displacement components is 1.3%. The hybrid element results are identical to their non-hybrid counterparts to the precision reported here. The tip displacement components are compared in the tables below.

Load	**Tip Displacement (U_x)**		
P_z	**B31(H)**	**B32(H)**	**B33(H)**
300.0	−7.097	−7.173	−7.188
450.0	−10.82	−10.92	−10.93
600.0	−13.62	−13.73	−13.74
3000.0	−24.99	−25.06	−25.04

Load	Tip Displacement (U_y)		
P_z	B31(H)	B32(H)	B33(H)
300.0	−12.13	−12.17	−12.19
450.0	−18.70	−18.74	−18.76
600.0	−23.78	−23.81	−23.85
3000.0	−47.70	−47.70	−47.78

Load	Tip Displacement (U_z)		
P_z	B31(H)	B32(H)	B33(H)
300.0	40.43	40.47	40.50
450.0	48.67	48.70	48.73
600.0	53.58	53.60	53.63
3000.0	68.55	68.46	68.52

Response predicted by Abaqus (element B31)



Input files

n3g5x32x_b31.inp	B31 elements.
n3g5x32x_b31h.inp	B31H elements.
n3g5x32x_b32.inp	B32 elements.
n3g5x32x_b32h.inp	B32H elements.
n3g5x32x_b33.inp	B33 elements.
n3g5x32x_b33h.inp	B33H elements.

3DNLG-6: Buckling of a flat plate when subjected to in-plane shear

Products: Abaqus/Standard

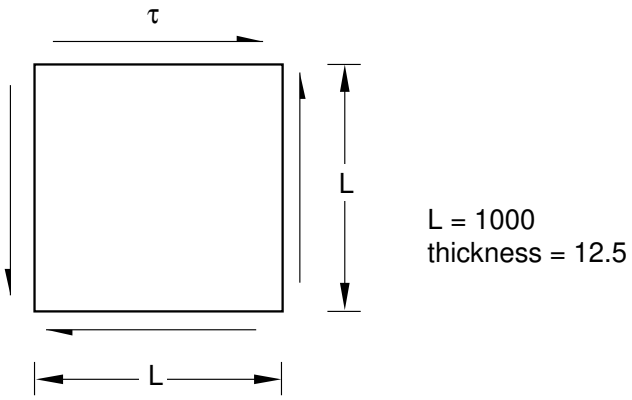
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

S3 S3R S4 S4R S4R5 S8R S8R5 S9R5
STRI3 STRI65
SC6R SC8R

Problem description



Material: Young's modulus = 6.4×10^6, Poisson's ratio = 0.3.

Boundary conditions: $U_z = 0$ along all edges. $U_x = U_y = \theta_z = 0$ at center node.

Loading: Uniform shear load around edges. The RIKS algorithm is used to increment the shear load to a maximum of 2.12×10^8 per edge.

Initial imperfection: The midsurface position of the plate is defined by

$$z = 0.5t \frac{f(x,y)}{f(L/2,L/2)},$$

where $f(x,y) = \Sigma_m \Sigma_n a_{mn} \sin \frac{(m\pi x)}{L} \sin \frac{(n\pi x)}{L}$, a_{11} = 1.0, a_{22} = 0.2897, a_{13} = –0.0706, a_{31} = –0.0691, a_{33} = 0.0384, and a_{42} = 0.0032.

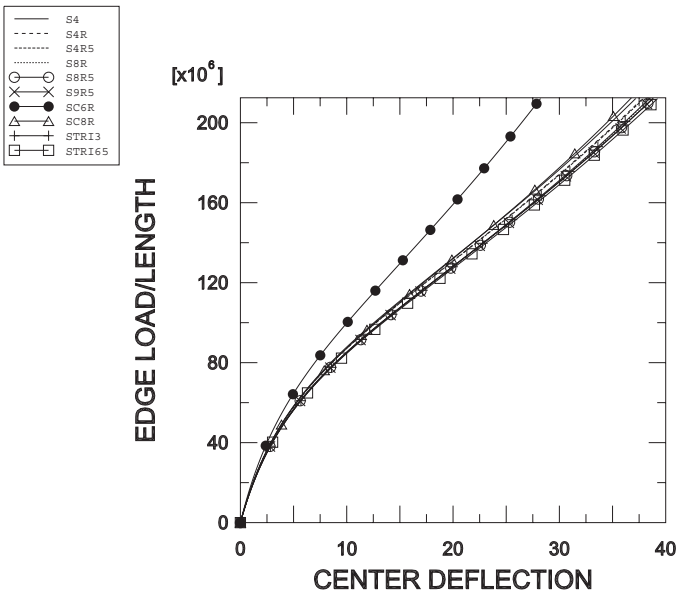
Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 3DNLG-6 from NAFEMS Publication R0024 “A Review of Benchmark Problems for Geometric Non-linear Behaviour of 3D Beams and Shells (SUMMARY).”

The published results of this problem were obtained with Abaqus. Thus, a comparison of Abaqus and NAFEMS results is not an independent verification of Abaqus. The NAFEMS study includes results from other sources for comparison that may provide a basis for verification of this problem.

Results and discussion

Very similar load-deflection curves are obtained for all the test cases. The SC6R model shows a slightly stiffer response than the other elements. The response of all elements tested is shown below.



Input files

n3g6xf3x.inp	S3/S3R elements.
n3g6xe4x.inp	S4 elements.
n3g6xf4x.inp	S4R elements.
n3g6x54x.inp	S4R5 elements.
n3g6x68x.inp	S8R elements.
n3g6x58x.inp	S8R5 elements.
n3g6x59x.inp	S9R5 elements.
n3g6x63x.inp	STRI3 elements.
n3g6x56x.inp	STRI65 elements.
nlg6_std_sc6r.inp	SC6R elements.
nlg6_std_sc8r.inp	SC8R elements.

3DNLG-7: Elastic large deflection response of a hinged spherical shell under pressure loading

Products: Abaqus/Standard

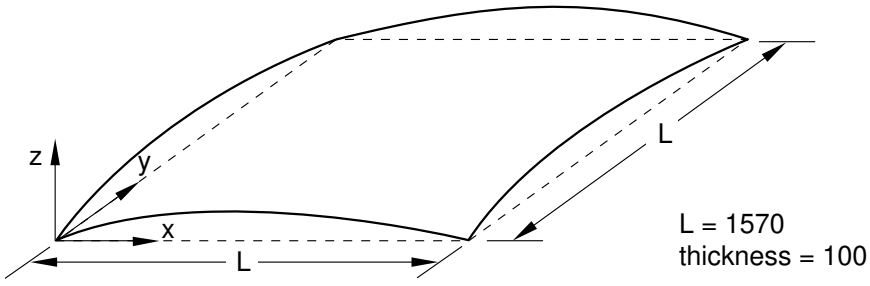
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CSS8
S3 S3R S4 S4R S4R5 S8R S8R5 S9R5
STRI3 STRI65
SC6R SC8R

Problem description



Model: The shell midsurface is defined in terms of global Cartesian coordinates, where

$$z = 2.0285 \times 10^{-4} \{x(L-x) + y(L-y)\}.$$

Material: Young's modulus = 69, Poisson's ratio = 0.3.

Boundary conditions: Simply supported along all edges ($U_x = U_y = U_z = 0$).

Loading: Evenly distributed follower pressure load normal to shell surface. p_{max} = 0.1.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 3DNLG-7 from NAFEMS Publication R0024 “A Review of Benchmark Problems for Geometric Non-linear Behaviour of 3D Beams and Shells (SUMMARY).”

The published results of this problem were obtained with Abaqus. Thus, a comparison of Abaqus and NAFEMS results is not an independent verification of Abaqus. The NAFEMS study includes results from other sources for comparison that may provide a basis for verification of this problem.

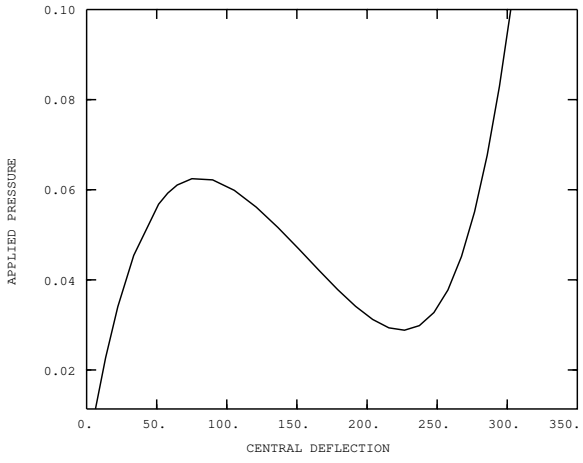
Results and discussion

In the following table, limit points 1 and 2 correspond to the peak and local minimum, respectively, of the load-displacement curve. All the meshes share the same nodal spacing.

Element	Limit point 1		Limit point 2	
	Pressure	U_z at center	Pressure	U_z at center
S3/S3R	0.06639	78.15	0.03300	220.9
S4	0.06581	78.84	0.03133	223.5
S4R	0.06581	78.98	0.03134	223.5
S4R5	0.06281	79.62	0.02956	224.4
S8R	0.06263	79.14	0.02849	223.6
S8R5	0.06261	79.21	0.02882	223.7
S9R5	0.06261	79.08	0.02883	223.4
STRI3	0.06397	78.61	0.03175	221.9
STRI65	0.06244	79.26	0.02886	224.0
SC6R	0.06749	81.0	0.03388	217.1
SC8R	0.06679	81.9	0.03213	217.6
CSS8	0.06619	83.9	0.03278	221.1

Response predicted by Abaqus

Similar load-displacement curves are obtained for all test cases. The response predicted using S8R5 elements is shown below.



Input files

n3g7xf3x.inp	S3/S3R elements.
n3g7xe4x.inp	S4 elements.
n3g7xf4x.inp	S4R elements.

n3g7x54x.inp	S4R5 elements.
n3g7x68x.inp	S8R elements.
n3g7x58x.inp	S8R5 elements.
n3g7x59x.inp	S9R5 elements.
n3g7x63x.inp	STRI3 elements.
n3g7x56x.inp	STRI65 elements.
nlg7_std_sc6r.inp	SC6R elements.
nlg7_std_sc8r.inp	SC8R elements.
nlg7_std_css8.inp	CSS8 elements.

3DNLG-8: Collapse of a straight pipe segment under pure bending

Products: Abaqus/Standard

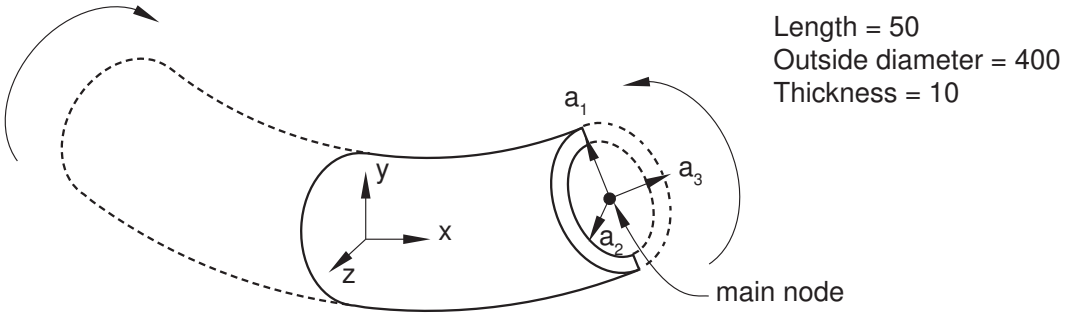
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

S3 S3R S4 S4R S4R5 S8R S8R5 S9R5
STRI3 STRI65

Problem description



Elastic and elastic-plastic cases of a beam under pure bending are tested. Nonlinear multipoint constraints and section ovalization are also tested.

Model: A quarter model is used because of symmetry.

Material: Young's modulus = 1.93×10^5, Poisson's ratio = 0.26, plastic (isotropic hardening).

Equiv. Stress	Equiv. Plastic Strain
272	0.0
346	0.00473
379	0.01264
404	0.02836
424	0.0491

Boundary conditions: Symmetry on plane $z = 0$ ($U_z = \theta_x = \theta_y = 0$); symmetry on plane $x = 0$ ($U_x = \theta_y = \theta_z = 0$); symmetry on rotated plane: i) Shell nodes must remain in rotated plane (Nonlinear MPC). ii) The z-rotation of shell nodes in the rotated plane must be the same as the main node (Linear MPC). iii) The component of rotation about the local $\boldsymbol{a_1}$-axis must be zero (Nonlinear MPC). The main node is constrained to move on the x-axis only.

Loading: *Z*-rotation applied to pipe end (θ_z^{max} = 0.004).

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 3DNLG-8 from NAFEMS Publication R0024 “A Review of Benchmark Problems for Geometric Non-linear Behaviour of 3D Beams and Shells (SUMMARY).”

The published results of this problem were obtained with Abaqus. Thus, a comparison of Abaqus and NAFEMS results is not an independent verification of Abaqus. The NAFEMS study includes results from other sources for comparison that may provide a basis for verification of this problem.

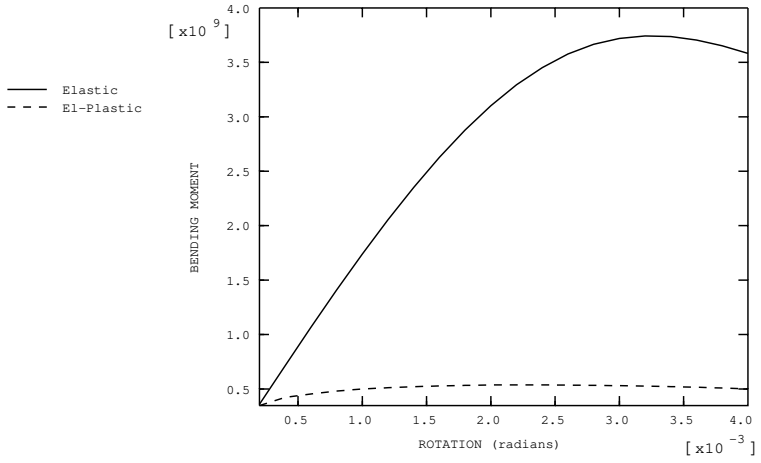
Results and discussion

In the following table, the end moments for the elastic and elastic-plastic cases are reported at three rotation values of the main node. All the meshes have the same nodal spacing.

Element	θ = 0.0012		θ = 0.0024		θ = 0.004	
	Moment × 10^{-9}		Moment × 10^{-9}		Moment × 10^{-9}	
	Elastic	Plastic	Elastic	Plastic	Elastic	Plastic
S3/S3R	2.056	0.5146	3.488	0.5402	3.704	0.5066
S4	2.032	0.5088	3.452	0.5350	3.676	0.5042
S4R	2.020	0.5080	3.436	0.5332	3.664	0.5024
S4R5	2.018	0.5078	3.420	0.5334	3.578	0.5008
S8R	2.054	0.5116	3.448	0.5380	3.568	0.5024
S8R5	2.054	0.5118	3.452	0.5380	3.582	0.5030
S9R5	2.054	0.5118	3.452	0.5380	3.582	0.5030
STRI3	2.056	0.5144	3.474	0.5390	3.662	0.5036
STRI65	2.054	0.5122	3.452	0.5388	3.592	0.5046

Response predicted by Abaqus

Essentially identical moment-rotation curves (at the main node) are obtained for all test cases. The response predicted using S8R5 elements is shown below.



Input files

Elasticity:

n3g8xf31.inp	S3/S3R elements.
n3g8xf31.f	User subroutine *MPC* used in n3g8xf31.inp.
n3g8xe41.inp	S4 elements.
n3g8xe41.f	User subroutine *MPC* used in n3g8xe41.inp.
n3g8xf41.inp	S4R elements.
n3g8xf41.f	User subroutine *MPC* used in n3g8xf41.inp.
n3g8x541.inp	S4R5 elements.
n3g8x541.f	User subroutine *MPC* used in n3g8x541.inp.
n3g8x681.inp	S8R elements.
n3g8x681.f	User subroutine *MPC* used in n3g8x681.inp.
n3g8x581.inp	S8R5 elements.
n3g8x581.f	User subroutine *MPC* used in n3g8x581.inp.
n3g8x591.inp	S9R5 elements.
n3g8x591.f	User subroutine *MPC* used in n3g8x591.inp.
n3g8x631.inp	STRI3 elements.
n3g8x631.f	User subroutine *MPC* used in n3g8x631.inp.
n3g8x561.inp	STRI65 elements.
n3g8x561.f	User subroutine *MPC* used in n3g8x561.inp.

Elastoplasticity:

n3g8xf32.inp	S3/S3R elements.
n3g8xf32.f	User subroutine *MPC* used in n3g8xf32.inp.
n3g8xe42.inp	S4 elements.
n3g8xe42.f	User subroutine *MPC* used in n3g8xe42.inp.
n3g8xf42.inp	S4R elements.
n3g8xf42.f	User subroutine *MPC* used in n3g8xf42.inp.
n3g8x542.inp	S4R5 elements.

n3g8x542.f	User subroutine *MPC* used in n3g8x542.inp.
n3g8x682.inp	S8R elements.
n3g8x682.f	User subroutine *MPC* used in n3g8x682.inp.
n3g8x582.inp	S8R5 elements.
n3g8x582.f	User subroutine *MPC* used in n3g8x582.inp.
n3g8x592.inp	S9R5 elements.
n3g8x592.f	User subroutine *MPC* used in n3g8x592.inp.
n3g8x632.inp	STRI3 elements.
n3g8x632.f	User subroutine *MPC* used in n3g8x632.inp.
n3g8x562.inp	STRI65 elements.
n3g8x562.f	User subroutine *MPC* used in n3g8x562.inp.

3DNLG-9: Large elastic deflection of a pinched hemispherical shell

Products: Abaqus/Standard

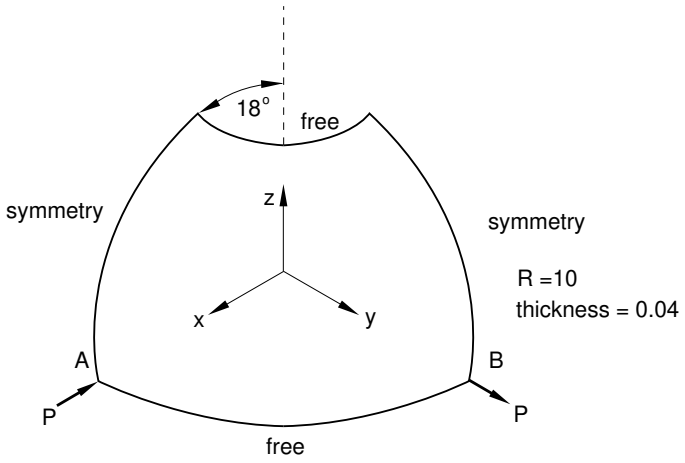
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

CSS8
S3 S3R S4 S4R S4R5 S8R S8R5 S9R5
STRI3 STRI65
SC6R SC8R

Problem description



Material: Young's modulus = 6.825×10^7, Poisson's ratio = 0.3.

Boundary conditions: Symmetry on plane $y = 0$ ($U_y = \theta_x = \theta_z = 0$), symmetry on plane $x = 0$ ($U_x = \theta_y = \theta_z = 0$), $U_z = 0$ at A to prevent rigid body motion.

Loading: Inward and outward diametrical point loads ($P = 100$).

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 3DNLG-9 from NAFEMS Publication R0024 “A Review of Benchmark Problems for Geometric Non-linear Behaviour of 3D Beams and Shells (SUMMARY).”

The published results of this problem were obtained with Abaqus. Thus, a comparison of Abaqus and NAFEMS results is not an independent verification of Abaqus. The NAFEMS study includes results from other sources for comparison that may provide a basis for verification of this problem.

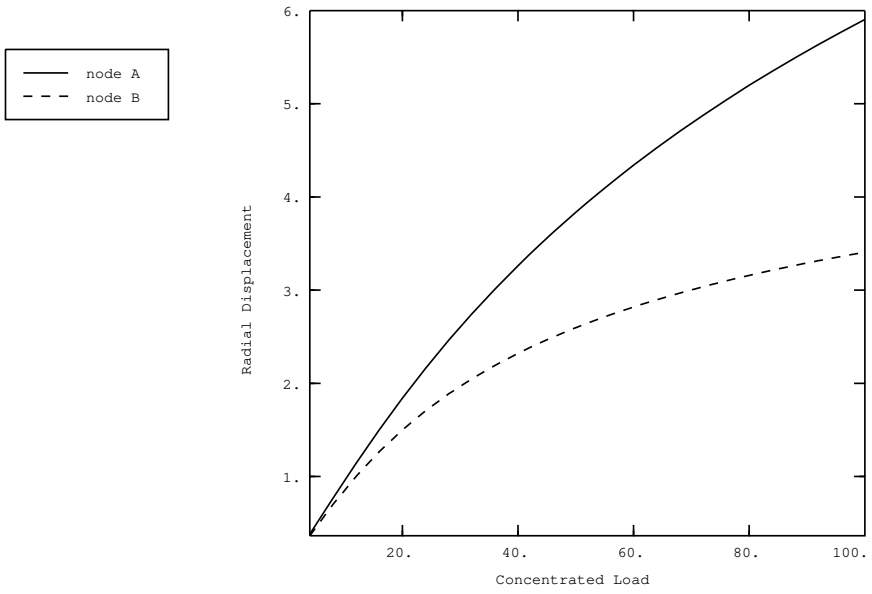
Results and discussion

The following table shows the radial displacements at points A and B at three load levels. All the meshes have the same nodal spacing.

Element	P = 40		P = 60		P = 100	
	$U_r\,(A)$	$U_r\,(B)$	$U_r\,(A)$	$U_r\,(B)$	$U_r\,(A)$	$U_r\,(B)$
S4R	–3.26	2.32	–4.34	2.82	–5.90	3.41
S4	–3.21	2.30	–4.26	2.79	–5.80	3.37
S4R5	–3.28	2.33	–4.36	2.83	–5.95	3.43
S8R	–3.16	2.23	–4.13	2.67	–5.47	3.19
S8R5	–3.23	2.32	–4.30	2.81	–5.83	3.40
S9R5	–3.23	2.32	–4.30	2.81	–5.83	3.40
S3/S3R	–3.12	2.27	–4.15	2.76	–5.67	3.34
STRI3	–3.16	2.29	–4.20	2.79	–5.76	3.37
STRI65	–3.13	2.27	–4.16	2.75	–5.63	3.31
SC6R	–3.06	2.21	–4.03	2.66	–5.42	3.21
SC8R	–3.25	2.30	–4.29	2.78	–5.80	3.35
CSS8	–3.15	2.27	–4.18	2.74	–5.64	3.30

Response predicted by Abaqus

Similar load-displacement curves are obtained for all test cases. The response predicted using S4R elements is shown below. The curve for node A is for the negative displacement and load values.



Input files

n3g9xf3x.inp	S3/S3R elements.
n3g9xe4x.inp	S4 elements.
n3g9xf4x.inp	S4R elements.
n3g9x54x.inp	S4R5 elements.
n3g9x68x.inp	S8R elements.
n3g9x58x.inp	S8R5 elements.
n3g9x59x.inp	S9R5 elements.
n3g9x63x.inp	STRI3 elements.
n3g9x56x.inp	STRI65 elements.
nlg9_std_sc6r.inp	SC6R elements.
nlg9_std_sc6r_sgs.inp	SC6R elements using **SHELL GENERAL SECTION*.
nlg9_std_sc8r.inp	SC8R elements.
nlg9_std_sc8r_stackdir_sphori.inp	SC8R elements with STACK DIRECTION=ORIENTATION and a spherical orientation system.
nlg9_std_css8.inp	CSS8 elements.

3DNLG-10: Elastic-plastic behavior of a stiffened cylindrical panel under compressive end load

Products: Abaqus/Standard

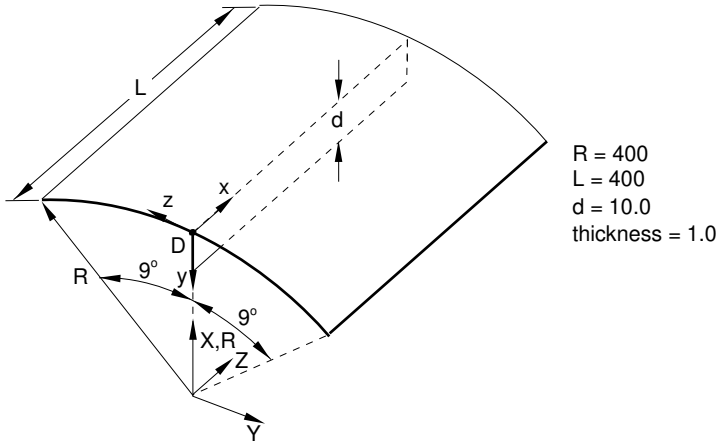
Overview

This problem provides evidence that Abaqus can reproduce the result from the benchmark defined by NAFEMS and cited as the reference solution.

Elements tested

S3 S3R S4 S4R S4R5 S8R S8R5 S9R5
STRI3 STRI65

Problem description



Model: There is an initial imperfection in both the cylindrical panel and the stiffener.

Cylindrical panel: $\Delta R = 0.569 \sin \frac{(\theta \pi^2 Z)}{7200}$, where $-9 \leq \theta \leq 9$ and $0 \leq Z \leq 200$.

Stiffener: $\Delta z = -0.3 \sin \frac{(\pi x)}{400} \sin \frac{(\pi y)}{20}$, where $0 \leq x \leq 200$ and $0 \leq y \leq 10$.

Material: Young's modulus = 2.1×10^5, Poisson's ratio = 0.3, yield stress = 350.

Boundary conditions: Tangential symmetry along edges 1 and 2 at $\theta = \pm 9°$ ($U_\theta = \theta_R = \theta_Z = 0$). Simply supported at end *A* for panel: ($U_R = U_\theta = \theta_R = \theta_Z = 0$), for stiffener: ($U_y = U_z = \theta_x = 0$). Symmetry at $Z = L/2$ for panel: ($U_Z = \theta_R = \theta_\theta = 0$), for stiffener: ($U_x = \theta_y = \theta_z = 0$).

Loading: Compressive, evenly distributed edge load. The RIKS algorithm is used to increment the load until the global z-displacement at point D exceeds 0.2.

Reference solution

This is a test recommended by the National Agency for Finite Element Methods and Standards (U.K.): Test 3DNLG-10 from NAFEMS Publication R0024 “A Review of Benchmark Problems for Geometric Non-linear Behaviour of 3D Beams and Shells (SUMMARY).”

The published results of this problem were obtained with Abaqus. Thus, a comparison of Abaqus and NAFEMS results is not an independent verification of Abaqus. The NAFEMS study includes results from other sources for comparison that may provide a basis for verification of this problem.

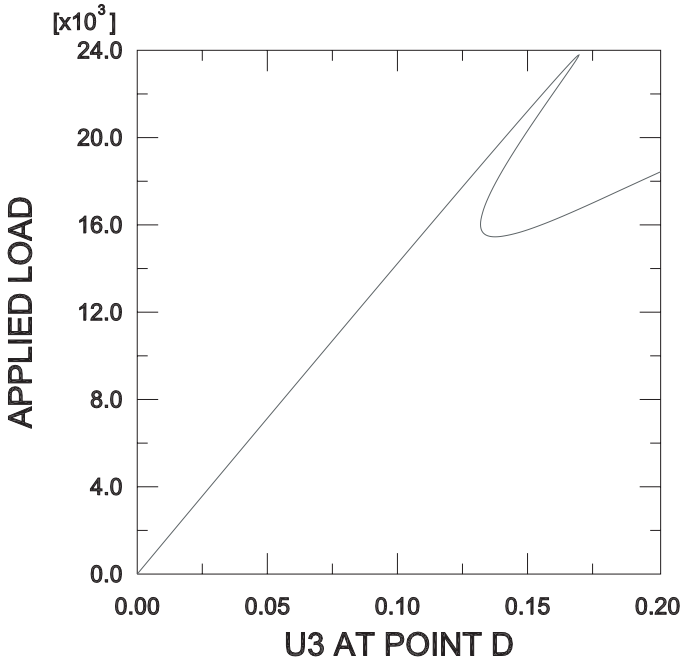
Results and discussion

In the following table, limit points 1 and 2 correspond to the peak and local minimum, respectively, of the load-displacement curve. To produce results comparable to those for S8R5/S9R5, the first-order elements require a finer mesh with half the nodal spacing in the curved direction of the panel. The limit loads for the thick shell element S8R are noticeably higher than others in this thin shell application, even when a fine mesh with the same nodal spacing as that used for the first-order elements is generated.

Element	Limit point 1		Limit point 2	
	End load	U_z at D	End load	U_z at D
S3/S3R	23357	0.167	15622	0.138
S4	23505	0.168	15097	0.135
S4R	23621	0.168	15126	0.136
S4R5	23540	0.168	15074	0.136
S8R	25318	0.180	16964	0.146
S8R5	23764	0.169	15453	0.138
S9R5	23789	0.170	15460	0.138
STRI3	22993	0.165	14809	0.134
STRI65	22534	0.161	15048	0.133

Response predicted by Abaqus

Similar load-displacement curves are obtained for all the test cases. The response predicted using S8R5 elements is shown below.



Input files

n3g10f3x.inp	S3/S3R elements.
n3g10e4x.inp	S4 elements.
n3g10f4x.inp	S4R elements.
n3g1054x.inp	S4R5 elements.
n3g1068x.inp	S8R elements.
n3g1058x.inp	S8R5 elements.
n3g1059x.inp	S9R5 elements.
n3g1063x.inp	STRI3 elements.
n3g1056x.inp	STRI65 elements.

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/Aqua*
- *Abaqus/Design*

Abaqus/Standard

This section lists the topics in this guide that apply to Abaqus/Standard.

Abaqus/Aqua analysis

Near bottom pipeline pull-in and tow

Pull-in of a pipeline lying directly on the seafloor

Slender pipe subject to drag: the “reed in the wind”

Acoustic analysis

A simple coupled acoustic-structural analysis

A simple steady-state dynamic acoustic analysis

Acoustic analysis of a duct with mean flow

Acoustic radiation impedance of a sphere in breathing mode

Acoustic scattering from a rigid sphere

Acoustic scattering from an elastic spherical shell

Acoustic-acoustic tie constraint in three dimensions

Acoustic-acoustic tie constraint in two dimensions

Acoustic-structural interaction in an infinite acoustic medium

Analysis of a point-loaded, fluid-filled, spherical shell

Coupled exterior acoustic eigenanalysis

Real exterior acoustic eigenanalysis

Buckling analysis

Buckling analysis of beams

Buckling of a cylindrical shell under uniform axial pressure

Buckling of a ring in a plane under external pressure

Buckling of a simply supported square plate

Lateral buckling of an L-bracket

Composite tests

R0031(1): Laminated strip under three-point bending

R0031(2): Wrapped thick cylinder under pressure and thermal loading

R0031(3): Three-layer sandwich shell under normal pressure loading

Connector elements

Dynamic response of a two degree of freedom system

Linear behavior of spring and dashpot elements

Continuum elements

Cantilever beam analyzed with CAXA and SAXA elements

Cook's membrane problem

Geometrically nonlinear analysis of a cantilever beam

Torsion of a hollow cylinder

Two-point bending of a pipe due to self weight: CAXA and SAXA elements

Coupled pore fluid flow and stress analysis

Demand wettability of a porous medium: coupled analysis

Desaturation in a column of porous material

Partially saturated flow in a porous medium

Wicking in a partially saturated porous medium

Design sensitivity analysis

Design sensitivity analysis for cantilever beam

Sensitivity analysis of modified NAFEMS problem 3DNLG-1: Large deflection of Z-shaped cantilever under an end load

Sensitivity of the stress concentration factor around a circular hole in a plate under uniaxial tension

Dynamic stress/displacement analysis

Crash simulation of a motor vehicle

Double cantilever elastic beam under point load

Explosively loaded cylindrical panel

Free ring under initial velocity: comparison of rate-independent and rate-dependent plasticity

Large rotation of a one degree of freedom system

Motion of a rigid body in Abaqus/Standard

Pipe whip simulation

Revolute MPC verification: rotation of a crank

Subspace dynamic analysis of a cantilever beam

Truss impact on a rigid wall

Elasticity

Anisotropic hyperelastic modeling of arterial layers

Fitting of elastomeric foam test data

Fitting of rubber test data

Transient thermal loading of a viscoelastic slab

Uniform strain, viscoplastic truss

Viscoelastic rod subjected to constant axial load

Electromagnetic analysis

Eigenvalue analysis of a piezoelectric cube with various electrode configurations

Induction heating of a cylindrical rod by an encircling coil carrying time-harmonic current

Modal dynamic analysis for piezoelectric materials

Steady-state dynamic analysis for piezoelectric materials

TEAM 2: Eddy current simulations of long cylindrical conductors in an oscillating magnetic field

TEAM 4: Eddy current simulation of a conducting brick in a decaying magnetic field

TEAM 6: Eddy current simulations for spherical conductors in an oscillating magnetic field

TEAM 13: Three-dimensional nonlinear magnetostatic analysis

Fracture mechanics

A penny-shaped crack under concentrated forces

Center slant cracked plate under tension

Contour integral evaluation: three-dimensional case

Contour integral evaluation: two-dimensional case

Ct-integral evaluation

Fully plastic J-integral evaluation

Nonuniform crack-face loading and J-integrals

Single-edged notched specimen under a thermal load

Fundamental tests of creep behavior

Test 1A: 2D plane stress – uniaxial load, secondary creep

Test 1B: 2D plane stress – uniaxial displacement, secondary creep

Test 2A: 2D plane stress – biaxial load, secondary creep

Test 2B: 2D plane stress – biaxial displacement, secondary creep

Test 3A: 2D plane stress – biaxial (negative) load, secondary creep

Test 3B: 2D plane stress – biaxial (negative) displacement, secondary creep

Test 4A: 2D plane stress – biaxial (double) load, secondary creep

Test 4B: 2D plane stress – biaxial (double) displacement, secondary creep

Test 4C: 2D plane stress – shear loading, secondary creep

Test 5A: 2D plane strain – biaxial load, secondary creep

Test 5B: 2D plane strain – biaxial displacement, secondary creep

Test 6A: 3D – triaxial load, secondary creep

Test 6B: 3D – triaxial displacement, secondary creep

Test 7: Axisymmetric – pressurized cylinder, secondary creep

Test 8A: 2D plane stress – uniaxial load, primary creep

Test 8B: 2D plane stress – uniaxial displacement, primary creep

Test 8C: 2D plane stress – stepped load, primary creep

Test 9A: 2D plane stress – biaxial load, primary creep

Test 9B: 2D plane stress – biaxial displacement, primary creep

Test 9C: 2D plane stress – biaxial stepped load, primary creep

Test 10A: 2D plane stress – biaxial (negative) load, primary creep

Test 10B: 2D plane stress – biaxial (negative) displacement, primary creep

Test 10C: 2D plane stress – biaxial (negative) stepped load, primary creep

Test 11: 3D – triaxial load, primary creep

Test 12A: 2D plane stress – uniaxial load, primary-secondary creep

Test 12B: 2D plane stress – uniaxial displacement, primary-secondary creep

Test 12C: 2D plane stress – stepped load, primary-secondary creep

Geometric nonlinear tests

3DNLG-1: Elastic large deflection response of a Z-shaped cantilever under an end load

3DNLG-2: Elastic large deflection response of a pear-shaped cylinder under end shortening

3DNLG-3: Elastic lateral buckling of a right angle frame under in-plane end moments

3DNLG-4: Lateral torsional buckling of an elastic cantilever subjected to a transverse end load

3DNLG-5: Large deflection of a curved elastic cantilever under transverse end load

3DNLG-6: Buckling of a flat plate when subjected to in-plane shear

3DNLG-7: Elastic large deflection response of a hinged spherical shell under pressure loading

3DNLG-8: Collapse of a straight pipe segment under pure bending

3DNLG-9: Large elastic deflection of a pinched hemispherical shell

3DNLG-10: Elastic-plastic behavior of a stiffened cylindrical panel under compressive end load

Heat transfer and thermal-stress analysis

Axisymmetric elemental cavity radiation view factor calculations

Convection and diffusion of a temperature pulse

Coupled temperature-displacement analysis: one-dimensional gap conductance and radiation

Freezing of a square solid: the two-dimensional Stefan problem

Quenching of an infinite plate

Radiation analysis of a plane finned surface

Three-dimensional elemental cavity radiation view factor calculations

Two-dimensional elemental cavity radiation view factor calculations

Infinite elements

Infinite elements: circular load on half-space

Infinite elements: the Boussinesq and Flamant problems

Spherical cavity in an infinite medium

Wave propagation in an infinite medium

Mode-based dynamic analysis

Analysis of a cantilever subject to earthquake motion

Double cantilever subjected to multiple base motions

Eigenvalue analysis of a beam under various end constraints and loadings

Eigenvalue analysis of a cantilever plate

Free and forced vibrations with damping

Free vibrations of a spherical shell

Linear analysis of a rod under dynamic loading

Random response of a cantilever subjected to base motion

Random response to jet noise excitation

Residual modes for modal response analysis

Response spectrum analysis of a simply supported beam

Verification of Rayleigh damping options with direct integration and modal superposition

Vibration of a cable under tension

Vibration of a rotating cantilever plate

Modeling discontinuities using XFEM

Crack propagation in a beam under impact loading simulated using XFEM

Crack propagation in a plate with a hole simulated using XFEM

Crack propagation of a single-edge notch simulated using XFEM

Dynamic shear failure of a single-edge notch simulated using XFEM

Propagation of hydraulically driven fracture using XFEM

Plasticity and creep

Biaxial tests on gray cast iron

Creep of a thick cylinder under internal pressure

Cylinder under internal pressure

Indentation of a crushable foam plate

Notched unreinforced concrete beam under 3-point bending

One-way reinforced concrete slab

Simple proportional and nonproportional cyclic tests

Simple tests on a crushable foam specimen

Slider mechanism with slip-rate-dependent friction

Stretching of a plate with a hole

Test of ORNL plasticity theory under biaxial loading

Triaxial tests on a saturated clay

Uniaxial tests on jointed material

Uniformly loaded, elastic-plastic plate

Verification of creep integration

Proposed forced vibration benchmarks

Test 5: Deep simply supported beam: frequency extraction

Test 5H: Deep simply supported beam: harmonic forced vibration

Test 5R: Deep simply supported beam: random forced vibration

Test 5T: Deep simply supported beam: transient forced vibration

Test 13: Simply supported thin square plate: frequency extraction

Test 13H: Simply supported thin square plate: harmonic forced vibration

Test 13R: Simply supported thin square plate: random forced vibration

Test 13T: Simply supported thin square plate: transient forced vibration

Test 21: Simply supported thick square plate: frequency extraction

Test 21H: Simply supported thick square plate: harmonic forced vibration

Test 21R: Simply supported thick square plate: random forced vibration

Test 21T: Simply supported thick square plate: transient forced vibration

Proposed nonlinear benchmarks

NL1: Prescribed biaxial strain history, plane strain

NL2: Axisymmetric thick cylinder

NL3: Hardening with two variables under load control

NL4: Snap-back under displacement control

NL5: Straight cantilever with end moment

NL6: Straight cantilever with axial end point load

NL7: Lee's frame buckling problem

Soils analysis

Consolidation around a cylindrical heat source

Consolidation of a triaxial test specimen

Finite deformation of an elastic-plastic granular material

Finite-strain consolidation of a two-dimensional solid

Limit load calculations with granular materials

The one-dimensional thermal consolidation problem

The Terzaghi consolidation problem

Special-purpose elements

Acoustic modes of an enclosed cavity

Delamination analysis of laminated composites

Standard benchmarks: free vibration tests

FV2: Pin-ended double cross: in-plane vibration

FV4: Cantilever with off-center point masses

FV12: Free thin square plate

FV15: Clamped thin rhombic plate

FV16: Cantilevered thin square plate

FV22: Clamped thick rhombic plate

FV32: Cantilevered tapered membrane

FV41: Free cylinder: axisymmetric vibration

FV42: Thick hollow sphere: uniform radial vibration

FV52: Simply supported “solid” square plate

Standard benchmarks: linear elastic tests

LE1: Plane stress elements—elliptic membrane

LE2: Cylindrical shell bending patch test

LE3: Hemispherical shell with point loads

LE4: Axisymmetric hyperbolic shell under uniform internal pressure

LE5: Z-section cantilever

LE6: Skew plate under normal pressure

LE7: Axisymmetric cylinder/sphere under pressure

LE8: Axisymmetric shell under pressure

LE9: Axisymmetric branched shell under pressure

LE10: Thick plate under pressure

LE11: Solid cylinder/taper/sphere—temperature loading

Standard benchmarks: linear thermo-elastic tests

T1: Plane stress elements—membrane with hot-spot

T2: One-dimensional heat transfer with radiation

T3: One-dimensional transient heat transfer

T4: Two-dimensional heat transfer with convection

Static stress/displacement analysis

Analysis of an anisotropic layered plate

Beam/gap example

Composite shells in cylindrical bending

Concrete slump test

Crushing of a pipe

Necking of a round tensile bar

Pressurized rubber disc

Radial stretching of a cylinder

Snap-through of a shallow, cylindrical roof under a point load

The Hertz contact problem

Thick composite cylinder subjected to internal pressure

Uniaxial stretching of an elastic sheet with a circular hole

Uniform collapse of straight and curved pipe segments

Steady-state transport analysis

Steady-state spinning of a disk in contact with a foundation

Steady-state transport analysis

Structural elements

Analysis of a twisted beam

Performance of continuum and shell elements for linear analysis of bending problems

Ribbon test for shells with applied moments

Shell bending under a tip load

Shell elements subjected to uniform thermal loading

Skew sensitivity of shell elements

The barrel vault roof problem

The pinched cylinder problem

The pinched sphere problem

Tip in-plane shear load on a cantilevered hook

Triangular plate-bending on three point supports

Twisted ribbon test for shells

Variable thickness shells and membranes

Substructures and mass diffusion analysis

Analysis of a frame using substructures

Thermomechanical diffusion of hydrogen in a bending beam

Two-dimensional test cases in linear elastic fracture mechanics

Test 1.1: Center cracked plate in tension

Test 1.2: Center cracked plate with thermal load

Test 2.1: Single edge cracked plate in tension

Test 3: Angle crack embedded in a plate

Test 4: Cracks at a hole in a plate

Test 5: Axisymmetric crack in a bar

Test 6: Compact tension specimen

Test 7.1: T-joint weld attachment

Test 8.1: V-notch specimen in tension

Underwater shock analysis

One-dimensional underwater shock analysis

Response of beam elements to a planar wave

Abaqus/Explicit

This section lists the topics in this guide that apply to Abaqus/Explicit.

Acoustic analysis

Acoustic radiation impedance of a sphere in breathing mode

Acoustic-acoustic tie constraint in three dimensions

Acoustic-acoustic tie constraint in two dimensions

Acoustic-structural interaction in an infinite acoustic medium

Adaptivity analysis

Adaptivity patch test with different materials

Advection in a rotating frame

Indentation with different materials

Propagation of a compaction wave in a shock tube

Water sloshing in a pitching tank

Wave propagation in a shock tube

Wave propagation with different materials

Buckling analysis

Buckling of a column with general contact

Composite tests

R0031(1): Laminated strip under three-point bending

R0031(2): Wrapped thick cylinder under pressure and thermal loading

R0031(3): Three-layer sandwich shell under normal pressure loading

Dynamic stress/displacement analysis

Compression of cylindrical shells with general contact

Explosively loaded cylindrical panel

Free ring under initial velocity: comparison of rate-independent and rate-dependent plasticity

Frictional braking of a rotating rigid body

Impact of a copper rod

Oblique shock reflections

Pipe whip simulation

Plate penetration by a projectile

Rigid body dynamics with Abaqus/Explicit

Steady-state slip of a belt drive

Truss impact on a rigid wall

Elasticity

Anisotropic hyperelastic modeling of arterial layers

Rubber under uniaxial tension

Transient thermal loading of a viscoelastic slab

Eulerian analysis

Deflection of an elastic dam under water pressure

Eulerian analysis of a collapsing water column

Heat transfer and thermal-stress analysis

Coupled temperature-displacement analysis: one-dimensional gap conductance and radiation

Freezing of a square solid: the two-dimensional Stefan problem

Quenching of an infinite plate

Infinite elements

Wave propagation in an infinite medium

Plasticity and creep

Biaxial tests on gray cast iron

Indentation of a crushable foam plate

Mixed-mode failure of a notched unreinforced concrete beam

Notched unreinforced concrete beam under 3-point bending

One-way reinforced concrete slab

Pressure on infinite geostatic medium

Pressurization of a thick-walled cylinder

Simple tests on a crushable foam specimen

Stretching of a plate with a hole

Proposed forced vibration benchmarks

Test 5T: Deep simply supported beam: transient forced vibration

Test 13T: Simply supported thin square plate: transient forced vibration

Test 21T: Simply supported thick square plate: transient forced vibration

Soils analysis

Finite deformation of an elastic-plastic granular material

Limit load calculations with granular materials

Special-purpose elements

Delamination analysis of laminated composites

Fluid filled rubber bladders

Standard benchmarks: linear elastic tests

LE1: Plane stress elements—elliptic membrane

LE2: Cylindrical shell bending patch test

LE3: Hemispherical shell with point loads

LE5: Z-section cantilever

LE6: Skew plate under normal pressure

LE7: Axisymmetric cylinder/sphere under pressure

LE8: Axisymmetric shell under pressure

LE10: Thick plate under pressure

Standard benchmarks: linear thermo-elastic tests

T2: One-dimensional heat transfer with radiation

T3: One-dimensional transient heat transfer

T4: Two-dimensional heat transfer with convection

Static stress/displacement analysis

Composite shells in cylindrical bending

Concrete slump test

Necking of a round tensile bar

Pressurized rubber disc

The Hertz contact problem

Structural elements

Analysis of a twisted beam

Performance of continuum and shell elements for linear analysis of bending problems

Shell bending under a tip load

Simulation of propeller rotation

Skew sensitivity of shell elements

The barrel vault roof problem

The pinched cylinder problem

The pinched sphere problem

Transient response of a shallow spherical cap

Variable thickness shells and membranes

Underwater shock analysis

Air-backed coupled plate response to a planar exponentially decaying wave

Coupled cylindrical shell response to a planar step wave

Coupled spherical shell response to a planar step wave

Cylindrical shell response to a planar exponentially decaying shock wave

Cylindrical shell response to a planar step shock wave

Fluid-filled spherical shell response to a planar step wave

One-dimensional underwater shock analysis

Plate response to a planar exponentially decaying shock wave

Response of beam elements to a planar wave

Spherical shell response to a planar exponentially decaying wave

Spherical shell response to a planar step wave

Spherical shell response to a spherical exponentially decaying wave

The one-dimensional cavitation problem

The submerged infinite cylinder problem

The submerged sphere problem

Water-backed coupled plate response to a planar exponentially decaying wave

Abaqus/CAE

This section lists the topics in this guide that apply to Abaqus/CAE.

Eulerian analysis

Deflection of an elastic dam under water pressure

Eulerian analysis of a collapsing water column

Modeling discontinuities using XFEM

Crack propagation in a plate with a hole simulated using XFEM

Crack propagation of a single-edge notch simulated using XFEM

Abaqus/Aqua

This section lists the topics in this guide that apply to Abaqus/Aqua.

Abaqus/Aqua analysis

Near bottom pipeline pull-in and tow

Pull-in of a pipeline lying directly on the seafloor

Slender pipe subject to drag: the “reed in the wind”

Abaqus/Design

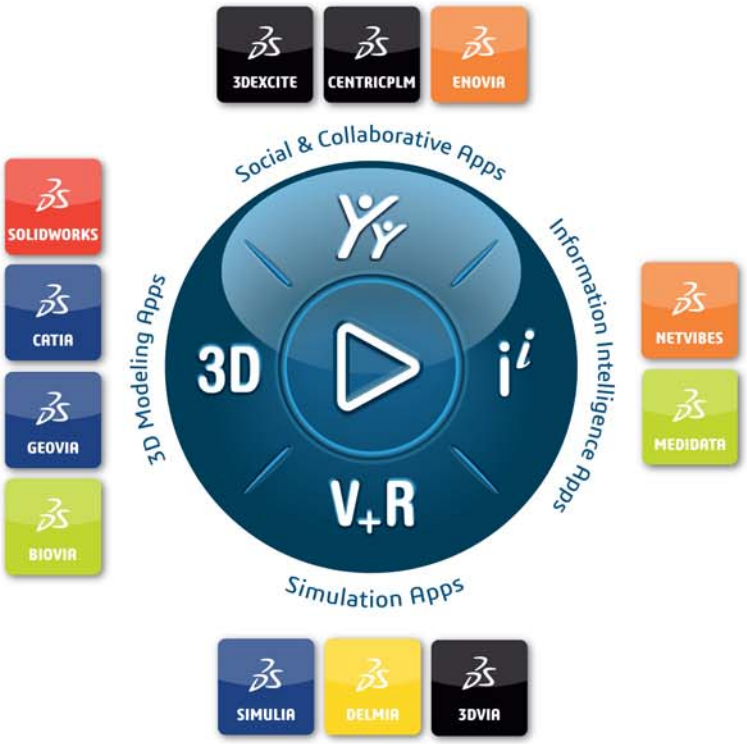
This section lists the topics in this guide that apply to Abaqus/Design.

Design sensitivity analysis

Design sensitivity analysis for cantilever beam

Sensitivity analysis of modified NAFEMS problem 3DNLG-1: Large deflection of Z-shaped cantilever under an end load

Sensitivity of the stress concentration factor around a circular hole in a plate under uniaxial tension



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.