Nearly incompressible elasticity — Cook's membrane

Table of contents

Unreleased Alpha Functionality

This example uses unreleased functionality intended for evaluation by advanced users. The formulation, activation workflow, and results may change without notice. Validate results against a reference solution or a conventional Abaqus discretization before using them for engineering decisions.

1 Overview

Nearly incompressible materials can cause volumetric locking in a displacement-only formulation, making the computed structure artificially stiff. Coreform's locking-free formulation introduces an independent pressure field for the volumetric response, while displacement represents the distortional response. The volume constraint is enforced in a weak, averaged sense rather than point by point.

This example applies the formulation to a quadratic, body-fitted Cook's membrane model undergoing large deformation. It uses a nearly incompressible material and Coreform's compressible Neo-Hookean material model.

Figure 1 shows a plan view of the trapezoidal geometry and applied boundary conditions. The left face is clamped, out-of-plane displacement is fixed throughout the body, and an upward resultant shear load is applied to the right face. The upper loaded corner is the displacement-probe location; the out-of-plane thickness t listed below is not shown. All dimensional values use the MMTS consistent unit system.

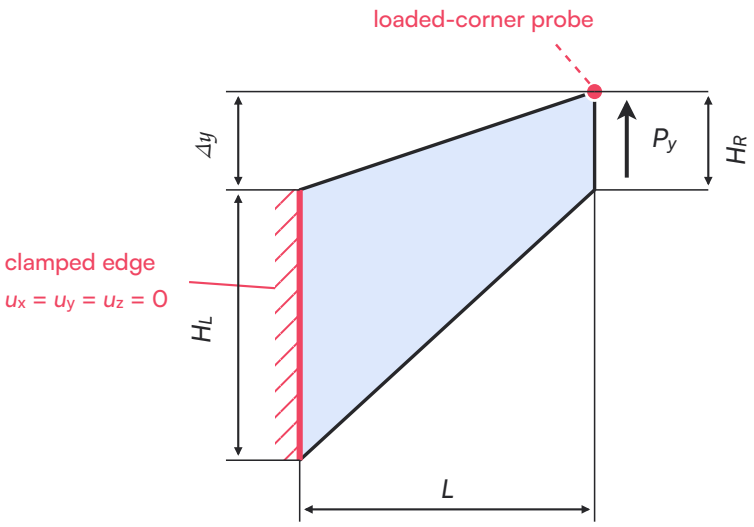

Figure 1: Plan view of Cook's membrane geometry and boundary conditions.

Property	Value
Horizontal span, L	$48\ mm$
Left height, H_L	$44\ mm$
Right height, H_R	$16\ mm$
Upper-edge vertical offset, Δy	$16\ mm$
Extruded thickness, t	$1.91516\ mm$
Young's modulus, E	$240.565\ MPa$
Poisson's ratio, ν	0.49999
Applied resultant shear load, P_y	$6.25\ N$
Geometric nonlinearity	On

1.1 Performance and limitations

The locking-free formulation is substantially slower than the regular formulation because the pressure field adds degrees of freedom to the global system. It also requires additional coupled residual and tangent assembly and uses unsymmetric matrix storage, increasing both solution time and memory use.

This example demonstrates unreleased alpha functionality, not a generally supported production workflow.

2 Formulation

By default, each integration element receives one element-local pressure degree of freedom. The pressure field supplies the hydrostatic stress and relieves the displacement field from the constraint that causes volumetric locking. Stabilization controls nonphysical pressure modes.

The implementation uses the stabilized mixed displacement-pressure method described by O. Klaas, A. Maniatty, and M. S. Shephard, "A stabilized mixed finite element method for finite elasticity: Formulation for linear displacement and pressure interpolation," *Computer Methods in Applied Mechanics and Engineering*, vol. 180, issues 1—2, pp. 65—79, 1999. doi:10.1016/S0045-7825(99)00059-6

2.1 Material behavior

The material implementation depends on the Abaqus geometric-nonlinearity setting:

Abaqus setting	Material behavior
`NLGEOM=OFF`	Uses the Abaqus material-model path through UELMAT with small-strain behavior.
`NLGEOM=ON`	Uses Coreform's in-house compressible Neo-Hookean model through UEL.

If any model step has `NLGEOM=ON`, the Coreform Neo-Hookean backend is selected. Enabling `NLGEOM` therefore changes both the kinematics and the material implementation used by this alpha feature.

3 Activate the locking-free formulation

Set `USE_LOCKING_FREE=1` before starting Coreform IGA for Abaqus, then create or regenerate the Coreform IGA job. The environment variable is read during export; setting it only when submitting an existing input deck is too late.

On Linux:

```
export USE_LOCKING_FREE=1
/opt/Coreform-IGA-2026.7/bin/coreform_iga
```

On Windows Command Prompt:

```
set USE_LOCKING_FREE=1
"C:\Program Files\Coreform IGA 2026.7\coreform_iga.bat"
```

Set `USE_LOCKING_FREE=0`, or remove the environment variable, to return to the regular formulation.

4 Download and run the example

Download the Abaqus/CAE journal file below to reconstruct the model and generate the input artifacts used by this example.

Run the journal from a new, writable directory so its output files are easy to identify. Set `USE_LOCKING_FREE=1` before starting Abaqus/CAE, as described above, and use Abaqus/CAE's `recover` option rather than File > Run Script. Abaqus recommends `recover` for `.jnl` files; running a journal as an ordinary script can produce an incomplete model database.

On Linux, the Coreform launcher changes to `RUNTIME_DIR` before starting Abaqus/CAE, so set it to the directory containing the downloaded journal:

```
mkdir -p cooks-membrane
cd cooks-membrane
# Save cooks_membrane.jnl in this directory before continuing.
export USE_LOCKING_FREE=1
export RUNTIME_DIR="$PWD"
/opt/Coreform-IGA-2026.7/bin/coreform_iga recover="$PWD/cooks_membrane.jnl"
```

On Windows Command Prompt:

```
mkdir cooks-membrane
cd /d cooks-membrane
rem Save cooks_membrane.jnl in this directory before continuing.
set USE_LOCKING_FREE=1
"C:\Program Files\Coreform IGA 2026.7\coreform_iga.bat" recover="%CD%
\cooks_membrane.jnl"
```

Abaqus/CAE executes the journal during startup. Wait for model recovery and the Coreform exports to finish before closing Abaqus/CAE. The journal produces the following files in the working directory:

File	Contents
`cooks_membrane.cae`	Reconstructed Abaqus/CAE model database, including the Coreform IGA mesh and job definitions.
`cooks_membrane.cf`	Coreform model export consumed by Coreform IGA Mesh.
`cooks_membrane.inp`	Abaqus simulation input exported from the CAE model for the subsequent interop step.
`cooks_membrane.jnl`	Abaqus/CAE recovery journal associated with the saved model database.

File	Contents
`abaqus.rpy`	Abaqus/CAE session replay log. Abaqus may add a numeric suffix if a replay log already exists.

These are model-generation artifacts, not analysis results.

4.1 Run Abaqus/Standard

Keep the recovered model open in the same Abaqus/CAE session and complete the analysis as follows:

1. Switch to the Job module.
2. Click the Coreform IGA Job Manager icon to open the Coreform IGA for Abaqus - Job Manager.
3. Select the `cooks_membrane` job and click Submit. Use the Coreform job manager, not the standard Abaqus job manager. The Coreform Submit action rebuilds the IGA mesh, runs the Abaqus interop translation, and then submits the translated `cooks_membrane.inp` deck to Abaqus/Standard. It is not necessary to click Build Mesh first.
4. Leave Abaqus/CAE running until the job status changes to Completed. A successful run prints messages similar to the following in the Abaqus/CAE message area:

```
Job cooks_membrane: Analysis Input File Processor completed successfully.
Job cooks_membrane: Abaqus/Standard completed successfully.
Job cooks_membrane completed successfully.
```

5. With `cooks_membrane` still selected in the Coreform job manager, click Results to open `cooks_membrane_iga.odb` in the Visualization module. The standard Abaqus output database, `cooks_membrane.odb`, can also be opened with File > Open.

The completed workflow produces these primary result artifacts:

File	Contents
`cooks_membrane.odb`	Standard Abaqus output database.
`cooks_membrane_iga.odb`	IGA results mapped to the visualization mesh.
`probe_data.json`	Probe histories, including the loaded-corner displacement plotted below.

The working directory also contains the Coreform mesh database and translated input-deck files, as well as the usual Abaqus/Standard status and diagnostic files such as `.dat`, `.msg`, and `.sta`.

5 Expected result

The vertical displacement at the loaded upper corner is $u_y = 6.95081$ mm.

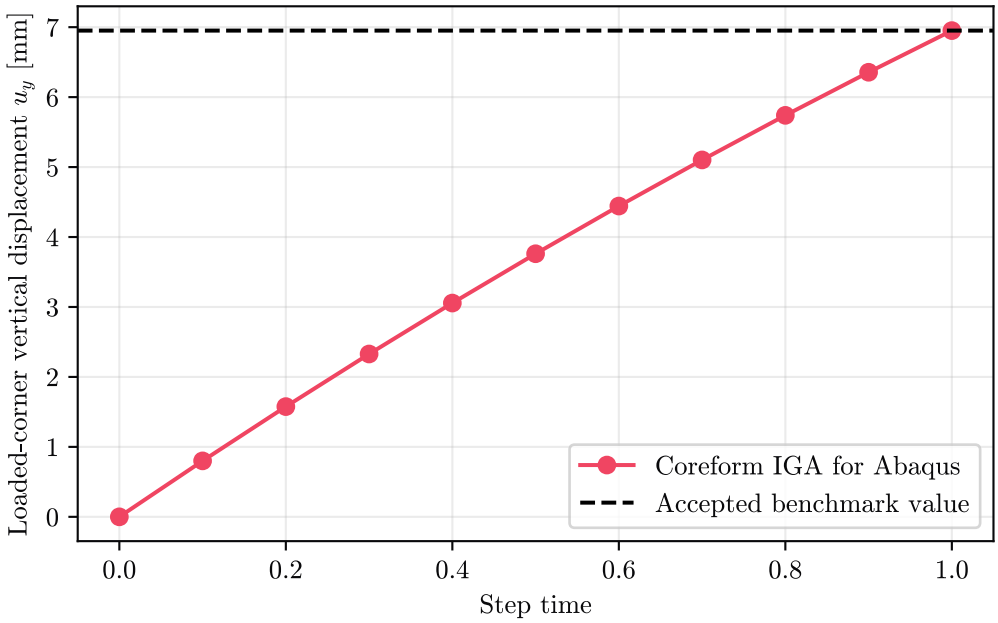

Figure 2: Loaded-corner vertical displacement over the nonlinear load step.