

# Overview

## Overview

*January 27, 2010*

### Overview of Goals and Objectives

In order to test the accuracy and speed of various elastic and viscoelastic finite element calculations using different codes on different platforms, we have developed the following benchmark comparisons. Information resulting from the benchmark comparisons will be used for the following purposes.

1. Confirming proper numerical implementation of the physics including rheological laws, fault constitutive laws, etc.
2. Testing the accuracy of the numerical implementations as a function of meshing scheme, number of nodes, element type, time-stepping scheme, code, etc.
3. Testing the computational efficiency of different codes, solvers, and modeling techniques as a function of meshing scheme, number of nodes, element type, time-stepping scheme, code, etc.
4. Comparing and evaluating available finite element codes.

Based on the comparisons, we would like to be able to 1) identify and correct any errors in numerical implementation which currently exist in any of the considered codes, 2) quantify differences in numerical solutions as a function of meshing scheme, number of nodes, element type, time-stepping scheme, code, etc., and 3) quantify and, if possible, minimize model induced uncertainties resulting from discretization, model boundaries, unexpected transients in time-dependent materials, etc.

### General Methodology

All benchmark descriptions assume a right-handed Cartesian coordinate system with the x-direction running east, the y-direction running north, and the z-direction running up. If a boundary condition is applied at a depth,  $d$ , this will correspond to  $z = -d$ . The surface is always assumed to be at  $z = 0$ . Use whatever coordinate system is most convenient for your program but please convert the results to the one defined here.

Benchmark meshes will be described at the coarsest level to be run. Because meshes may be either structured or unstructured, the mesh nodal spacing described refers to the average. If memory, time, and patience allow, also run models at 1/2, 1/4, 1/8, etc. the original coarse node spacing. This will make it possible to see how accuracy and speed scale with mesh spacing. If your code permits a variety of element types, also run models using various types of elements (linear vs. quadrilateral; hexahedral vs. tetrahedral, full vs. reduced integration). This will make it possible to see how accuracy and speed change with element type. Finally, variable mesh spacing degrades accuracy, but, for economy, we would like to employ a variable mesh (e.g., to resolve stress variations at the fault tips,

## OVERVIEW

---

etc.). If time permits investigate the trade-offs involved in using variable resolution meshes.

With regard to output, there are a number of parameters which should be noted for each model. For the purposes of determining accuracy, please record displacements at all nodes and stresses (all 6 independent components) at all Gauss points at each specified time. For the purposes of evaluating performance, please try to keep track of memory usage (including the size of stiffness matrix and mean execution time, etc. More details regarding the submission of your results for inclusion in the summary analysis can be found at the end of this document. To ease the burden on those who are compiling the data, your results will not be accepted unless they are in the specified format.

When available, analytical solutions for the various benchmarks will be given on the CIG web site at <http://XXXXXXXXXXXXX>. Whenever possible, please check to make sure your results are essentially correct before submitting them for the summary analysis.