

A proposal to the CIG for support towards the development of a crustal deformation community software suite

Prepared by Brad Hager, Mark Simons, Charles Williams, and Carl Gable
November 5, 2004

Introduction and Motivation

Over the past few years we have implemented a multi-year program to coordinate development and validation of 3D quasi-static, finite-element codes for modeling crustal deformation; develop deformation models with observed topography, fault geometries, rheological properties, geologic slip rates, geodetic motions, and earthquake histories; and use these models to infer fault slip, rheologic structure, and fault interactions through stress transfer. In order to leverage funding and take advantage of ongoing work, this effort is carried out in coordination with SERVO, GeoFEM, ACES, and other modeling efforts. We were given seed funding to begin to produce a community-modeling package that is engineered with software evolution and growth as design requirements, developing code able to run on multiple platforms, including Beowulf PC clusters and GRID-computing systems. An ultimate goal is to simulate deformation spanning many earthquake cycles over the past ~ 300,000 yrs to compare to geologic determinations of earthquake clustering and statistical studies of seismicity. This final step will include development of an adjoint model and data assimilation techniques.

This effort should provide the computational technology needed for efficient exploration of fault system dynamics. This technology is being designed with input from the SCEC Crustal Deformation Modeling (CDM) community, as well as other groups. (For more information, see <http://geoweb.mit.edu/fe/>.) Our goal is software that is well-documented, user-friendly, open source, modular, easily modified, and implemented for use on parallel computers (e.g., PC clusters, GRID-computing) in order that it be possible to carry out large computations efficiently and in order that it not be necessary for a scientist to have a dedicated machine to run it.

Strategy

A key component of our strategy for community building, as well as building software, is a series of workshops to involve deeply the SCEC CDM community in this process. The first CDM workshop, held at Caltech in June 2002, focused on assessing the accuracy, speed, and ability to modify software in use by members of the community. At this workshop the CDM Group developed its Mission Statement: 1) Build tools to understand the response to single earthquakes, and make geodetic comparisons, infer rheology, and constrain structures; 2) Build tools to simulate fault system interaction, regional strain and stress field evolution and produce results that assist in the estimation or modeling of fault slip and constrain physics; 3) Develop understanding of transient stress interaction among faults; and 4) Determine realistic predictions of geologic features (e.g., topography, fault slip).

The second annual workshop, in August 2003, was expanded in both length and participation. Los Alamos National Laboratory hosted the workshop with the locale chosen to enable SCEC scientists to benefit from attendance by Lab experts, particularly those with expertise in meshing. By leveraging SCEC, NASA, and LANL support, we were able to increase the number of students and senior researchers attending, as well as meet for a longer time. Because members of the NASA-sponsored SERVO Quakesim group participated in the workshop, there was significant interchange of ideas and software. One of the group objectives is verifying code accuracy, so significant effort was spent refining the preliminary benchmark problems. A highlight of the workshop was intense discussion of Computational Frameworks.

The highest priority of the CDM subgroup was to develop a quasi-static, parallelized finite element code that will eventually be able to represent the deformation and stress fields due to all major faults in southern California as provided by the Community Block Model, using realistic rheologies and fault behavior.

The third annual workshop was also held at LANL in August 2004, this time with support provided by NSF, NASA, SCEC, and LANL. The attendance at this workshop was nearly double that of previous years. At this workshop, we continued development of the community block models, providing at least a tetrahedral mesh of this model. We have tried to balance the need for broad community involvement in specifying code requirements with the need for highly focused work to produce a product of value to all. Our judgment is that the most effective strategy is to have a small group responsible for mesh development and beta testing, then to build community expertise and obtain feedback via a workshop. A significant difference in the 2004 workshop was the emphasis on training new users in the use of the available software (GeoFEM and Tecton/Pyre). By the end of the workshop everybody had downloaded, installed, compiled, and run one of these codes on their private machines. It is only by this kind of hands-on training, that we can increase the size of the user community in preparation of the EarthScope data stream as well as to provide more voices contributing to the design goals of the crustal dynamics community code development effort.

In tandem with all these activities, Charles Williams (RPI) leveraged SCEC, NSF/ITR, and Caltech resources to initiate the upgrade of his FEM code "Tecton" into a SCEC Community code. He has accomplished the initial phase of integration of the code into the Pyre framework, which immediately adds several new capabilities to the code, while easing the process of adding new features. The initial version of the code is now available as a dynamic shared library, callable via python (Pyre) function calls. A number of new element types have been added, including tetrahedral and transition elements, as well as quadratic versions of all elements. Work is progressing to integrate Tecton and "eqsim," Brad Aagaard's source physics and strong ground motion code, via the Pyre Framework. In addition, the Quakesim code "GeoFEST" (JPL) was impressive, with a parallel beta version about to be released. A stronger integration of this project with the SCEC effort was initiated, so we hope to soon have two parallel FEM solvers available.

Our current design issues include:

- 1) A computational mesh that includes realistic surfaces (faults, with fault slip either imposed or determined using rheological laws such as rate-state friction) and 3-D material variations (elastic moduli, viscoelastic parameters, and poroelasticity). These material parameters must conform to estimates of geologic structure and be easily modified to include improved estimates. Tetrahedral elements provide the natural match approach.
- 2) Accurate determination of the stress distribution associated with coseismic and postseismic deformation – Coulomb stress change, postseismic deformation with rate-state friction and nonlinear rheologies are all crucially dependent on accurate determination of stress. The geophysical importance of accurate estimates of stress seems to require higher-order elements, in which displacements are highly accurate and stress is continuous.
- 3) We must be able to run very large, highly resolved models for long periods of time. In particular, stress interaction studies and inferences from post-seismic observations rely on internally consistent states of stress. This internal consistency requires us to "spin up" a given model for many earthquake cycles before analyzing the system behavior. In addition, "data assimilation," or solving the adjoint problem (to address uncertain initial conditions), requires the ability to run a large number of models. These requirements argue for parallel code that can be run on multi-CPU machines such as Beowulf PC clusters or GRID infrastructure.

What we would like CIG to provide

Based on our ongoing activities, we have several immediate needs. These include:

1. Continued support for the annual workshop - perhaps coordinated with SCEC.
2. Archive/repository support for benchmarking activities: a web site, web management, easy I/O and collation of results. This is a general need for CIG.
3. Reworking of the parameter and field value I/O using modern interfaces including web forms for input and 3D visualization tools for the output. This is a general need for CIG.
4. Support for parallelizing Tecton using PETSc (a collaboration between Matt Knepley and Charles Williams).
5. Implementation of different fault friction laws and non-linear visco-elasto-plastic bulk rheologies.
6. Introduction of fully-coupled poroelasticity.
7. Addition of a diffusive/advective temperature solver. This is a general need for CIG.
8. Attacking the remeshing problem for long term evolution and/or spin up problem.
9. Testing using different compilers on different platforms. This is a general need for CIG.

Since CIG is new, it is not entirely clear what should be done under CIG as opposed to other avenues. We will rework these requests based on feedback from the CIG executive committee.