Using the GeoFEST faulted region simulation system

Jay Parker,[1] Greg Lyzenga[2], Andrea Donnellan[3], Michelle Judd[4], Teresa Baker[5], Charles Norton[6], Edwin Tisdale[7] and Peggy Li[8]

(1) Jet Propulsion Laboratory/California Institute of Technology, 4800 Oak Grove Drive, Pasadena, CA 91109, USA (e-mail: Jay.W.Parker@jpl.nasa.gov; phone: +1 818-354-6790). (2) Jet Propulsion Laboratory/California Institute of Technology, 4800 Oak Grove Drive, Pasadena, CA 91109, USA (e-mail: Gregory.A.Lyzenga@jpl.nasa.gov; phone: +1 818-354-6920). (3) Jet Propulsion Laboratory/California Institute of Technology, 4800 Oak Grove Drive, Pasadena, CA 91109, USA (e-mail: Andrea.Donnellan@jpl.nasa.gov; phone: +1 818-354-4737). (4) Jet Propulsion Laboratory/California Institute of Technology, 4800 Oak Grove Drive, Pasadena, CA 91109, USA (e-mail: Michele.A.Judd@jpl.nasa.gov; phone: +1 818-354-4994). (5) Massachusetts Institute of Technology, 77 Massachusetts Ave., Cambridge, MA 02139, USA, (e-mail: teresab@alum.mit.edu; phone: +1 818-484-2884). (6) Jet Propulsion Laboratory/California Institute of Technology, 4800 Oak Grove Drive, Pasadena, CA 91109, USA (e-mail: Charles.D.Norton@jpl.nasa.gov; phone: +1 818-393-3920). (7) Jet Propulsion Laboratory/California Institute of Technology, 4800 Oak Grove Drive, Pasadena, CA 91109, USA (e-mail: Edwin.R.Tisdale@jpl.nasa.gov; phone: +1 818-354-1295). (8) Jet Propulsion Laboratory/California Institute of Technology, 4800 Oak Grove Drive, Pasadena, CA 91109, USA (e-mail: P.P.Li@jpl.nasa.gov; phone: +1 818-354-1341).

Abstract

GeoFEST (the Geophyical Finite Element Simulation Tool) simulates stress evolution, fault slip and plastic/elastic processes in realistic materials, and so is suitable for earthquake cycle studies in regions such as Southern California. Many new capabilities and means of access for GeoFEST are now supported. New abilities include MPI-based cluster parallel computing using automatic PYRAMID/Parmetis-based mesh partitioning, automatic mesh generation for layered media with rectangular faults, and results visualization that is integrated with remote sensing data. The parallel GeoFEST application has been successfully run on over a half-dozen computers, including Intel Xeon clusters, Itanium II and Altix machines, and the Apple G5 cluster. It is not separately optimized for different machines, but relies on good domain partitioning for load-balance and low communication, and careful writing of the parallel diagonally preconditioned conjugate gradient solver to keep communication overhead low. Demonstrated thousand-step solutions for over a million finite elements on 64 processors require under three hours, and scaling tests show high efficiency when using more than (order of) 4000 elements per processor. The source code and documentation for GeoFEST is available at no cost from Open Channel Foundation. In addition GeoFEST may be used through a browser-based portal environment available to approved users. That environment includes semi-automated geometry creation and mesh generation tools, GeoFEST, and RIVA-based visualization tools that include the ability to generate a flyover animation showing deformations and topography. Work is in progress to support simulation of a region with several faults using 16 million elements, using a strain energy metric to adapt the mesh to faithfully represent the solution in a region of widely varying strain.

Introduction

Modeling the deformations due to the earthquake cycle in tectonically active regions requires considerable flexibility in the tools for specifying the geometry of faults, surfaces, and materials, as well as in the numerical solutions. Realistic simulations must allow for complex material modeling that includes systems of faults, layers of distinct materials, special lumpy volumes such as basins and mountains, and appropriate boundary conditions set by the tectonic motions bounding the region of study. The resulting deformation field will have regions of very high and very low strain, respectively near fault interaction zones and within near-rigid blocks or plates. These characteristics motivated a sustained effort to improve usability and performance of GeoFEST (an acronym for Geophysical Finite Element Simulation Tool), which employs the accuracy and flexibility of stress-strain finite elements in an environment tailored to setting up and solving the mechanical evolution of faulted crustal regions.

GeoFEST equations and features

GeoFEST supports elastic and viscoelastic simulation in regions with faults that slip in instantaneous events, and are driven by boundary displacements and forces. It solves the relevant physical relations using unstructured three-dimensional meshes, particularly with linear tetrahedral whose size may vary over more than an order of magnitude over the model domain. It is appropriate to use GeoFEST to study immediate and long-term deformations due to earthquakes, tectonic loading, and material relaxation. It can be used to test hypothetical ideas about fault physics, fault interaction from stress transfer, and scenario exploration with regards tectonic driving forces and material properties. A novel application under consideration is the numeric calculation of fault coupling green's functions that may be used in a very long-term fault interaction calculation.

Basic equations

In common with other finite element structural analysis programs, GeoFEST produces discrete numerical solutions to the continuum equations of elastostatics (e.g., [1]) and small-displacement, quasi-static viscoelasticity. The first of these basic physical relations can be briefly summarized in the tensor equation for elastostatic equilibrium,

$$\sigma_{ij,j} + f_i = 0 \qquad (1)$$

where f_i is the externally imposed volumetric body force and σ_{ij} is the Cauchy stress tensor, which for isotropic elastic media is linearly related to the strain tensor ε_{ij} through the elastic moduli λ and μ:

$$\sigma_{ij} = c_{ijkl}\varepsilon_{kl} \qquad (2)$$

where

$$c_{ijkl} = \mu\left(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk}\right) + \lambda\delta_{ij}\delta_{kl} \qquad (3)$$

Quasi-static viscoelasticity is implemented through a succession of time-domain updates of the stress field, which relaxes according to a viscoplastic effective strain prescription (as described by [2]):

$$\frac{\partial\sigma_{ij}}{\partial t} = c_{ijkl}\left(\frac{\partial\varepsilon_{kl}}{\partial t} - \frac{\partial\varepsilon_{kl}^{VP}}{\partial t}\right) \qquad (4)$$

where

$$\frac{\partial\varepsilon_{kl}^{VP}}{\partial t} = \beta_{ij}\left(\sigma_{ij}\right) \qquad (5)$$

In this expression, β is a specified functional relationship between stress and viscoplastic strain rate, conventionally taken as linear for the case of Newtonian viscoelasticity. The unconditionally stable implicit scheme described by [2] for stepping this relation forward in time is employed by GeoFEST, and represents its basis for time dependent simulation of deformation processes in which inertia is not important (that is, excluding such problems as wave propagation and dynamic rupture).

Slip on faults is supported through the split node formulation of [3]. Our implementation is described in the GeoFEST User's Guide, available at [4]. Briefly, a fault is a collection of nodes on a surface, and the nodes are designated "split" and given a slip magnitude, schedule, and the slip and fault orientation. This is sufficient to define the effects of slip at each node, which is implemented as an equivalent force on all adjacent nodes. The result is the correct stress and displacement fields in the neighborhood of the split node, exactly as if the node had split in two parts going opposite ways.

Materials and boundary conditions

In principal, every finite element could be assigned a different material; in practice, adjacent groups of elements are assigned a material according to the user's understanding of geophysical features such as layers or inclusions. As reflected in the equations above, isotropic elastic, Maxwell viscoelastic, and power-law nonlinear viscoelastic materials are supported. Boundary conditions of displacements, velocity, or force are implemented as component constraints at specified nodes.

Support for ease of design and use

GeoFEST has been integrated into a web portal environment at [5]. This portal includes tools for designing a full GeoFEST analysis. Fault geometry may be imported from the QuakeTables database, and edited to define the geometry of a layered model. A mesh generator is included that allows automated concentration of the unstructured mesh elements near faults. Boundary conditions may be simply specified on each model surface. The investigator may use the portal to initiate GeoFEST simulations, view simple images of the results, and download the simulation products for further analysis and visualization.

The entire GeoFEST source code is available for download from [6]. The download materials include a sequential version that is directly built with any common C compiler, and a parallel version that has been tested on many types of servers and supercomputers. There are instructions included for gathering additional freely available libraries that must be linked to form the MPI-based parallel executable, such as PYRAMID [7]. The user may opt to compile and run GeoFEST in this way, entirely apart from the portal environment. Users may generate and download mesh files from the portal, or write translator scripts to import files from other meshing applications. To assist such efforts, the input format for GeoFEST is fully documented in [4] and interactive hypertext examples [8].

Cluster computing performance

A test case 100x100x60 km with 3 faults representing the Landers M7.2 1992 earthquake slip and 500 years of viscoelastic relaxation is used to measure scaled performance. Three mesh densities are employed, resulting in problems with 82 thousand, 350 thousand, and 1.4 million tetrahedral finite elements. It is found that for one elastic fault-slip step followed by 1000 half-year time steps with 8 specified times of output of the full model displacement, that I/O was inconsequential, and that run time is dominated by the iterative solution of the finite element stiffness equations at each step. Each iteration employs global dot products and an exchange of shared boundary information between each adjacent pair of processors, where adjacency is in the sense of touching finite element blocks initially established by Pyramid's domain decomposition. This inner loop communication proved challenging in maintaining efficient scaling as more processors are employed, but a dot-product gather and orderly pair-wise exchange method resulted in the excellent scaling shown in Figure 1. The three curved in this plot indicate scaling in terms of operations/sec vs. number of processors employed, for the three problem sizes. The small case speeds up nearly linearly from 4 to 16 processors, but has insufficient work for efficient operation on 64 processors. The medium case matches the operations rate on 16 processors, and scales near-linearly to 64. The large case matches the operations rate of the medium case on 64 processors. This plot suggests efficiency and scaling are excellent so long as each processor has at least 4000 finite elements, and we expect this performance to continue as we attempt a 16 million simulation on 500 processors.

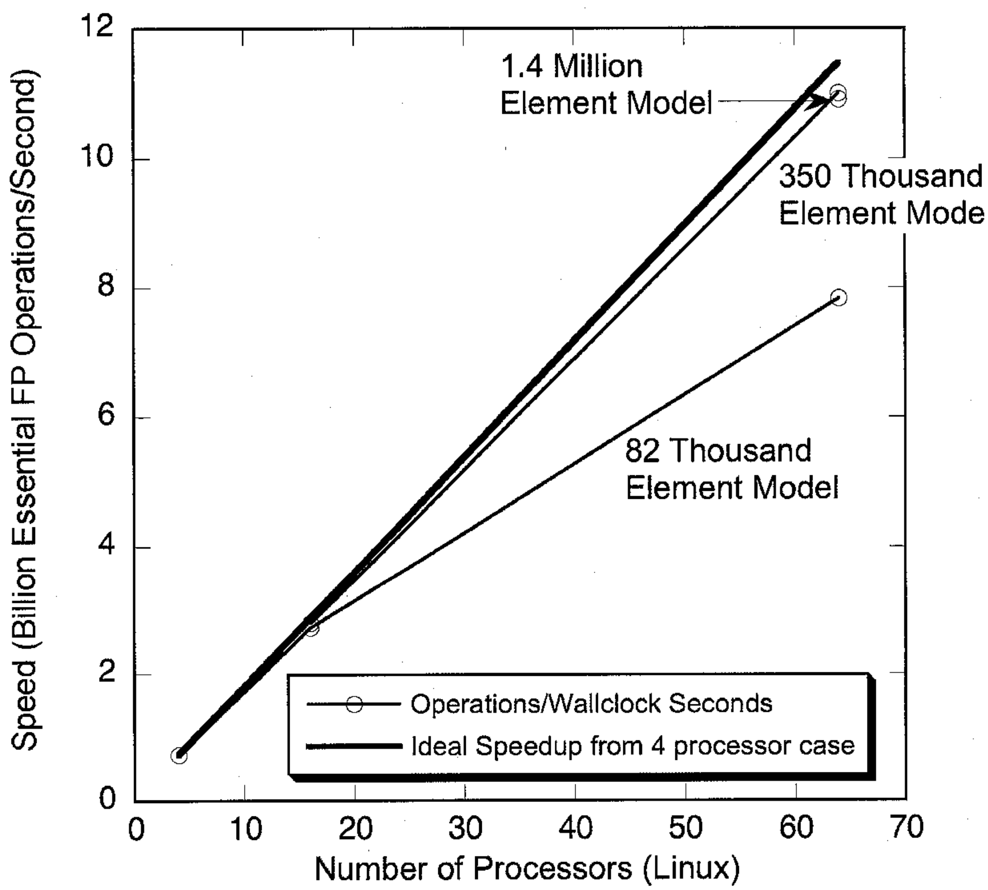

Figure 1: Scaling of three problems according to mesh size on Linux cluster.

Acknowledgments

This research was supported by the Computational Technologies Program of NASA's Earth Science Technology Office, and was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

References

[1] Thomas J. R. Hughes, "The Finite Element Method: Linear Static and Dynamic Finite Element Analysis." Dover, Publication, INC., Mineola, New York, 2000.

[2] Thomas J. R. Hughes and Robert Taylor, "Unconditionally Stable Algorithms For Quasi-Static Elasto-Plastic Finite Element Analysis." Computers & Structures, Vol. 8, pp. 169-173, 1978.

[3] H. J. Melosh and Raefsky, "A Simple and Efficient Method for Introducing Faults into Finite Element Computations." Bulletin of the Seismological Society of America, Vol. 71, No. 5, October, 1981.

[4] http://quakesim.jpl.nasa.gov/documentation.html

[5] http://complexity.ucs.indiana.edu:8282

[6] http://www.openchannelsoftware.org/projects/GeoFEST

[7] http://www.openchannelfoundation.org/projects/Pyramid

[8] http://www.physics.hmc.edu/GL/geofest/input/index.html, http://www.physics.hmc.edu/GL/geofest/input3/index.html