CitcomS: Compressible Mantle Convection

CitcomS, developed and maintained by the Computational Infrastructure for Geodynamics (CIG), is a finite element code designed to solve thermal convection problems relevant to Earth's mantle, and is released under the GNU General Public License. The software is now at version 3.0.3. All CIG software can be found via the CIG website, http://geodynamics.org.

E. Tan, M. Gurnis, S. Kientz, Computational Infrastructure for Geodynamics (U.S.)

NSF Award 0426271

Parallel Code Written in C, the code runs on a variety of parallel processing computers, including shared and distributed memory platforms. In an effort to increase the functionality of CitcomS to include greater control during simulations on large parallel systems, the software has been reengineered from previous versions of CitcomS to work with a Python-based modeling framework called Pyre. With Pyre, CitcomS can be dynamically coupled with other CitcomS simulations or with other codes such as SNAC, which solves crustal and lithospheric problems.

Full Spherical or Restricted Region Options CitcomS offers two variants, CitcomSFull and CitcomSRegional; the first solves for problems within a full spherical domain, and the second, for a restricted domain of a full sphere. Although the code is capable of solving many different kinds of convection problems using the flexibility of finite elements, there are aspects of CitcomS which make it well-suited for solving problems in which the plate tectonic history is incorporated. You easily access either geometry by simply changing command line options.

About CitcomS The fundamental basis for the numerical solution of any time-dependent convection problem is the sequential solution of an equation of motion and an energy equation. Convection problems are initially valued with boundary conditions, including all of the problems which are solved with CitcomS. The normal sequence of steps for the solution of convection problems starts with an initial temperature field. First, the momentum equation is solved. The solution of this equation gives us the velocity from which we then solve the advection-diffusion equation, giving us a new temperature. CitcomS uses this interleaved strategy. Variable viscosity, including temperature-, pressure-, position-, composition-, and stress-dependent viscosity are all possible, although they may not be fully implemented in the current version.

Governing Equations

With CitcomS, the mantle is treated as an anelastic, compressible, viscous spherical shell under Truncated Anelastic Liquid Approximation. With these assumptions, thermal convection is governed by the equations for conservation of mass, momentum, and energy:

$$(\rho u_i)_{,i} = 0 \qquad \text{[Eq. 1]}$$

$$-P_{,i} + \left(\eta(u_{i,j} + u_{j,i} - {}^2/_3\, u_{k,k}\,\delta_{ij})\right)_{,i} + \delta\rho g \delta_{ir} = 0 \qquad \text{[Eq. 2]}$$

$$\rho c_P (T_{,t} + u_i T_{,t}) = \rho c_P \kappa T_{,ii} + \rho\alpha g u_r (T + T_0) + \Phi + \rho\,(Q_{L,t} + u_i Q_{L,i}) = \rho H \qquad \text{[Eq. 3]}$$

where ρ is the density, u is the velocity, P is the dynamic pressure, η is the viscosity, δ_{ij} is the Kroneker delta tensor, $\delta\rho$ is the density anomaly, g is the gravitational acceleration, T is the temperature, T_0 is the temperature at the surface, c_P is the heat capacity, κ is the thermal diffusivity, α is the thermal expansivity, Φ is the viscous dissipation, Q_L is the latent heat, and H is the heat production rate. The expression X_y represents the derivative of X with respect to y, where i and j are spatial indices, r is the radial direction, and t is time. With phase transitions and temperature and composition variations, the density anomalies are:

$$\delta\rho = -\alpha\bar{\rho}\,(T - \bar{T}_a) + \delta\rho_{ph}\Gamma + \delta\rho_{ch} C \qquad \text{[Eq. 4]}$$

where $\bar{\rho}$ is the radial profile of density, $\bar{T}_a$ is the radial profile of adiabatic temperature, $\delta\rho_{ph}$ is the density jump across a phase change, $\delta\rho_{ch}$ is the density difference between the compositions, Γ is the phase function, and C is the composition.

References

1. Hughes, T.J.R. *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis.* Englewood Cliffs, New Jersey: Prentice-Hall, Inc.; 1987. 672 p.
2. Moresi, L.N., and M. Gurnis (1996), Constraints on the lateral strength of slabs from three-dimensional dynamic flow models, *Earth Planet. Sci. Lett., 138*, 15-28.
3. Ramage, A., and A.J. Wathen (1994), Iterative solution techniques for the Stokes and Navier-Stokes equations, *Int. J. Numer. Methods. Fluids, 19*, 67-83.
4. Brooks, A.N. A Petrov-Galerkin Finite Element Formulation for Convecton Dominated Flows. Unpublished doctoral thesis, California Institute of Technology, Pasadena, CA, 1981.
5. Zhong, S., M.T. Zuber, L.N. Moresi, and M. Gurnis (2000), The role of temperature-dependent viscosity and surface plates in spherical shell models of mantle convection, *J. Geophys. Res., 105*, 11,063-11,082.
6. Cahouet, J., and J.-P. Chabard (1988), Some fast 3D finite element solvers for the generalized Stokes problem, *Int. J. Numer. Methods. Fluids, 8*, 869-895.
7. Atanga, J., and D. Silvester (1992), Iterative methods for stabilized mixed velocity-pressure finite elements, *Int. J. Numer. Methods. Fluids, 14*, 71-81.

Numerical Methods

The governing equations are solved with the finite element method[1]. CitcomS employs an Uzawa algorithm to solve the momentum equation coupled with the continuity constraints[2,3]. The energy equation is solved with a Streamline-Upwind Petrov-Galerkin method[4]. Brick elements are used, such as eight velocity nodes with trilinear shape functions and one constant pressure node for each element. The use of brick elements in 3D (or rectangular elements in 2D) is important for accurately determining the pressure, such as dynamic topography, in incompressible Stokes flow[1].

The discrete form of Eq. 1 and 2 may be written in the following matrix form:

$$(\mathbf{B}^T + \mathbf{C})\, u = 0 \qquad \text{[Eq. 5]}$$

$$\mathbf{A}u + \mathbf{B}p = f \qquad \text{[Eq. 6]}$$

where $\mathbf{A}$ is the "stiffness" matrix, u is a vector of unknown velocities, $\mathbf{B}$ is the discrete gradient operator, $\mathbf{C}$ is the second term in Eq. 1, p is a vector of unknown pressures, and f is a vector composed of the body and boundary forces acting on the fluid. The individual entries of $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$ and f are obtained using a standard finite element formulation; see Zhong et al for the explicit entries[5].

In the incompressible case, $\mathbf{C}$ is zero. Eq. 6 can be transformed by premultiplying by $\mathbf{B}^T + \mathbf{A}^{-1}$ and using Eq. 5 to eliminate the velocity unknowns:

$$\mathbf{B}^T + \mathbf{A}^{-1}\,\mathbf{B}p = \mathbf{B}^T + \mathbf{A}^{-1}\, f \qquad \text{[Eq. 7]}$$

This equation is solved using the Uzawa algorithm, an established method for solving the minimization of a dual function[6], which simultaneously yields the velocity field. A conjugate gradient scheme[3,7] is used for this iteration and forms the basis for the technique used in CitcomS.

In the compressible case, there are two different strategies to solve Eq. 5 and 6. The first strategy is to add another layer of iterations when solving Eq. 7. The right-hand-side vector is updated by the velocity solution of the previous iteration. This equation can be solved using the same conjugate gradient scheme as the incompressible case.

$$\mathbf{B}^T + \mathbf{A}^{-1}\,\mathbf{B}p = \mathbf{B}^T + \mathbf{A}^{-1}\, f - \mathbf{C}u^{(i-1)} \qquad \text{[Eq. 8]}$$

The second strategy is to transform Eq. 6 by premultiplying by $(\mathbf{B}^T + \mathbf{C})\mathbf{A}^{-1}$ and using Eq. 5 to eliminate the velocity unknowns:

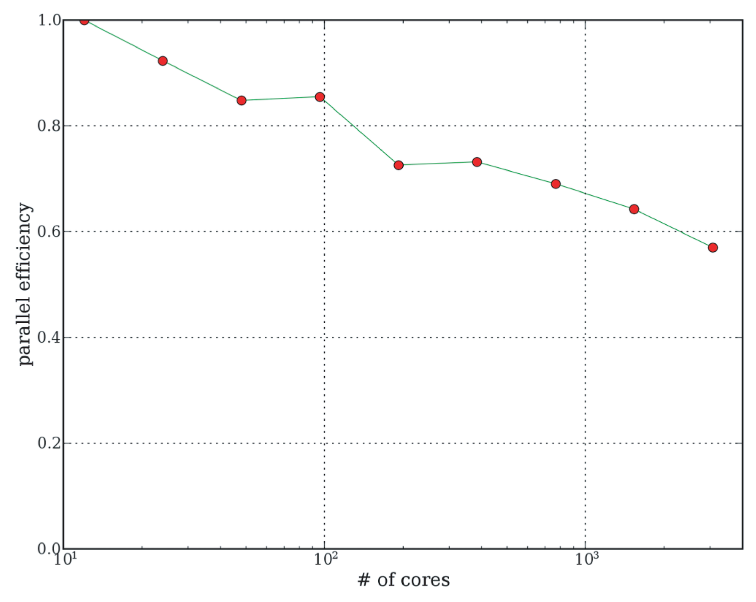
$$(\mathbf{B}^T + \mathbf{C})\mathbf{A}^{-1}\,\mathbf{B}p = (\mathbf{B}^T + \mathbf{C})\mathbf{A}^{-1} f \qquad \text{[Eq. 9]}$$

This equation is solved using a bi-conjugate gradient stabilized scheme.

The linear system is solved by either a full multigrid or a conjugate gradient solver. The solver uses the additive Schwarz method, in which each processor solves the sub-linear system within its domain and communicates with neighboring processors to obtain a global solution.

Parallel Performance

The parallel efficiency of the multigrid solver in CitcomS, scaling from 12 cores (the minimum required for a global model) to 3072 cores, is about 55%. The weak scaling is conducted with 30K elements per core on the TACC Lonestar cluster. The efficiency is measured in CPU time per Uzawa iteration, in which one multigrid solve is performed. The number of Uzawa iterations for each solve decreases with the core count, due to mesh-dependent convergence criterion.



geodynamics.org

Meshes and Geometry

There are two forms of meshes and geometries for CitcomS. By default CitcomS will produce a mesh within a regional geometry that is bound by lines of constant latitude and longitude. There is an option to generate a global mesh of a spherical shell. For a global mesh, CitcomS is also capable of generating a mesh for an entire spherical shell in which elements in map view are approximately equal in area. In the full spherical mode, CitcomS has 12 caps numbered 0 to 11 (see Fig. 1).

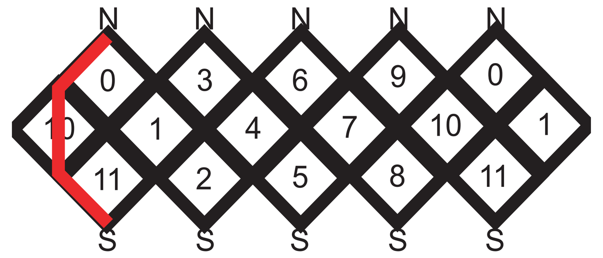
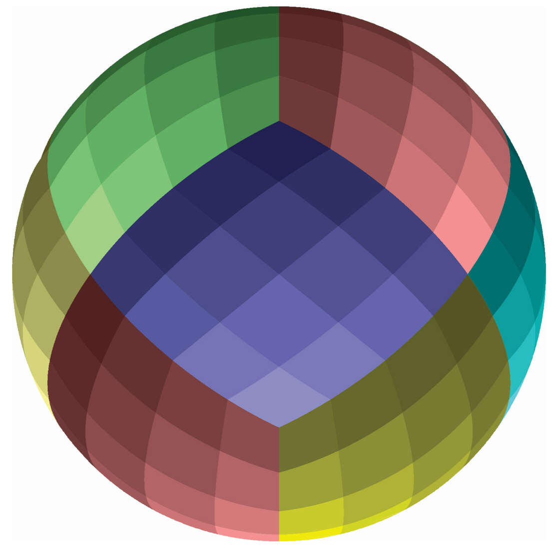


Fig. 1: Topological connectivity of the 12 caps. N is the north pole and S is the south pole. The red line marks the 0 degree meridian.



The caps are approximately square in map view so that the edges of the square are oriented diagonally with respect to latitude and longitude. The four corners of the domain are connected by great circles (Fig. 2). CitcomS can further generate meshes such that additional processors are used to divide caps uniformly along the two edges of the caps (Fig. 2) as well as in radius.

Fig. 2: Orthographic projection of processors from a full CitcomS mesh in which there are 16 processors in map view for each cap. The CitcomS cap is shown as distinct colors while the processor domains within the caps are indicated by the intensity of the color. This example was produced for a run with 2 processors in radius such that the total number of processors was 12x16x2=384.

Cookbook: Thermo-Chemical Convection

This example solves for thermo-chemical convection within a full spherical shell domain. Composition heterogenity exists in the Earth mantle. The density anomalies due to the composition heterogenity, as well as due to the thermal heterogeneity, drive the convection flow.

In this scenario, the mantle is initially layered. The bottom layer is compositionally distinct and is denser with a buoyancy number of 0.4. The model is purely bottom heated with a Rayleigh number 10^7. The boundary conditions are constant temperature and free-slip.

The results for this problem are shown in Fig. 3. The buoyancy ratio in this model is too low to stabilize the chemical layer. A few thermo-chemical plumes are rising from the lower mantle, especially the ones at the 4, 11, and 12 o'clock directions. The resolution of this model is fairly low. The composition isosurface is slightly discontinuous across the cap boundary. A model of higher resolution will not have this kind of artifact.

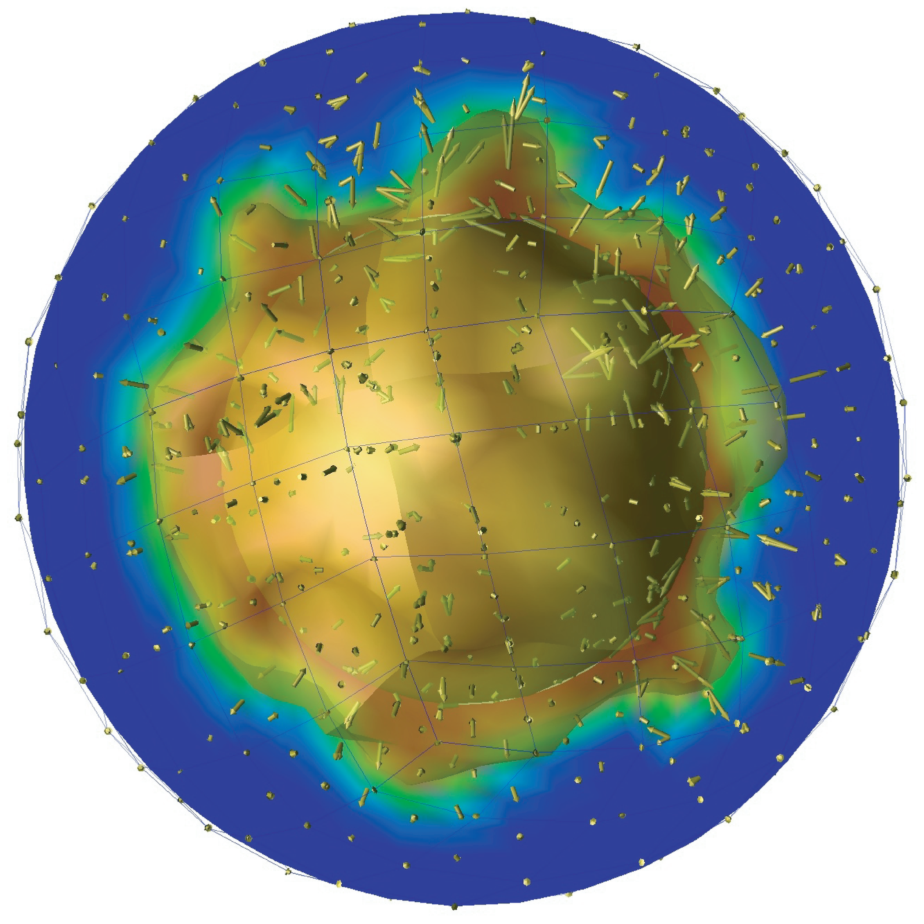


Fig. 3: Result of Thermo-Chemical Convection The composition and velocity field at the 20th step. The arrows are the velocity vectors. The composition field is shown in an isosurface of 0.7 and in a cross section.