Numerical simulation of geodynamic processes with the Portable Extensible Toolkit for Scientific Computation

R.F. Katz, M.G. Knepley, B. B.N. Smith, M. Spiegelman, E.T. Coon

▶ **To cite this version:**

HAL Id: hal-00532110
https://hal.science/hal-00532110

Submitted on 4 Nov 2010

Accepted Manuscript

Title: Numerical simulation of geodynamic processes with the Portable Extensible Toolkit for Scientific Computation

Authors: R.F. Katz, M.G. Knepley, B. Smith, M. Spiegelman, E.T. Coon
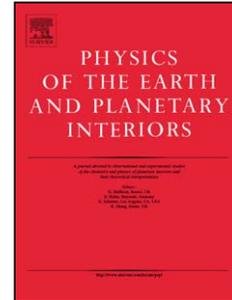
Please cite this article as: Katz, R.F., Knepley, M.G., Smith, B., Spiegelman, M., Coon, E.T., Numerical simulation of geodynamic processes with the Portable Extensible Toolkit for Scientific Computation, *Physics of the Earth and Planetary Interiors* (2007), doi:10.1016/j.pepi.2007.04.016

Numerical simulation of geodynamic processes with the Portable Extensible Toolkit for Scientific Computation

R. F. Katz [a],* M. G. Knepley [b] B. Smith [b] M. Spiegelman [c,d]

E. T. Coon [d]

[a] *University of Cambridge, Department of Applied Mathematics and Theoretical Physics, Wilberforce Road, Cambridge CB3 0WA, UK*

[b] *Mathematics and Computer Science Division, Argonne National Laboratory, USA*

[c] *Lamont-Doherty Earth Observatory, Columbia University, USA*

[d] *Department of Applied Physics and Applied Mathematics, Columbia University, USA*

Abstract

Geodynamics simulations are characterized by rheological nonlinearity, localization, three-dimensional effects, and separate but interacting length scales. These features represent a challenge for computational science. We discuss how a leading software framework for advanced scientific computing (the Portable Extensible Toolkit for Scientific Computation, PETSc) can facilitate the development of geodynamics simulations. To illustrate our use of PETSc, we describe simulations of (i) steady-state, non-Newtonian passive flow and thermal structure beneath a mid-ocean ridge, (ii) magmatic solitary waves in the mantle, and (iii) the formation of localized bands

of high porosity in a two-phase medium being deformed under simple shear. We highlight two supplementary features of PETSc, structured storage of application parameters and self-documenting output, that are especially useful for geodynamics simulations.

Key words: scientific computation, parallel computation, geodynamics, mid-ocean ridge, PETSc

PACS: 91.40.Jk, 91.40.St, 91.45.-c, 91.45.Fj, 47.11.Df, 02.60.Cb

1 Introduction

Two major challenges in solid-Earth geodynamics are the derivation of theoretical descriptions of geodynamic phenomena in terms of partial differential equations (PDEs) and the determination of solutions to these equations for appropriate boundary and initial conditions. Even minimally complex geodynamic models often contain important three-dimensional effects, sharp gradients in model variables, strong nonlinearities in material properties such as viscosity, and localization processes that lead to separated but interacting length scales. These features usually preclude analytical solution of the governing equations and enforce a reliance on numerical simulations. However, the same features also contribute to the difficulty of efficiently generating numerical solutions. The development of effective simulations can be facilitated by advanced numerical software libraries, of which the Portable, Extensible Toolkit for Scientific computation (PETSc) [Balay et al., 2001, 2004] is a

---

* Corresponding author.

  Email address: rfk22@cam.ac.uk (R. F. Katz).

  URL: www.damtp.cam.ac.uk/user/rfk22 (R. F. Katz).

2

leading example. The purpose of this paper is to demonstrate how PETSc has facilitated the development of large-scale computational simulations of a classic geodynamic theory for the coupled flow of solid and molten mantle rock [McKenzie, 1984].

Section 2 gives a general description of the PETSc package and greater detail on the parts of the package that we have employed for geodynamics simulations. Section 3 considers two example geodynamic simulations, the first of single-phase Stokes flow and the second of two-phase Darcy-Stokes flow. Section 4 describes special features of PETSc that are particularly useful for developing geodynamics simulations. Section 5 discusses both the future developments in PETSc and the challenges of future geoscience simulations.

1.1 Governing equations for mantle dynamics

Many of the expressions of mantle dynamics observable on the surface of the Earth are related to volcanoes. Major elements, trace elements and the isotopic chemistry of lavas, for example, are partially controlled by the spatial distribution of melting and the paths of melt transport. The position of volcanoes relative to plate boundaries as well as to other volcanoes is also a consequence of the coupled dynamics of magma and mantle rock. Theoretical models are required to interpret these observations in terms of fluid mechanical processes occurring at depth. A theory for the dynamics of the mantle should therefore describe both a solid, crystalline phase, which makes up the vast bulk of the mantle, and a liquid phase (magma or fluid), which is present in the mantle beneath volcanically active hotspots and tectonic plate-boundaries. In the limit of zero fluid fraction, this theory should reduce to the standard equations

3

used to describe mantle convection. In the limit of a rigid crystalline mantle, the theory should reduce to Darcy's law for flow in a permeable medium.

One derivation of such a theory is provided by McKenzie [1984], which considers the macroscopic conservation of mass, momentum, and energy for two interpenetrating continua consisting of a low-viscosity fluid in a high-viscosity deformable and permeable matrix. The equations for conservation of mass and momentum for both phases can be written as follows:

$$\frac{\partial \rho_f \phi}{\partial t} + \boldsymbol{\nabla} \cdot [\rho_f \phi \mathbf{v}] = \Gamma, \tag{1}$$

$$\frac{\partial \rho_s (1-\phi)}{\partial t} + \boldsymbol{\nabla} \cdot [\rho_s (1-\phi)\mathbf{V}] = -\Gamma, \tag{2}$$

$$\phi(\mathbf{v} - \mathbf{V}) = -\frac{K}{\mu} [\boldsymbol{\nabla} P - \rho_f \mathbf{g},] \tag{3}$$

$$\boldsymbol{\nabla} P = \boldsymbol{\nabla} \cdot \left(\eta \left[\boldsymbol{\nabla} \mathbf{V} + \boldsymbol{\nabla} \mathbf{V}^T \right] \right) + \boldsymbol{\nabla} \left[\left(\zeta - \frac{2}{3}\eta \right) \boldsymbol{\nabla} \cdot \mathbf{V} \right] + \bar{\rho} \mathbf{g}. \tag{4}$$

Here ϕ is porosity, ρ_f, ρ_s are the fluid and solid densities, $\mathbf{v}, \mathbf{V}$ are the fluid and solid velocity fields, Γ is the rate of mass transfer from solid to liquid (i.e. melting/crystallization rate), K is the permeability, μ is the melt viscosity, P is the fluid pressure, $\bar{\rho} = \phi \rho_f + (1-\phi)\rho_s$ is the phase-averaged density, $\mathbf{g}$ is the acceleration due to gravity, and η, ζ are the shear and bulk viscosities of the solid [see Spiegelman, 1993a,b, for further discussion].

Equations (1) and (2) conserve mass for the fluid and solid individually and allow mass-transfer between the phases through Γ. Equation (3) is an extended form of Darcy's law governing the separation of melt from solid. This separation flux is proportional to the permeability and fluid-pressure gradients in excess of hydrostatic. Equation (4) governs momentum conservation of the solid phase which is modeled as a compressible, inertia-free viscous fluid.

4

An important feature of equations (1)–(4) is that they consistently couple solid stresses and fluid pressure. The fluid pressure responds to solid deformation and gravity which drives fluid flow and changes the porosity. Variations in porosity and stress can then feed-back through the constitutive relations for the permeability and viscosity. Such feedbacks lead to a wide range of behavior including non-linear porosity waves [e.g., Scott and Stevenson, 1984, 1986, Barcilon and Richter, 1986, Barcilon and Lovera, 1989] and spontaneous flow localization [e.g., Stevenson, 1989, Katz et al., 2006]. In most cases solution of these equations require numerical methods because nonlinearities in the governing equations preclude analytical treatment. To find numerical solutions, we require software that is capable of solving large systems of coupled, non-linear algebraic equations; PETSc is a leading example of such software.

1.2 Methods for Solving Nonlinear PDEs

Discretization of nonlinear partial differential equations onto a mesh leads to a system of nonlinear algebraic equations. Such a system can be represented as

$$F(\tilde{u}) = 0 \tag{5}$$

where $\tilde{u} \in \mathbb{R}^n$ is a vector containing the exact solution of the problem and F is a nonlinear function of $\tilde{u}$ that maps $\mathbb{R}^N \to \mathbb{R}^N$. In practice, for large systems of equations, it is difficult to find a vector $\tilde{u}$ such that $F(\tilde{u})$ is exactly equal to the zero vector. More typically, one is satisfied with an approximation to the exact solution that satisfies the inequality

$$||F(u)|| < tol, \tag{6}$$

5

where *tol* is some specified tolerance and the vector norm $|| \cdot ||$ is chosen according to the context. The key to finding a good approximation to equation (5) is to employ an iterative method that reduces $||F(u)||$ on each iteration. In describing iterative methods, u^n is used to represent the approximate solution after n iterations of the solution method.

The most straightforward iteration solution strategy is a fixed-point iteration for the equation $u = u + F(u)$. An initial guess for the solution u^0 is used to determine a residual $F(u^0)$, which is then applied as an update to the guess, $u^1 = u^0 + F(u^0)$. If successive iterates converge, they will converge to the solution of $F(u) = 0$.

A more robust method, referred to as Picard iteration in the groundwater literature [e.g., Paniconi and Putti, 1994, Mehl, 2006], can be used to linearize a nonlinear system of equations, making it amenable to solution by a readily available linear solver. This conversion is accomplished by substitution of best current guesses for field variables into nonlinear terms of the governing equations. We employ this method below to split a set of coupled PDEs into two parts, which are solved separately (see section 3.9). Paniconi and Putti [1994] and Mehl [2006] have quantitatively compared this successive substitution method with Newton's method.

Newton's method, which is described in more detail below, can converge faster than Picard for nonlinear systems derived from PDEs if the initial guess u^0 is close to the solution. To be practical, however, Newton's method requires ready evaluation of the Jacobian as well as some control feature such as line-search or trust-region techniques [Steihaug, 1983] to make it more robust. PETSc provides these features. In the next section we give a general description of

6

PETSc and highlight its capabilities relevant for solving problems resulting from systems of nonlinear PDEs.

2 PETSc basics

The Portable Extensible Toolkit for Scientific Computation is designed to assist the development of "full physics" simulations. Its goal is to eliminate implementation as the bottleneck in developing and running complex, multi-physics simulations at a scale that allows the rapid advancement in science through simulation. PETSc's focus is on the numerical solution of the algebraic systems arising when using implicit methods based on conventional finite-difference, finite-element, and finite-volume techniques for PDEs. Moreover, the modular design fosters reuse of scientific components, such as a Navier-Stokes solver, among different simulations. PETSc also supports a limited but growing set of tools for managing the data distribution and discretization techniques needed to construct the algebraic systems.

Several excellent general-purpose software environments for numerical computation exist; perhaps the best known of these is Matlab. In addition, easy-to-use "scripting" languages such as Python allow the rapid prototyping of numerical simulations. However, many scientific simulations in geodynamics require (and will require even more in the future) resolution at a scale in both time and space leading to a system size that is unavailable in these environments. Only parallel (multiprocessor) computing systems can tackle these large problems. In fact, many realistic simulations will require computer systems with thousands of processors. Thus, simulations developed in these general-purpose environments must often be completely redeveloped by using

7

hand-coded parallel computational numerics, using the foregoing system only for visualization and post processing for statistics.

2.1 Background and Philosophy

A wide variety of "parallelization" techniques have been proposed over the years to achieve both efficient use of parallel machines and ease of programming for scientists. They have all been, essentially, failures. The result is that the message-passing-model is the standard parallelization approach for engineering and scientific simulation. One fortunate result of research and development in parallel computing is the development the Message Passing Interface (MPI), which provides a powerful common interface on all hardware systems [Message Passing Interface Forum, 1994, 1998].

Message-passing parallel programming has sometimes been called the assembly language of parallel computing. The programmer must manage every detail of the parallelism and data movement between processes. With this come great power and flexibility—in fact, an overwhelming amount of flexibility. One goal of PETSc is to eliminate the direct use of MPI programming from numerical simulations involving the solution of PDEs. Specifically, the PETSc libraries are used to manage the details of the communication and the user is left to orchestrate the overall flow of the communication and computation, plus the detailed physics modules.

2.2 Distributed Arrays and Parallelism

[Fig. 1 about here.]

8

PETSc is predicated on mesh decomposition, also called domain decomposition, in order to partition data and computational work among the processes. Each process is typically assigned a contiguous portion of the mesh, as in Fig. 1. The physical data for this portion of the mesh is stored in local process memory, and computations on that portion of the mesh are performed by this process. Of course, these local calculations will require data from neighboring partitions, which are termed ghost points, shown in Fig. 2.

[Fig. 2 about here.]

Managing the data movement and coordination between processes is the bulk of the difficulty with parallel computing. This effort is vastly simplified by using the local physics/global algebraic solver paradigm in PETSc. The fields are restricted to a local representation, including the ghost points; local physics computations are performed; and the result is used to update the global representation. The conversion between local and global data layouts, as well as communication, is handled automatically by PETSc. Thus the algebraic solvers (Newton's method, linear iterative solvers, etc.) see only a global algebraic representation of the problem, while the physics modules see only the data on a local portion of the mesh. The beauty is that at this abstract level, the data management is independent of the particular physical model, be it fluid dynamics, structural mechanics, MHD, or the like.

The `DA` construct in PETSc is the special case of this paradigm for computation on structured tensor product grids. It combines specification of the topology, geometry, and process interconnection. The user supplies only a routine to evaluate the nonlinear residual and optionally the Jacobian over a given local piece of the mesh. The local fields are presented to the user code,

9

not as abstract objects, but as the more familiar multidimensional array in both C and Fortran, giving expressions that conform to stencil indexing (i.e., $u(i,j,k)$) instead of vector (i.e., $u(I)$) indexing. If the Jacobian is also user provided, PETSc provides additional support for index translation with the `MatSetValuesStencil()` method. It automatically translates `(i,j,k)` mesh coordinates to global matrix indices needed for the algebraic solver. This approach works equally well for tensor product finite-element, finite-volume, or finite-difference formulations. The key point again is that the physics modules need only know about the local mesh representation, never the global algebraic solver representation.

In addition, there is support for geometric multigrid with automatic or custom interpolation and coarse-grid operators. Since the user has provided an evaluation routine for a general grid patch, coarse representations of the operator can be obtained directly. They may also be obtained automatically by PETSc via the algebraic Galerkin process.

2.3 Solvers

The PETSc algebraic solvers always work with the global algebraic representation of the fields. This allows the solver software to be used in virtually any application, independent of the particular physics, discretization, or even local representation of the fields.

PETSc uses (truncated, approximate) Newton's method to solve the nonlinear algebraic equations. That is,

$$u^{n+1} = u^n + \delta u^n, \tag{7}$$

10

where δu^n is obtained by approximately solving

$$J(u^n)\delta u^n = -F(u^n), \tag{8}$$

where $F(u^n)$ is the global residual at iteration n and J is the (approximate) Jacobian. The computation of $F(u^n)$ is done as described above in the local mesh representation. The `DA` PETSc infrastructure automatically manages the translation of the results of the local physics modules into the global representation used by algebraic solvers.

For non-matrix-free Newton's method, PETSc computes the Jacobian matrix using finite differences via $J_{*j}(u^n) = \nabla_{u_j} F(u^n) \approx (F(u^n + he_j) - F(u^n))/h$, where h is computed dynamically to provide the best approximation (unless the Jacobian is provided by a subroutine in the application code). Naively, one would need N computations of the residual to compute all the columns of J; again N is the total number of unknowns. Fortunately, because of the sparsity of J, all the columns of the J that do not share a common row can be computed by using the same discrete residual evaluations. This computation reduces the number of discrete residual calculations to the number of colors of a particular graph of the matrix J that is bounded and independent of N, the size of the problem. In addition, since the perturbations needed in the differencing are local, one can perform all of these function evaluations without parallel communications, dramatically decreasing the cost.

To specify the problem to PETSc, the user defines a solution tolerance and provides call-back functions that generate an initial guess of the solution and calculate each component of the residual vector $r^n = F(u^n)$ given a vector of field variables u^n. All of the physics of the problem resides in the latter

11

of these two call-back functions. We have found that incomplete LU precon-
ditioned GMRES [Demmel, 1997] gives robust and scalable performance on
these particular problems, and we have used it in generating the simulation
results described below.

PETSc provides a range of linear solvers (see `www.mcs.anl.gov/petsc/petsc-as/`
`documentation/linearsolvertable.html` for the current comprehensive list).
The solvers are generally a combination of a fixed-point solver (called a pre-
conditioner) such as Gauss-Seidel or multigrid, plus a Krylov method that
accelerates the convergence of the preconditioner, such as the conjugate gra-
dient method or GMRES. Since the use of these solvers is independent of the
particular physics, they may all be selected at runtime. This approach allows
the optimal solver for a particular application to be determined rapidly via a
series of runs, without requiring recompiles between the changes in the solvers.

3   Simulations of One and Two-phase Mantle Dynamics

3.1   Single Phase Limit

In the limit that porosity, ϕ, and melting rate, Γ, are both zero, equations (1)–
(4) reduce to Stokes flow for an incompressible fluid. Coupled with an equation
for the conservation of energy, this system can be solved for the thermal and
flow structure of a convecting fluid. Many authors have used these equations
to study the kinematically driven flow of mantle rock in mid-ocean ridge and
subduction zone settings [e.g., van Keken et al., 2002, Kelemen et al., 2002,
Gerya and Yuen, 2003]. The challenge in performing such calculations arises
from nonlinearities of the constitutive equation that describes the viscosity

12

of mantle rock. Over geologic timescales, the mantle behaves as a fluid with a non-Newtonian, temperature-dependent viscosity [Karato and Wu, 1993, Kelemen et al., 1997a]:

$$\eta = A_0 \exp\left(\frac{E^* + PV^*}{RT} - \alpha\phi\right)\dot{\epsilon}_{II}^{\frac{1-n}{n}}, \tag{9}$$

where A_0 is a constant of proportionality, E^* and V^* are the activation energy and activation volume, R is the gas constant, $\alpha \approx 27$ is an empirically determined constant [Hirth and Kohlstedt, 1995a,b, Mei et al., 2002], ϕ is the porosity in volume fraction, $\dot{\epsilon}_{II}$ is the second invariant of the strain rate tensor, and n is a constant describing the strain-rate dependence of viscosity. For dislocation creep $n \approx 3.5$, while for diffusion creep $n = 1$.

Neglecting the temperature and strain-rate dependencies (i.e., assuming a constant viscosity) with certain kinematically prescribed boundary conditions allows for analytic solution of incompressible Stokes for the "corner flow" solution [Batchelor, 1967]. This solution has been used extensively to model the mantle at tectonic plate boundaries [e.g. McKenzie, 1969, Spiegelman and McKenzie, 1987]. Kelemen et al. [2002], however, showed that isoviscous models cannot meet constraints derived from petrologic and heat-flow data at subduction zones. Variable viscosity models of mantle flow can meet these constraints, but such models generally preclude analytic solution because the rheology introduces nonlinearity into the governing equations. We have employed PETSc to generate numerical solutions for variable-viscosity flow and thermal structure in ridge [Katz et al., 2004] and arc [Knepley et al., 2006] settings.

13

Next we detail the discretization and solution strategy that we have employed to model the three-dimensional, incompressible, single-phase flow and thermal structure of the mantle beneath mid-ocean ridges. We discuss parallel performance of this code and briefly explore an example of the solutions that we obtain.

3.2 Discretization

We use a finite volume discretization of the governing equations on a staggered mesh, shown in Fig. 3, to avoid spurious grid-scale oscillations of pressure [Patankar, 1980]. Calculating fluid velocities on the control volume boundaries leads to simple discrete representations of the governing equations. For example, the continuity equation $\boldsymbol{\nabla} \cdot \mathbf{V} = 0$ is integrated over a control volume giving

$$\frac{u_{ijk} - u_{i-1jk}}{\Delta x} + \frac{v_{ijk} - v_{ij-1k}}{\Delta y} + \frac{w_{ijk} - w_{ijk-1}}{\Delta z} = P^r_{ijk}, \tag{10}$$

where u, v, w is the velocity in the x, y, and z-directions and the cell dimensions are given by $\Delta x, \Delta y, \Delta z$. P^r_{ijk} is the residual of the continuity equations in cell Ω_{ijk} and corresponds to the pressure variable in our approach, even though equation (10) does not contain the pressure. Pressure in cell Ω_{ijk} is constrained by the momentum equations for u, v, w in Ω_{ijk} and its immediate neighbors. The residuals for the discrete momentum equations are assigned to u^r_{ijk}, v^r_{ijk}, and w^r_{ijk}. These equations are presented in detail in Albers [2000].

[Fig. 3 about here.]

14

Temperature is governed by the conservation of enthalpy equation

$$\rho c_P \mathbf{V} \cdot \boldsymbol{\nabla} T = \boldsymbol{\nabla} \cdot k \boldsymbol{\nabla} T, \tag{11}$$

which requires that, in steady state, advection of heat be balanced by diffusion of heat. Here T is the mantle potential temperature, and k, ρ, and c_P are the thermal conductivity, density and specific heat of the solid mantle. This equation is discretized (see Albers [2000]) to give an expression for the residual T_{ijk}^r. Following Trompert and Hansen [1996] we use a Fromm upwind advection scheme.

To completely specify the discrete problem, we must specify boundary conditions. The grid of control volumes is constructed so that the domain boundaries fall on the edges of cells and there are a sufficient number of buffer cells outside the domain boundary to accommodate the boundary stencil. The vertical velocity, w, has a mesh position that coincides with the domain boundary when a cell is adjacent to a horizontal boundary. Thus, to impose $w = 0$ on the top boundary of the domain, we must specify that $w_{ij0}^r = w_{ij0}$. We use linear interpolation to enforce Dirichlet boundary conditions on variables that, because of staggering of the mesh, do not fall on the domain boundaries. For example, the residual for temperature on the top boundary is given by $T_{ij0}^r = T_{ij0} + T_{ij1}$. When $T_{ij0}^r = 0$, we have successfully imposed $T(x, y, 0) = 0$ (T represents the non-dimensional temperature in this case). The complete set of boundary conditions for the domain is given in Table 1.

[Table 1 about here.]

Each control volume Ω_{ijk} has five degrees of freedom $\{u, v, w, P, T\}$ and their corresponding residuals. The total set of degrees of freedom and residuals

15

can be assembled into two vectors, u and r, of length $N = 5N_c$, where N_c is the total number of grid cells. The approximate solution to the discrete nonlinear system $F(\tilde{u}) = 0$ is then given by an unknown vector u such that $||F(u)|| = ||r|| \leq tol$, where tol is a tolerance specified by the user and is typically chosen to be 1×10^{-5} or less.

3.3   Solution Strategy: Continuation

In general, the convergence of Newton's method requires a good initial guess. We have certainly found this situation to be true in the case of variable viscosity flow beneath a mid-ocean ridge. While the 2D analytic corner flow solution for constant viscosity is a good starting guess for 2D simulations, it is clearly not good for 3D flow beneath a ridge with transform offsets. We therefore adopt a continuation method, forcing the variation in viscosity to go from zero to the full predicted variance over a set of iterations of the nonlinear solve [Knepley et al., 2006]. To smoothly control the variation of viscosity, we set an upper limit on viscosity and use this limit to normalize it to a range between zero and one. The viscosity field η_m^* used in iteration m of the continuation loop is then given by

$$\eta_m^* = \eta^{\alpha_m}, \quad \alpha_m \in [0, 1], \tag{12}$$

where $m = 1, 2, ..., M$. In the first iteration $\alpha_1 = 0$, giving the solution to the isoviscous case. This solution is then used as a guess for the next solve with $\alpha_2 > \alpha_1$. The iteration loop ends with the solution at $\alpha_M = 1$, which has full variation in viscosity. For diffusion creep viscosity (Newtonian, temperature-dependent), between 5 and 10 continuation steps are required, where dislo-

16

cation creep (non-Newtonian, temperature-dependent) typically requires between 10 and 30 iterations. For iterations with $m < M$, a relaxed nonlinear tolerance may be used (e.g., $tol = 10^{-3}$) to reduce the number of Newton steps that are necessary and hence to speed the continuation method.

3.4 Parallel Scaling

Although most 2D simulations can be run in serial on a single processor, 3D simulations with reasonable spatial resolution typically cannot. The simulations discussed above, which require a grid resolution of 1–5 km in a domain of order 300 km in each direction, must be run in parallel. Using PETSc facilitates the transition from a serial platform to a parallel supercomputer in terms of code development: in most cases, application code contains no explicit interprocess communications. A properly written application code that uses PETSc-provided data structures and methods is inherently parallel. This parallelization, however, does not guarantee perfect scaling. How the performance of the application code scales to large numbers of processors depends on the chosen solvers and on the structure of the application code itself.

Figure 4 shows scaling results for the 3D ridge simulation code compared with ideal scaling for a fixed problem size. The application uses an additive Schwartz method for domain decomposition with a ILU preconditioner on each block [Cai et al., 1997].

[Fig. 4 about here.]

17

3.5   3D Ridge Results

Many authors have developed 3D numerical models of mantle flow beneath mid-ocean ridges with transform faults. Early examples used constant viscosity [Phipps Morgan and Forsyth, 1988], layered viscosity [M. Rabinowicz and Rosemberg, 1993], and temperature and pressure dependent viscosity [Shen and Forsyth, 1992]. Sparks and Parmentier [1993] considered 3D convection beneath a mid-ocean ridge. Georgen and Lin [2002] investigated 3D isoviscous flow beneath a ridge triple junction. The model described here differs from earlier work in that it incorporates the effects of non-Newtonian dislocation creep viscosity on the flow and thermal structure beneath a mid-ocean ridge.

While a thorough investigation of the behavior of the 3D ridge model is beyond the scope of this paper, a representative result is shown in Fig. 5. In this case, we have chosen the ridge geometry and spreading rate to mimic the region around the Clipperton and Siqueiros transform faults of the Eastern Pacific Rise at about 9° north latitude. This is a fast-spreading ridge with a robust melt supply [Crawford and Webb, 2002].

[Fig. 5 about here.]

Grid resolution for this simulation is 4 kilometers in each direction. The computation, performed on 512 nodes of the Blue Gene/L at Argonne National Laboratory, solved for 2.8 million degrees of freedom (however this simulation was not used for the scaling test, shown above). The deformation rheology used in the simulation combines diffusion and dislocation creep and predicts mantle viscosities as low as 5×10^{18} Pa-s in the region beneath the spreading center.

18

Figure 5a shows isosurfaces of upwelling velocity. These surfaces suggest that upwelling beneath a fast-spreading ridge is uniform beneath most of a ridge segment. Near the ends of segments bounded by large transform offsets, upwelling rates are diminished, and mantle is drawn into the spreading zone laterally from across the transform fault. Vertically integrated melt production, shown in Fig. 5b, reflects this variation in upwelling, with lower melt production rates beneath segment ends than beneath segments centers. Note, however, that a small offset in the ridge has little effect on upwelling or melt production. The low predicted viscosity could permit buoyant convection beneath the ridge, driven by horizontal gradients in porosity [Buck and Su, 1989]. This would lead to a time-dependent solution and should be explored in the context of two-phase models.

For this 3D model, melting is calculated by using the lherzolite solidus of Katz et al. [2003] and an adiabatic productivity of 0.4% per kilometer of upwelling, chosen to give a reasonable crustal thickness [Katz et al., 2004]. Melt focusing to the ridge is parameterized as a process of melt percolation up a sloping solidus [e.g., Sparks and Parmentier, 1991, Magde and Sparks, 1997]. The maximum melt focusing distance is arbitrarily set to 80 km in Fig. 5. Changing this distance affects the predicted crustal production rate. For the simulation shown in Fig. 5, changing the maximum melt focusing distance from 80 km to 30 km reduces the predicted crustal thickness from about 7.5 to 4.9 km. Changes in melt focusing may also affect crustal thickness asymmetry across transform faults on migrating ridges [Carbotte et al., 2004, Katz et al., 2004]. Simple parameterization of melt transport are useful for gaining insight into mid-ocean ridge processes, but more quantitative models will depend on solutions to the equations governing two-phase mantle/magmatic flow.

19

3.6   Two Phase Flow

The 3D ridge models provide the large-scale framework for understanding the solid flow field and thermal structure beneath mid-ocean ridges. A more complete description of these regions, however, includes the production and migration of partially molten rock. Fortunately, equations (1)–(4) provide a consistent and tractable extension of solid-state mantle convection to include magma dynamics. These equations can be rewritten in a form that is amenable to numerical solution as a coupling of compressible Stokes with Darcy's law. The surprising feature of this coupled system is that it has considerably richer behavior than either of the two subproblems alone. In particular, these equations are unstable to the spontaneous formation of time-dependent, small-scale coherent structures such as solitary waves or "melt bands" (see below).

The key to the new formulation is to partition the total pressure P into three components,

$$P = P_l + \mathcal{P} + P^*, \tag{13}$$

where $P_l = \rho_s^0 gz$ is the reference background "lithostatic" pressure, $\mathcal{P} = (\zeta - 2\eta/3)\boldsymbol{\nabla} \cdot \mathbf{V}$ is the "compaction" pressure due to expansion or compaction of the solid, and P^* includes all remaining contributions to the fluid pressure (particularly the dynamic pressure due to viscous shear of the matrix).

With these definitions and a bit of algebra, we can eliminate the melt velocity $\mathbf{v}$ from the equations using the same basic manipulations as in Spiegelman [1993a] [see also Spiegelman et al., 2001, Spiegelman and Kelemen, 2003]. If we approximate ρ_f, ρ_s to be constant (but not equal), we can rewrite the

20

equations as

$$\frac{D\phi}{Dt} = (1 - \phi)\frac{\mathcal{P}}{\xi} + \Gamma/\rho_s \tag{14}$$

$$-\boldsymbol{\nabla}\cdot\frac{K}{\mu}\boldsymbol{\nabla}\mathcal{P} + \frac{\mathcal{P}}{\xi} = \boldsymbol{\nabla}\cdot\frac{K}{\mu}\left[\boldsymbol{\nabla}P^* + \Delta\rho\mathbf{g}\right] + \Gamma\frac{\Delta\rho}{\rho_f\rho_s} \tag{15}$$

$$\boldsymbol{\nabla}\cdot\mathbf{V} = \frac{\mathcal{P}}{\xi}, \tag{16}$$

$$\boldsymbol{\nabla}P^* = \boldsymbol{\nabla}\cdot\eta\left(\boldsymbol{\nabla}\mathbf{V} + \boldsymbol{\nabla}\mathbf{V}^T\right) - \phi\Delta\rho\mathbf{g} \tag{17}$$

where $D\phi/Dt = \partial\phi/\partial t + \mathbf{V}\cdot\boldsymbol{\nabla}\phi$ is the material derivative of porosity in the frame of the solid, $\xi = (\zeta - 2\eta/3)$ and $\Delta\rho = \rho_s - \rho_f$.

Equation (14) is an evolution equation for porosity in a frame following the solid flow. In this frame, porosity changes are driven by the balance of physical volume changes ($\mathcal{P}/\xi \equiv \boldsymbol{\nabla}\cdot\mathbf{V}$) and melting. Equation (15) is a modified Helmholtz equation for the compaction pressure $\mathcal{P}$, which reduces to Darcy flow in rigid porous media in the limit $\xi \to \infty$. This equation is responsible for much of the novel behavior in this system and has been discussed in detail in Spiegelman [1993a,b]. Equation (16) relates the divergence of the solid flow field to the compaction pressure, and Eq. (17) is Stoke's equation for the solid velocity and P^* with porosity-driven buoyancy. Given $\phi, \mathcal{P}, P^*$ and $\mathbf{V}$, the melt flux is reconstructed as

$$\phi\mathbf{v} = \phi\mathbf{V} - \frac{K(\phi)}{\mu}\left[\boldsymbol{\nabla}(P^* + \mathcal{P}) + \Delta\rho\mathbf{g}\right]. \tag{18}$$

All of these equations are in forms readily amenable to analytic and numerical techniques. Equations (14)–(17) form a coupled hyperbolic-elliptic set of equations for porosity, pressure and solid flow. To solve these problems requires initial conditions and inflow conditions for the porosity and boundary

21

conditions on pressure, solid velocity or stress. Most natural boundary conditions for the compaction pressure can be written in terms of the melt flux and tend to be Neumann conditions on $\mathcal{P}$.

3.7 Magmatic Solitary Waves

The novel features of these equations arise from equations (14)–(15). In the limit of small porosity $\phi << 1$, constant viscosities and neglecting melting or large-scale solid shear, Equations (14)–(15) can be written in dimensionless form as

$$\frac{D\phi}{Dt} = \mathcal{P} \tag{19}$$

$$-\boldsymbol{\nabla}\cdot\phi^n\boldsymbol{\nabla}\mathcal{P} + \mathcal{P} = \boldsymbol{\nabla}\cdot\phi^n\hat{\mathbf{g}} \tag{20}$$

These equations have been shown to produce nonlinear solitary waves of porosity in 1, 2, and 3 dimensions that propagate over a uniform porosity background with fixed form and constant speed [e.g., Scott and Stevenson, 1984, Scott et al., 1986, Scott and Stevenson, 1986, Richter and McKenzie, 1984, Barcilon and Richter, 1986, Barcilon and Lovera, 1989, Wiggins and Spiegelman, 1995].

These waves are a natural consequence of the ability of the matrix to dilate or compact in response to variations in melt flux. Perhaps more importantly, they provide an excellent benchmark test for computational methods. Given a single solitary wave of the appropriate dimension, it should propagate with unchanging form and constant phase velocity. Any other behavior is an artifact of the numerical method. We have developed several PETSc-based examples

22

for 2D solitary waves that incorporate a highly accurate spectral solution for individual solitary waves in all dimensions (G. Simpson, pers. comm. 2006). The PETSc codes are solved on a regular mesh that takes advantage of the `DA` abstraction as well as geometric multi-grid preconditioners and a semi-Lagrangian method of characteristics solver extension to PETSc to accurately solve the hyperbolic advection components.

3.8   Localization and the Formation of Melt Bands

The solitary wave solutions represent excellent verification tests for simulations. However, a more difficult problem arises in solving the full system of two-phase equations where shear deformation of the solid couples with volumetric deformation and the evolution of porosity. Such problems are motivated by experiments described by Zimmerman et al. [1999] and Holtzman et al. [2003a] that demonstrate spontaneous melt localization in a deforming, partially molten two-phase aggregate. The pattern of melt bands observed in these experiments may have important implications for melt transport and seismic anisotropy in the mantle [Holtzman et al., 2003b]. These experiments provide both direct validation of the equations of magma dynamics and a considerable computational challenge for solving coupled non-linear PDEs.

Katz et al. [2006] describes PETSc-based computational models of the experiments that are solutions of the equations of magma dynamics assuming no melting or buoyancy. These simulations require a matrix viscosity that depends on both porosity and strain-rate (equation (9) above) to reproduce the observed pattern of melt bands at low angles ($\sim 20°$) to the shear plane. Linearized stability analysis that assume only porosity dependent viscosity [e.g.

23

Stevenson, 1989, Spiegelman, 2003]) predict melt bands oriented at 45° to the shear plane. Katz et al. [2006] extended this analysis to non-Newtonian viscosity and provided full non-linear solutions that demonstrated how the strain-rate dependence of viscosity controls the angle at which melt bands emerge.

The simulations were developed by using a discretization of equations (14)–(17) on a 2D staggered mesh similar to that shown in Fig. 3. In this mesh, horizontal and vertical velocities reside on cell boundaries, while the porosity and pressures ($\mathcal{P}$ and P^*) reside on the cell centers. The mesh is periodic in the x-direction. The discrete equations are derived with finite-difference or finite-volume approximations that conform to the mesh layout. For example, the discrete form of continuity, equation (10), becomes

$$\frac{U_{ij} - U_{i-1j}}{\Delta x} + \frac{V_{ij} - V_{ij-1}}{\Delta y} - \frac{\mathcal{P}_{ij}}{\xi_{ij}} = P_{ij}^{*r}. \tag{21}$$

Note that as in equation (10), the residual in equation (21) corresponds to a pressure variable that does not appear in the equation. P^* is in fact constrained by the momentum balance equation (17) and the compaction rate equation (15).

The time-dependence in the melt band simulation is derived from the advection-compaction equation for porosity, (14), which is hyperbolic, while the other governing equations are elliptic. We employ a semi-implicit time discretization of (14) and, as above for the ridge simulation, a Fromm scheme for advection. Because of the time-dependence, our strategy for solving the discrete equations derived from (14)–(17) differs from our approach in the 3D ridge simulation, where we bundled all the discrete equations into one system of

24

algebraic equations and solved them simultaneously. Here we split the discrete advection-compaction equation from the others and solve it separately, iterating at each timestep between solution of the hyperbolic equation and the elliptic equations. The advantages and disadvantages of this approach are discussed in Section 3.9.

While a complete description of the physics of this model of shear band formation is beyond the scope of the paper, a short summary follows. As described by Katz et al. [2006], the angle of melt bands in a deforming two-phase aggregate results from a balance between two modes of rheological weakening. When $n = 1$ in equation (9), the viscosity does not depend on strain rate. In this case, simulations show that melt bands weakened by the porosity dependence of viscosity emerge at 45° to the plane of shear. These bands grow fastest because they are perpendicular to the direction of maximum extension in simple shear. For a strain-rate-dependent viscosity ($n > 1$), however, concentrated shear deformation further weakens the bands, allowing them to more easily decompact under extension. Enhanced shear strain is largest for porosity bands oriented at zero and 90° and goes to zero for 45° bands (see Katz et al. [2006]). Thus two competing processes affect the preferred angle of melt bands. Linear analysis by Katz et al. [2006] suggests that the balance between favorable orientation for extension (45°) and favorable orientation for concentrating shear (0 and 90°) is controlled by the factor $(1 - n)/n$.

In simulations, the contrast in porosity between melt bands and the compacted regions between them grows with time. Concurrent with this localization of porosity is a localization of shear strain. For $n > 1$, strain becomes localized in bands that are roughly coincident with porosity bands in space. The combination of high porosity and enhanced shear over narrow regions produces sharp

25

gradients in viscosity. Figure 6 shows the evolution of the maximum viscosity gradient for simulations with different values of n and different initial porosity conditions. The increase of viscosity gradients within the domain is associated with a breakdown in convergence of the Newton solver and is sometimes accompanied by an increase in the condition number of the Jacobian matrix of the discretized elliptic equations. As shown by the terminal values along lines in Fig. 6, when the maximum dimensionless viscosity gradient reaches $O(10^3)$, the Newton solver fails to converge for the set of elliptic equations (15)–(17), although the linear solve for the correction δu typically continues to converge. At this point the simulation cannot be integrated further. No clear relationship exists between the maximum viscosity gradient and the condition number of the Jacobian matrix.

[Fig. 6 about here.]

3.9 Elliptic-Hyperbolic Solver Iteration

The set of coupled equations that describe magma/mantle interaction contains one hyperbolic, time-dependent equation (14). The other three equations, (15)–(17), represent instantaneous balances. One approach to handling the discrete versions of these equations is to combine them into a single system of nonlinear equations that can (ideally) be solved with a single call to PETSc's nonlinear solver. A more flexible and efficient method is to split the equations into smaller systems. In the time-dependent simulations described above, we have separated the hyperbolic equation from the others.

In practice, two issues arise with the single-solve approach. First, PETSc treats

26

all variables as unconstrained. This practice can lead to problems when solving for porosity, which is physically constrained to lie between zero and one. Even for constitutive equations that, in theory, prevent porosity from becoming negative, practice shows that a discrete version of the equations over a finite time-step can produce porosities slightly below zero. In this case, floating point exceptions caused by negative porosity can cause a simulation to fail. A second problem concerns simulation efficiency. The simplest magma-dynamics system has one evolution equation for porosity, but more complex systems may contain several (e.g., temperature, concentrations of various chemical species). These variables are governed by advection-diffusion-reaction equations that, when discretized, result in diagonally dominant Jacobian matrices. Such matrices are associated with linear systems that can be solved with about three iterations of GMRES, as opposed to the Jacobian of elliptic equations that requires many more. Combining parabolic and elliptic variables in a single system of equations leads to a Jacobian matrix in which some rows are diagonally dominant. Furthermore, because the computational work of GMRES scales as $O(N^2)$ where N is the number of unknowns, dividing the problem into parts can produce significant speedup.

Splitting the system of equations has costs, however. Of these, the most significant is the need to iterate between the solves at each timestep. Typically we iterate twice, although a more rigorous approach would be to check the residual of the *combined* system of equations after each iteration. There is also a cost in code complexity and in the number of communications between processors in parallel computations. The latter has an insignificant effect on overall code performance for fast cluster interconnection networks.

PETSc provides a set of tools for splitting the fields of unknowns with the

27

`Fieldsplit` preconditioner. This allows one to apply different linear solvers to different sets of fields. For example, consider a three component problem where the first two components are coupled elliptic equations. One can specify, on the command line when invoking the executable, `-pc_type fieldsplit -pc_fieldsplit_0_fields 0-1 -pc_fieldsplit_1_fields 2 -fieldsplit_0_pc_type boomeramg -fieldsplit_1_pc_type jacobi`. The first two fields are pre-conditioned as coupled system using the Hypre BoomAMG algebraic multigrid pre-conditioner (see `www.llnl.gov/CASC/linear_solvers`), while the third field is pre-conditioned with a simple Jacobi step. A slightly more powerful tool is the `PCComposite` preconditioner; this allows one to easily string together several preconditioners, where each is selected to handle particular parts of the solution space.

4   PETSc Special Features

4.1   Parameter Handling Using PetscBag

One characteristic common to many geodynamics models is the large number of physical parameters that must be included. These parameters are associated with material properties such as rheology, buoyancy, and phase change; boundary conditions; and control of the numerical solution. PETSc provides a convenient set of data structures and methods that facilitate management of these large sets of program parameters for code usability

Specifically the `PetscBag` object manages user parameters such that the actual data structure is opaque to the user. The user must declare a variable of type `PetscBag` and may then use functions to put parameters of any standard type (integer, real, Boolean, character strings, etc.) "into the bag". Example

28

code demonstrating this interface is shown in section 4.3. For each parameter the user calls `PetscBagRegisterXXX()`, replacing `XXX` with a suffix corresponding to the variable type. The purpose of registering a parameter is to simultaneously (i) set a default value, (ii) provide an identifier string to precede and identify the parameter when it is set upon program execution, and (iii) provide a descriptive "help" string that is printed with the variable's command-line identifier and default value when requested. This help string is useful to remind the user of the meaning of a parameter, including its units and type.

The `PetscBag` object may be written to a binary file with other PETSc objects for both output and the documentation of simulation input. When writing the `PetscBag` to a file, the application code need not specify the size or contents of the bag; as an application code evolves and new parameters are added, the input/output modules are unchanged.

4.2   Self-Documenting Output

As the code of a simulation evolves to incorporate new physical models, new constitutive equations, or new solution strategies, the output files generated by that code may also evolve in their structure and content. With each change, new scripts to load, post process, and visualize the output files would be needed, making it difficult to load old output files from earlier versions of a code. Self-documenting output is a way to avoid this difficulty. A common solution to this problem is provided by XML, which has a self-documenting structure. However, it is ill-suited to the large, structured data sets produced by PETSc simulations and is difficult to integrate into existing analysis or

29

visualization environments such as Matlab.

PETSc includes an output mode that combines the creation of a binary data file, which contains the `Vec`, `Mat`, and `PetscBag` data output by the user, with the generation of an ASCII descriptor file, which contains a script for reading the binary file into Matlab. This script, when executed in Matlab, creates a struct with fields containing the data from the associated binary file. The field names of this structure are specified in the simulation source code. These field names allow the user who loads the simulation output to understand the nature of the data, even if this user has no knowledge of the simulation source code. Code that demonstrates the use of self-documenting PETSc output is shown in Section 4.3.

The pair of files comprising of the binary output file and its plain text descriptor file is independent of any changes to the simulation code; hence there is no ambiguity about the content or structure of the binary file and no issue of backward compatibility. This integration of binary output with a descriptor file provides a self-contained method for quickly and easily reading simulation data.

4.3 Example Code

The following program, written in C, demonstrates the use of self-documenting output and the `PetscBag` object, as discussed in the preceding subsections. A struct, `Parameter`, is defined to contain the needed data. Then the bag is allocated and data registered, including default values, identifiers, and help strings. Once the simulation is completed, a viewer of type `PetscViewerBinaryMatlab`

30

is used to generate both the binary data file and descriptor file. These files can later be read into Matlab, using a script provided with the PETSc distribution, `PetscReadBinaryMatlab.m`. For function documentation and more specific usage information, see Balay et al. [2001].

---

static char help[] = "Demonstrates use of PetscBag and PetscViewerBinaryMatlab\n\n";

#include "petscbag.h"

#include "petscda.h"

/* *Define a C struct that will contain a bag's parameters.* */

typedef struct {

  char   filename[PETSC_MAX_PATH_LEN];

  PetscReal rho;

  PetscInt nx, ny;

} Parameter;

/* *Example program* */

int main(**int** argc,**char** **argv)

{

  DA   da;

  Vec   vec;

  PetscBag   bag;

  Parameter *params;

  PetscViewer viewer;

  /* *Initialize PETSc* */

  PetscInitialize(&argc, &argv, (**char** *)0, help);

31

```
/* Create a bag and register variables */

PetscBagCreate(PETSC_COMM_WORLD, sizeof(Parameter), &bag);

PetscBagGetData(bag, (void **) &params);

PetscBagSetName(bag, "ParameterBag", "demonstration");

PetscBagRegisterString(bag, &params->filename, PETSC_MAX_PATH_LEN,

                       "output", "filename", "Name of output file");

PetscBagRegisterReal(bag, &params->rho, 3.0, "rho", "Density, kg/m^3");

PetscBagRegisterInt(bag, &params->nx, 17, "nx", "number of points in x");

PetscBagRegisterInt(bag, &params->ny, 9, "ny", "number of points in y");


/* Create a DA and a global vector. Note that naming the field(s) is required! */

DACreate2d(PETSC_COMM_WORLD, DA_NONPERIODIC, DA_STENCIL_STAR,

           -params->nx, -params->ny, PETSC_DECIDE, PETSC_DECIDE,

           1, 1, 0, 0, &da);

DASetFieldName(da, 0, "rho");

DACreateGlobalVector(da, &vec);

VecSet(vec, params->rho);


/*

   Create the viewer, output a bag and vector to viewer, and close it.

   The resulting file can be read into Matlab using the provided script,

   PetscReadBinaryMatlab.m

*/

PetscViewerBinaryMatlabOpen(PETSC_COMM_WORLD, params->filename, &viewer);

PetscViewerBinaryMatlabOutputBag(viewer, "parameters", bag);

PetscViewerBinaryMatlabOutputVecDA(viewer, "fields", vec, da);


/* Clean up and exit */
```

32

```
    PetscViewerBinaryMatlabDestroy(viewer);

    PetscBagDestroy(bag);

    VecDestroy(vec);

    PetscFinalize();

    return 0;
}
```

5 Conclusions

5.1 Future of PETSc

The key concepts that organize PETSc's `DA` interface may be generalized to a treatment of unstructured meshes in arbitrary dimension. In this more general setting, however, one must clearly separate the various concerns. The PETSc `Mesh` object is based on the Sieve topology interface presented by Knepley and Karpeev [2005]. Under Sieve, geometry is described as merely another field over the mesh. Local discretization information can be provided by the FIAT system [Kirby, 2004], which is used to generate quadrature information for arbitrary finite elements. The local physics computations are still segregated from the global communication and numbering.

Moreover, under this model the local physics/global solver paradigm can be greatly extended. Finite elements themselves fit this model, as local approximations for restricted fields are then used to update a global field. Multigrid also obeys this model, with restriction and updating operating between grid resolutions. Structuring the algorithms and interface in this way allows global

33

data management and its attendant hardships (numbering, indexing, communication, etc.) to be divorced from the specificities of local computation (dimension, element shape, finite element order, etc.). Future versions of PETSc will incorporate such developments and hence facilitate the development of finite element simulations on domains with complex geometry.

5.2   Future of Geodynamics Simulations

Geochemical and petrological analyses of volcanically derived rocks provide a powerful constraint on the dynamics of magmatic systems. A wealth of such analyses exist and have been used to develop hypotheses about the distribution and style of melting beneath plate boundaries [e.g., Kelemen et al., 1997b, Langmuir et al., 2004, Kelley et al., 2006]. Tectonic-scale simulations of magma/mantle interaction can provide quantitative tests of these hypotheses and can generate insight into melt transport processes. In order to produce predictions testable by comparison to geochemical and petrologic data, simulations must generate self-consistent solutions for mantle flow, melting, and magmatic transport of mass, energy, and chemistry. The development of such models represents a major computational challenge.

Past simulations used to generate predictions of lava chemistry have typically simplified either the melt transport or the mantle flow field. van Keken et al. [2001] used parameterized melt transport to calculate helium degassing of the mantle in a global model of mantle convection with spherical geometry. Spiegelman and Kelemen [2003] calculated trace element signatures of reactive melt channels using detailed models of magma/mantle interaction that neglect large-scale deformation of the host rock. Spiegelman and Reynolds [1999] made

34

predictions of the spatial distribution of lava chemistry at mid-ocean ridges by solving for both solid mantle deformation and magmatic flow; however, this work neglected reactive melting, melt-lithosphere interaction, and the variable-viscosity of mantle rock. Choblet and Parmentier [2001] simulated 3D, variable viscosity mantle flow, melting, and melt transport beneath a mid-ocean ridge but did not calculate geochemical signatures of the flow. A simulation that self-consistently combines a current understanding of magma dynamics and large-scale mantle flow to make geochemical predictions is still lacking.

Some of the challenges of developing such a model stem from the difficulties associated with simulations such as those described in section 1.1. These simulations seek to isolate one part of the system, for example, solid mantle flow or two-phase flow but in a simple geometry with no melting. In both cases, non-Newtonian viscosity results in strong nonlinearity, sharp viscosity contrasts, and, ultimately, near-singular systems of linear equations that are not easily solved. In models of magma dynamics that include shear [Katz et al., 2006] and reactive flow [Aharonov et al., 1997, Spiegelman et al., 2001], localization of porosity into small-scale features contributes to the difficulty of the problem.

Additional complexities arise when mantle flow simulation is combined with magma dynamics. Localization leads to a hierarchy of length scales from the tectonic scale of ∼100 km to smaller than the compaction scale (the relevant length scale for magma localization) of ∼1 km. Resolving the compaction scale in a domain that is hundreds of kilometers on each side requires significant computational power. Rheological and thermal interaction between the liquid and solid phases in such simulations means that the two flow fields cannot be decoupled and solved separately. A self-consistent set of fluid dynamical

35

equations such as those given in Section 1.1, plus a set of equations governing the evolution of chemistry [Aharonov et al., 1997] and temperature in the two-phase region, is required. These equations should handle both partially melted ($\phi > 0$) and melt-free ($\phi = 0$) zones. Together these requirements represent a significant challenge to computational scientists and geodynamicists.

These challenges are being addressed by the Computational Infrastructure for Geodynamics (www.geodynamics.org). CIG is an NSF-sponsored partnership between computational scientists and solid-earth scientists to develop the next generation of modeling software for the geodynamics community for investigating a wide range of geodynamics problems, including magma migration, mantle convection, short and long-term lithospheric deformation, computational seismology, and the geodynamo. The long-term goal for CIG is to develop an interoperable software suite for investigating a range of multiphysics problems in geodynamics. An important design criterion for all new software, however is to leverage as much as possible from existing high-quality computational software projects. In particular PETSc and the affiliation with Argonne is a fundamental component for solver technologies in both the new earthquake physics codes PyLith and a Lithospheric Deformation code GALE and will continue to be the principal development platform for magma-dynamics problems.

We have demonstrated how PETSc, a leading example of an advanced scientific computation library, can be used to facilitate simulation of complex non-linear systems with localization instabilities and large memory requirements. Advanced scientific computation libraries will become increasingly important to the success of magma-dynamics simulation as researchers tackle problems that connect fluid-dynamical processes occurring at depth in the mantle with

36

geochemical and petrologic observations made at the surface. Such models will provide powerful tools for interpreting chemical data and for developing an understanding of volcanic source regions.

<center>37</center>

References

E. Aharonov, M. Spiegelman, and P. Kelemen. Three-dimensional flow and reaction in porous media: Implications for the Earth's mantle and sedimentary basins. *J. Geophys. Res.*, 102:14821–14833, 1997.

M. Albers. A local mesh refinement multigrid method for 3-D convection problems with strongly variable viscosity. *J. Comput. Phys.*, 160:126–150, 2000.

S. Balay, K. Buschelman, W. Gropp, D. Kaushik, M. Knepley, L. McInnes, B. Smith, and H. Zhang. PETSc home page. http://www.mcs.anl.gov/petsc, 2001.

S. Balay, K. Buschelman, V. Eijkhout, W. D. Gropp, D. Kaushik, M. G. Knepley, L. C. McInnes, B. F. Smith, and H. Zhang. PETSc users manual. Technical Report ANL-95/11 - Revision 2.1.5, Argonne National Laboratory, 2004.

V. Barcilon and O. Lovera. Solitary waves in magma dynamics. *J. Fluid Mech.*, 204:121–133, 1989.

V. Barcilon and F. M. Richter. Non-linear waves in compacting media. *J. Fluid Mech.*, 164:429–448, 1986.

G. K. Batchelor. *An Introduction to Fluid Dynamics*. Cambridge University Press, Cambridge, 1967.

W. Buck and W. Su. Focused mantle upwelling below mid-ocean ridges due to feedback between viscosity and melting. *Geophys. Res. Letts.*, 16(7): 641–644, 1989.

X. Cai, D. Keyes, and V. Venkatakrishnan. Newton-Krylov-Schwarz: An implicit solver for CFD. In *Proceedings of the Eighth International Conference on Domain Decomposition Methods*, pages 387–400, 1997.

38

S. Carbotte, C. Small, and K. Donnelly. The influence of ridge migration on the magmatic segmentation of mid-ocean ridges. *Nature*, 429:743–746, 2004.

G. Choblet and E. Parmentier. Mantle upwelling and melting beneath slow spreading centers: Effects of realistic rheology and melt productivity. *Earth Planet. Sci. Lett.*, 184:589–604, 2001.

W. Crawford and S. Webb. Variations in the distribution of magma in the lower crust and at the Moho beneath the East Pacific Rise at 9 degrees-10 degrees N. *Earth Plan. Sci. Lett.*, 203(1):117–130, 2002.

J. W. Demmel. *Applied Numerical Linear Algebra*. SIAM, 1997.

J. Georgen and J. Lin. Three-dimensional passive flow and temperature structure beneath oceanic ridge-ridge-ridge triple junctions. *Earth Plan. Sci. Lett.*, 204(1–2):115–132, 2002.

T. Gerya and D. Yuen. Rayleigh-taylor instabilities from hydration and melting propel 'cold plumes' at subduction zones. *Earth Plan. Sci. Lett.*, 212 (1–2):47–62, 2003.

G. Hirth and D. L. Kohlstedt. Experimental constraints on the dynamics of the partially molten upper mantle: Deformation in the diffusion creep regime. *J. Geophys. Res.*, 100(2):1981–2002, Febuary 1995a.

G. Hirth and D. L. Kohlstedt. Experimental constraints on the dynamics of the partially molten upper mantle 2. deformation in the dislocation creep regime. *J. Geophys. Res.*, 100(8):15441–15052, August 1995b.

B. K. Holtzman, N. J. Groebner, M. E. Zimmerman, S. B. Ginsberg, and D. L. Kohlstedt. Stress-driven melt segregation in partially molten rocks. *Geochem. Geophys. Geosyst.*, 4, 2003a. Art. No. 8607.

B. K. Holtzman, D. L. Kohlstedt, M. Zimmerman, F. Heidelbach, T. Hiraga, and J. Hustoft. Melt segregation and strain partitioning: Implications for seismic anisotropy and mantle flow. *Science*, 301(5637):1227–1230, 2003b.

39

S. Karato and P. Wu. Rheology of the upper mantle: A synthesis. *Science*, 260(5109):771–778, 1993.

R. Katz, M. Spiegelman, and C. Langmuir. A new parameterization of hydrous mantle melting. *Geochem. Geophys. Geosys.*, 4(9), 2003.

R. Katz, M. Spiegelman, and S. Carbotte. Ridge migration, asthenospheric flow and the origin of magmatic segmentation in the global mid-ocean ridge system. *Geophys. Res. Letts.*, 31:L15605, 2004.

R. Katz, M. Spiegelman, and B. Holtzman. The dynamics of melt and shear localization in partially molten aggregates. *Nature*, 442:676–679, 2006. doi:10.1038/nature05039.

P. Kelemen, G. Hirth, N. Shimizu, M. Spiegelman, and H. Dick. A review of melt migration processes in the adiabatically upwelling mantle beneath oceanic spreading ridges. *Phil. Trans. R. Soc. London A*, 355(1723):283–318, 1997a.

P. Kelemen, G. Hirth, N. Shimizu, M. Spiegelman, and H. Dick. A review of melt migration processes in the adiabatically upwelling mantle beneath oceanic spreading ridges. *Phil. Trans. R. Soc. London A*, 355(1723):283–318, 1997b.

P. Kelemen, M. Parmentier, J. Rilling, L. Mehl, and B. Hacker. Thermal convection of the mantle wedge. *Geochim. Cosmochim. Acta*, 66(15A):A389–A389, 2002.

K. Kelley, T. Plank, T. Grove, E. Stolper, S. Newman, and E. Hauri. Mantle melting as a function of water content beneath back-arc basins. *J. Geophys. Res.*, 111(B9), 2006.

R. C. Kirby. Algorithm 839: Fiat a new paradigm for computing finite element basis functions. *ACM Trans. Math. Software*, 30(4):502–516, 2004.

M. Knepley, R. Katz, and B. Smith. Developing a geodynamics simulator with

40

PETSc. In A. Bruaset and A. Tveito, editors, *Numerical Solution of Partial Differential Equations on Parallel Computers*, volume 51 of *Lecture Notes in Computational Science and Engineering*, pages 413–438. Springer-Verlag, 2006.

M. G. Knepley and D. A. Karpeev. Flexible representation of computational meshes. Preprint ANL/MCS-P1295-1005, Argonne National Laboratory, October 2005. URL `ftp://info.mcs.anl.gov/pub/tech_reports/reports/P1295.pdf`.

C. Langmuir, P. Michael, K. Lehnert, S. Goldstein, G. Soffer, E. Gier, and J. Snow. Constraints from the Gakkel ridge on the origin of MORB. *Geochim. Cosmochim. Acta*, 68(11):A690–A690, 2004.

J. S. M. Rabinowicz, S. Rouzo and C. Rosemberg. Three-dimensional mantle flow beneath mid-ocean ridges. *J. Geophys. Res.*, 98:7851–7869, 1993.

L. Magde and D. W. Sparks. Three-dimensional mantle upwelling, melt generation and melt migration beneath segmented slow-spreading ridges. *J. Geophys. Res.*, 102:20571–20583, 1997.

D. McKenzie. Speculations on consequences and causes of plate motions. *Geophys. J.*, 18:1–32, 1969.

D. McKenzie. The generation and compaction of partially molten rock. *J. Petrol.*, 25:713–765, 1984.

S. Mehl. Use of Picard and Newton iteration for solving nonlinear ground water flow equations. *Ground Water*, 44(4):583–594, 2006.

S. Mei, W. Bai, T. Hiraga, and D. Kohlstedt. Influence of melt on the creep behavior of olivine-basalt aggregates under hydrous conditions. *Earth Planet. Sci. Lett.*, 201:491–507, 2002.

Message Passing Interface Forum. MPI: A Message-Passing Interface standard. *International Journal of Supercomputer Applications*, 8(3/4):165–414,

41

1994.

Message Passing Interface Forum. MPI2: A Message Passing Interface standard. *International Journal of High Performance Computing Applications*, 12(1–2):1–299, 1998.

C. Paniconi and M. Putti. A comparison of Picard and Newton iteration in the numerical solution of multidimenional variably saturated flow problems. *Water Resources Res.*, 30(12):3357–3374, 1994.

S. Patankar. *Numerical Heat Transfer and Fluid Flow*. Hemisphere Publishing Corp., 1980.

J. Phipps Morgan and D. Forsyth. 3-d flow and temperature perturbations due to transform offset: effects on oceanic crustal and upper mantle structure. *J. Geophys. Res.*, 93:2955–2966, 1988.

F. M. Richter and D. McKenzie. Dynamical models for melt segregation from a deformable matrix. *J. Geol.*, 92:729–740, 1984.

D. R. Scott and D. Stevenson. Magma solitons. *Geophys. Res. Lett.*, 11: 1161–1164, 1984.

D. R. Scott and D. Stevenson. Magma ascent by porous flow. *J. Geophys. Res.*, 91:9283–9296, 1986.

D. R. Scott, D. Stevenson, and J. Whitehead. Observations of solitary waves in a deformable pipe. *Nature*, 319:759–761, 1986.

Y. Shen and D. Forsyth. The effects of temperature- and pressure-dependent viscosity on three-dimensional passive flow of the mantle beneath a ridge-transform system. *Geophys. J. Int.*, 97:19717–19728, 1992.

D. Sparks and E. Parmentier. Melt extraction from the mantle beneath spreading centers. *Earth Planet. Sci. Lett.*, 105(4):368–377, 1991.

D. Sparks and E. Parmentier. The structure of three-dimensional convection beneath oceanic spreading centers. *Geophys. J. Int.*, 112:81–91, 1993.

42

M. Spiegelman. Linear analysis of melt band formation by simple shear. *Geochem. Geophys. Geosyst.*, 4(9), 2003. Article 8615, doi:10.1029/2002GC000499.

M. Spiegelman. Flow in deformable porous media, part 1. Simple analysis. *J. Fluid Mech.*, 247:17–38, 1993a.

M. Spiegelman. Flow in deformable porous media. part 2. Numerical analysis—The relationship between shock waves and solitary waves. *J. Fluid Mech.*, 247:39–63, 1993b.

M. Spiegelman and P. B. Kelemen. Extreme chemical variability as a consequence of channelized melt transport. *Geochem. Geophys. Geosyst.*, 4(7), 2003. Article 1055, doi:10.1029/2002GC000336.

M. Spiegelman and D. McKenzie. Simple 2-D models for melt extraction at mid-ocean ridges and island arcs. *Earth Planet. Sci. Lett.*, 83:137–152, 1987.

M. Spiegelman and J. Reynolds. Combined theoretical and observational evidence for convergent melt flow beneath the EPR. *Nature*, 402:282–285, 1999.

M. Spiegelman, P. B. Kelemen, and E. Aharonov. Causes and consequences of flow organization during melt transport: The reaction infiltration instability in compactible media. *J. Geophys. Res.*, 106(B2):2061–2077, 2001. www.ldeo.columbia.edu/~mspieg/SolFlow/.

T. Steihaug. The conjugate gradient method and trust regions in large scale optimization. *SIAM J. Numer. Anal.*, 20:626–637, 1983.

D. J. Stevenson. Spontaneous small-scale melt segregation in partial melts undergoing deformation. *Geophys. Res. Lett.*, 16(9):1067–1070, 1989.

R. Trompert and U. Hansen. The application of a finite volume multigrid method to 3d flow problems in a highly viscous fluid with variable viscosity. *Geophys. Astrophys. Fluid Dyn.*, 83:261, 1996.

43

P. van Keken, C. Ballentine, and D. Porcelli. A dynamical investigation of the heat and helium imbalance. *Earth Planet. Sci. Lett.*, 188:421–434, 2001.

P. van Keken, B. Kiefer, and S. Peacock. High-resolution models of subduction zones: implications for mineral dehydration reactions and the transport of water into the deep mantle. *Geochem. Geophys. Geosys.*, 3, 2002.

C. Wiggins and M. Spiegelman. Magma migration and magmatic solitary waves in 3-D. *Geophys. Res. Lett.*, 22(10):1289–1292, May 15 1995.

M. E. Zimmerman, S. Q. Zhang, D. L. Kohlstedt, and S. Karato. Melt distribution in mantle rocks deformed in shear. *Geophys. Res. Lett.*, 26(10): 1505–1508, 1999.

44

Copyright transfer   The submitted manuscript has been created by UChicago Argonne, LLC, operator of Argonne National Laboratory ("Argonne") under Contract No. DE-AC02-06CH11357 with the U.S. Department of Energy. The U.S. Government retains for itself, and others acting on its behalf, a paid-up, nonexclusive, irrevocable worldwide license in said article to reproduce, prepare derivative works, distribute copies to the public, and perform publicly and display publicly, by or on behalf of the Government.

45

List of Figures

46

6   The maximum gradient in viscosity as a function of strain. At each timestep the magnitude of the viscosity gradient is calculated at each point in the domain as $||\boldsymbol{\nabla}\eta|| = \sqrt{\left(\frac{\partial \eta}{\partial x}\right)^2 + \left(\frac{\partial \eta}{\partial z}\right)^2}$. Values on the y-axis of this plot are determined by taking the maximum value of $||\boldsymbol{\nabla}\eta||$ over the domain at each step in strain. Line colors represent the value of n used in each simulation, as given in the plot legend. Solid lines represent simulations in which the initial porosity perturbation is white noise on the grid with amplitude 0.1%. Dashed lines represent simulations initiated with a porosity field perturbed by noise with an amplitude spectrum that decreases with $|\tilde{k}|^{-0.9}$, where $\tilde{k}$ is a two-dimensional wavevector. For simplicity, in these simulations we have used a constant bulk viscosity, $\zeta_0 = 10\eta_0$.   53
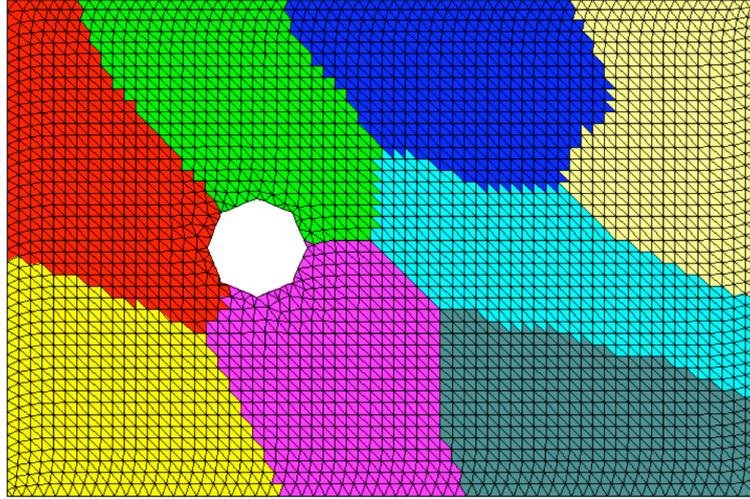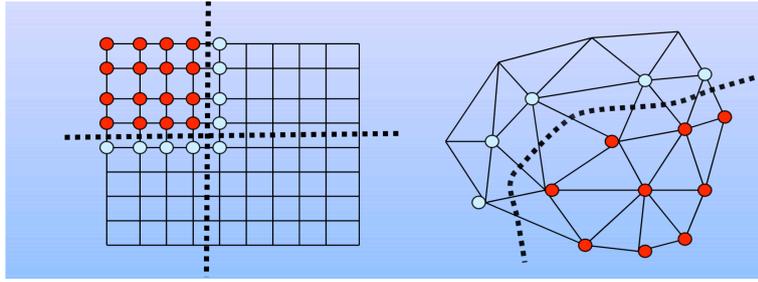
47

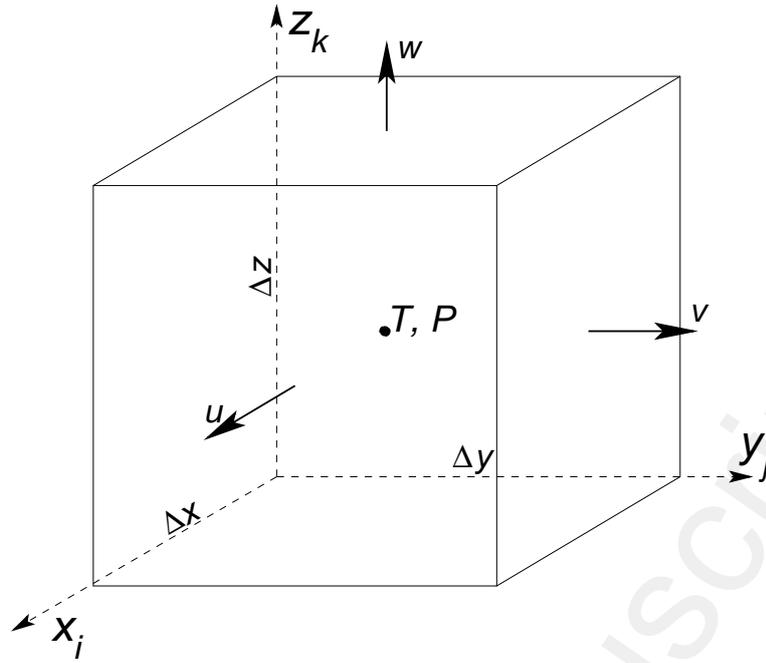Fig. 1. Partition of mesh across processors.

48

Fig. 2. Examples of Ghost Points.

49

Fig. 3. Control volume Ω_{ijk}, labeled to illustrate the position of variables on the staggered mesh. Adapted from Albers [2000].
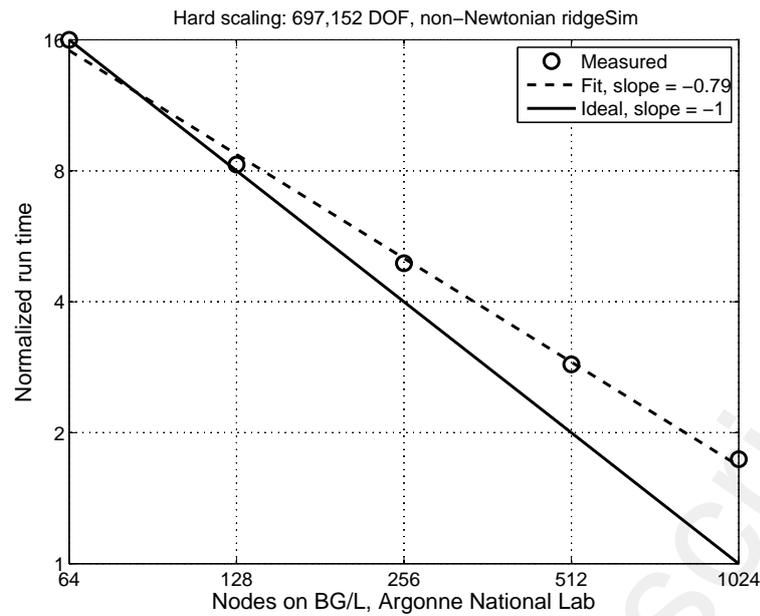
50

Fig. 4. Normalized run times as a function of the number of processors for solution of the 3D, single-phase, steady-state flow and thermal structure problem of a mid-ocean ridge. The problem size is held constant as the number of processors increases. (The grid spacing is 2.5 km in each direction for a total of $\sim 7 \times 10^5$ degrees of freedom.) Ideally, doubling the number of processors would reduce the run time by a factor of one-half; the data show that this application achieves about 80% of ideal scaling. The calculations were performed on an IBM Blue Gene/L system with 1024 dual PowerPC compute nodes. Code was compiled with IBM's BG/L-optimized C compiler. More information on this system is available at `www.bgl.mcs.anl.gov`.
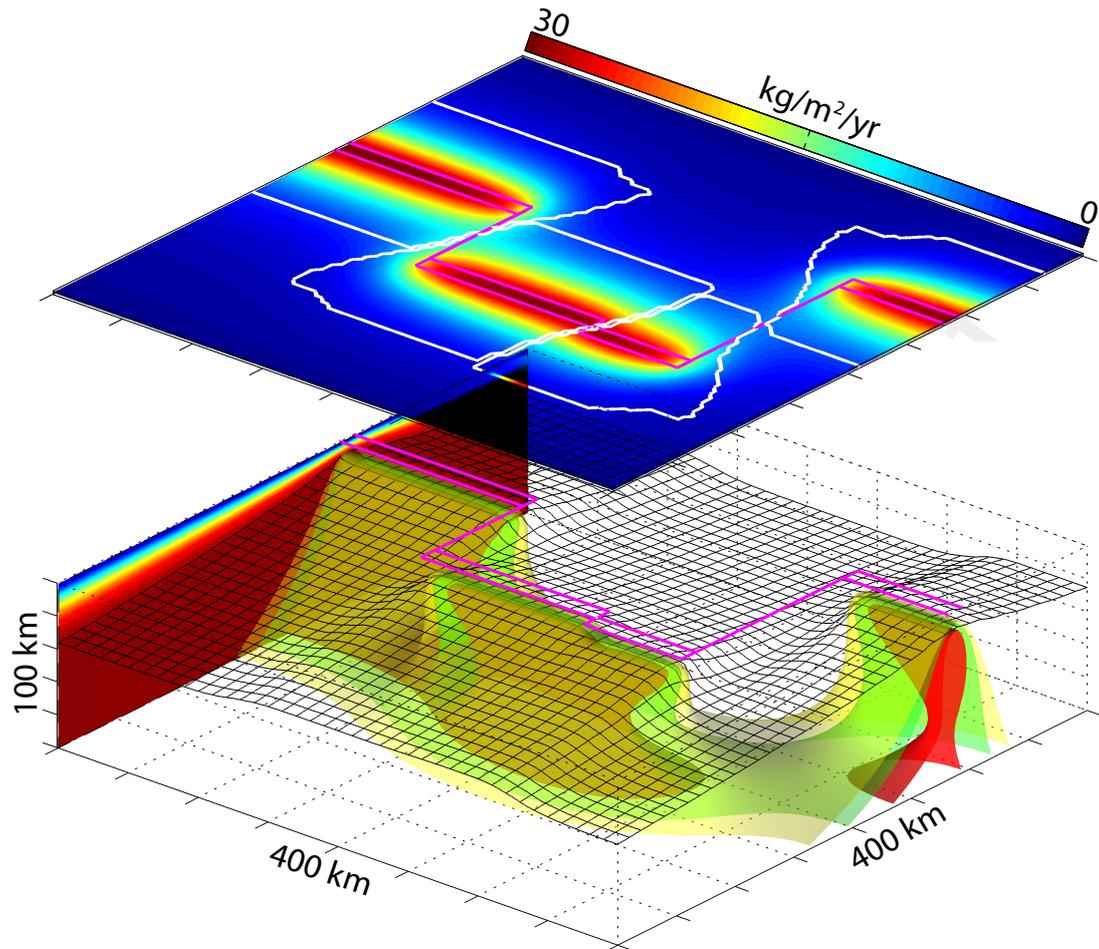
51

Fig. 5. Output from a mid-ocean ridge simulation with a full spreading rate of 10.8 cm/yr. The pattern of ridge segments and transform faults, shown by the magenta lines, approximates the geometry of the Eastern Pacific Rise around the Siqueiros and Clipperton transform faults at about 9° N. **(a)** is a three-dimensional representation of upwelling. The transparent, colored isosurfaces connect points of constant upwelling rate. Yellow, green, and red surfaces denote the 3.0, 3.5, and 4.05 cm/yr isosurfaces, respectively. A 2D slice through the potential temperature field of the model is shown on the back-left panel of the graph. On this panel, blue is 0 °C and red is 1300 °C, the assigned potential temperature of the deep mantle. The base of the thermal boundary layer is the depth at which the potential temperature reaches 1300 °C. This depth is shown, over the interior of the 3D domain, by the mesh surface. **(b)** shows the vertically integrated melting rate in color. Regions over which melt is focused to each ridge are bounded by white lines. The ridge trace is shown in magenta. Calculations performed to determine the generation and focusing of melt are described in the main text.
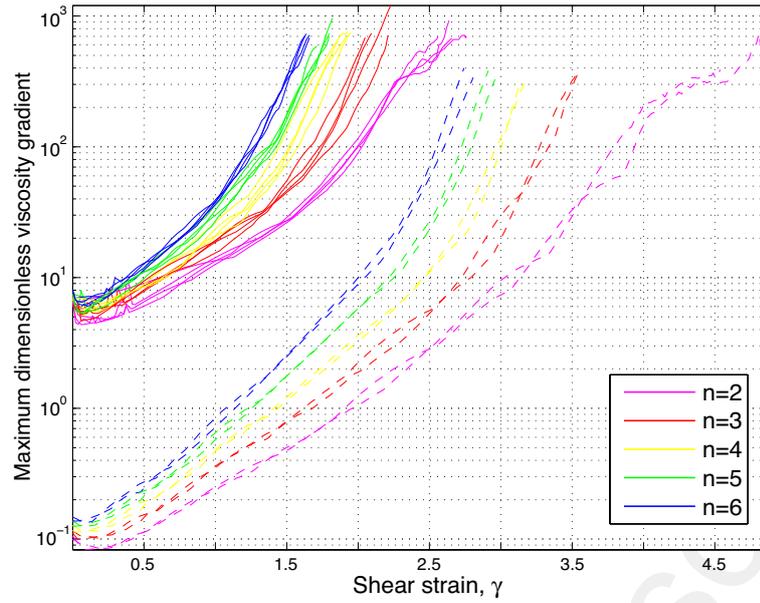
52

Fig. 6. The maximum gradient in viscosity as a function of strain. At each timestep the magnitude of the viscosity gradient is calculated at each point in the domain as $||\boldsymbol{\nabla}\eta|| = \sqrt{\left(\frac{\partial\eta}{\partial x}\right)^2 + \left(\frac{\partial\eta}{\partial z}\right)^2}$. Values on the y-axis of this plot are determined by taking the maximum value of $||\boldsymbol{\nabla}\eta||$ over the domain at each step in strain. Line colors represent the value of n used in each simulation, as given in the plot legend. Solid lines represent simulations in which the initial porosity perturbation is white noise on the grid with amplitude 0.1%. Dashed lines represent simulations initiated with a porosity field perturbed by noise with an amplitude spectrum that decreases with $|\tilde{k}|^{-0.9}$, where $\tilde{k}$ is a two-dimensional wavevector. For simplicity, in these simulations we have used a constant bulk viscosity, $\zeta_0 = 10\eta_0$.

53

List of Tables

54

Boundary	Variable	Boundary Condition
$z = 0$	u	$u(x, y, 0) = U(x, y)$ where U is the imposed plate motion.
	v	$v = 0$ on top boundary
	w	$w = 0$ on top boundary
	P	This BC is irrelevant for the interior of domain
	T	dimensionless $T = 0$ on top boundary
$z = D$	u	$u = 0$ on bottom boundary.
	v	$v = 0$ on bottom boundary
	w	$dw/dz = 0$ on bottom boundary
	P	$P = 0$ on bottom boundary
	T	dimensionless T equals the (dimensionless) mantle potential temperature
$x = 0, L$	u	u satisfies equation (10).
	v	$\sigma_{xy} = 0$
	w	$\sigma_{xz} = 0$
	P	This BC is irrelevant for the interior of the domain
	T	$dT/dx = 0$
$y = 0, W$	all	von Neumann (reflection) condition

Table 1

Boundary conditions for the 3-D ridge simulation. The domain is size $x \in [0, L]$, $y \in [0, W]$, $z \in [0, D]$. The spreading ridge is parallel to the y-axis, transform faults, and the spreading rate vector parallel to the x-axis.

55