Geochemistry Geophysics Geosystems G3

Published by AGU and the Geochemical Society

Article
Volume 12, Number 6
8 June 2011
Q06001, doi:10.1029/2011GC003551
ISSN: 1525-2027

Fluidity: A fully unstructured anisotropic adaptive mesh computational modeling framework for geodynamics

D. Rhodri Davies
Department of Earth Science and Engineering, Imperial College London, London, SW7 2AZ, UK (Rhodri.Davies@imperial.ac.uk)

Cian R. Wilson
Institute of Shock Physics and Department of Earth Science and Engineering, Imperial College London, London, SW7 2AZ, UK

Now at Lamont-Doherty Earth Observatory, Columbia University, Palisades, New York 10964, USA

Stephan C. Kramer
Institute of Shock Physics and Department of Earth Science and Engineering, Imperial College London, London, SW7 2AZ, UK

[1] We present a new computational modeling framework, Fluidity, for application to a range of two- and three-dimensional geodynamic problems, with the focus here on mantle convection. The approach centers upon a finite element discretization on unstructured simplex meshes, which represent complex geometries in a straightforward manner. Throughout a simulation, the mesh is dynamically adapted to optimize the representation of evolving solution structures. The adaptive algorithm makes use of anisotropic measures of solution complexity, to vary resolution and allow long, thin elements to align with features such as boundary layers. The modeling framework presented differs from the majority of current mantle convection codes, which are typically based upon fixed structured grids. This necessitates a thorough and detailed validation, which is a focus of this paper. Benchmark comparisons are undertaken with a range of two- and three-dimensional, isoviscous and variable viscosity cases. In addition, model predictions are compared to experimental results. Such comparisons highlight not only the robustness and accuracy of Fluidity but also the advantages of anisotropic adaptive unstructured meshes, significantly reducing computational requirements when compared to a fixed mesh simulation.

Components: 9800 words, 9 figures, 3 tables.

Keywords: mesh adaptivity; geodynamics; mantle convection; finite element methods; benchmark; model validation and verification.

Index Terms: 0545 Computational Geophysics: Modeling (1952, 4255, 4316); 1213 Geodesy and Gravity: Earth's interior: dynamics (1507, 7207, 7208, 8115, 8120); 8120 Tectonophysics: Dynamics of lithosphere and mantle: general (1213).

Received 7 February 2011; **Revised** 18 March 2011; **Accepted** 4 April 2011; **Published** 8 June 2011.

Davies, D. R., C. R. Wilson, and S. C. Kramer (2011), Fluidity: A fully unstructured anisotropic adaptive mesh computational modeling framework for geodynamics, *Geochem. Geophys. Geosyst.*, *12*, Q06001, doi:10.1029/2011GC003551.

Copyright 2011 by the American Geophysical Union

1. Introduction

[2] Mantle convection can be approximated as a fluid dynamical process, involving a number of complicated physical effects [*Turcotte and Oxburgh*, 1967; *McKenzie et al.*, 1974]. These include the following: (1) strain localization and brittle failure at the surface; (2) exothermic and endothermic mineralogical phase changes; (3) potential chemical stratification/lateral heterogeneity; (4) a complex rheology that depends strongly on pressure, temperature, composition, stress, grain size, volatile content, and material phase; and (5) both internal heating from radioactivity and substantial basal heating from the core [e.g., *Bunge et al.*, 1997]. Several of these effects are becoming well constrained through experimental procedures, observational data and numerical simulation. However, others remain poorly understood. This is largely because realistic simulations of global mantle convection, incorporating all necessary dynamical complexities, are often unfeasible, due to limitations in numerical methods and computational resources. For example, excluding a few notable exceptions [e.g., *Stadler et al.*, 2010; *Alisic et al.*, 2011], global mantle convection models do not fully utilize constraints on mantle and plate boundary rheology, as numerical simulation at the required resolution is not possible with most current methods [e.g., *Zhong et al.*, 2000; *Tackley*, 2008; *Davies and Davies*, 2009]. Similar problems occur with global adjoint/data assimilation techniques [*Bunge et al.*, 2003]: detailed regional geological constraints are often ignored as the underlying computational grid is too coarse to reliably assimilate the information. As a result, most adjoint studies, thus far, have utilized high-resolution regional models, which have limitations [e.g., *Liu et al.*, 2008]. The examples presented highlight two common themes within the geodynamical modeling community: (1) the need to resolve a range of length and time scales, within a single model, and (2) the failure of most current solution methods at doing so.

[3] Adaptive mesh refinement methods, where the underlying computational grid is modified as the simulation evolves, provide a means to address these issues. Their applicability to geodynamics has already been examined: in 2-D, *Davies et al.* [2007, 2008] show that unstructured adaptive mesh refinement methods significantly improve the efficiency of thermal and thermochemical mantle convection simulations. *Burstedde et al.* [2008] extend this work to 3-D petascale parallel domains, via structured adaptive mesh refinement, thus allowing for global-scale mantle convection simulations with a local resolution of 1 km [*Stadler et al.*, 2010; *Alisic et al.*, 2011]. Although built upon different foundations (unstructured versus structured discretizations), the success of these studies indicates that adaptive mesh refinement methods have a fundamental role to play in future geodynamical simulations. However, it should be noted that such methods are only advantageous in simulations where (1) regions of dynamic significance are of limited spatial extent and (2) the location and/or extent of these regions varies in time. In simulations requiring high-resolution globally, fixed uniform meshes are often the most suitable. Alternatively, if the location of dynamically significant regions is known a priori and such regions are of limited spatial extent, fixed nonuniform meshes often suffice [e.g., *Syracuse et al.*, 2010].

[4] In this paper, we present Fluidity, a new computational framework for geodynamical modeling, principally developed by the Applied Modelling and Computation Group at Imperial College London [e.g., *Pain et al.*, 2001; *Piggott et al.*, 2008]. The code has several state-of-the-art features that offer significant benefits for geodynamical simulations: (1) it uses an unstructured mesh, which enables straightforward multiresolution representation of complex geometry domains; (2) it dynamically optimizes this mesh, providing increased resolution in areas of dynamic importance, thus allowing for accurate simulations, across a range of length scales, within a single model. Mesh optimization is enhanced using anisotropic elements; (3) it utilizes implicit solvers, which allow for large time steps (advection at Courant numbers greater than one) with minimal loss of accuracy. These attributes allow Fluidity to simulate mantle convection accurately and efficiently.

[5] The goals of this paper are threefold: (1) to introduce Fluidity, and its unique numerical capabilities, to the community, (2) to validate the code against known benchmark and laboratory solutions, and (3) to further demonstrate the applicability of mesh adaptivity for geodynamical flows. Although Fluidity is optimized to run on parallel architectures and performs parallel mesh adaptivity (the subdomains used in parallel computing automatically adjust themselves to balance the computational load on each processor, as the mesh evolves [*Catalyurek et al.*, 2007; *Piggott et al.*, 2008; *Gorman et al.*, 2009]), we focus here upon serial simulations. The structure of the paper is as follows: we begin by presenting an overview of Fluidity, covering the discretizations and solution strategies employed within the code, in addition to

the mesh optimization methodologies. This is followed by results from 2-D [*Blankenbach et al.*, 1989] and 3-D [*Busse et al.*, 1994] benchmark problems. The new computational framework is subsequently applied to simulations of laboratory generated plumes [*Vatteville et al.*, 2009]. In addition to providing a thorough validation of the code, which complements standard benchmark tests, these final simulations demonstrate the significant benefits of adaptivity for simulating mantle convection and geodynamical flows in general.

2. Fluidity: Equations, Discretization, and Solution Strategies

2.1. Governing Equations

[6] The equations governing mantle convection are derived from conservation laws of mass, momentum and energy. The simplest mathematical formulation assumes incompressibility and the Boussinesq approximation [*McKenzie et al.*, 1974]. Under this formulation, the nondimensional momentum and continuity equations are

$$\sum_j \partial_j \sigma_{ij} + Ra_0 T \hat{k}_i = 0, \tag{1}$$

$$\sum_j \partial_j u_j = 0, \tag{2}$$

where u_i, σ_{ij} and T are the velocity, stress and temperature, respectively, $\hat{k}_i$ is the unit vector in the direction opposite to gravity and Ra_0 denotes the dimensionless Rayleigh number, which quantifies the vigor of convection:

$$Ra_0 = \frac{\rho_0 \alpha \Delta T g d^3}{\mu_0 \kappa}. \tag{3}$$

Here, ρ_0 denotes reference density, α is the thermal expansion coefficient, ΔT is the characteristic temperature change across the fluid layer, g is the gravitational acceleration, d is the characteristic length, μ_0 is the reference dynamic viscosity and κ is the thermal diffusivity. Note that the above nondimensional equations are obtained from the following characteristic scales: length d; time d^2/κ; and temperature ΔT.

[7] When simulating incompressible flow, it is convenient to decompose the full stress tensor, σ_{ij}, into deviatoric and volumetric components, according to

$$\sigma_{ij} = \tau_{ij} - p\delta_{ij}, \tag{4}$$

where τ_{ij} is the deviatoric stress tensor, p is dynamic pressure and δ_{ij} is the Kronecker delta function. Substituting (4) into (1) and utilizing the following constituative relation, which relates the deviatoric stress tensor, τ_{ij}, to the strain rate tensor, $\dot{\varepsilon}_{ij}$:

$$\tau_{ij} = 2\mu\dot{\varepsilon}_{ij} = \mu\left(\partial_j u_i + \partial_i u_j\right) \tag{5}$$

yields

$$\sum_j \partial_j \left[\mu\left(\partial_j u_i + \partial_i u_j\right)\right] - \partial_i p + Ra_0 T \hat{k}_i = 0. \tag{6}$$

The viscous flow problem can therefore be posed in terms of pressure, p, velocity, u_i, and temperature, T. The evolution of the thermal field is controlled by the following advection-diffusion equation:

$$\frac{\partial T}{\partial t} + \sum_i u_i \partial_i T = \sum_i \partial_i (\kappa \partial_i T). \tag{7}$$

These governing equations are sufficient to solve for the three unknowns, together with adequate boundary and initial conditions.

2.2. Discretization Schemes

[8] Analytical solutions to this coupled system of equations are generally unavailable, except in simple scenarios. However, their complexity can be reduced by discretizing the equations and approximating their solutions on a mesh of points. We focus here on finite element discretization methods, which form the heart of Fluidity and are ideally suited to unstructured meshes.

2.2.1. Finite Element Discretization of the Stokes Equations

[9] The Galerkin finite element discretization, applied to (6) and (2) starts from the weak form of the equations. We choose a collection of functions, M, to test the continuity equation, and vector valued functions, N_i, to test the momentum equation, and require that

$$\int \sum_{i,j} \left[\left(\partial_j N_i\right)\mu\left(\partial_j u_i + \partial_i u_j\right) + N_i \partial_i p\right] = \int \sum_i N_i Ra_0 T \hat{k}_i, \tag{8}$$

$$\int \sum_j \left(\partial_j M\right) u_j = 0, \tag{9}$$

for all test functions N_i and M. Note that we have integrated by parts the viscosity term in the

momentum equation and the divergence term in the continuity equation, but left out the boundary terms

$$\int_{\partial\Omega} \sum_{i,j} N_i \mu (\partial_j u_i + \partial_i u_j) n_j \tag{10}$$

$$\text{and} \quad \int_{\partial\Omega} \sum_{j} M u_j n_j, \tag{11}$$

respectively. Here, n_j is the outward pointing normal to the boundary $\partial\Omega$. These boundary terms are dealt with in combination with the boundary conditions.

[10] For no-slip boundary conditions all velocity components, u_i, are strongly enforced to be zero. This means that all test and trial functions with nonzero components at the boundary are eliminated from the test and trial space and, hence, both (10) and (11) disappear. For free-slip conditions combined with a no-normal flow condition, this is done only in the normal direction and (11) disappears. However, tangential velocity components are left free in both test and trial space. In this case, the no-normal stress condition, $\sum_j n_j \sigma_{ij} = 0$, is substituted in (10) and again the integral disappears.

[11] Finite element discretization then proceeds by restricting the solution fields, u_i and p, to a finite-dimensional function space, the *trial space*. Trial spaces are constructed by generating a mesh, thus dividing the domain into a set of polygonal elements, and restricting the functions to be polynomials of a certain degree within each element. Fluidity offers a range of element pairs. However, in this paper, we exclusively use the P2-P1 discretization, which consists of piecewise quadratic functions (P2) for velocity and piecewise linear functions (P1) for pressure. The P1 basis is denoted by M_a, where a is the index of the associated vertex, while the basis of P2 is denoted by N_b, where b ranges over all quadratic nodes in an element. The discrete velocity and pressure solutions can then be written as linear combinations of N_b and M_a, respectively:

$$u_i = \sum_b u_{b_i} N_b \quad \text{and} \quad p = \sum_a p_a M_a, \tag{12}$$

where the coefficients u_{bi} and p_a represent values of these functions at the associated nodes.

[12] Substitution of these trial functions into (8) and (9) and using the same N_b and M_a as test functions, generates a linear system of the form

$$\begin{pmatrix} \mathrm{K} & \mathrm{G} \\ \mathrm{G}^T & 0 \end{pmatrix} \begin{pmatrix} \underline{u} \\ \underline{p} \end{pmatrix} = \begin{pmatrix} \underline{f} \\ \overline{0} \end{pmatrix}, \tag{13}$$

where $\underline{u}$ and $\underline{p}$ are vectors of the coefficients u_{bi} and p_a. The matrices K and G and right-hand side vector $\underline{f}$ are given by

$$\begin{aligned} \mathrm{K}_{b_i c_j} &= \int (\partial_j N_b) \mu (\partial_i N_c) + \sum_k (\partial_k N_b) \mu (\partial_k N_c) \delta_{ij}, \\ \mathrm{G}_{a b_j} &= \int (\partial_j M_a) N_b, \\ \underline{f}_{b_i} &= \int N_b Ra_0 T \hat{k}_i. \end{aligned}$$

2.2.2. Solution of the Discretized System

[13] This system of equations is indefinite and, hence, specialized solution algorithms are required (for an excellent overview of some popular solution schemes, see *May and Moresi* [2008]). The method adopted by the Stokes solver within Fluidity is based upon a full-projection/pressure-correction approach. In this algorithm, the velocity and pressure solutions are obtained as follows. Note that it is assumed we know the state of all variables at the nth time step and that we wish to calculate their value at the $(n + 1)$th step.

[14] 1. Given an initial pressure field p_0 (we use the pressure from the previous time step or iteration), solve for a preliminary velocity $\underline{u}*$ in

$$\mathrm{K}\underline{u}* + G\underline{p}_0 = \underline{f}. \tag{14}$$

[15] 2. The resulting velocity will, in general, not be divergence free. To project back to a divergence free solution, the pressure is corrected $\underline{p}^{n+1} = \underline{p}_0 + \Delta\underline{p}$, where $\Delta\underline{p}$ is obtained from the following pressure correction equation:

$$\mathrm{G}^T \mathrm{K}^{-1} \mathrm{G} \Delta\underline{p} = \mathrm{G}^T \underline{u}*. \tag{15}$$

[16] 3. A velocity correction is performed, given by

$$\underline{u}^{n+1} = \underline{u}* - \mathrm{K}^{-1} \mathrm{G} \Delta\underline{p}. \tag{16}$$

[17] It is easily verified that the resulting $\underline{u}^{n+1}$ and $\underline{p}^{n+1}$ are solutions of (13). This solution procedure is equivalent to the Schur Complement Reduction method in the study by *May and Moresi* [2008].

[18] In Fluidity, iterative methods are utilized to solve these equations. Due to the slow convergence of stationary iterative methods, we consider preconditioned Krylov subspace methods only [e.g., *Elman et al.*, 2005; *May and Moresi*, 2008; *Geenen et al.*, 2009]. For this we rely on the Portable

Extensible Toolkit for Scientific Computation (PETSc) library [*Balay et al.*, 1997].

[19] For the evaluation of (14) and (16) we solve a linear system with the symmetric positive definite matrix K. For this we use the Conjugate Gradient (CG) method with an algebraic multigrid (AMG) preconditioner, based on the smoothed aggregation method of *Vanek et al.* [1996]. Fluidity implements its own version of this algorithm [*Kramer et al.*, 2010].

[20] The pressure correction equation (15) is solved using the FGMRES Krylov method [*Saad*, 1993], preconditioned with a pressure mass matrix, scaled by the local inverse of viscosity (see *Verfurth* [1984], *Benzi et al.* [2005], and *May and Moresi* [2008] for further details). In solving (15), we adopt a matrix-free representation of $G^T K^{-1} G$, as an explicit construction of this Schur complement matrix would be expensive due to the presence of K^{-1}. This requires one additional inner solve of K, for every iteration of the FGMRES method, for which we again use CG preconditioned with AMG.

2.2.3. Discretization and Solution of the Advection-Diffusion (Temperature) Equation

[21] When applied to (7), the Galerkin finite element discretization yields the following matrix equation:

$$\mathrm{M}\underline{T} + \mathrm{A}(u_i)\underline{T} + \mathrm{D}\underline{T} = 0. \tag{17}$$

Here M, A and D are the mass, advection and diffusion matrices, respectively, given by

$$\mathrm{M}_{ab} = \int N_a N_b, \tag{18}$$

$$\mathrm{A}(u_i)_{ab} = \int \sum_i N_a u_i \partial_i N_b, \tag{19}$$

$$\mathrm{D}_{ab} = \int \sum_i (\partial_i N_a)\kappa(\partial_i N_b), \tag{20}$$

and $\underline{T}$ is the vector of coefficients, T_a, such that $T = \sum_a T_a N_a$. Although available in Fluidity, the results presented here were found to be sufficiently smooth that no stabilization of the Galerkin discretization (e.g., SUPG [*Hughes and Brooks*, 1982]) was required, whilst also allowing the use of a P2 discretization for T. Dirichlet boundary conditions are imposed strongly and homogeneous Neumann boundary conditions $\sum_j n_j \partial_j T = 0$ weakly so that no additional boundary terms are necessary in A and D.

[22] Temporal discretization is achieved via a classic θ scheme:

$$\mathrm{M}\frac{\underline{T}^{n+1} - \underline{T}^{n}}{\Delta t} + \mathrm{A}\left(u_i^{n+\theta_{\mathrm{nl}}}\right)\underline{T}^{n+\theta_T} + \mathrm{D}\underline{T}^{n+\theta_T} = 0. \tag{21}$$

Here

$$\underline{T}^{n+\theta_T} = \theta_T \underline{T}^{n+1} + (1-\theta_T)\underline{T}^{n}, \quad 0 \leqslant \theta_T \leqslant 1. \tag{22}$$

[23] Equation (21) depends on the solution of (13) for the advective velocity. Vice versa the resulting temperature feeds back into (13) via the buoyancy term, and in some cases the viscosity. The nonlinear coupling between these equations is solved via a simple Picard iteration, where, in each iteration, the latest available values for the nonlinear terms are used. For the solution of (21), if $\tilde{u}_i^{n+1}$ is the velocity field obtained in the previous solution of (13), the advective velocity is given by

$$u_i^{n+\theta_{\mathrm{nl}}} = \theta_{\mathrm{nl}}\tilde{u}_i^{n+1} + (1-\theta_{\mathrm{nl}})u_i^{n}, \quad 0 \leqslant \theta_{nl} \leqslant 1. \tag{23}$$

[24] Using equation (22), (21) can be rearranged as a single matrix equation for the unknown vector $\underline{T}^{n+1}$:

$$\left[\mathrm{M} + \theta_T \Delta t\left(\mathrm{A}\left(u_i^{n+\theta_{\mathrm{nl}}}\right) + \mathrm{D}\right)\right]\underline{T}^{n+1} = \left[\mathrm{M} - (1-\theta_T)\Delta t\left(\mathrm{A}\left(u_i^{n+\theta_{\mathrm{nl}}}\right) + \mathrm{D}\right)\right]\underline{T}^{n}. \tag{24}$$

We use a GMRES Krylov subspace method to solve this system [*Saad*, 1993], with successive over relaxation (SOR) preconditioning. In all results that follow, $\theta_{\mathrm{nl}} = \theta_T = 0.5$. Our time stepping algorithm therefore reduces to the well-known Crank-Nicholson scheme.

3. Anisotropic Mesh Adaptivity

3.1. Background and Motivation

[25] Once the discretization underlying the modeling framework is in place, a robust method to adapt the computational mesh is required. Mesh optimization is used to best represent evolving solution structures and to reduce discretization errors. Our approach is motivated by the need to identify anisotropy so that such information can be built into the eventual adapted mesh.

[26] Isotropic unstructured meshes enforce the same grid point spacing in all directions. Certain dynamical features, however, are highly anisotropic, as material properties vary only in specific directions.

For example, at a boundary layer, properties vary rapidly normal to the boundary, but are often relatively smooth along the boundary. The most suitable grid point spacing would therefore be smallest normal to the boundary layer. However, an isotropic mesh will also impose this grid point spacing along the length of the boundary, thus increasing the computational requirements of the simulation. Allowing resolution to vary in different directions means that the grid point spacing can be at its smallest normal to a boundary layer, but larger along its length. Such unstructured, anisotropic meshes aim to represent boundary layers and one-dimensional flow features with the minimum required number of grid points, thus providing a significant advantage over structured/isotropic meshes, such as those utilized by *Burstedde et al.* [2008].

[27] The approach described here is a metric based formulation, built upon the Hessian matrix. This allows element size/edge length to be controlled by a specified interpolation error. Hessian based interpolation errors are cheap to compute, yield directional information and give a good guide to problem complexity [e.g., *Lohner et al.*, 1985; *Peraire et al.*, 1987; *Wu et al.*, 1990; *Pain et al.*, 2001; *Davies et al.*, 2007; *Piggott et al.*, 2008]. Following construction of the metric, a functional, which gauges mesh quality in terms of element size and shape, is optimized, by locally adjusting mesh connectivity and nodal positions. The final result is a mesh that represents the solution to the desired interpolation error, within a given tolerance.

3.2. Estimates of Discretization Error: The Metric

[28] To optimize the mesh a solution-dependent error indicator, which varies in space and time and yields the desired mesh resolution, is required. In Fluidity, this is achieved via the construction of a metric tensor field that is built upon the Hessian matrix for each solution component and user-defined definitions of (potentially spatially varying) desired errors. In 1-D, the interpolation error, ε, may be approximated by

$$\varepsilon = h_e^2 \left| \frac{\partial^2 \psi}{\partial x^2} \right|, \tag{25}$$

where h_e is the length of element e and ψ is the solution variable. In multiple dimensions

$$\varepsilon = \sum_{i,j} v_j |H|_{ij} v_i \tag{26}$$

with a Hessian matrix, $H_{ij} = \partial_i \partial_j \psi$, and vector, v_i, representing the length and desired direction, which may be arbitrarily oriented. Note that once H_{ij} is evaluated (see section 3.3), signs of the curvatures are removed by taking absolute values of the eigenvalues of H_{ij} (only the magnitude of the curvature is significant).

[29] For a specified interpolation error, $\hat{\varepsilon}$, a metric, $\mathcal{M}_{ij}$, is defined such that an element size (edge length) is unity if it has the desired interpolation error. Thus

$$\mathcal{M}_{ij} = \frac{1}{\hat{\varepsilon}} |H|_{ij}. \tag{27}$$

The length of v_i with respect to $\mathcal{M}_{ij}$ is then calculated from

$$\|v_i\|_{\mathcal{M}_{ij}} = \left(\sum_{i,j} v_j \mathcal{M}_{ij} v_i \right)^{1/2}. \tag{28}$$

Thus, in 1-D, for an element e, of length h_e in Euclidian space, the length, $\|h_e\|_{\mathcal{M}_{ij}}$, with respect to $\mathcal{M}_{ij}$ is

$$\|h_e\|_{\mathcal{M}_{ij}} = \left(h_e^2 \frac{1}{\hat{\varepsilon}} \left| \frac{\partial^2 \psi}{\partial x^2} \right| \right)^{1/2}. \tag{29}$$

Thus, if $\|h_e\|_{\mathcal{M}_{ij}}$ is less than one, ε is lower than the desired interpolation error, $\hat{\varepsilon}$, and the mesh may be locally coarsened. Conversely, if $\|h_e\|_{\mathcal{M}_{ij}}$ is greater than one the desired interpolation error is not attained and the mesh must be refined.

[30] The metric, $\mathcal{M}_{ij}$, is subsequently modified to take into account maximum and minimum element sizes, in addition to bounds on the aspect ratio of elements. The metric also incorporates controls on edge length gradation (i.e., the maximum allowable jump in edge length from element to element) and it is possible to advect the metric forward in time, thus providing an estimate of mesh requirements in the future and reducing the number of adaptive steps required as the simulation evolves [*Wilson*, 2009; *Hiester et al.*, 2011].

[31] For the problems examined herein, and those typically examined in geodynamics, more than one solution field exists (e.g., temperature, velocity and pressure). Each field has its own metric, reflecting its own required mesh resolution. Each field can also have its own error requirements, which will, in turn, be reflected in the metric associated with that field. In simulations where the metric is constructed for more than one field, metrics are superimposed, so that the adaptivity algorithm only deals with a

single metric field. This is achieved by locally prescribing the finest mesh resolution, in a particular direction, dictated by any of these fields under consideration (for further information, see *Pain et al.* [2001]).

3.3. Hessian Calculation

[32] Galerkin methods are repeatedly applied to calculate the first derivatives of a solution field, ψ. At each vertex a of the mesh, first derivatives are approximated by

$$\partial_i \psi|_a \approx q_{a_i} = \mathrm{L}_a^{-1} \int M_a \partial_i \psi, \tag{30}$$

where M_a are the P1 finite element basis functions defined in section 2.2.1 and L_a is the lumped P1 mass matrix [see *Zienkiewicz and Taylor*, 1991]. The second-order terms of the Hessian are then approximated by applying the same procedure to the first derivative field, $q_j = \sum_b M_b q_{bj}$ as follows:

$$\partial_i \partial_j \psi|_a \approx H_{a_{ij}} = \mathrm{L}_a^{-1} \int \sum_b M_a \partial_i M_b q_{b_j}, \tag{31}$$

to obtain the numerical discretization, $H_{a_{ij}}$, of the Hessian at node a.

3.4. Mesh Quality Functional

[33] Following construction of the metric, a functional is formed that is optimized to improve mesh quality. Multiple choices are available for such a mesh quality functional. When one considers anisotropic mesh optimization, it is common to form a functional that depends on both the shape and size of elements forming the mesh. In the two-dimensional results presented here, the objective functional of *Vasilevski and Lipnikov* [1999] is used. The functional of *Pain et al.* [2001] is utilized for all three-dimensional cases (the three-dimensional algorithm is the focus of all subsequent discussion). This takes the form

$$F_{e|\mathcal{M}} = \frac{1}{2} \sum_{l \in \mathcal{L}_e} \left(\delta_{l|\mathcal{M}}\right)^2 + \left(q_{e|\mathcal{M}}\right)^2, \tag{32}$$

where F_e is the functional associated with element e, and $\mathcal{L}_e$ is the set of edges of element e. The first term in this functional gauges the size of an element, whilst the second term gauges its shape. For the mesh quality functional to yield variable-resolution anisotropic meshes, the element quantities $\delta_{l|\mathcal{M}}$ and $q_{e|\mathcal{M}}$ are actually measured with respect to the non-Euclidean metric, $\mathcal{M}_{ij}$ (hence the presence of the subscript $\mathcal{M}$ in (32)). In this equation, $\delta_{l|\mathcal{M}}$ for edge l is defined as

$$\delta_{l|\mathcal{M}} = r_{l|\mathcal{M}} - 1, \tag{33}$$

where $r_{l|\mathcal{M}}$ is the length, with respect to the metric, of edge l, whilst $q_{e|\mathcal{M}}$ in (32), is defined as

$$q_{e|\mathcal{M}} = \left(\frac{\beta}{r_{s|\mathcal{M}}} - 1\right), \tag{34}$$

where $\beta = 1/(2\sqrt{6})$ and $r_{s|\mathcal{M}}$ is the radius of the inscribed sphere of element e, with respect to the metric. The scalar β is chosen such that $q_{e|\mathcal{M}} = 0$ for an ideal element (aspect ratio of unity in relation to the metric). Similarly, when all edge lengths are unity, with respect to $\mathcal{M}$, $\delta_{l|\mathcal{M}} = 0$. Minimization of (32) thus ensures a mesh with appropriately shaped and sized elements: they are equilateral elements in metric space.

3.5. Mesh Optimization

[34] Having calculated the mesh quality functional, an optimization procedure, which visits elements in the mesh and performs topological operations on local connectivity, is then utilized to drive down this functional and, hence, adapt the mesh. The operations permitted on the mesh are as follows (see Figure 1 for illustration).

[35] 1. Edge split is when a node is inserted on a preexisting edge in the mesh and surrounding elements, with improved shape/size characteristics compared with the original elements, are created (for example, in 2-D, two triangular elements become four). While the location of this new node along the preexisting edge can be optimized, it is common to simply split it at its midpoint.

[36] 2. Edge collapse is the inverse of edge splitting; this involves the removal of a node and the merging of elements. In cases where one of the nodes defines the domain geometry or an internal structure, the nodes are collapsed to that point. If both nodes define some geometrical structure this operation is not permitted.

[37] 3. Edge-edge swap preserves the number of nodes, but manipulates the edge lengths and element shapes by changing the configuration of an edge between two elements (two dimensions)/four elements (three dimensions).

[38] 4. Edge-face swap is a three-dimensional operation: if two tetrahedra share a common face, and provided their combined interior is convex, the face

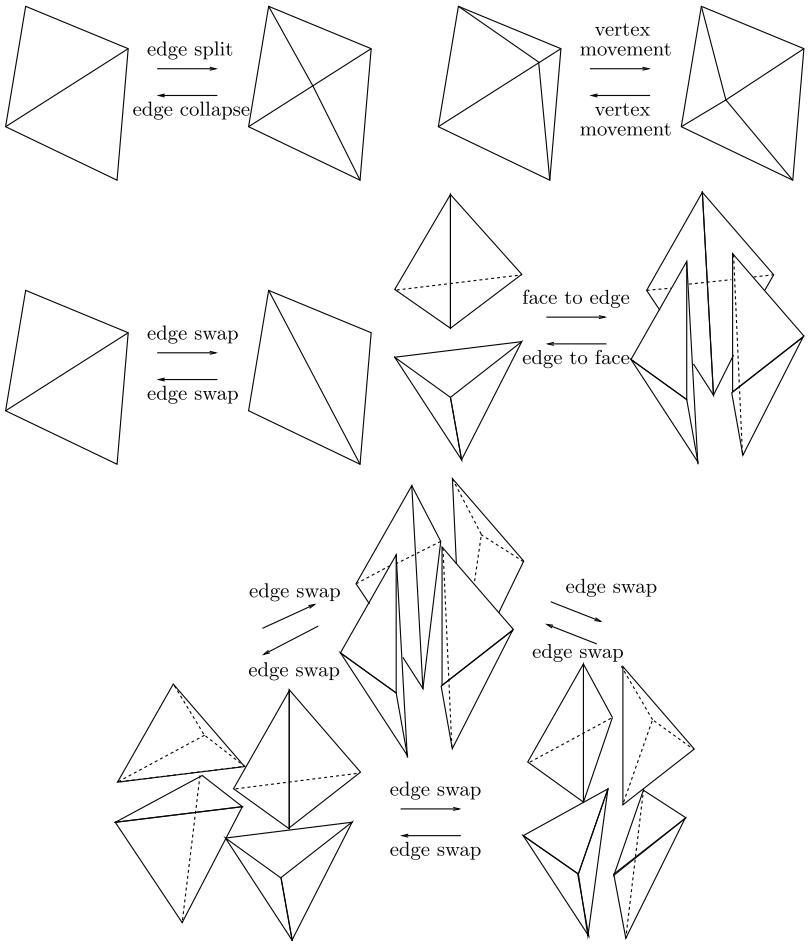

Figure 1. Topological operations performed on a local cluster of elements to optimize a mesh. See section 3.5 for further details.

is deleted and a new edge is introduced between the two nodes not shared, thus producing three tetrahedra with different alignments. The inverse operation, where an edge is replaced by a face, is also allowed.

[39] 5. Node/vertex movement is when the local topology of the mesh is preserved, but mesh quality is improved by visiting each node and applying a hill climbing algorithm to relocate the node to a position that minimizes the maximum functional value of the surrounding elements.

[40] The mesh optimization process proposes a new configuration, which is a small change in the mesh, using one of the above operations. This is accepted as the current mesh configuration if the change in the maximum functional associated with all elements affected by the change is negative and less than a smallness parameter, γ. In addition, if the maximum functional value of all elements that would be affected by a local mesh transformation is less than a certain threshold value, $\mathbb{F}_t$, then this mesh transformation is not considered. In all three-dimensional results that follow, $\gamma = -0.01$ and $\mathbb{F}_t = 0.15$. Further discussion of the adaptivity algorithms in two and three dimensions can be found in the studies by *Vasilevski and Lipnikov* [1999] and *Pain et al.* [2001], respectively.

3.6. Interpolation

[41] Following mesh optimization, two distinct meshes exist: the old mesh (mesh A) and a new mesh (mesh B). Mesh A was optimal for conditions that existed at a previous period in time and contains up to date information about fields at the current time. Mesh B, on the other hand, has just been created and is optimal for the current condition of the fields. Other than describing the same domain, the meshes will, in general, be unrelated. However, for the simulation to proceed, it is necessary to transfer old information from mesh A to mesh B. Although several options are available in Fluidity (see *Farrell et al.* [2009] and *Farrell and*

Maddison [2010] for further details), consistent interpolation is utilized in all adaptive simulations presented herein [*Hassan et al.*, 1995; *Davies et al.*, 2007]: information is transferred using the underlying basis functions of each field. Solution nodes of mesh B are located in mesh A using the R tree search spatial index library [*Manolopoulos et al.*, 2006]. Field values are then interpolated from the solution nodes of mesh A to these locations, using the field basis functions of mesh A as interpolants. As noted in section 2, for all simulations examined herein, P2 basis functions are used for velocity and temperature. Pressure, which is discretized using P1 functions, is not interpolated.

4. Validation and Verification

[42] In this section we compare the numerical predictions of Fluidity with well-established two- and three-dimensional cartesian geometry benchmark results. Note that, although spherical shell geometry presents additional challenges when representing domain boundaries and interpolating between successive meshes, unstructured, anisotropic adaptive mesh refinement techniques are equally applicable in such domains. For direct quantitative comparisons with the benchmark results of *Blankenbach et al.* [1989] and *Busse et al.* [1994], we calculate the Nusselt Number (Nu) and RMS velocity (V_{RMS}), once a steady state has been achieved (the variation in the infinity norm of the velocity, pressure and temperature fields is $<10^{-7}$, between consecutive time steps). The Nusselt Number is defined as

$$Nu = -z_{\max}\frac{\int_{z=z_{\max}}\sum_i n_i\partial_i T}{\int_{z=0} T}, \tag{35}$$

where $z_{\max}$ is the maximum z coordinate of the domain, $\int_{z=z\max}$ denotes the integral over the top surface of the domain and $\int_{z=0}$ denotes the integral over the bottom surface of the domain. The RMS velocity is given by

$$V_{RMS} = \sqrt{\frac{1}{V}\int\sum_i u_i^2}. \tag{36}$$

Here, V denotes the domain volume/area.

4.1. Two-Dimensional Convection Benchmarks

[43] The first examples considered are steady state convection in a two-dimensional square domain, of unit dimensions. Boundary conditions for temperature are $T = 0$ at the surface ($z = 1$), $T = 1$ at the base ($z = 0$), with insulating (homogeneous Neumann, $\partial_x T = 0$) sidewalls. For velocity, free-slip and no-normal flow boundary conditions are specified at all boundaries.

[44] Three isoviscous cases are examined, at $Ra_0 = 10^4$, 10^5 and 10^6 (cases 1a–1c: *Blankenbach et al.* [1989]), on both structured (uniform resolution) and adaptive unstructured meshes. A variable viscosity case is also examined, where viscosity varies as a function of temperature ($Ra_0 = 10^4$, case 2a: *Blankenbach et al.* [1989]), according to the relation

$$\mu = \mu_0 \exp[\ln(1000)T], \tag{37}$$

where μ_0 is the reference viscosity. For the adaptive simulations, specified minimum edge lengths, maximum edge lengths and desired interpolation errors are varied, depending on the requirements of the simulation. However, for all adaptive cases, the maximum aspect ratio of elements is limited to 10. Separate metrics are constructed for velocity, pressure and temperature fields and these are superimposed, as described in section 3.2.

[45] Results are presented in Figures 2–3 (note that, excluding the location of upwelling flow at $x = 0$ or $x = 1$, these results are insensitive to the initial condition). They show excellent agreement with the benchmark predictions of *Blankenbach et al.* [1989]. On uniform, structured meshes, solution accuracy improves with increased resolution, as expected, with all high-resolution cases achieving solutions that lie within 0.5% of benchmark values. On adaptive, unstructured meshes, solution accuracy is improved, when compared to uniform meshes with a similar number of grid points.

[46] The metric and associated optimization algorithm focusses resolution around zones of high solution curvature (Figures 4a–4c and 5a–5c). Elements also become highly anisotropic, with their long edges aligning parallel to the direction of one-dimensional flow features (Figures 4d and 5a–5c). As noted previously, such anisotropic meshes aim to represent the problem with the minimum required number of grid points. Comparison with a near-isotropic mesh illustrates this point (Figures 5d–5f): placing an aspect-ratio bound of 1 (isotropic) on the metric increases the number of nodes required to represent the problem (case 1c: *Blankenbach et al.* [1989]), from 16,699 to 94,133. Anisotropic meshes therefore dramatically reduce the computational cost of a simulation. Note that the benefits of unstructured mesh optimization increase with convective vigor (Ra_0) and varying viscosity, owing to finer scale, more localized higher-gradient fea-

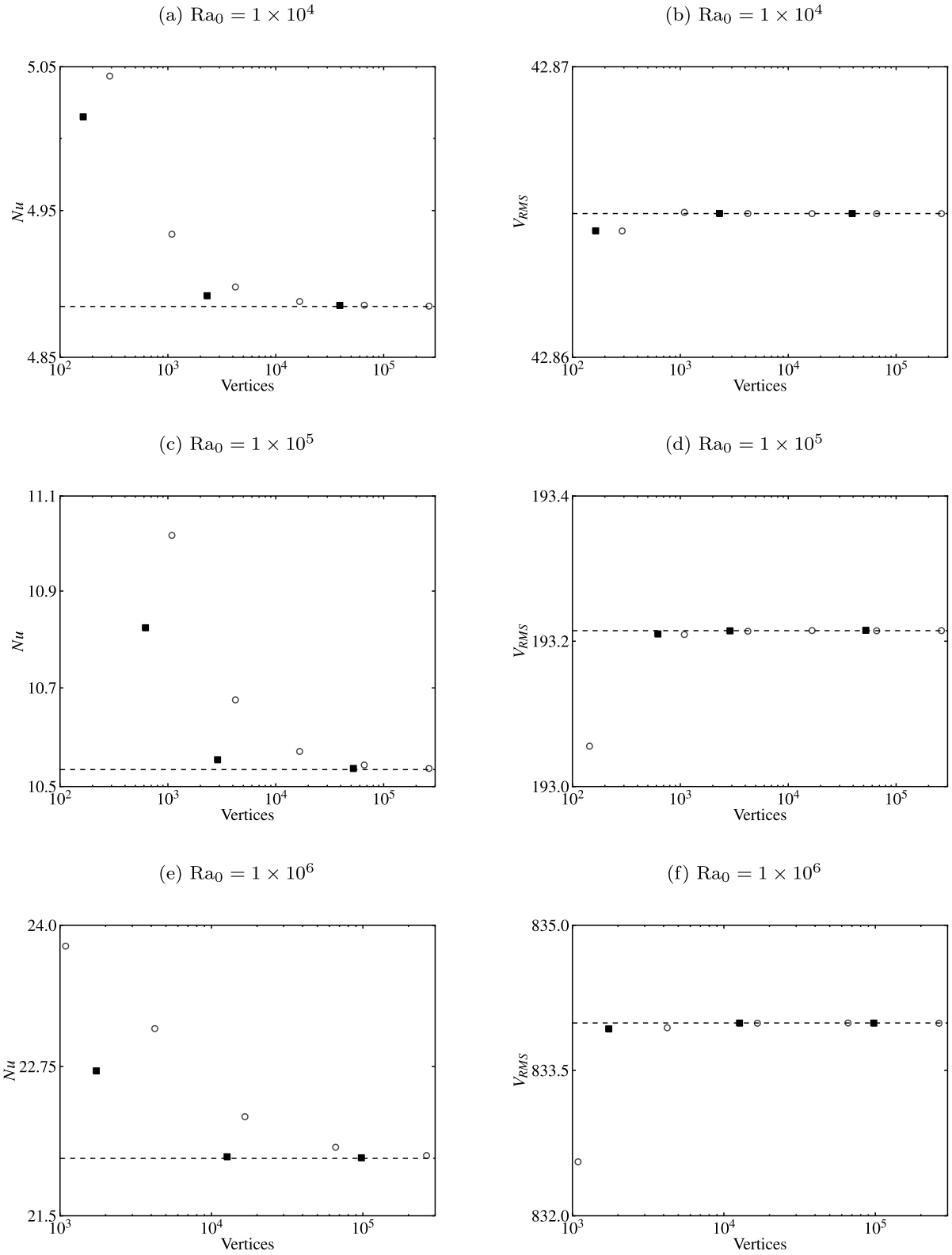

Figure 2. Results from 2-D, isoviscous square convection benchmark cases. (a) Nusselt number versus number of triangle vertices, at $Ra_0 = 1 \times 10^4$ (case 1a: *Blankenbach et al.* [1989]), for a series of uniform, structured meshes (open circles) and adaptive unstructured meshes (filled squares); (b) RMS velocity versus number of triangle vertices, at $Ra_0 = 1 \times 10^4$; (c, d) as with Figures 2a and 2b but at $Ra_0 = 1 \times 10^5$ (case 1b: *Blankenbach et al.* [1989]); (e, f) $Ra_0 = 1 \times 10^6$ (case 1c: *Blankenbach et al.* [1989]). Benchmark values are denoted by horizontal dashed lines.

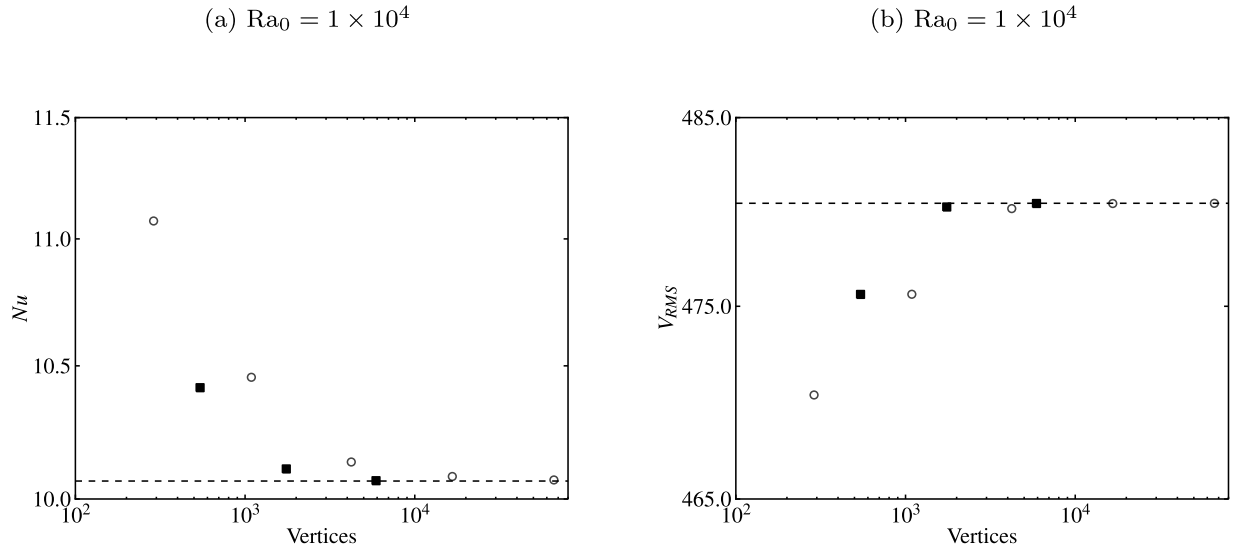

Figure 3. As in Figure 2 but for a variable viscosity benchmark case, on a series of uniform, structured meshes (open circles) and adaptive unstructured meshes (filled squares): $Ra_0 = 1 \times 10^4$ (case 2a: *Blankenbach et al.* [1989]). (a) Nusselt number versus number of triangle vertices; (b) RMS velocity versus number of triangle vertices.

tures, which are more difficult to resolve on coarser/uniform grids. In addition, this optimization, which clearly has a positive effect on global measures (i.e., V_{RMS}), has a more dramatic effect on the accuracy of local diagnostics (i.e., *Nu*). This is consistent with the findings of *Davies et al.* [2007].

4.2. Three-Dimensional Convection Benchmarks

[47] We next compare our results with three-dimensional benchmarks, for steady state cases with both constant (case 1a from *Busse et al.* [1994]) and temperature-dependent viscosity (case 2 from *Busse et al.* [1994]). The domain is a box of dimensions $a \times b \times 1$. Boundary conditions for temperature are $T = 0$ at the surface ($z = 1$) and $T = 1$ at the base ($z = 0$), with insulating (homogeneous Neumann) sidewalls. No-slip velocity boundary conditions are specified at the top surface and base of the domain, with free-slip and no-normal flow boundary conditions on all sidewalls. Initial conditions are chosen to produce a single ascending and descending flow, at $x = y = 0$ and ($x = a$), ($y = b$), respectively. For the constant viscosity case, $a = 1.0079$, $b = 0.6283$, whilst $Ra_0 = 3 \times 10^4$. For the variable viscosity case, the domain is a cube of unit dimensions ($a = b = 1$), whilst $Ra_0 = 2 \times 10^4$, defined by the viscosity at $T = 0.5$. For the viscosity, we adopt the following law:

$$\mu = \mu_0 \exp\left[\frac{Q}{T+G} - \frac{Q}{0.5+G}\right], \tag{38}$$

where $Q = [255/\ln(r)] - 0.25 \ln(r)$, $G = [15/\ln(r)] - 0.5$, and $r = 20$. Note that the final steady state solution showed strong sensitivity to the initial condition for this variable viscosity case. Results are presented in Figures 6 and 7, demonstrating an excellent agreement with benchmark values. As expected, and consistent with the two-dimensional cases, simulations on anisotropic adaptive unstructured meshes display more accurate solutions when compared to simulations on uniform meshes with a similar number of grid points.

[48] From these benchmark comparisons, we conclude that Fluidity is accurate for both two- and three-dimensional, steady state, isoviscous and variable viscosity simulations. In addition, anisotropic adaptive unstructured meshes represent the problem with significantly fewer grid points than uniform, structured meshes. However, the true value of adaptive meshes only becomes apparent for time-dependent problems, where the mesh can evolve with the dynamics of the underlying flow field. The applicability of Fluidity to such a problem is demonstrated in the following section.

5. Application to Mantle Plumes

[49] *Vatteville et al.* [2009] performed laboratory experiments initiating thermal plumes in a square tank of silicone oil at several heater powers. Approximating the domain to be cylindrically

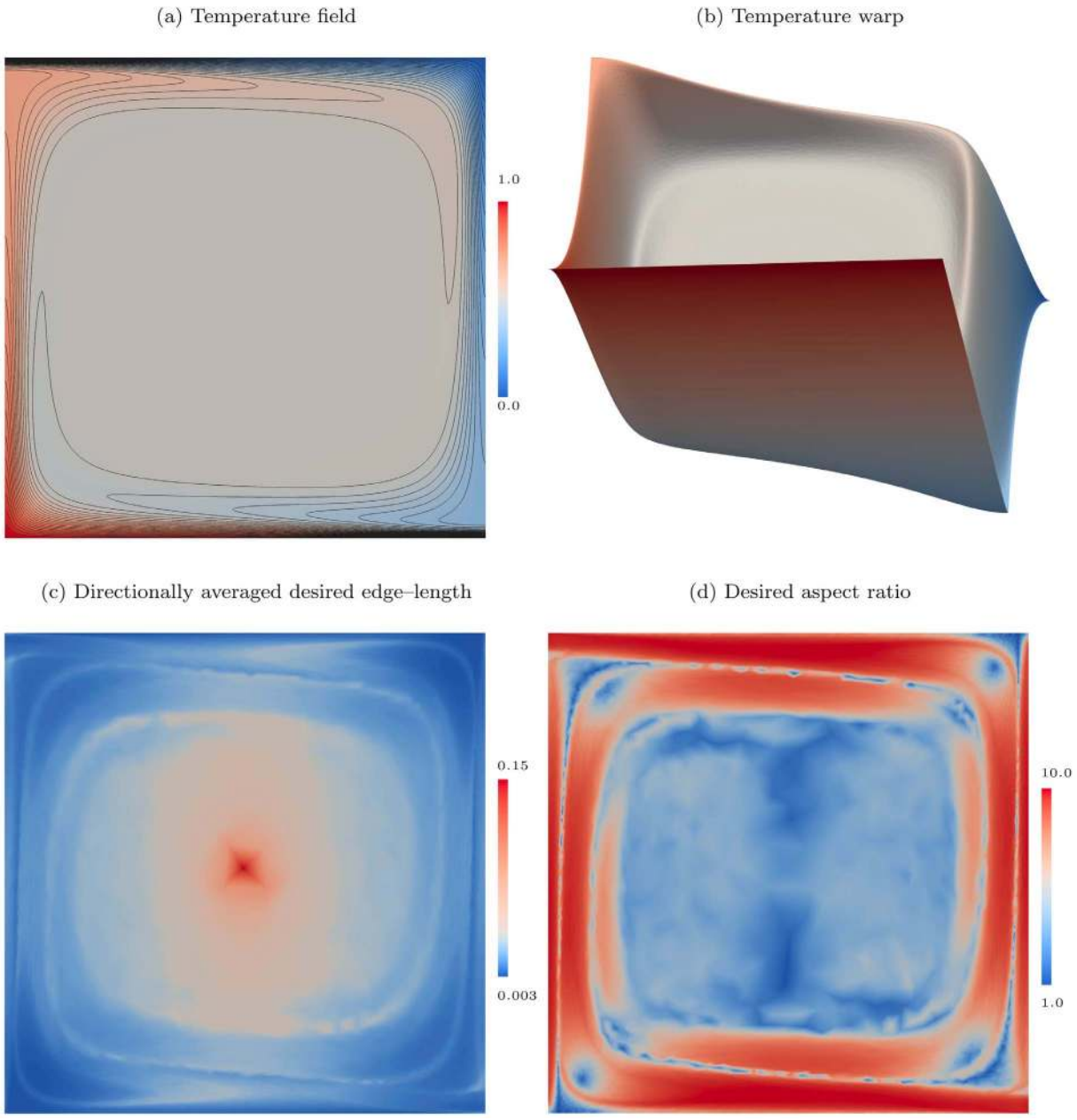

Figure 4. (a) Steady state temperature field from an isoviscous simulation at $Ra_0 = 1 \times 10^6$ (case 1c: *Blankenbach et al.* [1989]). Contour intervals = 0.025; (b) warped illustration of the temperature field; data defined on the $x - z$ plane is perturbed in the y direction, in proportion to temperature. The point at the right-hand limit of Figure 4b lies at $x = z = 1$. Regions of strong solution curvature become apparent. (c) The mesh optimization algorithm associates such regions with a small desired minimum edge length. Note that for this simulation, no artificial bounds were applied for minimum and maximum edge lengths. (d) The desired aspect ratio of elements. Where the desired edge length showed a strong directional variation, the element aspect ratio was limited to 10. The resultant underlying adapted mesh is shown in Figure 5a.

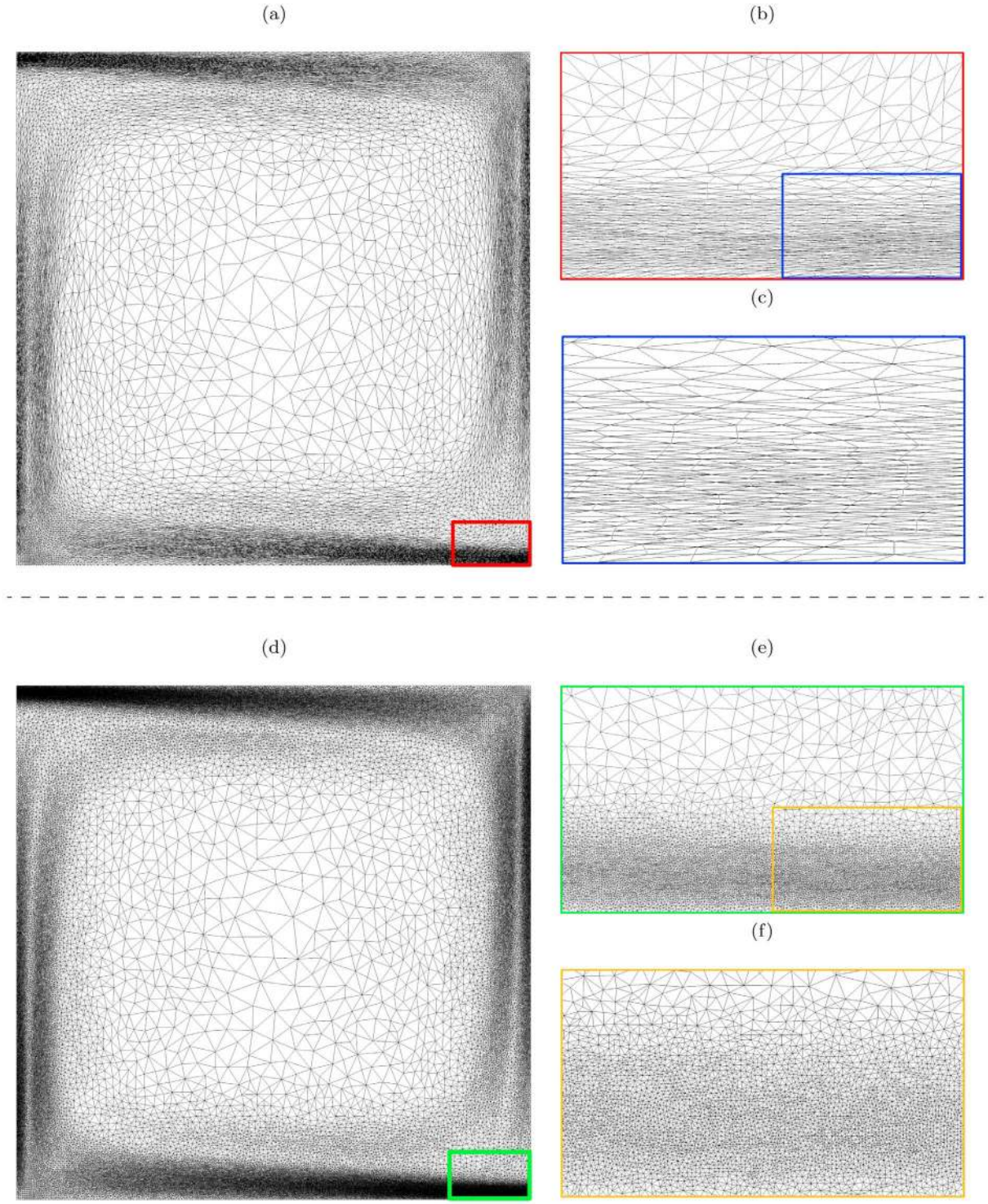

Figure 5. (a) Anisotropic, adapted unstructured mesh (maximum element aspect ratio = 10, number of triangle vertices = 16,699), from the simulation shown in Figure 4; (b, c) high-resolution images of specific mesh sections, illustrating anisotropic elements within the thermal boundary layer; (d) a near-isotropic unstructured mesh (maximum element aspect ratio = 1, number of triangle vertices = 94,133), with the same minimum edge length as the mesh displayed in Figure 5a; (e, f) high-resolution images of specific mesh sections. Placing an aspect ratio bound of 1 (isotropic) on the metric dramatically increases the number of nodes required to represent the problem.

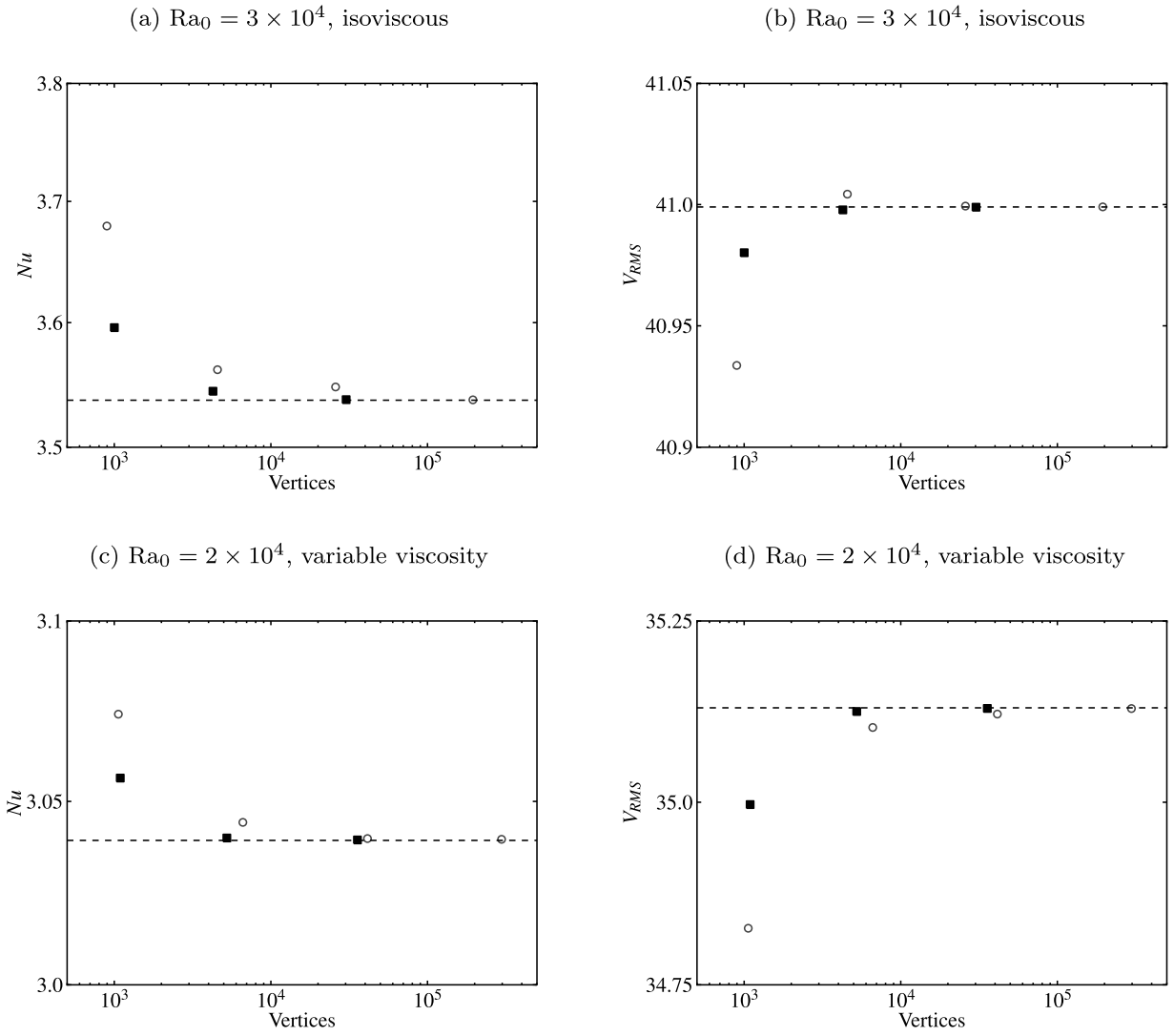

Figure 6. Results from 3-D benchmark cases, on a series of uniform, structured meshes (open circles) and adaptive unstructured meshes (filled squares). (a, b) An isoviscous case, at $Ra_0 = 3 \times 10^4$ (case 1a: *Busse et al.* [1994]); (c, d) a variable viscosity case, at $Ra_0 = 2 \times 10^4$ (case 2: *Busse et al.* [1994]). Benchmark values are denoted by horizontal dashed lines.

symmetric, they demonstrated that the experimental results could be reproduced numerically in two dimensions. In this section, we validate Fluidity against these combined laboratory and numerical experiments, presenting both fixed and adaptive mesh simulations of the experiments, with a more accurate description of the tank geometry. This direct comparison provides thorough validation and verification of the solvers and adaptive strategies employed by Fluidity, in a time-dependent, three-dimensional simulation, where regions of dynamic significance, and hence, for adaptive simulations, the mesh, are constantly evolving.

[50] Model parameters are summarized in Tables 1 and 2. We exploit some symmetries of the system and simulate a quarter (0.075 m × 0.075 m × 0.161 m) of the full experimental tank (0.15 m × 0.15 m × 0.161 m), with a 0.009 m radius and 0.003 m high heater (located in the center of the full tank). Our simulations are fully three-dimensional. We examine four cases, at different heater power levels (0.6, 1.0, 1.7 and 3.3 W, respectively; see Table 2). Boundary conditions are chosen to mimic those of the laboratory tank: no-slip conditions are specified on the bottom (including the heater) and external sidewalls of the

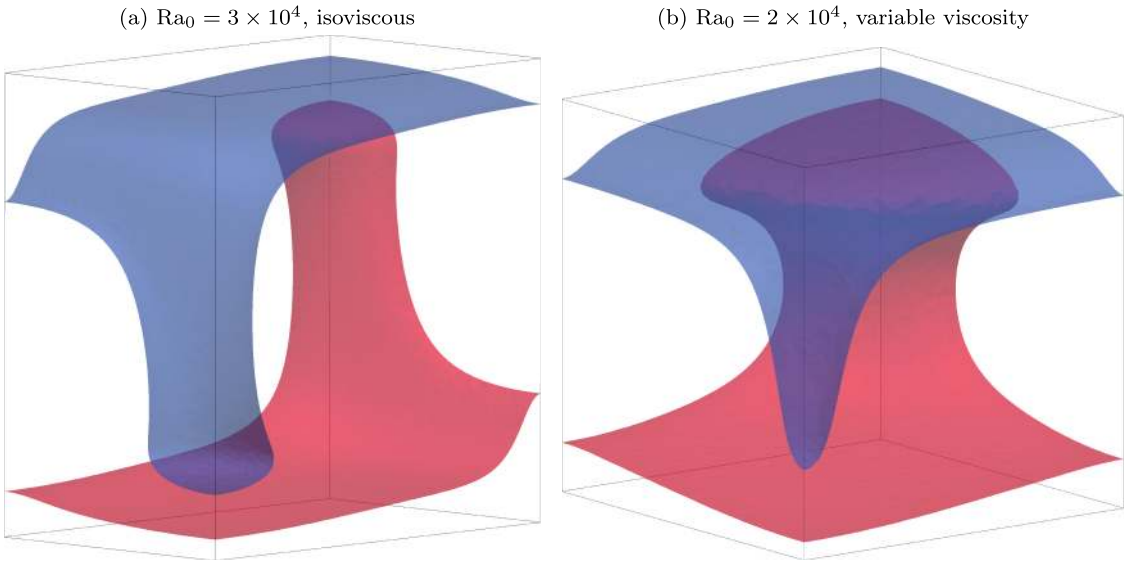

Figure 7. Temperature isosurfaces (red = 0.7; blue = 0.3) for 3-D benchmark cases. (a) Case 1a: *Busse et al.* [1994], constant viscosity; (b) case 2: *Busse et al.* [1994], a modestly temperature-dependent viscosity.

domain, whilst free-slip and no-normal flow conditions are specified at the top and internal reflecting sidewalls (planes of mirror symmetry). For temperature, the external sidewalls and top boundary are kept at room temperature. The bottom boundary is also held at room temperature, except at the heater itself, where we prescribe the measured time-dependent evolution of heater temperature, from the laboratory experiments (this is available as supplementary material to the study by *Vatteville et al.* [2009]). Homogeneous Neumann conditions are specified for all internal reflective sidewalls. We consider the temperature dependence of viscosity, following the relation in Table 1.

[51] Simulations are performed using both fixed and adaptive unstructured meshes, allowing for separate validation of both strategies. For adaptive simulations, we optimize the mesh every 5 time steps and restrict the minimum edge length to that of the fixed mesh cases (0.0004 m), thus allowing for simple comparison. Other controlling parameters of interest are: $\hat{\varepsilon}(u_i) = 0.0002$ m s^{-1}; $\hat{\varepsilon}(p) = 0.25$ Pa; $\hat{\varepsilon}(T) = 0.4$°C. Adaptive time stepping targets a maximum Courant number of 2.5. Note that only one nonlinear Picard iteration was used for the results presented: it was verified that increasing the number of iterations had a negligible influence on results.

[52] Results are presented in Figures 8 and 9. In Figure 8a, we plot the maximum velocity along the plume conduit, as a function of time, for the experimental data and for fixed and adaptive mesh simulations. The experimentally measured velocity field is slightly noisy, due to the statistical nature of Particle Image Velocimetry (PIV), but compares quantitatively well with the velocity field predicted by Fluidity, over a range of supplied powers and, hence, over a range of heater temperatures. We consistently observe that the near-steady plume conduit velocity predicted numerically is higher than the laboratory measurements. Identical discrepancies were observed between the numerical and laboratory results of *Vatteville et al.* [2009].

[53] One critical aspect of the laboratory measurements is that the PIV method uses an averaging window that is necessary to compile statistically

Table 1. Material Properties of Fluid (Silicone Oil) Used in Laboratory Plume Experiment[a]

Parameter	Value	Units
ρ_0	991	kg m^{-3}
α	9.4×10^{-4}	K^{-1}
μ	exp(−7.11 + 1892./T)	Pa s
k	0.17	W m^{-1} K^{-1}
C_p	1460	J kg^{-1} K^{-1}
Pr	4.5×10^3	—

[a]Values for thermal expansivity, α, dynamic viscosity, μ, and thermal conductivity, k, are measured at 20°C. Reference density, ρ_0, is measured at 0°C. Note that in the dynamic viscosity law, the temperature, T, is absolute temperature, in Kelvin. C_p denotes specific heat capacity, whilst Pr is the Prandtl number.

Table 2. Characteristics of Laboratory Plume Simulations[a]

Parameter	Heater Power (W)			
	0.6	1.0	1.7	3.3
Room T (°C)	21.47	19.56	19.69	20.63
Max T (°C)	42.4	45.8	56.2	74.2
Viscosity at Room T (Pa s)	0.504	0.526	0.524	0.513
Viscosity at Max T (Pa s)	0.329	0.309	0.256	0.19
Rayleigh Number	1.23×10^7	1.47×10^7	2.06×10^7	3.08×10^7

[a]As expected, the temperature difference, Rayleigh number and viscosity contrast increase with increasing power. The height of the fluid above the heater (158 mm) is used in defining the Rayleigh number (3).

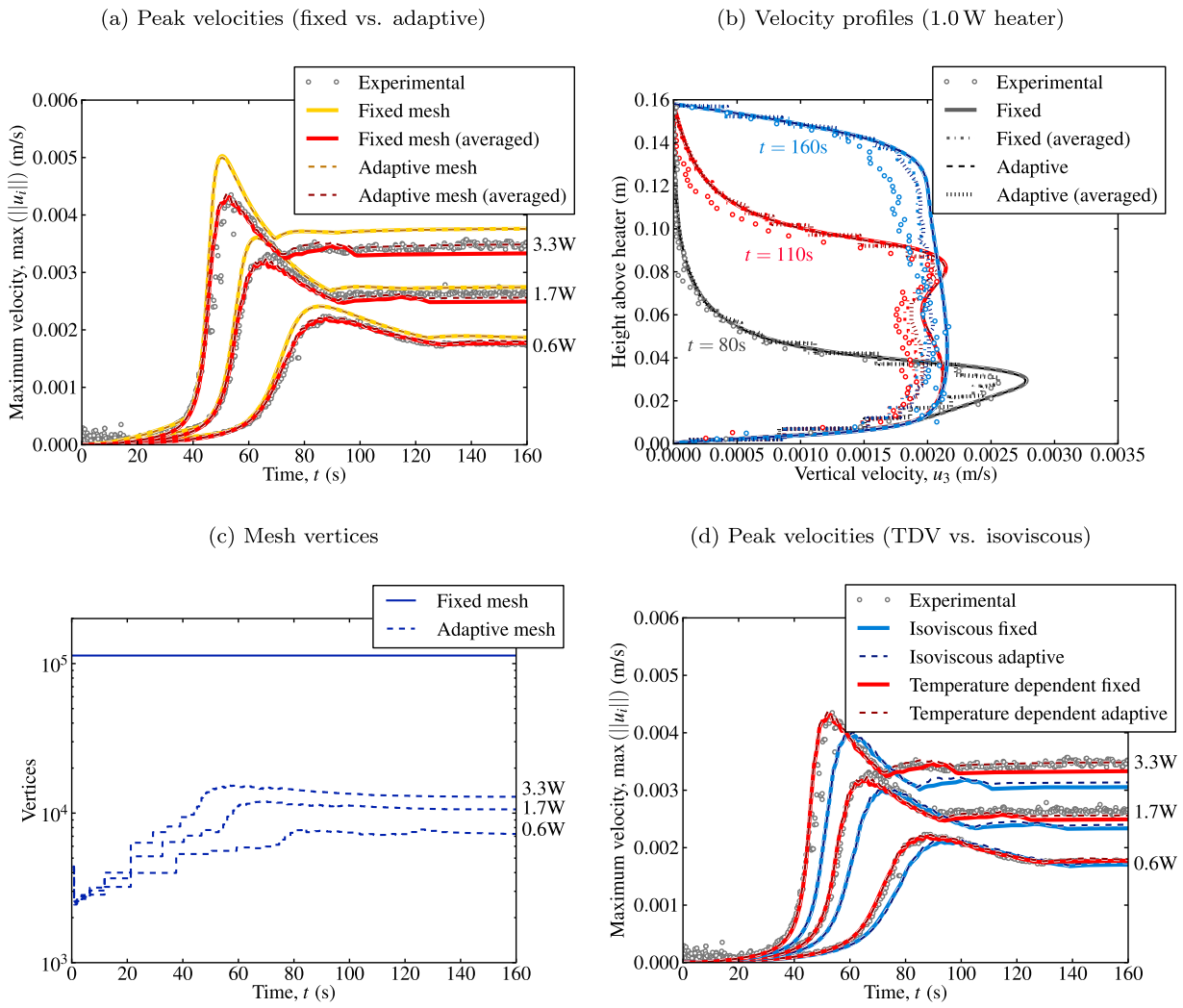

Figure 8. (a) Maximum velocities versus time in the laboratory experiments (data taken from *Vatteville et al.* [2009]) and the fixed and adaptive mesh numerical simulations using Fluidity. Following *Vatteville et al.* [2009], the numerical data is averaged in 5 mm squares over one of the internal reflective sidewalls to mimic the effect of laboratory data collection. The effect of this averaging is also visible in (b) velocity profiles of the numerical and laboratory data above the heater. In Figures 8a and 8b, excellent agreement can be seen both between the numerical and laboratory data and between (c) the fixed and adaptive mesh simulations, despite the adaptive simulations having significantly fewer vertices (note the logarithmic scale). (d) Averaged peak velocities from fixed and adaptive mesh isoviscous and temperature-dependent viscosity (TDV) simulations. The temperature dependence of the viscosity is small (for details see *Vatteville et al.* [2009]); however, it must be included to match the laboratory data set.

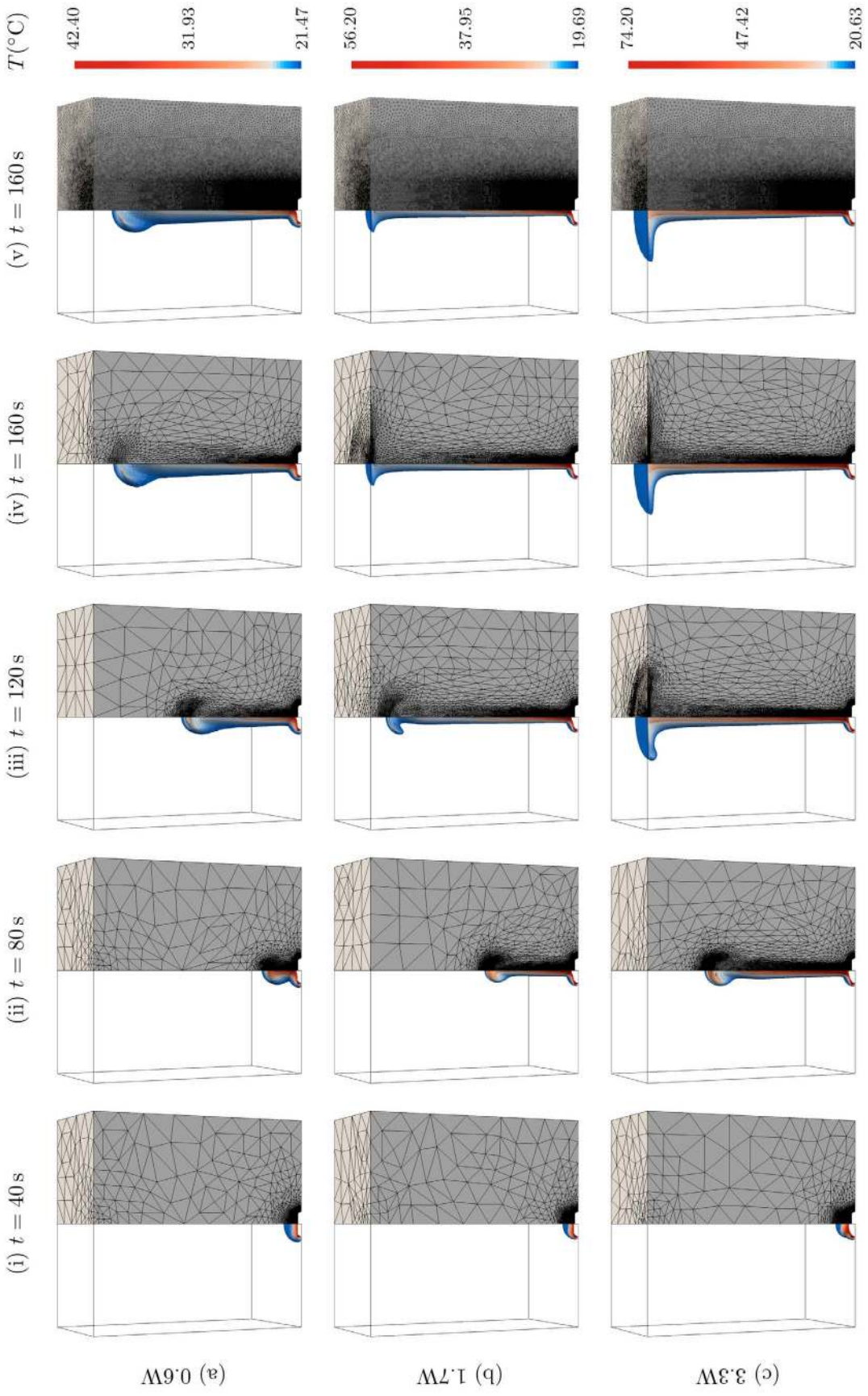
T(°C)
42.40
31.93
21.47
56.20
37.95
19.69
74.20
47.42
20.63
(v) t = 160 s
(iv) t = 160 s
(iii) t = 120 s
(ii) t = 80 s
(i) t = 40 s
(a) 0.6W
(b) 1.7W
(c) 3.3W

Figure 9

Table 3. Number of Vertices, P2 Nodes, and Time Steps in Each Simulation[a]

Heater Power (W)	Mesh	Vertices			P2 Nodes	Time Steps	Wall Time Adaptivity %
		Average	Minimum	Maximum	Average		
0.6	Fixed	113,158	113,158	113,158	825,990	626	—
0.6	Adaptive	6650	2447	7856	46,024	275	1.9
1.0	Fixed	113,158	113,158	113,158	825,990	728	—
1.0	Adaptive	7989	2484	9192	55,856	341	1.9
1.7	Fixed	113,158	113,158	113,158	825,990	956	—
1.7	Adaptive	10,306	2536	12,020	73,042	492	1.7
3.3	Fixed	113,158	113,158	113,158	825,990	1295	—
3.3	Adaptive	13,062	2580	15,367	93,476	776	1.8

[a]In adaptive simulations the average number of vertices and P2 nodes are per time step. The number of vertices as a function of simulation time can be seen in Figure 8c. The percentage of wall time taken by adaptivity (including metric assembly, mesh optimization, and subsequent interpolation) is also shown for the adaptive cases.

meaningful velocities (see *Vatteville et al.* [2009] for further details). To mimic the effects of this averaging, we postprocess numerical results by averaging in 5 mm squares over one of the internal reflective sidewalls. This two-dimensional plane mimics the laser sheet used for the PIV measurements in the experiments. Averaging has the effect of reducing the discrepancy between experimentally sampled and numerically predicted velocities (Figure 8b). However, as the average is based on the number of data points within a square, slight differences are introduced between postprocessed fixed and adaptive mesh results (Figure 8a).

[54] Figure 8d demonstrates the importance of including a temperature-dependent viscosity, by comparing the averaged peak velocities of fixed and adaptive mesh isoviscous and temperature-dependent simulations. As expected, the differences between isoviscous, where the viscosity is taken as the value at room temperature (see Table 2) throughout the domain, and temperature-dependent viscosity results increase with increasing power and temperature. At the highest power (3.3 W) there is a significant mismatch in the final velocity and plume development is delayed.

[55] Figure 9 illustrates plume development and evolution, for adaptive meshes, at three different power levels. For comparison, the final time frame from a fixed mesh simulation, using the same minimum element edge length as the adaptive mesh simulation, and a similar, though extruded, mesh to *Vatteville et al.* [2009], is shown. An excellent agreement is observed between the fixed and adaptive mesh simulations, despite the adaptive simulations having significantly fewer nodes (see Figure 8c). Adaptivity (including metric assembly, mesh optimization and subsequent interpolation) took an insignificant portion of the run time (~2%, see Table 3).

[56] While the minimum, maximum and average number of vertices (see Table 3) do not vary significantly for the majority of the simulation, the spatial distribution of resolution changes dramatically (see Figure 9 (i-iv)). High-resolution zones are based on the location of maxima in the field curvatures, which do not necessarily coincide with the field maxima themselves. Additionally, though the minimum edge length is limited to allow comparison with the fixed mesh results, achieving the desired interpolation error does not require attaining this lowermost bound throughout the entire simulation. These considerations, coupled with a Courant number defined time step, lead to a decrease in the number of time steps required (see Table 3), further reducing the computational cost.

[57] Considering the reduction in both the number of nodes and time steps as well as the insignificant

Figure 9. Several time frames showing isosurfaces of the temperature field, T, and underlying computational mesh, from the simulation of a thermal plume using Fluidity. A quarter of the experimental tank (0.075 m × 0.075 m × 0.161 m with a 0.009 m radius and 0.003 m high heater) is simulated, in three dimensions, using an adaptive mesh (i-iv) at three example heater power levels: (a) 0.6 W, (b) 1.7 W, (c) and 3.3 W. For comparison, a time frame from a fixed mesh simulation using the same minimum element edge length as the adaptive mesh simulations and an equivalent resolution to *Vatteville et al.* [2009] is shown (v). In the adaptive simulations elements become anisotropic, with their long edges aligned in the direction of upwelling flow, reducing the number of nodes required to represent the plume conduit (note that the number of vertices utilized for adaptive simulations are presented in Figure 8c). Isosurfaces are shown at the same temperatures in all time frames, for all heater power levels.

cost of mesh optimization, these results clearly demonstrate the advantages of mesh adaptivity for time-dependent problems, dramatically reducing the computational cost of a simulation, whilst maintaining solution accuracy.

6. Conclusions

[58] We have presented Fluidity, a finite element based computational modeling framework, which has several attributes ideally suited to geodynamical modeling: (1) the code uses an unstructured mesh, which enables the straightforward multi-resolution representation of complex geometry domains; (2) it dynamically optimizes this mesh, providing increased resolution in areas of dynamic importance, allowing for accurate simulations, across a range of length scales, within a single model. Mesh optimization is enhanced via anisotropic elements, which, as demonstrated herein, resolve one-dimensional flow features efficiently; (3) it uses implicit solvers, thus allowing for large time steps (advection at Courant numbers greater than one) with minimal loss of accuracy.

[59] Fluidity differs substantially from the majority of current mantle convection models, which are typically based upon fixed structured grids. Nonetheless, in this paper we have demonstrated the robustness and accuracy of Fluidity for simulations of mantle convection, via comparisons with a range of benchmark and experimental solutions. Furthermore, the significant advantages of anisotropic adaptive unstructured meshes have been shown: they dramatically reduce the computational cost of a simulation, whilst maintaining solution accuracy.

Acknowledgments

[60] DRD was funded by a Fellowship from the Royal Commission for the Great Exhibition of 1851. The authors would like to acknowledge support from the Applied Modelling and Computation Group (AMCG) at Imperial College London. Numerical simulations were undertaken at the following: (1) the Imperial College High Performance Computing (ICT-HPC) center and (2) HECToR, the UK's national high-performance computing service, which is provided by UoE HPCx Ltd at the University of Edinburgh, Cray Inc, and NAG Ltd and is funded by the Office of Science and Technology through EPSRC's High End Computing Program. The authors also benefited from discussion with Matthew Piggott, Gareth Collins, Saskia Goes, Marc Spiegelman, Carolina Lithgow-Bertelloni, William Newsome, Huw Davies, and Peter van Keken. Two anonymous reviewers are thanked for detailed and constructive reviews.

References

Alisic, L., M. Gurnis, G. Stadler, C. Burstedde, L. C. Wilcox, and O. Ghattas (2011), Slab stress and strain rate as constraints on global mantle flow, *Geophys. Res. Lett.*, *37*, L22308, doi:10.1029/2010GL045312.

Balay, S., W. D. Gropp, L. C. McInnes, and B. F. Smith (1997), Efficient management of parallelism in object-oriented numerical software libraries, in *Modern Software Tools for Scientific Computing*, pp. 163–202, Birkhauser, Cambridge, Mass.

Benzi, M., G. H. Golub, and J. Liesen (2005), Numerical solution of saddle point problems, *Acta Numer.*, *14*, 1–137.

Blankenbach, B., et al. (1989), A benchmark comparison for mantle convection codes, *Geophys. J. Int.*, *98*(1), 23–38, doi:10.1111/j.1365-246X.1989.tb05511.x.

Bunge, H.-P., M. A. Richards, and J. R. Baumgardner (1997), A sensitivity study of three-dimensionsal spherical mantle convection at 10^8 Rayleigh number: Effects of depth-dependent viscosity, heating mode and an endothermic phase change, *J. Geophys. Res.*, *102*(2), 11,991–12,007.

Bunge, H.-P., C. R. Hagelberg, and B. J. Travis (2003), Mantle circulation models with variational data assimilation: inferring past mantle flow and structure from plate motion histories and seismic tomography, *Geophys. J. Int.*, *152*(2), 280–301, doi:10.1046/j.1365-246X.2003.01823.x.

Burstedde, C., O. Ghattas, M. Gurnis, G. Stadler, E. Tan, T. Tu, L. C. Wilcox, and S. Zhong (2008), Scalable adaptive mantle convection simulation on petascale supercomputers, paper presented at 2008 ACM/IEEE Conference on Supercomputing, Inst. of Electr. and Electron. Eng., Austin, Tex.

Busse, F. H., et al. (1994), 3D convection at infinite Prandtl number in cartesian geometry: A benchmark comparison, *Geophys. Astrophys. Fluid Dyn.*, *75*(1), 39–59, doi:10.1080/03091929408203646.

Catalyurek, U., E. Boman, K. Devine, D. Bozdag, R. Heaphy, and L. Riesen (2007), Hypergraph-based dynamic load balancing for adaptive scientific publications, paper presented at 21st International Parallel and Distributed Processing Symposium (IPDPS'07), Inst. of Electr. and Electron. Eng., Long Beach, Calif.

Davies, D. R., and J. H. Davies (2009), Thermally driven mantle plumes reconcile multiple hot-spot observations, *Earth Planet. Sci. Lett.*, *278*(1–2), 50–54, doi:10.1016/j.epsl.2008.11.027.

Davies, D. R., J. H. Davies, O. Hassan, K. Morgan, and P. Nithiarasu (2007), Investigations into the applicability of adaptive finite element methods to two-dimensional infinite Prandtl number thermal and thermochemical convection, *Geochem. Geophys. Geosyst.*, *8*, Q05010, doi:10.1029/2006GC001470.

Davies, D. R., J. H. Davies, O. Hassan, K. Morgan, and P. Nithiarasu (2008), Adaptive finite element methods in geodynamics: Convection dominated mid-ocean ridge and subduction zone simulations, *Int. J. Numer. Methods Heat Fluid Flow*, *18*(7–8), 1015–1035, doi:10.1108/09615530810899079.

Elman, H. C., D. J. Silvester, and A. J. Wathen (2005), *Finite Elements and Fast Iterative Solvers: With Applications in Incompressible Fluid Dynamics*, Oxford Univ. Press, New York.

Farrell, P. E., and J. R. Maddison (2010), Conservative interpolation between volume meshes by local Galerkin projection, *Comput. Methods Appl. Mech. Eng.*, *200*, 89–100, doi:10.1016/j.cma.2010.07.015.

Farrell, P. E., M. D. Piggott, C. C. Pain, G. J. Gorman, and C. R. Wilson (2009), Conservative interpolation between unstructured meshes via supermesh construction, *Comput. Methods Appl. Mech. Eng.*, *198*, 2632–2642.

Geenen, T., M. ur Rehman, S. P. MacLachlan, G. Segal, C. Vuik, A. P. van den Berg, and W. Spakman (2009), Scalable robust solvers for unstructured FE geodynamic modeling applications: Solving the Stokes equation for models with large localized viscosity contrasts, *Geochem. Geophys. Geosyst.*, *10*, Q09002, doi:10.1029/2009GC002526.

Gorman, G. J., C. C. Pain, M. D. Piggott, A. P. Umpleby, P. E. Farrell, and J. R. Maddison (2009), Interleaved parallel tetrahedral mesh optimisation and dynamic load-balancing, in *Adaptive Modeling and Simulation*, edited by P. Bouillard and P. Diez, pp. 101–104, Int. Center for Numer. Methods in Eng., Barcelona, Spain.

Hassan, O., E. J. Probert, K. Morgan, and J. Peraire (1995), Mesh generation and adaptivity for the solution of compressible viscous high speed flows, *Int. J. Numer. Methods Eng.*, *38*, 1123–1148.

Hiester, H. R., M. D. Piggott, and P. A. Allison (2011), The impact of mesh adaptivity on the gravity current front speed in a two-dimensional lock-exchange, *Ocean Modell.*, doi:10.1016/j.ocemod.2011.01.003, in press.

Hughes, T. J. R., and A. N. Brooks (1982), Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations, *Comput. Methods Appl. Mech. Eng.*, *32*, 199–259.

Kramer, S. C., C. J. Cotter, and C. C. Pain (2010), Solving the poisson equation on small aspect ratio domains using unstructured meshes, *Ocean Modell.*, *35*, 253–263.

Liu, L., S. Spasojevic, and M. Gurnis (2008), Reconstructing Farallon plate subduction beneath North America back to the late Cretaceous, *Science*, *322*, 934–938, doi:10.1126/science.1162921.

Lohner, R., K. Morgan, and O. Zienkiewicz (1985), An adaptive finite element procedure for compressible high speed flows, *Comput. Methods Appl. Mech. Eng.*, *51*(1–3), 441–465, doi:10.1016/0045-7825(85)90042-8.

Manolopoulos, Y., A. Nanopoulos, A. Papadopoulos, and Y. Theodoridis (2006), *R-Trees: Theory and Applications*, Springer, London.

May, D., and L. Moresi (2008), Preconditioned iterative methods for Stokes flow problems arising in computational geodynamics, *Phys. Earth Planet. Inter.*, *171*(1–4), 33–47, doi:10.1016/j.pepi.2008.07.036.

McKenzie, D. P., J. M. Roberts, and N. O. Weiss (1974), Convection in the Earth's mantle: Towards a numerical simulation, *J. Fluid Mech.*, *62*(3), 465–538, doi:10.1017/S0022112074000784.

Pain, C. C., A. P. Umpleby, C. R. E. de Oliveira, and A. J. H. Goddard (2001), Tetrahedral mesh optimisation and adaptivity for steady-state and transient finite element calculations, *Comput. Methods Appl. Mech. Eng.*, *190*, 3771–3796, doi:10.1016/S0045-782500294-2.

Peraire, J., M. Vahdati, K. Morgan, and O. Zienkiewicz (1987), Adaptive remeshing for compressible flow computations, *J. Comput. Phys.*, *72*(2), 449–466, doi:10.1016/0021-9991(87)90093-3.

Piggott, M. D., G. J. Gorman, C. C. Pain, P. A. Allison, A. S. Candy, B. T. Martin, and M. R. Wells (2008), A new computational framework for multi-scale ocean modelling based on adapting unstructured meshes, *Int. J. Numer. Methods Fluids*, *56*(8), 1003–1015, doi:10.1002/fld.1663.

Saad, Y. (1993), A flexible inner-outer preconditioned GMRES algorithm, *SIAM J. Sci. Comput.*, *14*(2), 461–469.

Stadler, G., M. Gurnis, C. Burstedde, L. C. Wilcox, L. Alisic, and O. Ghattas (2010), The dynamics of plate tectonics and mantle flow: From local to global scales, *Science*, *329*, 1033–1038.

Syracuse, E. M., P. E. van Keken, and G. A. Abers (2010), The global range of subduction zone thermal models, *Phys. Earth Planet. Int.*, *183*, 73–79, doi:10.1016/j.pepi.2010.02.004.

Tackley, P. J. (2008), Modelling compressible mantle convection with large viscosity contrasts in a three-dimensional spherical shell using the yin-yang grid, *Phys. Earth Planet. Int.*, *171*, 7–18, doi:10.1016/j.pepi.2008.08.005.

Turcotte, D. L., and E. R. Oxburgh (1967), Finite amplitude convection cells and continental drift, *J. Fluid Mech.*, *28*, 29–42.

Vanek, P., J. Mandel, and M. Brezina (1996), Algebraic multigrid by smoothed aggregation for second and fourth order elliptic problems, *Computing*, *56*(3), 179–196, doi:10.1007/BF02238511.

Vasilevski, Y. V., and K. N. Lipnikov (1999), An adaptive algorithm for quasioptimal mesh generation, *Comp. Math. Math. Phys.*, *39*, 1468–1486.

Vatteville, J., P. E. van Keken, A. Limare, and A. Davaille (2009), Starting laminar plumes: Comparison of laboratory and numerical modeling, *Geochem. Geophys. Geosyst.*, *10*, Q12013, doi:10.1029/2009GC002739.

Verfurth, R. (1984), A combined conjugate gradient-multigrid algorithm for the numerical solution of the Stokes problem, *IMA J. Numer. Anal.*, *4*(4), 441–455, doi:10.1093/imanum/4.4.441.

Wilson, C. R. (2009), Modelling multiple-material flows on adaptive unstructured meshes, Ph.D thesis, Imperial College London, London.

Wu, J., J. Z. Zhu, J. Szmelter, and O. C. Zienkiewicz (1990), Error estimation and adaptivity in Navier-Stokes incompressible flows, *Comput. Mech.*, *6*(4), 259–270, doi:10.1007/BF00370106.

Zhong, S., M. T. Zuber, L. Moresi, and M. Gurnis (2000), Role of temperature-dependent viscosity and surface plates in spherical shell models of mantle convection, *J. Geophys. Res.*, *105*(B5), 11,063–11,082, doi:10.1029/2000JB900003.

Zienkiewicz, O. C., and R. L. Taylor (1991), *The Finite Element Method*, vol. 2, 4th ed., McGraw-Hill, New York.