



Ellipsis 3D: A particle-in-cell finite-element hybrid code for modelling mantle convection and lithospheric deformation

Craig O'Neill, Louis Moresi, Dietmar Muller, Rich Albert, Frédéric Dufour

► To cite this version:

Craig O'Neill, Louis Moresi, Dietmar Muller, Rich Albert, Frédéric Dufour. Ellipsis 3D: A particle-in-cell finite-element hybrid code for modelling mantle convection and lithospheric deformation. Computers & Geosciences, 2006, 32, pp.1769-1779. 10.1016/j.cageo.2006.04.006 . hal-00334536

HAL Id: hal-00334536

<https://hal.science/hal-00334536>

Submitted on 15 Jun 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Ellipsis 3D: A particle-in-cell finite-element hybrid code for modelling mantle convection and lithospheric deformation[☆]

Craig O'Neill^{a,*}, Louis Moresi^b, Dietmar Müller^a, Rich Albert^c, Frédéric Dufour^d

^a*School of Earth Sciences, The University of Sydney, Sydney, NSW, 2006, Australia.*

^b*School of Mathematical Sciences, Monash University, Victoria, Australia*

^c*Exxon Mobil Exploration Co., Houston, TX, USA*

^d*Ecole Centrale de Nantes, France*

We have extended the two-dimensional geodynamics finite-element code “Ellipsis” to three-dimensions. Ellipsis uses a hybrid particle in a cell scheme, which combines a fixed mesh of computational points and a dense arrangement of mobile material points. The fixed Eulerian mesh allows very fast computation, performed in Ellipsis via a multigrid iteration method, while the Lagrangian particle reference frame allows the tracking of material interfaces and history-dependent properties such as strain history for strain-softening materials. The method is exceptionally useful in very large deformation analyses, where purely Lagrangian approaches would be severely hampered by the need for remeshing to minimize element distortion. The Gnu Public Licensed Ellipsis3D code lends itself to combined 2D/3D model prototyping, and has proven to be an excellent geodynamics teaching tool for modelling, covering mantle convection, lithospheric extension and plume–lithosphere interaction.

Keywords: Mantle convection; Finite-element; Particle-in-cell; Lithosphere; Mixed scheme

1. Introduction

The finite-element method has been a popular approach for problems involving mantle convection and lithospheric deformation, due to its robustness and the existence of many efficient and reliable solvers. For study of lithospheric deformation, where deformation is viscoelastic/plastic with his-

tory-dependent rheology, Lagrangian finite-element formulations (in which the mesh is embedded in and deforms with the material) are ideal because they allow simple tracking of history variables (e.g., damage parameters). However, for problems on the mantle scale, where deformation is extremely large, Lagrangian finite-element methods are limited by severe mesh distortions. While this can be overcome by re-meshing, the computational expense of recalculating the mesh can be high, and history variables need to be interpolated to points in the new elements. Eulerian finite-element methods, in which the mesh is fixed in space and the material flows relative to the mesh, are the preferred

[☆]Code available from server at <http://www.iamg.org/CGEditor/index.htm>.

*Corresponding author. Present address: Department of Earth Sciences, Rice University, Houston, TX, USA.
Tel.: +1 713 348 4567; fax: +1 713 348 2356.

E-mail address: cjoneill@rice.edu (C. O'Neill).

alternative when modelling thermal convection. However, significant numerical diffusion (Lenardic, 1993; Van Keken et al., 1997) of interfaces limits the usefulness of this method in modelling several materials at once, and difficulties arise in tracking material-dependent properties, such as strain weakening or depletion (Tackley, 1998). Moresi et al. (2003) introduced a particle-in-cell finite-element code (Ellipsis) to overcome these difficulties. Ellipsis is based on the Eulerian finite-element method. However, instead of basing the integration points on a standard Gaussian quadrature scheme, the integration points are chosen to coincide with material points which are advected through the mesh. This enables the use of standard, proven, structured grid solvers, such as the multigrid method. It also enables the modelling of numerous, interacting materials, without the concern of numerical diffusion. History-dependent properties, such as strain, strain rate, or depletion due to melt extraction, are considered particle properties, not properties of the mesh, which alleviates problems with the standard Eulerian formulation.

In this paper, we extend this methodology into three dimensions with Ellipsis3D. The code is able to handle viscoelastic-plastic rheologies with strain weakening and depletion, with complicated, evolving boundary conditions. We briefly present the mathematical and numerical formulation of Ellipsis3D, together with an introduction to the code. We also present the results of numerical benchmarks, and four examples highlighting the capabilities of Ellipsis3D in modelling crustal-scale to mantle-scale processes.

2. Mathematical framework

Mantle convection simulations involve the solution to the coupled equations of momentum and energy (Moresi and Solomatov, 1998), subject to an incompressibility constraint. The momentum equation is generally written as

$$\nabla \cdot \sigma = \mathbf{g} \rho_0 \alpha T,$$

$$\sigma_{ij} = 2\eta D_{ij} - p \delta_{ij}. \quad (2.1)$$

Here η is the viscosity, $D_{ij} = (\partial v_i / \partial x_j + \partial v_j / \partial x_i) / 2$ is the rate of deformation tensor, p is the dynamic pressure, ρ_0 the reference density, \mathbf{g} the gravitational acceleration, α the thermal expansivity, and T the

temperature. The above equation is essentially the Navier–Stokes equation, without the inertial terms (i.e., Stokes equation). This assumption is justified given the large viscosity of the mantle, and the short timescale for the re-equilibration of body forces compared to convection timescales. To simplify the solution, the mantle is considered incompressible, i.e.,

$$\nabla \cdot \mathbf{v} = 0. \quad (2.2)$$

The assumption that the flow is incompressible apart from terms where density provides buoyancy forces is known as the Boussinesq approximation. It is an appropriate assumption for most mantle convection studies (Hansen and Yuen, 2000).

The time dependence of the problem is introduced through the energy equation:

$$\frac{D T}{D t} = \kappa \nabla^2 T + Q. \quad (2.3)$$

Here D/Dt is the material time derivative (taken at a point moving with respect to the fluid), κ the thermal diffusivity, and Q the volumetric heat production.

These equations describe the fundamental physics required for modelling mantle convection; however, their implementation depends on the scheme employed. The dominant influence on the solution of this coupled set of equations is the temperature, pressure, stress and history dependence of the viscosity term, η . The strong spatial variations in viscosity, mainly due to the temperature dependence, suggest the use of specialized solution methods optimized for problems with strong gradients in material coefficients, for which more research has been done in the context of Eulerian formulations. The history-dependent rheology, on the other hand, promotes the use of a Lagrangian reference frame.

Ellipsis (Moresi, et al., 2003; Moresi, et al., 2001) uses a particle-in-cell finite-element scheme which attempts to combine these two requirements. Its implementation is described in the following sections.

3. Particle-in-cell finite-element formulation of mantle convection

The motivation for Ellipsis was to combine the functionality of a particle method, in particular features like tracking material interfaces, tracking

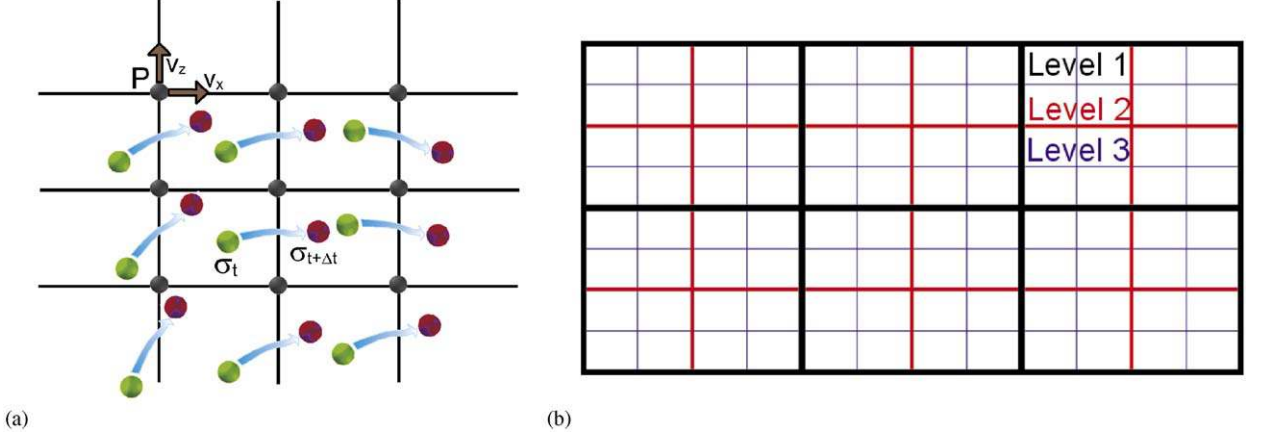


Fig. 1. Representation of the particle-in-cell in finite-element scheme: A conventional Eulerian finite-element grid is used, but instead of a Gaussian quadrature scheme to determine the distribution of integration points, the integration points coincide with material points, which are advected with the flow. Representation of the multigrid solver: A solution is found at the coarsest grid size (level 1), and the nodal solutions interpolated to successively finer grids as an initial solution for the next iteration.

material damage and degree of depletion, with a robust, reliable multigrid solver (Moresi, et al., 2003, see Fig. 1).

It is beyond the scope of this paper to introduce fully the finite-element method, and its application to convection problems; a number of good references exist (e.g., Hughes, 1987). Additionally, the implementation of a particle-based finite-element method for mantle convection problems is detailed in Moresi et al. (2003). Here we focus here on a working understanding of the grid-particle inter-relationship in Ellipsis3D.

The finite-element method always begins by transforming the conservation Eqs. (2.1)–(2.3) to a weak or variational form. The weak form takes the form of an integral over the solution domain. The advantages of this are: (1) problems with integrable discontinuities can be treated in a straightforward manner; (2) flux boundary conditions become a natural part of the problem; and most importantly (3) the “global integral” can be expressed naturally as a sum of sub-integrals over sub-domains, or elements (Hughes, 1987), that is, the problem domain is split into smaller elements—in Ellipsis3D, this takes the form of a Cartesian grid. The “weak-form” integrals, over continuous variables, are then replaced by discrete approximations, for the purposes of numerical computation. This is done by introducing interpolation functions, which interpolate unknowns in the element interiors from the values at the element vertices. The integrals are then replaced with weighted sums over a small number of

sample points within the elements, i.e.,

$$\int_{\Omega_e} \psi \, d\Omega_e \sum_{p=1}^P w_p \psi(x_p). \quad (3.1)$$

Here Ω_e is the element volume, ψ is the integrand evaluated at the sample points x_p , P is the number of sample points within the element volume (also known as integration points), and w_p are the weights of each integration point. So far, this approach is the same for standard finite-element methods, and Ellipsis3D. However, in conventional finite-element methods, the integration points are usually picked according to some scheme to get the most accurate integral for the smallest number of points (an example is the Gaussian quadrature scheme). In the particle-in-cell finite-element method, instead of optimally chosen integration points, the sample points are actually the particle positions. These particles move with respect to the mesh. The particle weights are varied to provide the correct integration, with a method described in more detail in Moresi et al. (2003). However, since the positions of the particles are not, as a rule, optimal, several times more particles per element are required for accuracy, compared to Gaussian quadrature points in an equivalent finite-element scheme. This is a disadvantage, but it is offset by an improved accuracy in integrating material properties at the actual material points where the information is tracked, which avoids the need to interpolate information to fixed integration points.

The discrete representation of the integral equations described above produces a matrix equation in a form which is common to most finite-element codes designed to solve problems of mechanical equilibrium. For details of the matrix form see Moresi et al. (2003). The pressure and velocity are solved using an Uzawa scheme employing a conjugate gradient method, eliminating the need for a separate back substitution to obtain velocity (Moresi & Solomatov, 1995).

The Uzawa iteration is coupled to a multigrid solver. Here, an approximate solution is calculated on the coarsest mesh, and is interpolated to a finer mesh, and used as the starting point for the iteration. The energy equation is solved using explicit time stepping of a Petrov-Galerkin upwind method (Moresi and Solomatov, 1995).

4. Ellipsis3D

4.1. Input

Ellipsis3D has been developed in C, and has been extensively tested under Linux and Macintosh operating systems. The problem to solve is defined in an input ASCII file, several of which are provided with the online documentation of this paper. The input file is fairly detailed, allowing a great deal of flexibility in the problem set-up.

The input file is divided into a number of subsections. Within it, the user specifies the standard output required, e.g., the advection-diffusion and solver-related parameters. The grid is assumed Cartesian, with the z -dimension being positive downward by default. The user specifies the coarsest grid units, and the number of times to subdivide this grid (i.e., the number of multigrid levels). The grid can be refined in regions where greater detail is required.

In the initial conditions, the user specifies the initial temperature distribution, including spatial perturbations. The user can also define the initial material, temperature and strain, by defining the vertices of rectangular or triangular prisms, or by defining the position and radii of spheres. A material type is assigned to each distribution. If the distributions overlap, the last defined distribution is effective. Nodal or particle files from previous runs can be input as initial temperature, plastic strain, or material distributions. The initial distribution of integration points, or tracers, and their density are also defined under the initial conditions.

Stress, velocity, and temperature boundary conditions are implemented in Ellipsis3D. The boundary conditions can be rectangular, circular, harmonic, or polynomial. All boundaries are free-slip by default, and the side boundaries can be periodic if required. The grid itself can be defined to change through time, e.g., expanding in an extensional model. It is possible to define time-varying deformation rates, in varying dimensions (see our example input file).

Ellipsis outputs a number of ascii files, ppm image files, and binary files, depending on what is requested in the input template. Nodal information, such as velocity, temperature, and pressure, can be exported to an ascii data file containing this information at each node. Similarly, a binary particle file can be created containing particle data at each particle position. Ellipsis3D can also create ascii files containing horizontal averages or surface observables of required data at each timestep. A time-log file of large-scale averages such as the rms velocity (total or directional, or at the surface), the Nusselt number, the surface or basal heat flow can also be created.

Sampling tracers can be used to create profiles in a given direction, or construct histories for a particle. The tracers can be Eulerian or Lagrangian, and can sample such properties as temperature, velocity, pressure, stress or strain. They can also be plotted on image files generated as the simulation progresses.

The standard image files generated by Ellipsis3D are ppms (Portable Pixel Map)—one the simplest image file formats to output. Fields to be plotted include temperature, viscosity, stress, pressure, strain, strain rate, depletion etc. The ppm colouring scheme is defined under the material properties. Data from nodes and particles can also be output to plot in alternative programs.

The material set-up is defined in the initial conditions. The materials themselves are defined individually, so that physical properties such as density, and thermal properties such as thermal expansivity, thermal diffusivity, heat capacity, and internal heat generation, can vary for different materials.

4.2. Physical models within Ellipsis

4.2.1. Viscosity

The effective viscosity of the mantle is a combination of both diffusion creep and dislocation

creep processes, which are described by the Arrhenius formulation (Karato and Wu, 1993):

$$\eta = B \exp \left[\frac{E + PV}{RT} \right] \sigma^{1-n}. \quad (4.1)$$

Here E is the activation energy, P the pressure, V the activation volume, R the universal gas constant, T the temperature, σ a scalar measure of stress (such as the differential or second invariant of deviatoric), and n the power-law exponent. The mantle can flow either by diffusion creep or dislocation creep, and the effective viscosity of the mantle is the smaller viscosity of the two. However, we only include one viscosity formulation at a time in Ellipsis3D. A simplification of this relationship, which is useful for purely temperature-dependent convection:

$$\eta = A \exp[-T_1 T]. \quad (4.2)$$

This is known as the Frank–Kamenetski approximation (Frank-Kamenetskii, 1969), the constants A and T_1 define the reference viscosity and viscosity range over the range of temperatures experienced, respectively. An arbitrary number of deformation mechanisms can be constructed for each material in the mesh. These are combined as follows:

$$\frac{1}{\eta} = \frac{1}{\eta_1} + \frac{1}{\eta_2} + \frac{1}{\eta_3} + \dots + \frac{1}{\eta_n} \quad (4.3)$$

to form the final effective viscosity for each material point under its current conditions.

4.2.2. Brittle deformation

A further mechanism of deformation relevant for near-surface deformation is brittle failure (Fig. 2). This behaviour occurs once the material's stress

attains a yield stress, defined by

$$\tau_{\text{yield}} = C_0 + \mu \sigma_n. \quad (4.4)$$

Here C_0 is the cohesion, μ the coefficient of friction, and σ_n the stress normal to the plane of failure, which is approximated by the lithostatic pressure (Moresi and Solomatov, 1998). The actual failure mechanism is modelled by means of an effective viscosity:

$$\eta_{\text{yield}}(D) = \frac{\tau_{\text{yield}}}{2D}. \quad (4.5)$$

D is the second invariant of the deviatoric strain-rate tensor. Thus the method of failure at a point is determined by the material stress; for values of stress below the yield stress at a given pressure, the viscosity is given by Eq. (4.2), and the fluid is purely viscous. For stresses that attain the yield stress, the material fails, and the deformation is governed by Eq. (4.4).

4.2.3. Strain and strain-rate weakening

Tackley (1998) has shown that the realistic generation of plates in convection simulations requires some sort of strain or strain-weakening mechanism, to stabilize the zones of failure, and allow for the integrity of rigid, non-deforming plates. These damage terms are incorporated in Ellipsis3D by modifying Eq. (4.4):

$$\tau_{\text{yield}} = (C_0 + \mu \sigma_n) \times f(\varepsilon, \dot{\varepsilon}), \quad (4.6)$$

where ε and $\dot{\varepsilon}$ are the strain and strain rate, respectively, and f is given by

$$f(\varepsilon, \dot{\varepsilon}) = 1 - (1 - E_a)(\varepsilon/\varepsilon_0)^{n_1} - (1 - E_a)(\dot{\varepsilon}/\dot{\varepsilon}_0)^{n_2}. \quad (4.7)$$

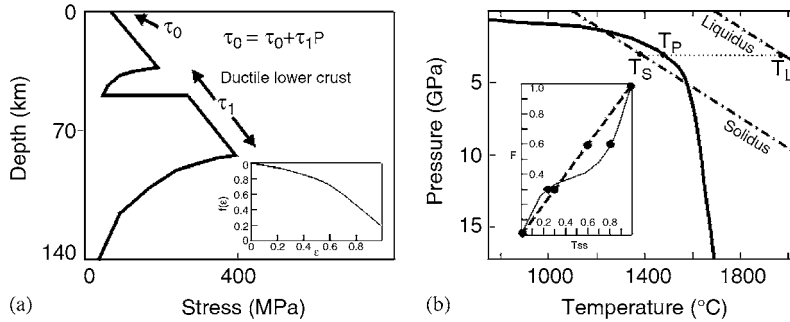


Fig. 2. (a) Strength of lithosphere as a function of depth: For shallow depths, the lithosphere is cold and the viscosity is high. However, the yield stress τ is low at low pressures, and the lithosphere deforms by brittle deformation. At higher temperatures, the viscosity is lower, and the material deforms viscously (lower crust and lower lithosphere). *Inset*: strain weakening term $f(\varepsilon)$ as defined in Eq. (4.7), as a function of strain ε (similar for strain rate). (b) Illustrative geotherm and linear solidus and liquidus, and particle temperature (T_p) as used in Eq. (4.11). *Inset*: relationship between supersolidus temperature T_{ss} and melt fraction F defined in Eq. (4.12).

Here ε_0 and $\dot{\varepsilon}_0$ are the strain and strain-rate weakening terms, respectively. E_a is defined as the ratio of $f(0,0)/f(\varepsilon_0,0)$, and E_s is the ratio of $f(0,0)/f(0,\dot{\varepsilon}_0)$. The exponents n_1 and n_2 describe the non-linearity of the response of f to strain and strain-rate. A visualization of this relationship is shown in Fig. 2.

4.2.4. Elasticity

Ellipsis3D uses a Maxwell viscoelastic model, where the strain rate tensor (Eq. (2.1)) is assumed to be the sum of the elastic and viscous strain rate tensors, D_e and D_v :

$$\frac{\nabla}{2\mu} + \frac{\tau}{2\eta} = \hat{D}_v + \hat{D}_e = \hat{D}, \quad (4.8)$$

where τ is the stress tensor, μ is the shear modulus, and η is the shear viscosity. D is the deviatoric part of the strain rate tensor. To maintain mechanical objectivity, Ellipsis incorporates the Jaumann corotational stress rate, $\dot{\tau}$, for a material point, given by

$$\dot{\tau} = \dot{\tau} + \tau W - W \tau, \quad (4.9)$$

where W is the spin tensor,

$$W_{ij} = \frac{1}{2} \left(\frac{\partial V_i}{\partial x_j} - \frac{\partial V_j}{\partial x_i} \right). \quad (4.10)$$

See Moresi et al. (2003) for details of the numerical implementation.

4.2.5. Melt depletion

A parameterization of melt production, assuming batch melt extraction and depletion of the mantle residuum similar to de Smet et al. (1998, 1999), has also been incorporated in Ellipsis3D. We define a supersolidus temperature T_{ss} , where

$$T_{ss} = \frac{T - T(P)_s}{T(P)_l - T(P)_s}, \quad (4.11)$$

where T is the temperature of a particle tracer, P is the pressure, $T(P)_s$ and $T(P)_l$ are the solidus and liquidus temperatures at pressure P , for the peridotite solidus of (Takahashi and Kushiro, 1983) and the liquidus of (McKenzie and Bickle, 1988). McKenzie and Bickle (1988) demonstrated a relationship between the T_{ss} and melt fraction F , given as

$$F = 2.0684T_{ss} - 4.0564T_{ss}^2 + 2.988T_{ss}^3. \quad (4.12)$$

The production of melt results in an additional latent heat term in the energy equation, which

becomes

$$\frac{DT}{Dt} = \kappa \nabla^2 T + Q - T \frac{\Delta S DF}{c_p \frac{DT}{Dt}}. \quad (4.13)$$

Here T is the temperature, t the time, κ the thermal diffusivity, Q a volumetric heat source term, ΔS the entropy change on melting, c_p the thermal capacitance, and F the fraction of partial melting. The effect on the density of the residuum has been modelled by assuming an extra term in the constitutive relationship:

$$\rho = \rho_0(1 - \alpha \Delta T - \beta F). \quad (4.14)$$

Here ρ and ρ_0 are the density and reference density, α the thermal expansivity, ΔT the difference between the temperature and the reference temperature, F the degree of melt depletion, and F the density of change of material upon full melt depletion.

5. Benchmarks

A three-dimensional convection benchmark was outlined in Busse et al. (1993). We have benchmarked Ellipsis3D against their solutions, and the results are shown in Fig. 3. We tested the Nusselt number, heat flux at various points on the surface, the root mean square (RMS) velocity, the velocity at various mid-depth points, and the temperature at mid depths, against the benchmarked solution, for the configuration shown in Fig. 3(a). The configuration is an example of isoviscous bimodal convection at a Rayleigh number of 10^4 , in an $a \times b \times 1$ box, where $a = 1.0079$ and $b = 0.6283$. The particle density for all cases is 64 (i.e., $4 \times 4 \times 4$ particles in the finest element). For smaller particle densities, the integration in the finest multigrid level may not be accurate. However, we have found that the improvement in accuracy for higher particle densities places a greater constraint on performance than increasing the multigrid level, and $4 \times 4 \times 4$ particles per finest element is the optimal trade-off between speed and accuracy. The top and bottom surfaces are no-slip (i.e., zero velocity), and the side boundaries are assumed to have mirror symmetry (the default in Ellipsis). The initial conditions are chosen so that we have ascending flow at $(x, y) = (0, 0)$ and descending flow at $(x, y) = (a, b)$. The simulation was allowed to run to a statistical steady state. As shown in Fig. 3(b), (c) and (d) the nodal velocities, surface heat flux and temperatures all converge towards their benchmark solutions for increasing grid resolution. The resolution we could

reach is limited in this problem; however, by the necessity of loading and dynamically storing information on all the particles, a $24 \times 24 \times 24$ grid is probably the current problem limit on a single processor.

We also performed a test of our viscoelastic rheology also by comparing our modelled results for a two-dimensional compressible square, and three-dimensional compressible cube, with an analytical

solution. The configuration is shown in Fig. 3(e). The analytical solution is derived from the constitutive equation for pressure:

$$\frac{\dot{P}}{K_e} + \frac{P}{\xi} = -\varepsilon_{yy} = -\frac{V_y}{h(t)} = -\frac{V_y}{h(0) + V_y t}. \quad (5.2)$$

Here K_e and ξ are the elastic (storage) modulus, and the viscosity (loss) modulus respectively, and V_y is the vertical velocity (see Fig. 3). When the velocity is not zero, a variational method can be used to obtain:

$$P(t) = \lambda(t)e^{-(K_e/\xi)/t}, \quad (5.3)$$

where

$$\lambda(t) = - \int_0^t \frac{K_e V_y}{h(t)} e^{(K_e/\xi)t} dt. \quad (5.4)$$

This is integrated numerically to obtain the analytical solution for compression. A comparison between analytical solution, and our results for two and three dimensions, are shown in Fig. 3(f), and agree well over the range explored.

6. Examples

The following examples are designed to highlight the capabilities of the code, and provide starting templates for potential users of laboratory exercises. These examples are of low resolution, and take about an hour of real time to complete on a

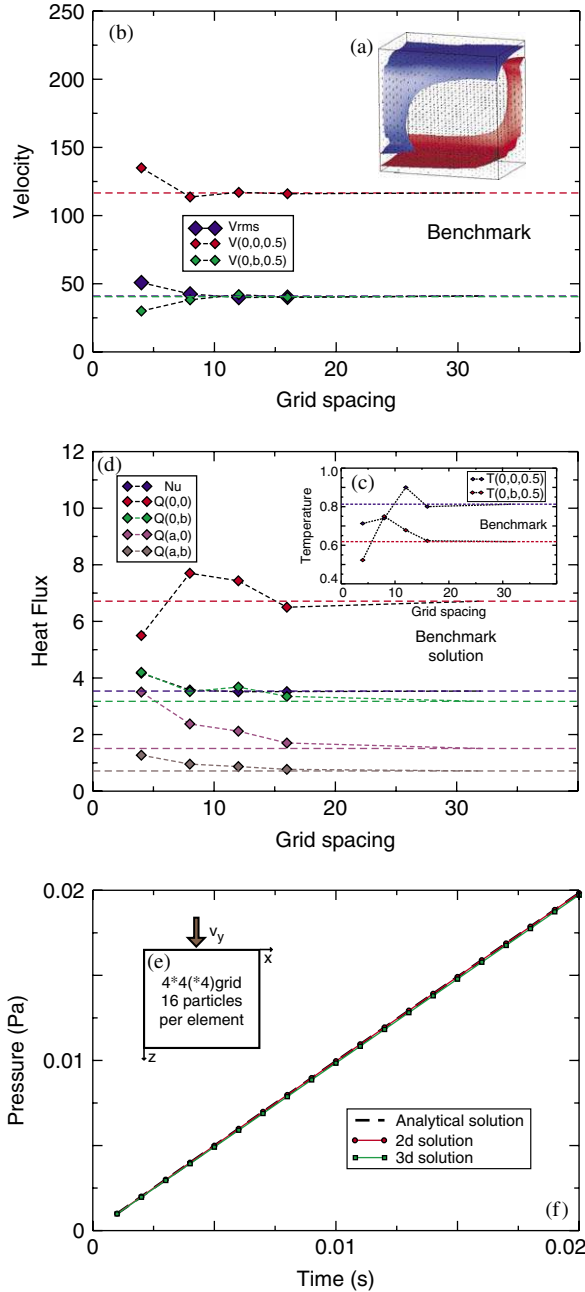


Fig. 3. (a) (*Inset*) 3D isoviscous benchmark example of Busse et al. (1993) benchmark, in an $a \times b \times 1$ geometry ($a = 1.0079$, $b = 0.6283$). This example uses $12 \times 12 \times 12$ grid cells, with a particle density of 64 per cell. Shown are isosurfaces of hot ($T > 0.7$, red) and cold ($T < 0.3$, blue) temperatures. Arrows represent the velocity field. The Rayleigh number is 10^4 , the top and bottom surfaces are no-slip (i.e., zero velocity), and the side boundaries are assumed to have mirror symmetry. (b) Steady-state velocities vs. grid size for the configuration shown in (a). Grid size increases from $4 \times 4 \times 4$, to $8 \times 8 \times 8$, to $12 \times 12 \times 12$ and $16 \times 16 \times 16$. Particle density was held constant at 64 per finest element (higher grid resolutions were not possible at this particle density). We measured the RMS velocity of the system, and the vertical nodal velocities at two points at mid depth. The solution converges to Busse et al. (1993) benchmark solution (horizontal dashed lines) for increasing grid size. (c) (*Inset*) Temperature vs. grid size, for temperatures for two points at mid depth ($z = 0.5$). (d) Heat flux (Q), or Nusselt number (Nu) for increasing grid size, for the Busse et al. (1993) benchmark example shown in (a). Heat flux is measured at different points on the surface ($z = 0$). (e) Schematic of our elastic benchmark of a compressed square. (f) Pressure vs. time for the analytical, 2D and 3D case.

Pentium IV processor using Linux. Ellipsis3D is, at this stage, is only a serial code, and each problem can be run on only one processor.

6.1. Chemical plume

Figure 4 shows the material field for a chemical plume. The visualization of all 3D output is accomplished using the freely available OpenDX software. The nodal velocities are shown as small arrows. The input files for the simulations described here, as well as instructions and scripts for plotting the results with OpenDX, are included in the code distribution. Both the plume material, and background mantle material are isoviscous in this example, with the plume having a viscosity 100 times less than that of the surrounding mantle. The large plume head, followed by a thinner conduit, shows fairly typical behaviour for plumes observed in laboratory experiments, which have lower viscosities than the material surrounding them (see van Keken, (1997), for comparison. Benchmark is similar to this example, but not performed due to the resolution required). The plume's density was 100 times less than that of the surrounding mantle.

The grid used in this example is a $3 \times 3 \times 3$ grid, with three multigrid levels. For this problem, the resolution of the problem takes about an hour to run on a desktop PC using Linux. Two tracer regions were defined, with nine tracers per element in the background material, and 15 tracers per element within the plume material. A linear temperature gradient exists at the start of the simulation, and the boundary conditions are free slip, and reflective (zero normal velocity) at the sides.

6.2. Oblique-forced subduction

Figure 5 shows an example of forced subduction. In this example, we have defined two materials, mantle and lithosphere (purple). The nodal velocities are shown as arrows. The surface colourfield shows the second invariant of the deviatoric strain rate tensor, indicating regions of high-accumulated distortional strain. This example contains two velocity conditions: one, rectangular velocity condition extending from 0.42 to 0.99 in the x -direction, and 0.01–0.99 in the y -direction. It has a magnitude of -1 , and corresponds to the red arrows in Fig. 5. The second velocity condition is the circular region of zero velocity, situated at $x = 0$, $y = 0$. These two

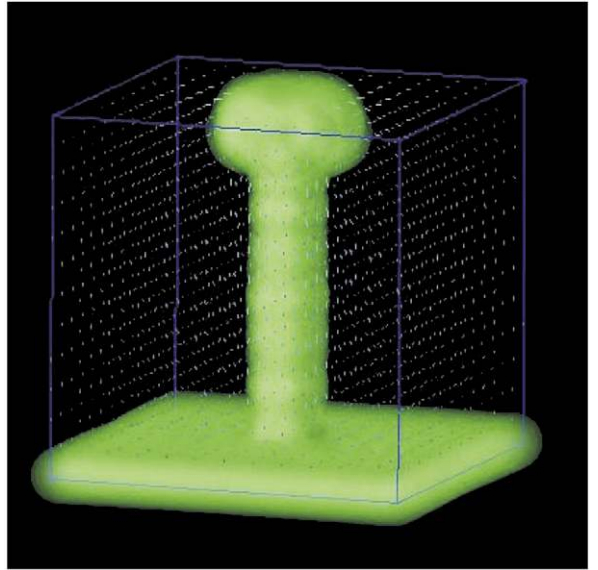


Fig. 4. A chemical plume: A $3 \times 3 \times 3$ grid is used with three multigrid levels. The plume and mantle are isoviscous, with a viscosity contrast of 100, and a variable tracer density.

velocity conditions serve to force subduction of the lithospheric material in the region between them.

The grid in this example is a $4 \times 4 \times 4$ grid with two multigrid levels, and 64 tracers per finest element. The side boundary conditions in the x -direction are periodic, those in the y are reflecting. The lithosphere is more dense than the underlying mantle. The mantle is purely isoviscous, while the lithosphere deforms viscoplastically, using a pressure-dependent yield stress, and including strain weakening.

6.3. Two-layered extension model

Figure 6 shows a three-layered crustal extension model, with a brittle upper crust, ductile lower crust, and compressible air layer. The particles constituting the upper crust are shown as orange spheres, those of the lower crust as yellow spheres. Plastic strain is also shown, either as a superposed translucent grid on the right side of the box, and as a red translucent contour surface delineating the areas with a plastic strain of 0.2. The air layer is not shown. A compressible material is necessary in the simulations in order to satisfy continuity; i.e., as the mesh extends and the crust is pulled apart, the air, which has a positive bulk viscosity, expands to fill the gap. The tracers may split and merge under deformation and mesh extension to ensure that the

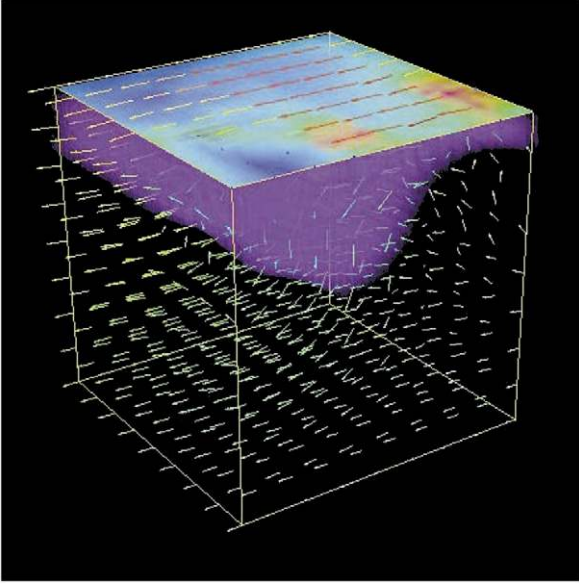


Fig. 5. Oblique forced subduction: Lithosphere is shown as purple, and the nodal velocities are shown as arrows. The surface colourfield indicates regions of high accumulated distortional strain. Two velocity conditions are used (see text).

modelling domain is sufficiently resolved at all times, the details of whose interactions are covered in Moresi et al. (2003).

The lower crust has a low viscosity, and deforms in a purely viscous manner. The upper crust, in contrast, has a viscosity 1000 times greater than the lower crust. The stresses generated in the upper crust by the extension of the box reach the yield stress for the material, and failure occurs. Areas undergoing plastic strain in this example are plotted as a red surface, and also the sidewall of the box consists of a coloured grid of the plastic strain at that place.

Ellipsis3D allows for time-dependent mesh extension, which was used in this example. The currently available time dependence involves combinations of up to three linear segments (with or without piecewise continuity). For example, the extension of the mesh might start with a linear increase in boundary velocity from zero to a target velocity for the first segment. This is followed immediately by a period of constant (or perhaps still increasing) velocity during the second segment, immediately followed by a linear decrease in velocity from the current velocity back to zero. Separate sequences or sets of these three segment linear functions are allowed, permitting isolated periods of extension/

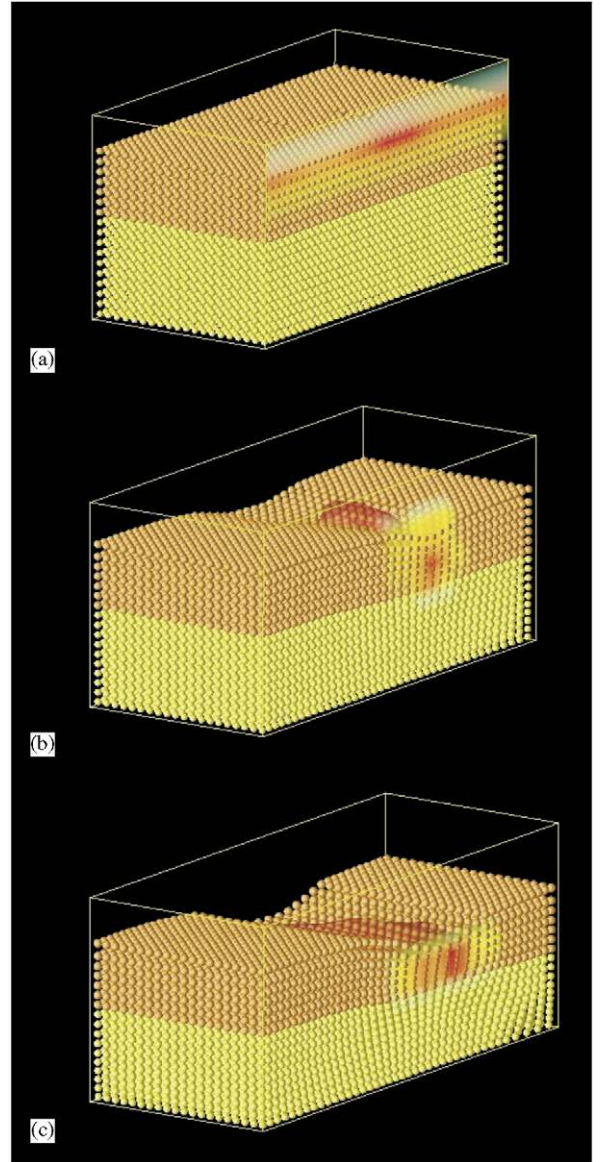


Fig. 6. A three-layered crustal extension model, with a brittle upper crust, ductile lower crust, and compressible air layer. Orange spheres denote the upper crust, and yellow spheres denote the lower crust. Plastic strain is also shown, either as a superposed translucent grid on the right side of the box, and as a red translucent contour surface delineating the areas with a plastic strain of 0.2.

compression interrupted by times of tectonic quiescence.

6.4. Plate flexure

Figure 7 shows the flexure of a viscoelastic plate under an applied load, in this case the weight of a

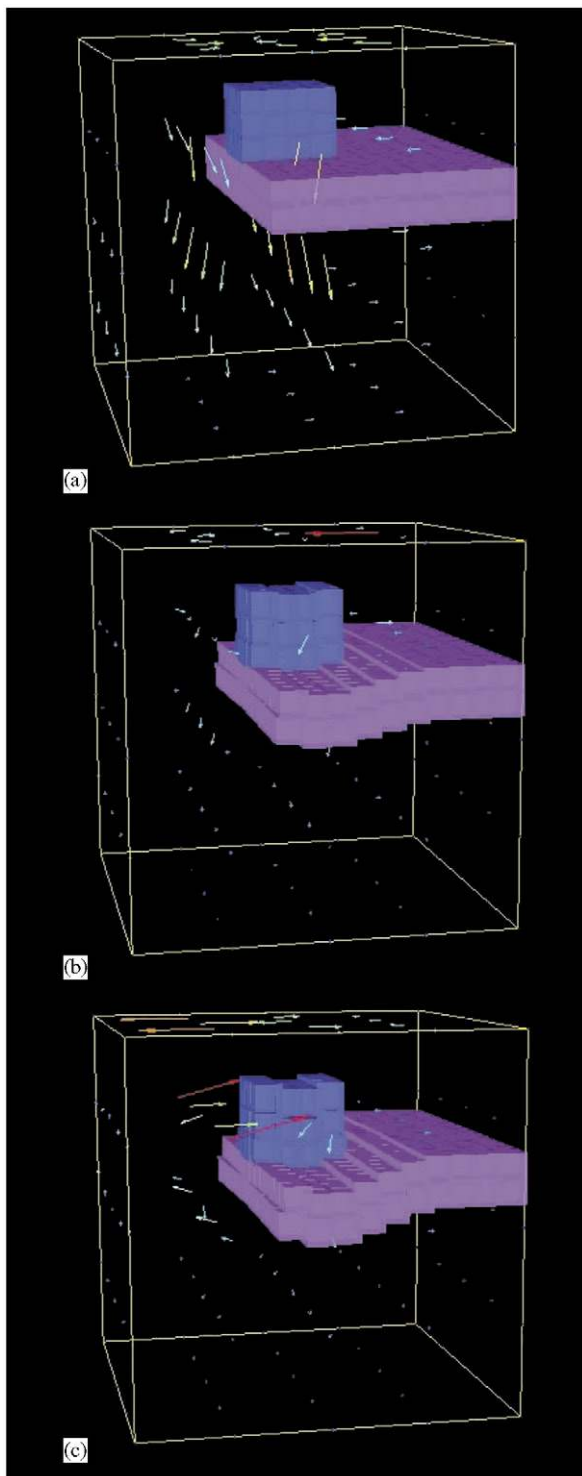


Fig. 7. Flexure of a viscoelastic plate (plate is purple): The plate has a density of 3300 kg/m^3 . The load (blue) has similar physical properties, but with a density of 8000 kg/m^3 . Arrows denote the velocity.

dense block on top of the free edge of the plate. This example has parallels in geodynamics, e.g., the loading of oceanic plate by a seamount, near a plate boundary. The units in this example are non-dimensionalized for simplicity. In this case, the viscoelastic plate is shown as purple, and has a density of 3300. The load (blue) has similar physical properties, but with a density of 8000. The surrounding area is filled with air. The plate is held in place at its attached edge by two rectangular velocity conditions. The plate has a viscosity of $1e4$, and an elastic shear modulus of $1e5$.

Also shown is the velocity at the nodes (arrows—the scale tends to be dominated by the large velocities in the low viscosity air). In this example, the coarsest grid is of two elements in the x -, y -, and z -directions, with four multigrid levels, so that at the finest multigrid level there are 16 elements. Regarding the integration point density, there are 64 tracers per finest element. All boundaries are reflective.

7. Conclusions

Ellipsis 3D is a robust, particle-in-cell finite-element code for modelling mantle convection and lithospheric deformation. It combines the standard advantages of the Eulerian formulation for large-deformation problems, and Lagrangian integration points for tracking materials and history-dependent properties. Ellipsis3D can model materials with viscoelastic-plastic rheologies, and include the effects of strain and strain-rate weakening, and depletion. Released under the GNU Public License, Ellipsis3D is a powerful tool for a range of geodynamic research and teaching applications.

References

- Busse, F.H., Christensen, U., Clever, R., Cserepes, I., Gable, C., Giannandrea, E., Guillou, I., Houseman, G., Nataf, H.C., Ogawa, M., Parmentier, M., Sotin, C., Travis, B., 1994. 3D convection at infinite Prandtl number in Cartesian geometry—a benchmark comparison. *Geophysical and Astrophysical Fluid Dynamics* 75 (1), 39–59.
- De Smet, J.H., van den Berg, A.P., Vlaar, N.J., 1998. Stability and growth of continental shields in mantle convection models including recurrent melt production. *Tectonophysics* 296, 15–29.
- De Smet, J.H., van den Berg, A.P., Vlaar, N.J., 1999. The evolution of continental roots in numerical thermo-chemical mantle convection models including differentiation by partial melting. *Lithos* 48, 152–170.
- Frank-Kamenetskii, D.A., 1969. *Diffusion and Heat Transfer in Chemical Kinetics*. Plenum, New York.

- Hansen, U., Yuen, D.A., 2000. Extended-Boussinesq thermal--chemical convection with moving heat sources and variable viscosity. *Earth and Planetary Science Letters* 176, 401–411.
- Hughes, T.J.R., 1987. *The Finite Element Method*. Prentice Hall Inc., Englewood Cliffs, NJ., 803pp.
- Lenardic, A., Kaula, W.M., 1993. A numerical treatment of geodynamic viscous flow problems involving the advection of material interfaces. *J. Geophys. Res.* 98, 8243–8269.
- McKenzie, D.P., Bickle, M.J., 1988. The volume and composition of melt generated by extension of the lithosphere. *Journal of Petrology* 29, 625–679.
- Moresi, L., Solomatov, V., 1995. Numerical investigation of 2D convection with extremely large viscosity variations. *Physics of Fluids* 7, 2154–2162.
- Moresi, L., Solomatov, V., 1998. Mantle convection with a brittle lithosphere: thoughts on the global tectonic styles of Earth and Venus. *Geophysics Journal International* 133, 669–682.
- Moresi, L., Mühlhaus, H.-B., Dufour, F., 2001. Particle in cell solution for creeping viscous flows with internal interfaces. In: Mühlhaus, H., Dyskin, A., Pasternak, E. (Eds.), *Proceedings of the Fifth International Workshop on Bifurcation and Localisation*. Balkema, Perth.
- Moresi, L., Dufour, F., Mühlhaus, H.-B., 2003. A Lagrangian integration point finite element method for large deformation modeling of viscoelastic geomaterials. *Journal of Computational Physics* 184, 476–497.
- Karato, S.I., Wu, P., 1993. Rheology of the upper mantle: a synthesis. *Science* 260, 771–778.
- Tackley, P.J., 1998. Self-consistent generation of tectonic plates in three-dimensional mantle convection. *Earth and Planetary Science Letters* 157, 9–22.
- Takahashi, E., Kushiro, I., 1983. Melting of dry peridotite at high pressure and basalt magma genesis. *American Mineralogist* 68, 859–879.
- Van Keken, P.E., 1997. Evolution of starting mantle plumes: a comparison between laboratory and numerical models. *Earth and Planetary Science Letters* 148, 1–14.
- Van Keken, P.E., King, S., Schmeling, H., Christensen, U., Neumeister, D., Doin, M.P., 1997. A comparison of methods for the modeling of thermochemical convection. *Journal of Geophysical Research* 102, 22477–22496.