

ASPECT Hackathon 2017 Final report

Timeline	3
Participants and areas of interest	4
Report on projects the participants worked on	7
Analytical solution for incompressible Stokes flow in a hollow sphere	7
C. Thieulot	7
Analytical solution for incompressible Stokes flow in an annulus	8
C. Thieulot	8
Dynamic topography benchmark (pressure smoothing vs CFB) based on annulus benchmark (work in progress)	11
C. Thieulot & Ian Rose	11
Geodynamic simulations from a self-consistent 1D mineral physics model	12
Bob Myhill, Juliane Dannberg, Timo Heister	12
Comparison of Four Advection Algorithms in ASPECT	13
Ying He, Harsha Lokavarapu, and Gerry Puckett	13
The Volume-of-Fluid Interface Tracking Algorithm	14
Jonathan Robey	14
The Volume-of-Fluid AMR Refinement Algorithm	14
Gerry Puckett, Harsha Lokavarapu, and Jonathan Robey	14
Time Independent Active Particle Benchmarks	15
Harsha Lokavarapu, Gerry Puckett, and Ying He	15
Extension to dynamic topography postprocessor to also calculate topography on the lower boundary	18
Jacky Austermann, Ian Rose	18
Dynamic topography and geoid cookbook	19
Jacky Austermann, Ian Rose, Shangxin Liu	19
Visualization of relationships between cookbooks, geophysical models, and benchmarks	20
Lorraine J. Hwang	20
3D initial strain distribution for rift initiation	21
Anne Glerum	21
Chunk boundary objects and manifolds	22
Anne Glerum, Rene Gassmoeller, Wolfgang Bangerth	22
Chunk geometry and initial topography	23
Anne Glerum, Rene Gassmoeller	23
Testing framework	23
Timo Heister	23
New Melt Solver (Melt 3.0)	24
Ryan Grove, Juliane Dannberg, Timo Heister	24

Melt and Plasticity	25
Juliane Dannberg, Anne Glerum, John Naliboff, Time Heister, Cedric Thieulot	25
Operator Splitting for models with melting	25
Juliane Dannberg, Timo Heister	25
User defined input coordinate systems and conditions	26
Matt Weller	26
Newton Solver	27
Menno Fraters, Wolfgang Bangerth	27
The consistent-boundary-flux method for dynamic topography computation	28
Ian Rose, Jacky Austermann, Cedric Thieulot	28
Particles and Melting Processes	30
Rene Gassmoeller, Joe Schools	30
Viscoelastic Rheology	30
John Naliboff, Cedric Thieulot	30
Strain weakening using the full finite strain tensor	31
John Naliboff	31
Incorporation of operators into Initial Temperature and Initial Composition Interface	31
Bob Myhill, Rene Gassmoeller	31
New Material Model Constructor	32
Paul Bremner, Ian Rose	32
Multicomponent Material Model Clarification	32
Paul Bremner, Rene Gassmoeller, Ian Rose	32
Seismic Model Input into Initial Temperature Overhaul	32
Paul Bremner, Jacky Austermann	32
3D subduction example	32
Bob Myhill, Anne Glerum, Juliane Dannberg, Rene Gassmoeller	32
Dynamic Friction Material Model	33
John Naliboff, Arushi Saxena, Cedric Thieulot	33
Geoid Postprocessor and Benchmark	33
Shangxin Liu, Ian Rose	33
Jupyter Notebook for ASPECT	34
Lorraine J. Hwang, Timo Heister, and Ian Rose	34
Statistics about ASPECT's growth during the hackathon	34

Introduction

To further the development of the mantle convection code ASPECT and to grow and foster its user community, 21 users and developers of ASPECT worked side-by-side over a 10 day period in Blue Ridge, Georgia in May 2017.

The ASPECT community made significant progress in various areas of development of the code. The additions include the implementation of a new and more accurate method to compute dynamic topography (the Consistent Boundary Flux method), an interface to the mineral physics toolkit BurnMan, and improvements of models with melt migration, such as better solvers and the option to use particles to track melt. Moreover, participants made substantial progress integrating a Newton solver into ASPECT, crucially improving the performance of nonlinear problems, in particular those with a visco-plastic rheology. To facilitate modeling of more realistic lithosphere-scale problems, users and developers also started working on material models and solver schemes allowing for a visco-elastic rheology. Through the efforts of many different participants, ASPECT now also supports more options and easier and more flexible ways to

assign model initial conditions, including an initial topography and distribution of finite strain. Furthermore, a large number of tutorial programs, documentation, tests and benchmark cases were added during the hackathon, including comparisons between different advection algorithms and tracer particles.

During the course of the hackathon, every participant contributed source code to the project. Together, users and developers added a total of almost 6000 lines of code, arising from 187 individual contributions, and including 40 new tests. These numbers are a significant increase over the previous hackathon.

Below is the timeline and a log of the individual contributions.

Timeline

Saturday, 05/06	Arrival
Sunday, 05/07	9am: House rules, Organization, Introduction (Timo) 11am: Git Pull Requests (Wolfgang) 3 pm: Melt solvers v 2.0 (Ryan)
Monday, 05/08	10am: Git pull requests and best practices (Timo) 1pm: Rene: Using profilers and optimizing the code (20 min) & Precompiling headers and unity builds (20 min)
Tuesday, 05/09	10am: GUI (Rene)

Wednesday, 05/10	1pm: Ying, Harsha & Gerry Presentation of a comparison of four methods for advecting a property in ASPECT
Thursday, 05/11	Day off
Friday, 05/12	
Saturday, 05/13	Bob & Timo: Burnman presentation
Sunday, 05/14	½ Day off
Monday, 05/15	Ian: CBF topography method, Lorraine: Software statistics, citation, IPython notebook
Tuesday, 05/16	Menno: Newton solver scheme
Wednesday, 05/17	checkout

Participants and areas of interest

Name, affiliation, email	Goals and interests for this hackathon
Wolfgang Bangerth Colorado State University bangerth@colostate.edu	<ol style="list-style-type: none"> 1. Help others 2. Review most code submissions 3. Fix bugs others may find 4. Get people involved
Timo Heister Clemson University heister@clemson.edu	<ol style="list-style-type: none"> 1. Help others, code review 2. Melt solver improvements 3. deal.II & ASPECT on Mac OSX 4. Coupling BurnMan & ASPECT
Rene Gassmoeller Colorado State University rene.gassmoeller@mailbox.org	<ol style="list-style-type: none"> 1. Help others, code review 2. Geometry/Manifold improvements 3. Particle improvements 4. Geoid postprocessor
Juliane Dannberg Colorado State University judannberg@gmail.com	<ol style="list-style-type: none"> 1. Help others 2. Magma/mantle dynamics 3. Coupling ASPECT & thermodynamic databases (especially with melt) 4. Melt solver improvements
Lorraine Hwang UC Davis ljhwang@ucdavis.edu	<ol style="list-style-type: none"> 1. Den mother 2. Management stuff :(3. Something with Jupyter (ASPECT and/or Burnman) 4. Bake cookies
Bob Myhill University of Bristol bob.myhill@bristol.ac.uk	<ol style="list-style-type: none"> 1. Linking mineral physics and geodynamics through BurnMan ↔ ASPECT 2. Subduction zone deformation from grain to regional scale (and links with seismology)
Jacky Austermann Cambridge University ja629@cam.ac.uk	<ol style="list-style-type: none"> 1. Adjoint equations for Stokes flow for dynamic topography and gravity 2. Setup inversion scheme (conjugate gradient method?) 3. Dynamic topography (+geoid?) benchmark 4. Look into consistent boundary flux method and self-gravity for DT - geoid calculation
Anne Glerum GFZ Potsdam / UU acglerum@gfz-potsdam	<ol style="list-style-type: none"> 1. 2D/3D chunk rift models with strain weakening, melting, Winkler bottom and initial temperature perturbation or noise on strain 2. Geometry/Manifold improvement 3. Layered chunk with initial topography

	<ol style="list-style-type: none"> 4. Also interested in elasticity, using tomography for temperature/density initial conditions
<p>Matt Weller University of Texas at Austin Institute for Geophysics mbweller@ig.utexas.edu</p>	<ol style="list-style-type: none"> 1. Global Magma/Mantle/Chemical/Tectonic evolution 2. ASPECT & thermodynamic databases (especially with melt) 3. Melt tracking and chemical evolution 4. Citcom vs Aspect speed and tests 5. Elasticity/strain tracking
<p>Menno Fraters Utrecht University menno.fraters@outlook.com</p>	<ol style="list-style-type: none"> 1. Developing and merging the Newton solver 2. Working on the world generator
<p>Ying He UC Davis yinghe@math.ucdavis.edu</p>	<ol style="list-style-type: none"> 1. Periodic boundary condition for DG solver 2. Time dependent particle benchmarks 3. Particle interpolation scheme
<p>Harsha Lokavarapu UC Davis hlokavarapu@ucdavis.edu</p>	<ol style="list-style-type: none"> 1. Particle interpolation schemes 2. Develop SolCx, SolKz, and Inclusion compositional field and active particle benchmarks 3. Profiling and optimization of code
<p>Joe Schools University of Maryland jschools@umd.edu</p>	<ol style="list-style-type: none"> 1. Melt processes in the lithosphere, plume/adiabatic melting 2. Linking/assessing ASPECT and thermodynamic/petrologic calculators (i.e. MELTS, pMELTS)
<p>Paul Bremner University of Florida pbremner@ufl.edu</p>	<ol style="list-style-type: none"> 1. Improve seismic tomography model input options. 2. Burnman interface. 3. Improve multicomponent material model interface. 4. General contribution wherever able.
<p>Shangxin Liu Virginia Tech sxliu@vt.edu</p>	<ol style="list-style-type: none"> 1. Geoid postprocessor 2. Geoid benchmark 3. 3D spherical shell time-dependent benchmark 4. Different formulations in ASPECT 5. Functions to transfer between spatial domain and spherical harmonic domain in 3D spherical shell.

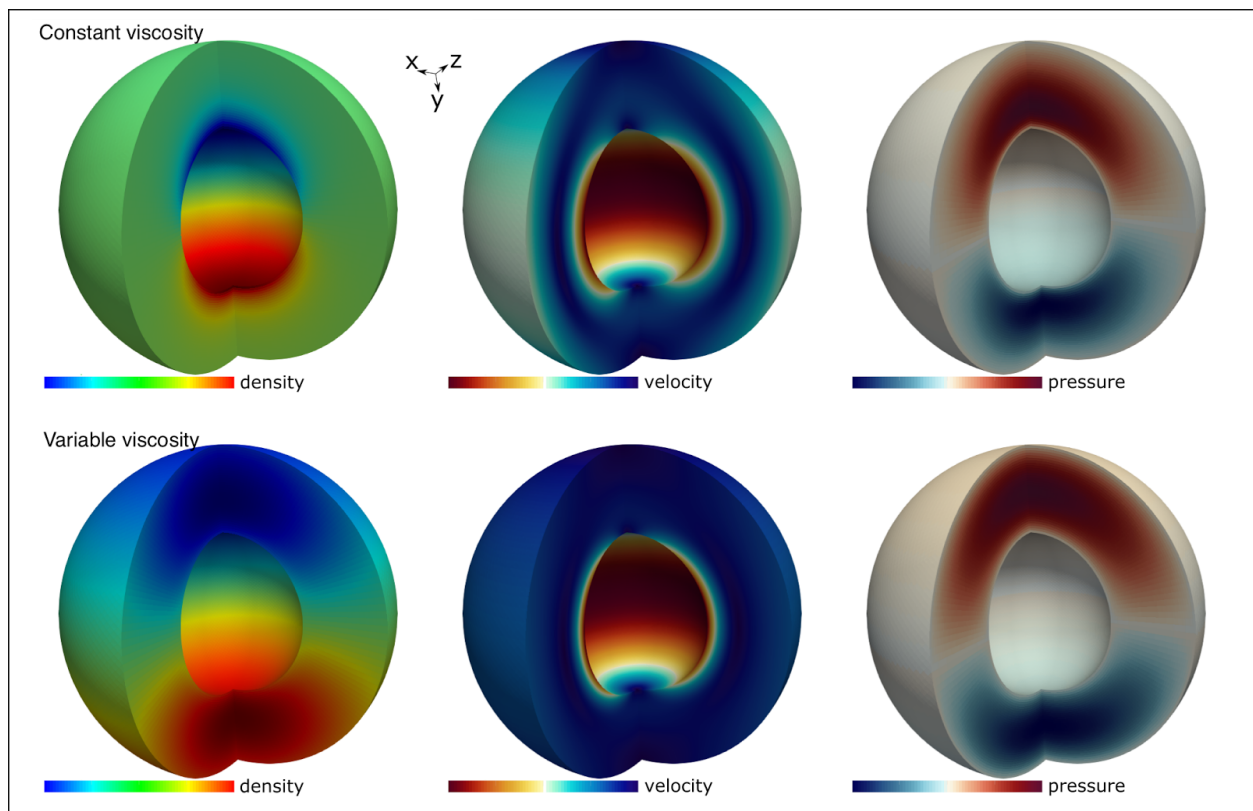
<p>Ryan Grove Clemson University rgrove@g.clemson.edu</p>	<ol style="list-style-type: none">1. Melt solvers2. Melt analysis3. Parallel computing4. Helping others
<p>Jonathan Robey (remote) UC Davis jmrobey@ucdavis.edu</p>	<ol style="list-style-type: none">1. Interface tracking2. General contributions where possible

Report on projects the participants worked on

Analytical solution for incompressible Stokes flow in a hollow sphere

C. Thieulot

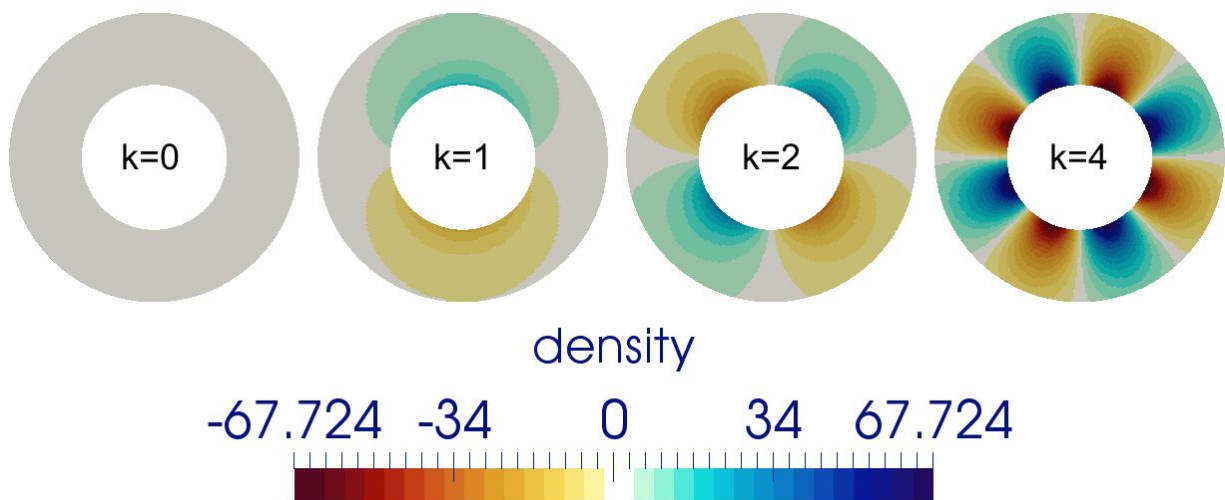
Prior to the hackathon, I have worked on deriving a new family of analytical flow solutions to the incompressible Stokes equations in a spherical shell. The velocity is tangential to both inner and outer boundaries, the viscosity is radial and of power-law type, and the solution has been designed so that the expressions for velocity, pressure, and body force are simple polynomials and therefore simple to implement in (geodynamics) codes. Various flow average values, e.g. the root mean square velocity, are analytically computed.

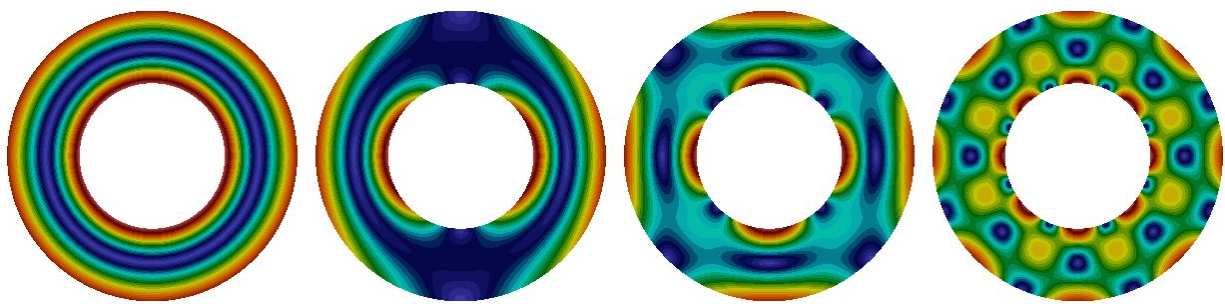


Analytical solution for incompressible Stokes flow in an annulus

C. Thieulot

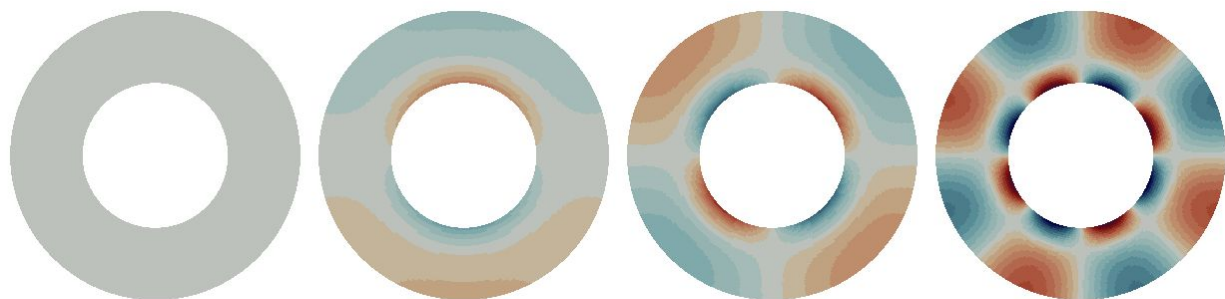
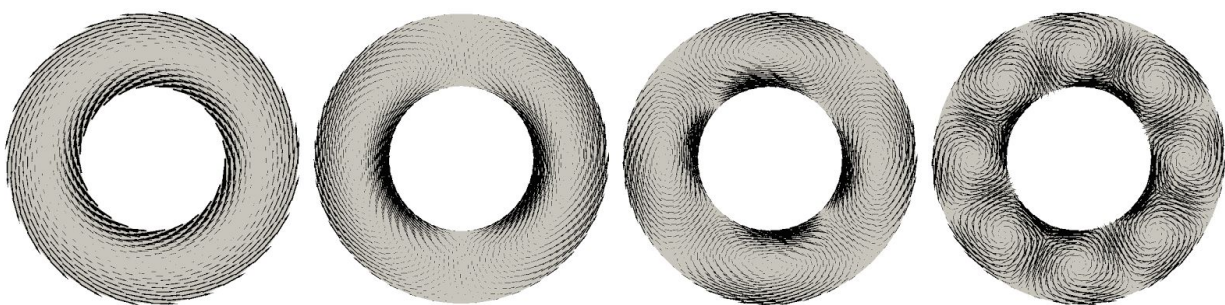
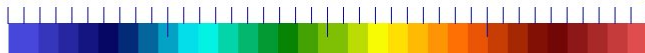
Prior to the hackathon, I have worked on deriving a new family of analytical flow solutions to the incompressible Stokes equations in an annulus. The velocity is tangential to both inner and outer boundaries, the viscosity is constant, and the solution has been designed so that the expressions for velocity, pressure, and body force are reasonably simple functions and therefore simple to implement in (geodynamics) codes. By changing the value of the parameter k , one can decide how many convection cells are present in the domain. The density, pressure and velocity fields are shown hereunder, alongside convergence rate plots for both velocity and pressure.





velocity

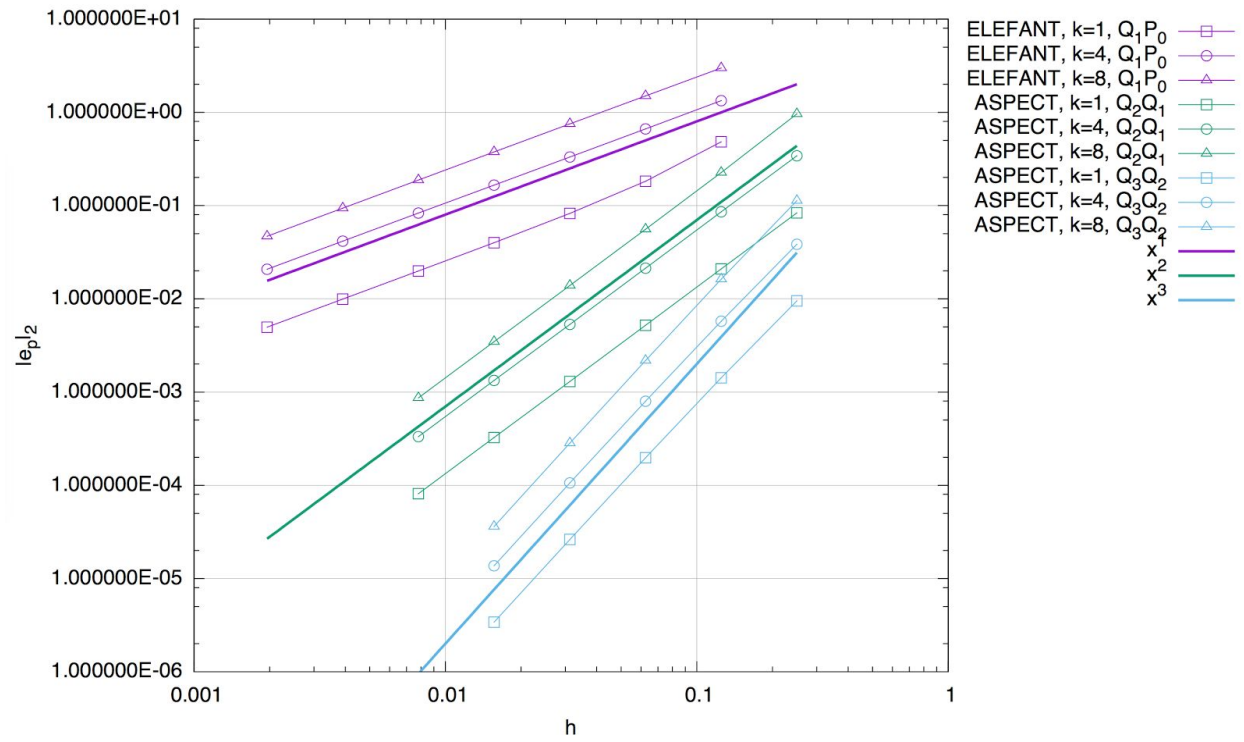
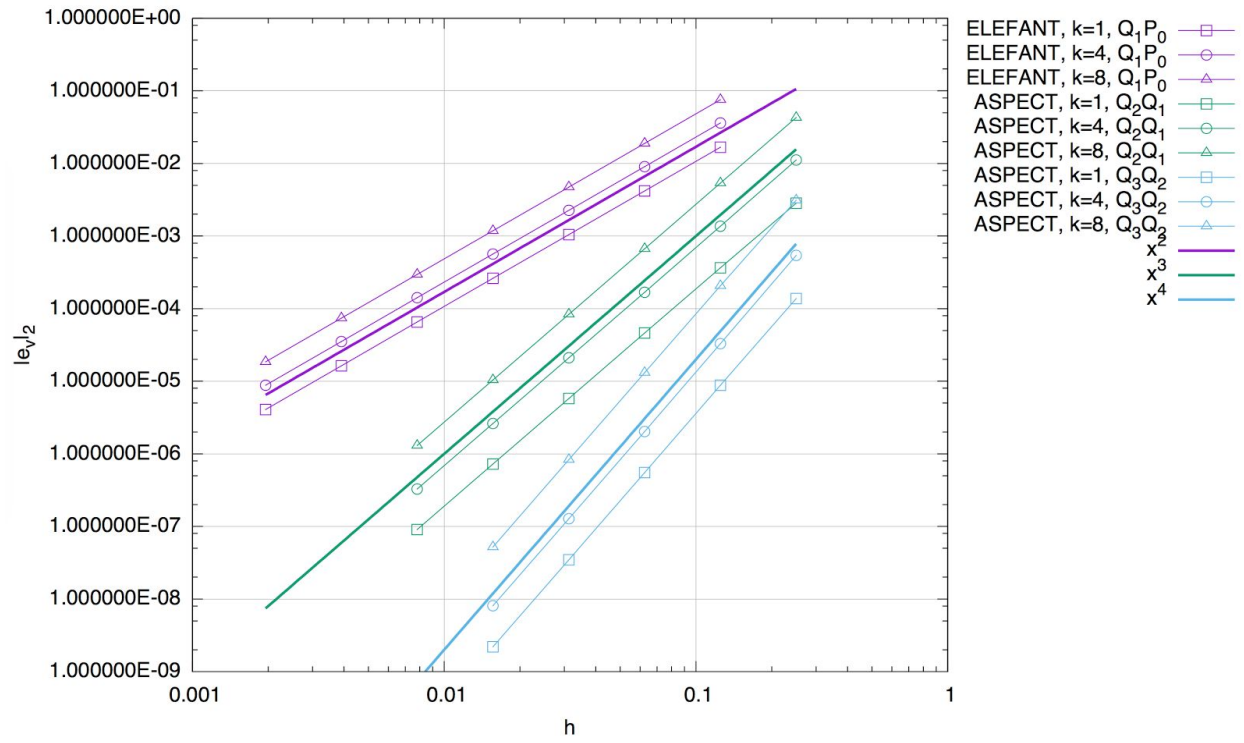
0.006 0.59 1.2 1.7 2.328



pressure

-8.502 -4 0 4.3 8.502

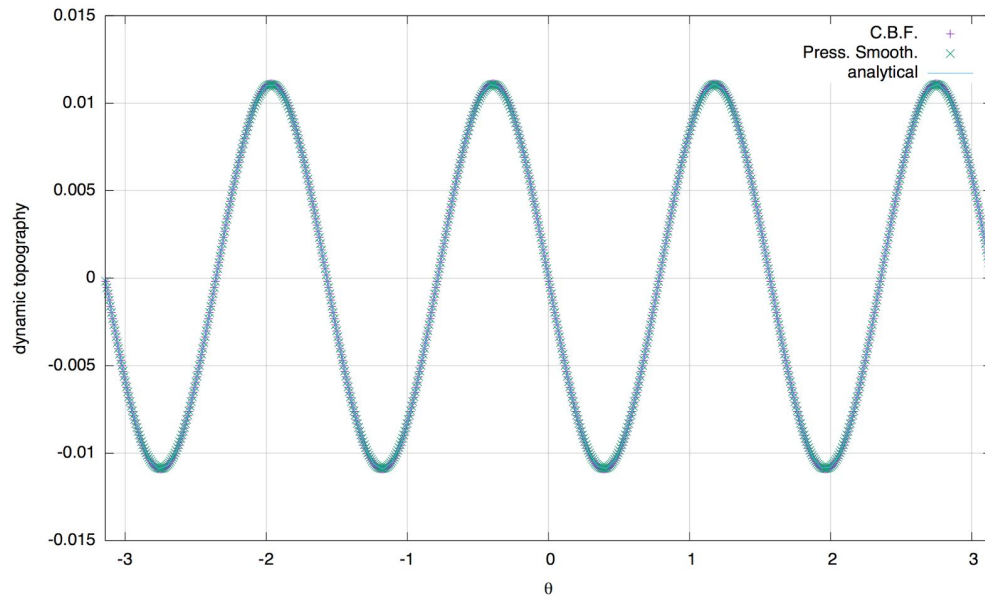




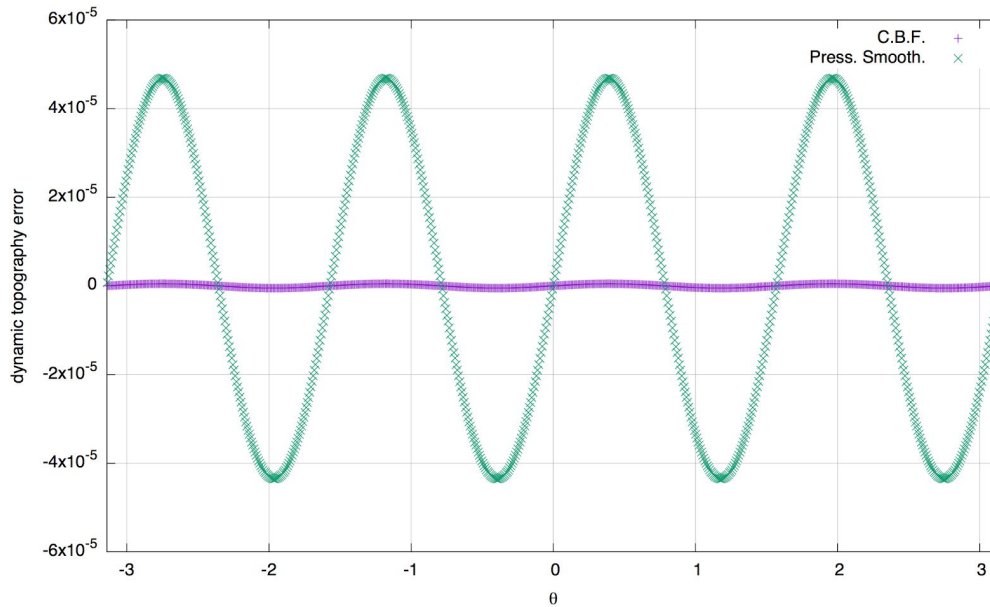
Dynamic topography benchmark (pressure smoothing vs CBF) based on annulus benchmark (work in progress)

C. Thieulot & Ian Rose

The following plots show the dynamic topography measurements at the surface of the annulus for both old (pressure smoothing) and new (CBF) methods along with the analytical solution.



This second plot shows the error between both methods and the analytical solution:



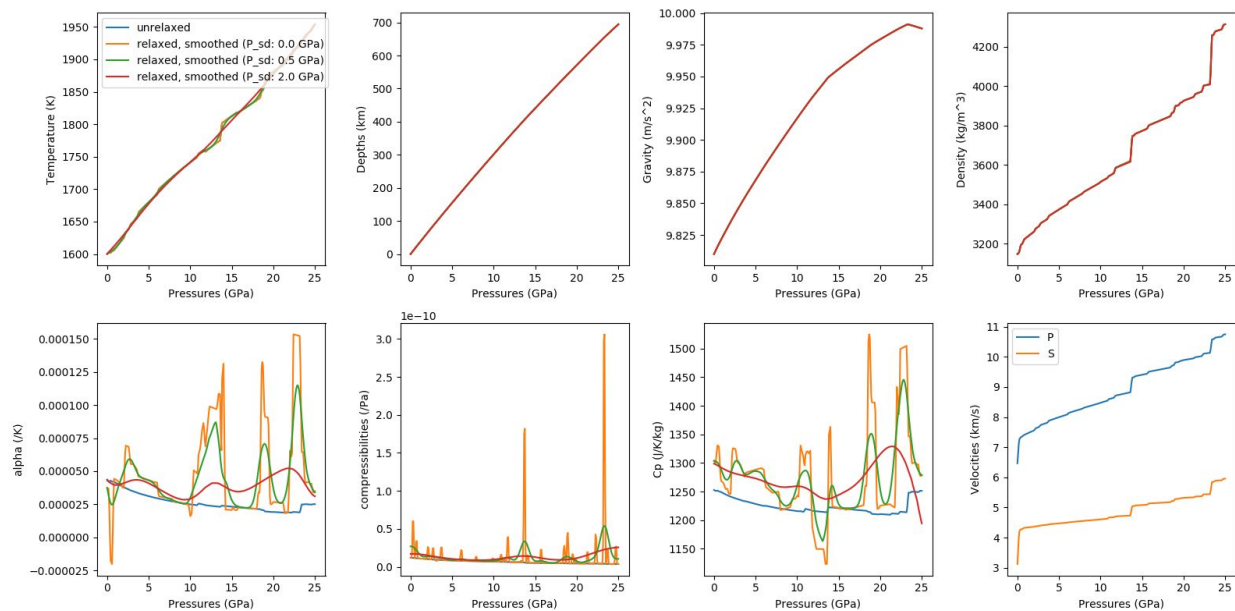
Geodynamic simulations from a self-consistent 1D mineral physics model

Bob Myhill, Juliane Dannberg, Timo Heister

We have implemented an online ipython notebook demonstrating the creation of ASPECT-readable 1D mineral physics profiles calculated self-consistently by the BurnMan software. The profiles incorporate the effects of mineral reactions, i.e. latent heat, enhanced compressibility and thermal expansivity. The notebook includes the option to smooth volume and entropy before calculating these properties, which is useful for geodynamics simulations where mesh size is coarse compared with the width of phase transitions.

The ipython notebook has interactivity for the following:

- Surface gravity and planetary radius (to self-consistently calculate pressure profiles)
- Mantle potential temperature and maximum pressure
- Adjustable parameters for entropy and volume smoothing

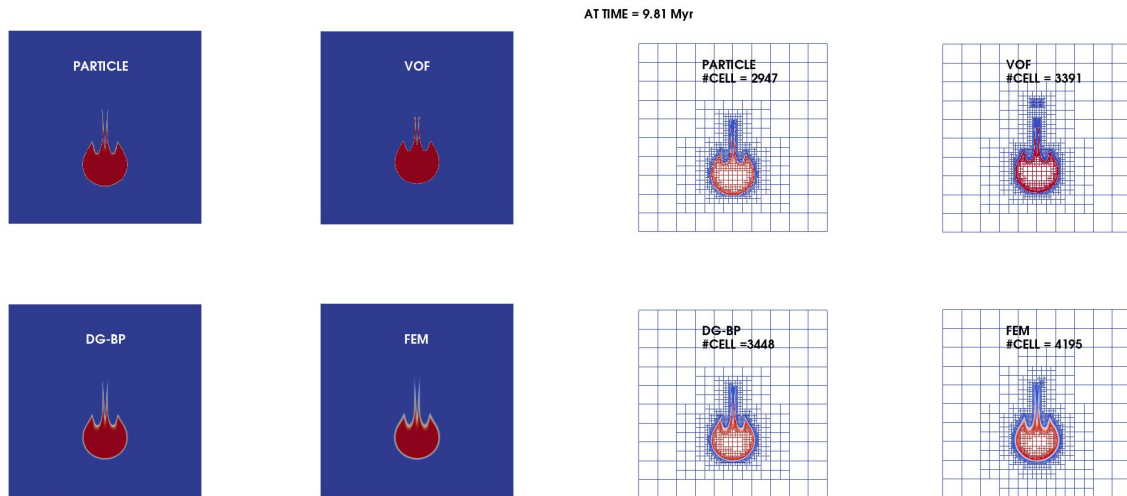


The figure above shows an example set of profiles along the same adiabat. Depth and gravity profiles are calculated self-consistently by integration from the surface. Physical properties are derived from a pyrolitic model, the mineral models from Stixrude and Lithgow-Bertelloni (2011) as calculated by PerpleX and processed by BurnMan. Blue: unrelaxed properties, corresponding to properties seen when perturbations in pressure and temperature are rapid compared with mineral reactions. Orange: Relaxed properties, more applicable to geodynamic simulations. Green and blue: Relaxed properties after smoothing the entropy and volume. Some smoothing is necessary to capture the effects of latent heat in geodynamic simulations; typically a reaction should span >4 cells to be captured reasonably in ASPECT, so a finer mesh means that less smoothing is required.

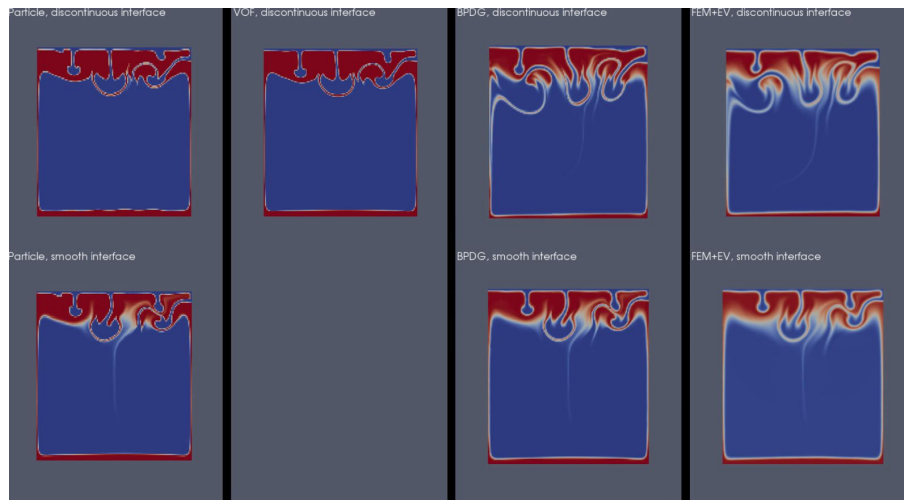
Comparison of Four Advection Algorithms in ASPECT

Ying He, Harsha Lokavarapu, and Gerry Puckett

We have been comparing the advection of a density field with the Bound Preserving Discontinuous Galerkin method (DGBP), PARTICLE, Volume-of-fluid (VOF), and the the original Finite Element (FEM) advection algorithm in ASPECT with Entropy Viscosity (EV), which we abbreviate FEM-EV.



The figure above shows the Gerya-Yuen falling box benchmark with a viscosity ratio of 1 on an adaptive mesh with minimum $h = 1/320$ and maximum $h = 1/10$. After applying the newly implemented composition approximate gradient mesh refinement strategy and same other mesh refinement parameters for all Particle, DG, FEM methods, the number of active cells required by FEM-EV algorithm has been reduced to **4195**, whereas the DGBP algorithm has **3448** active cells, the Particle algorithm has **2947** active cells and the VOF algorithm has **3391** active cells.



The figure above shows the van Keken problem, computed with an initially discontinuous interface (top), and computed with an initially smooth interface (bottom).

The Volume-of-Fluid Interface Tracking Algorithm

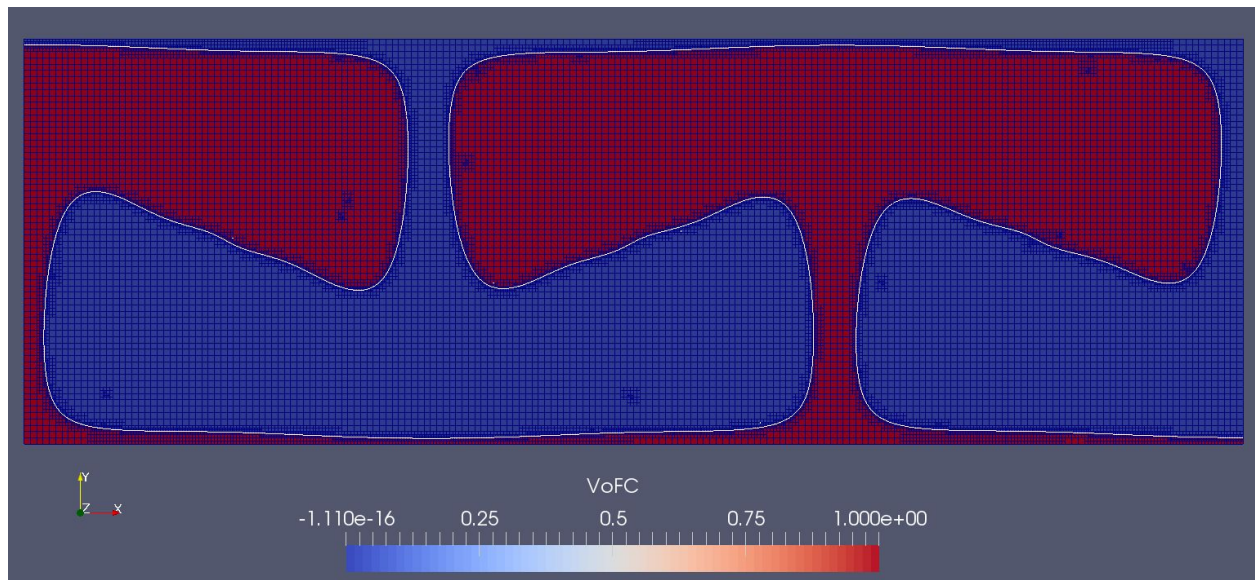
Jonathan Robey

We continued work on the Volume-of-Fluid algorithm including work on Refinement schemes (II) and alteration to correctly handle periodic boundaries during both reconstruction and advection.

The Volume-of-Fluid AMR Refinement Algorithm

Gerry Puckett, Harsha Lokavarapu, and Jonathan Robey

We are working on modifying the algorithm for placing the refined mesh cells only along an interface that is being propagated by the volume-of-fluid (VOF) interface tracking algorithm. In particular, before the Hackathon, not all refined cells that should have been coarsened were being coarsened as shown in Figure II.1 below.

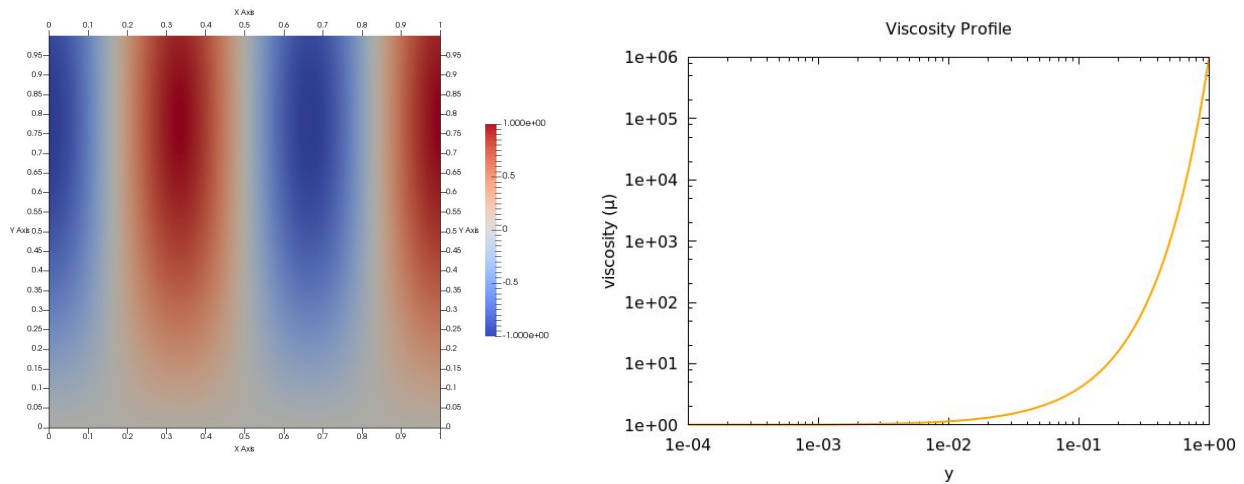


The figure shows a VOF computation with AMR of the Density Stratified Flow (DSF) Problem with $Ra = 1e5$ and $B = 0.2$. The refinement criterion is set to refine all cells that contain the interface and all cells immediately adjacent to cells that contain the interface. Note the presence of some refined cells that should have been coarsened after the interface left these cells. We are modifying the coarsen and refine algorithm to eliminate this problem.

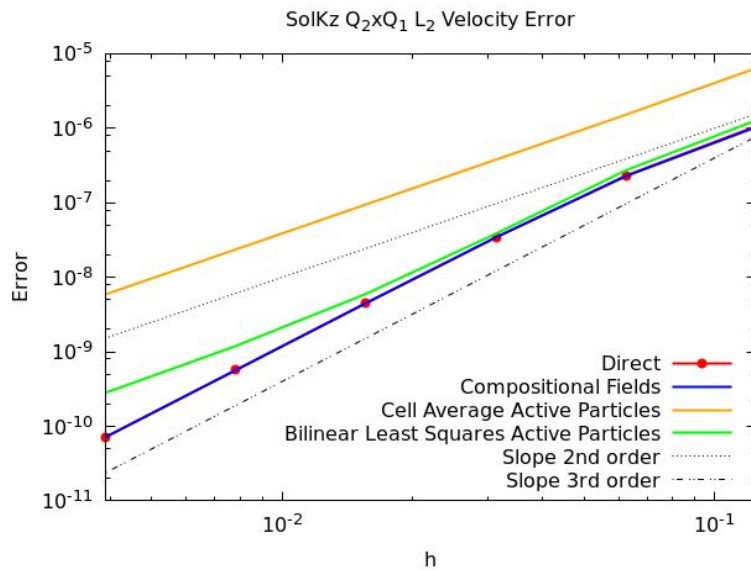
Time Independent Active Particle Benchmarks

Harsha Lokavarapu, Gerry Puckett, and Ying He

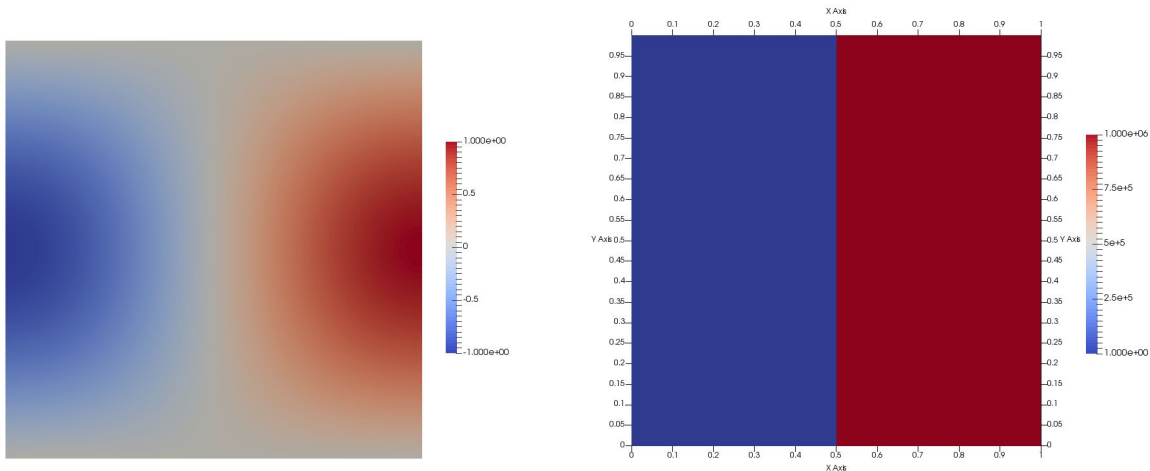
We added the SolKz, SolCx, and SolVI (inclusion) benchmarks using compositional fields as well as active particles. We also implemented a bilinear interpolation method that is based on least squares for determining the interpolant's coefficients. We compared this to the cell average particle interpolation methods. For both particle interpolation methods, we generate 16 particles per cell and used a Q2_Q1 element for the underlying 'particle compositional field'. Note that to reach 3rd order convergence of the bilinear least squares interpolation scheme in SolKz, as the resolution increases, so must the number of particles per cell.



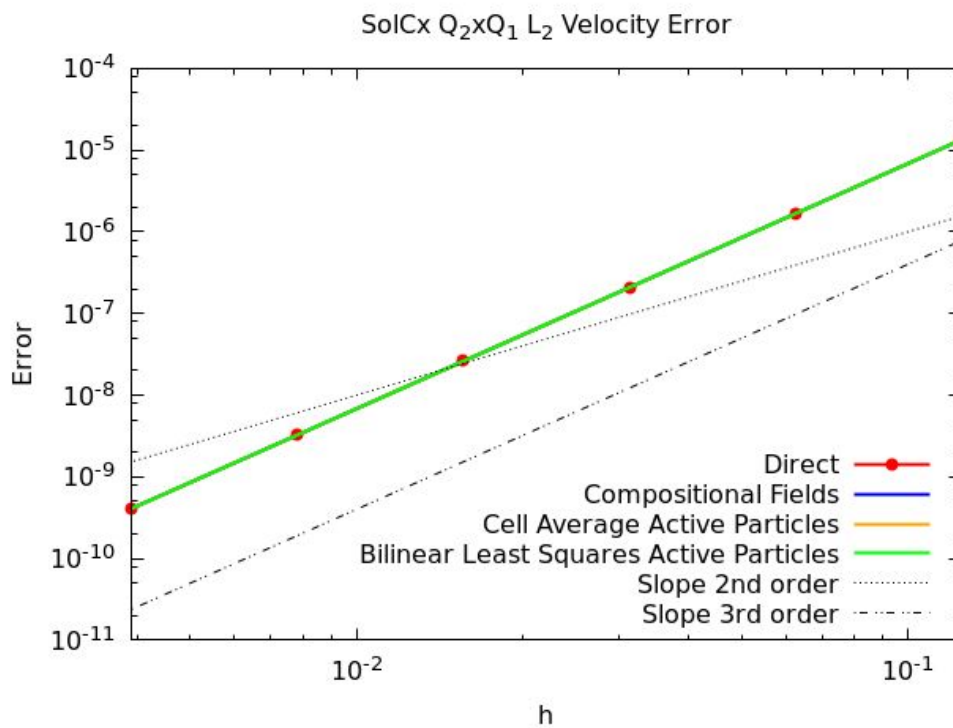
SolKz benchmark: Density (left) and viscosity profile (right).



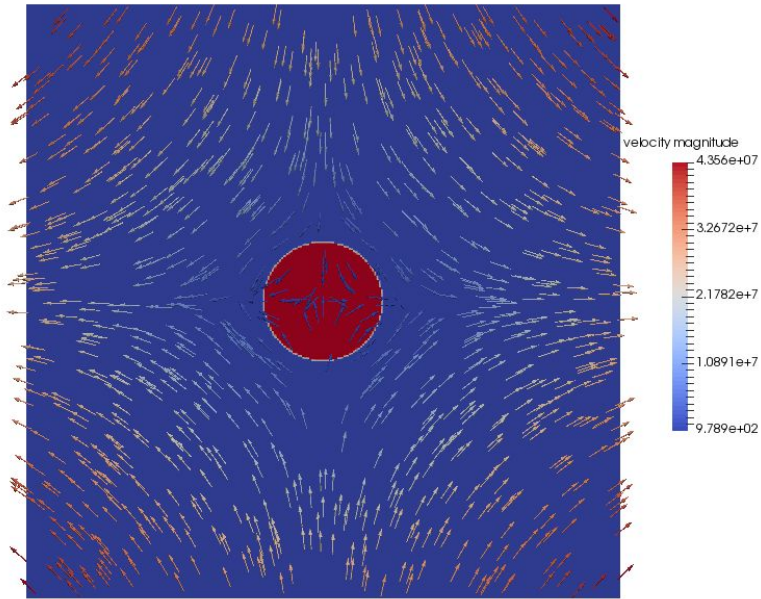
SolKz benchmark: Convergence rates of the various algorithms for modeling the SolKz benchmark. *Note:* We expect fully third-order convergence of the bilinear interpolation but it is not shown here. This is probably because the number of particles needs to be increased as $h \rightarrow 0$. Harsha is currently investigating this issue.



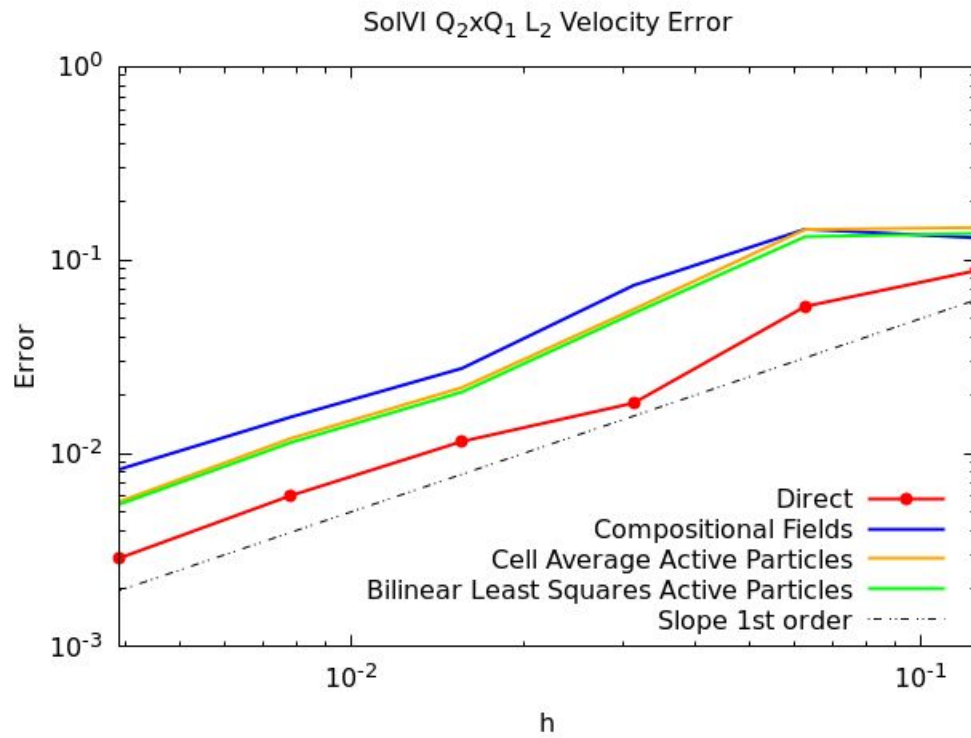
Setup of the SolCx benchmark (above).



SolCx benchmark: Convergence Rates of the various algorithms (as described above for modeling the SolKz benchmark). *Note:* We expect third-order convergence of the bilinear interpolation and are not getting it. This may be because the number of particles needs to be increased as $h \rightarrow 0$. Harsha is currently investigating this issue.



SoIVI benchmark: Viscosity profile with velocity glyphs (above).

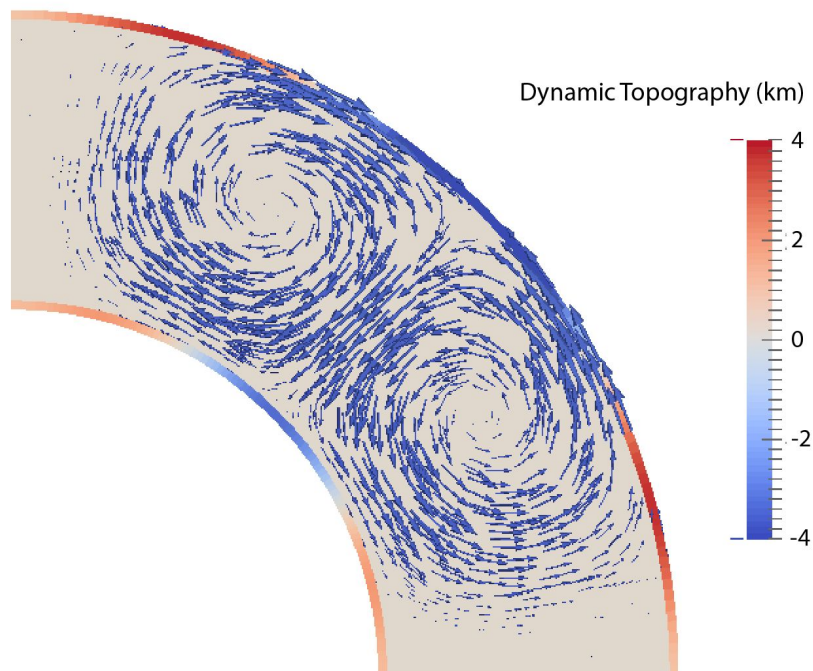


SoIVI benchmark: Convergence Rates of the various algorithms (above).

Extension to dynamic topography postprocessor to also calculate topography on the lower boundary

Jacky Austermann, Ian Rose

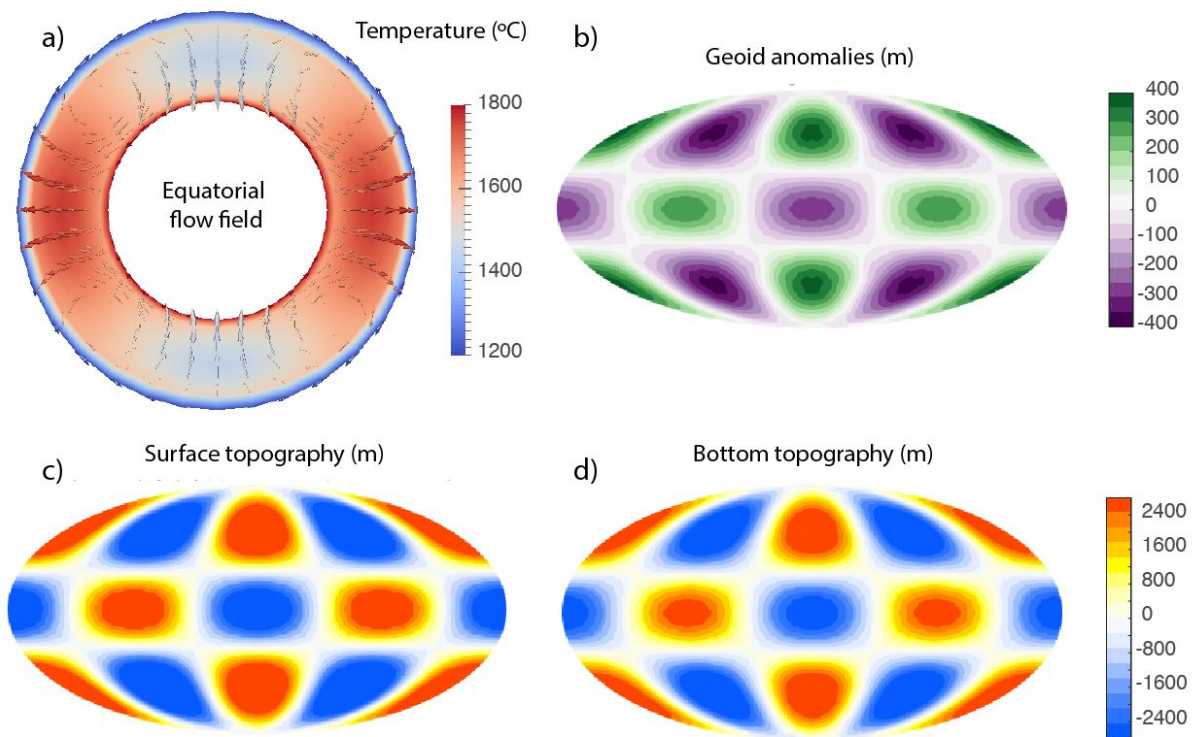
We included the calculation of topography on the lower boundary in the existing dynamic topography postprocessor. This topography is not a free surface but a stress based topography (outward stresses are balanced by excess topography accounting for the density contrast across this surface). In the visualization postprocessor the upper and lower boundary topographies are now both calculated and plotted. In the postprocessor that writes the topography into a text file we now output the lower surface topography in a separate file 'bottom_dynamic_topography'.



Dynamic topography and geoid cookbook

Jacky Auermann, Ian Rose, Shangxin Liu

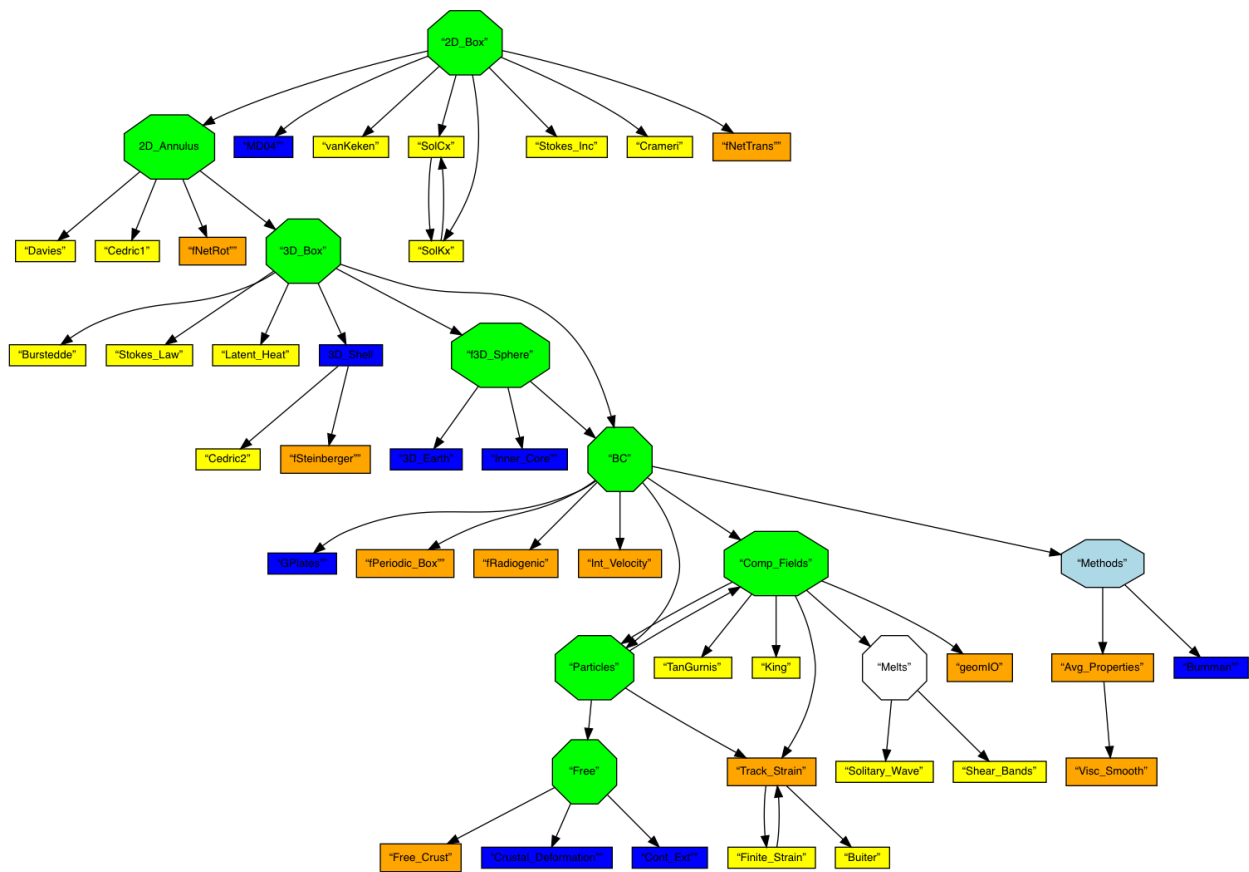
We put together a simple cookbook that is a good intermediate step between the “Simple convection in a spherical 3D shell” and the “3D convection with an Earth-like initial condition” cookbooks. This new cookbook uses a simple harmonic perturbation initial condition and introduces the dynamic topography and geoid postprocessor. We will also add the bottom topography and geoid postprocessors (and figures thereof) to the “3D convection with an Earth-like initial condition” cookbook.



Visualization of relationships between cookbooks, geophysical models, and benchmarks

Lorraine J. Hwang

ASPECT has an ever growing collection of cookbooks, geophysical models, and benchmarks. As an aid to new as well as experienced users, we visualized this set of examples as a networked collection. A new user should begin at the top of the tree, "2D_Box" and work through the green octagons to become familiarized with each major feature. Advanced and more in depth examples are listed below each.



Key:

- green hexagons - basic techniques
- orange - cookbooks
- blue - geophysical setups
- yellow - benchmarks as defined in the manual or found in the repository
- White - needed .prms

Files prepended with "f" are .prms in the `~cookbooks/future` folder and have not been written-up and are not in the manual. In addition, not all benchmarks have been written-up.

Methods - includes miscellaneous topics e.g. different mathematical/numerical techniques to stabilize solutions and enhancements to the physics or geophysical models.

This is a strawman date as of 10 May 2017. Please feel free to suggest cookbooks needed as well as changes in how examples are organized.

Implementation will require modification of .prm files for automatic generation (*Wolfgang Bangerth*)

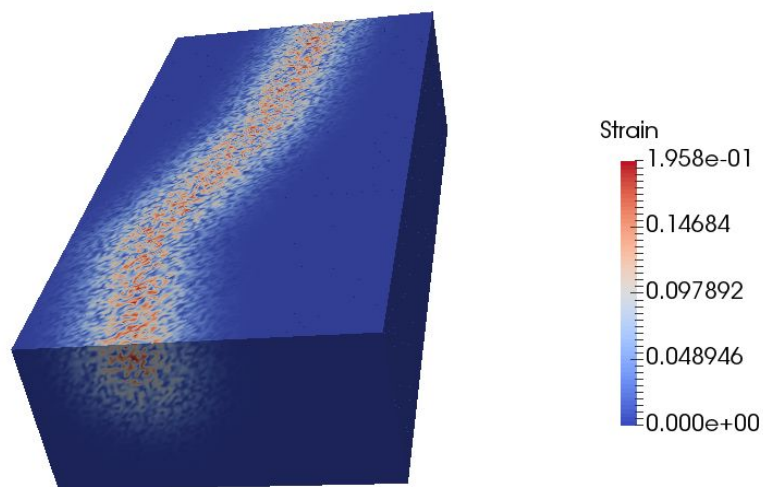
Styled after: [:http://www.dealii.org/8.5.0/doxygen/deal.II/Tutorial.html#list](http://www.dealii.org/8.5.0/doxygen/deal.II/Tutorial.html#list)

Uses: <http://www.graphviz.org/>

3D initial strain distribution for rift initiation

Anne Glerum

We worked on an plugin that provides initial conditions for the second strain invariant in (for now) a 2d/3d box. It creates random noise in an area around the future rift axis that is specified by a polygon. The maximum amplitude of the noise is controlled by a user-specified Gaussian distribution around the rift axis that is also smoothed out to zero in depth around a user-specified depth. This allows us to do distributed rift models in ASPECT, without an initial weak seed to initiate the rift. An example setup is shown in the figure below.

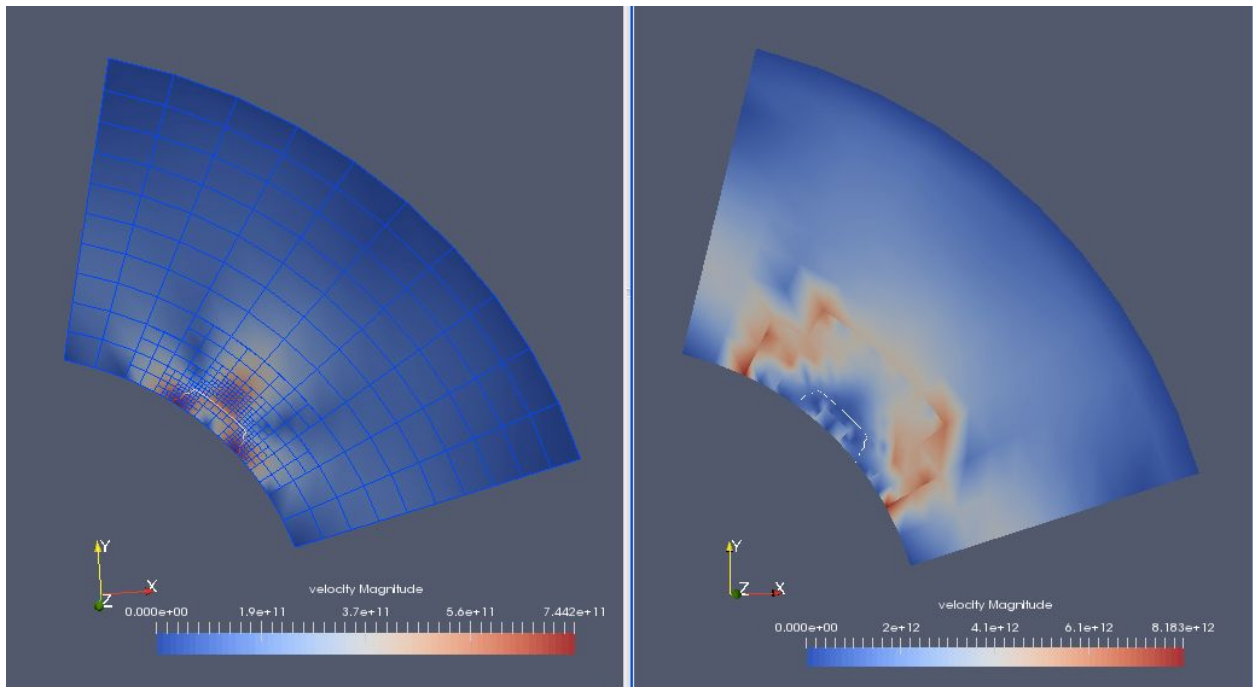


It shows the random distribution of noise orthogonal to the rift polygon that forms the initial condition for the compositional field representing the strain.

Chunk boundary objects and manifolds

Anne Glerum, Rene Gassmoeller, Wolfgang Bangerth

We worked on setting the right boundary objects for the chunk geometry model in combination with dealii prior to version 9 and setting the manifold and its `push_forward_gradient` function for aspect in combination with dealii 9. As the figure below shows, this removes spurious velocities along grid refinement levels.



The figure above shows the velocity field for the new chunk geometry (left) and the old chunk geometry (right). On the left, the mesh is also shown to demonstrate the spurious velocities along the same mesh refinement levels in the right figure.

Chunk geometry and initial topography

Anne Glerum, Rene Gassmoeller

The chunk geometry can now also include initial topography described by an ascii data table (for deal.II > 9.0), as demonstrated for an extreme case below:

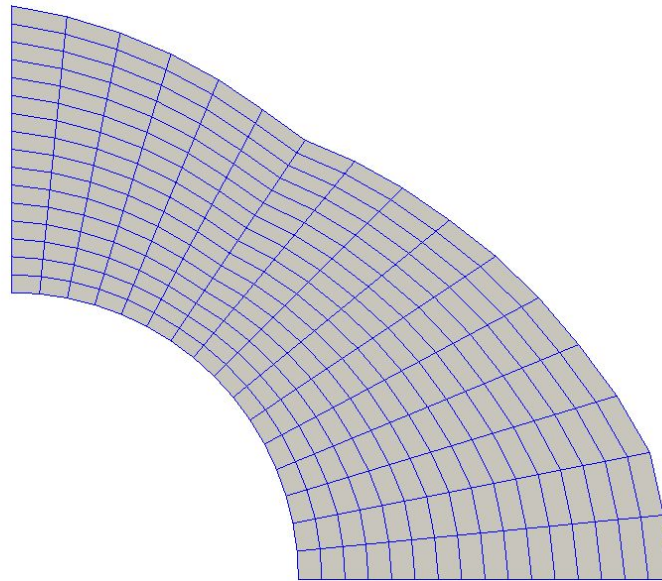
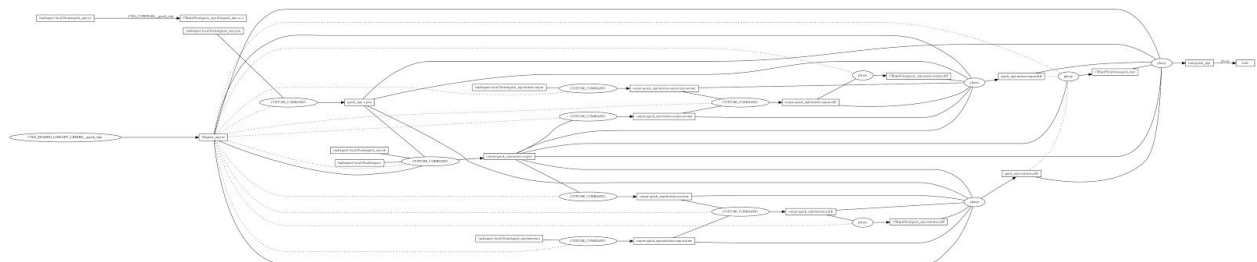


Figure above: An initial chunk geometry mesh perturbed by a topography given by an ascii table.

Testing framework

Timo Heister

I rewrote the docker images for running automated testing and redesigned the testsuite and the individual test dependencies. The testsuite is now a separate cmake project and the new dependencies track all inputs and outputs specifically to allow for “ninja” support instead of “make”. Overall, this avoid subtle bugs present before and improves performance significantly. Here is an example dependency graph for a single test:



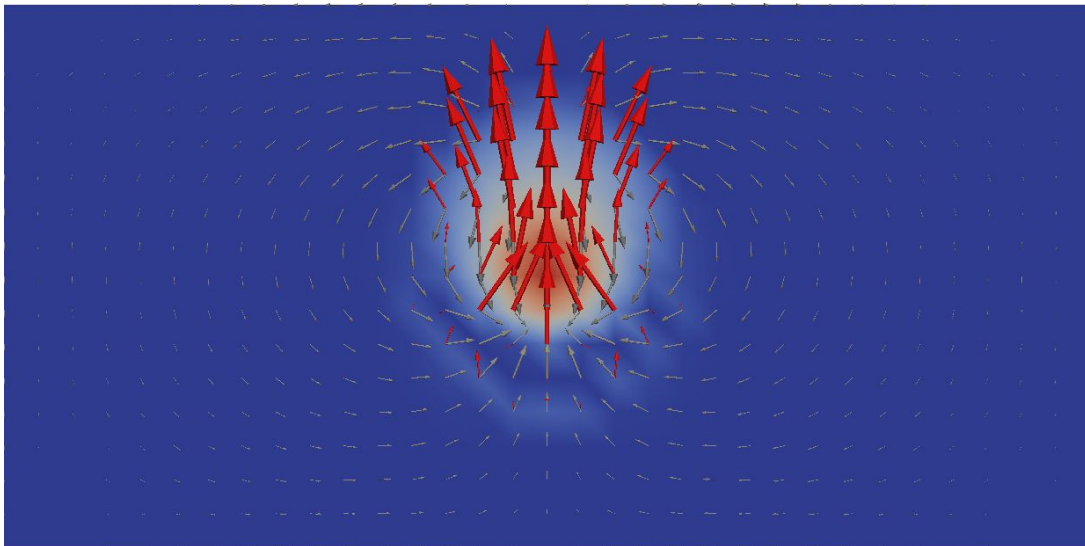
New Melt Solver (Melt 3.0)

Ryan Grove, Juliane Dannberg, Timo Heister

We worked on an exciting new update to the melt solver. In standalone work, it gives us less iteration counts for the harder inner A computations and a few more iteration counts for easier inner S computations. We worked on replicating these results in ASPECT and are almost satisfied that we have done so. We only have a big 3D run to go before we are convinced that it is better and put it into ASPECT.

The new features include:

- Constrain compaction pressure if k_d is zero,
- Add rescaled p_c by $\sqrt{k_d}$ to make it well-defined,
- Correct Schur complement handling.



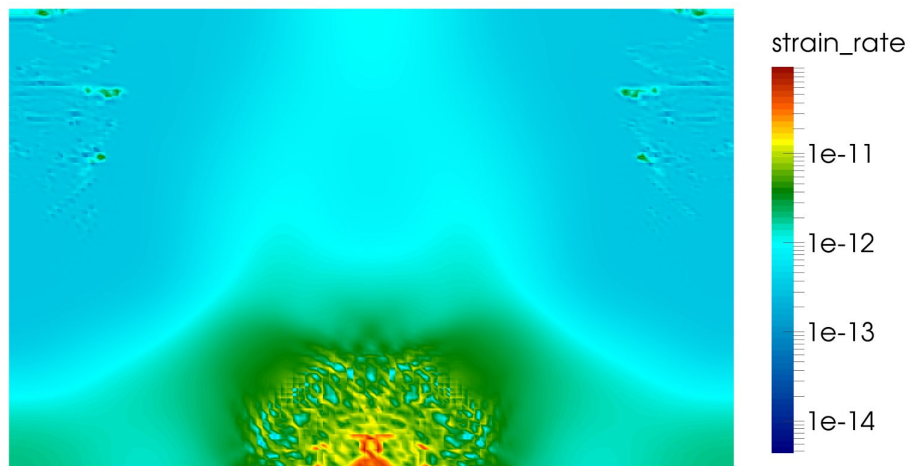
The figure above shows the porosity field and vectors of the velocity field in a simple test case for models with melt transport.

Melt and Plasticity

Juliane Dannberg, Anne Glerum, John Naliboff, Timo Heister, Cedric Thieulot

We worked on implementing a plastic material model that works with melt migration (using the formulation of Keller et al., 2013) and testing it.

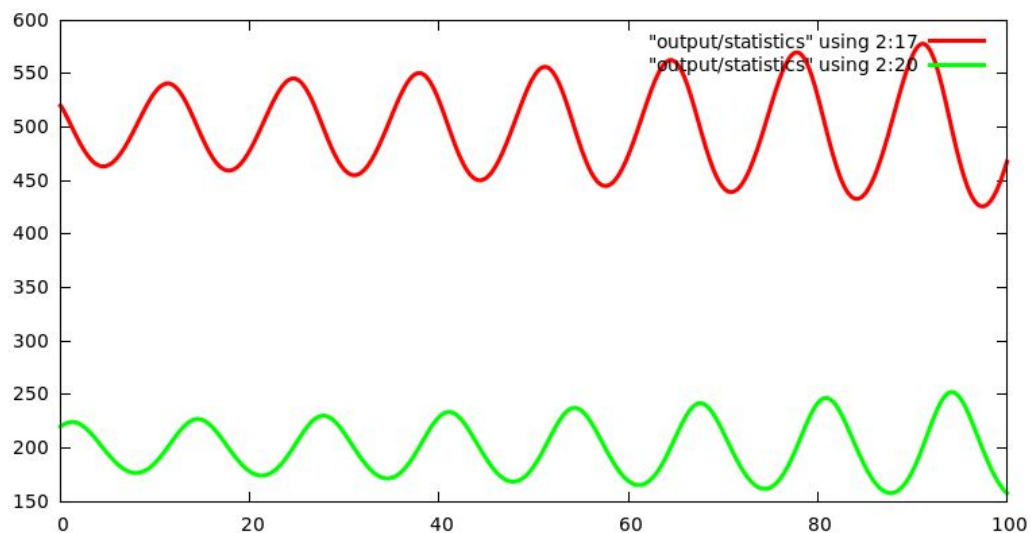
The figure below shows a first try to reproduce the setup from Keller et al., 2013.



Operator Splitting for models with melting

Juliane Dannberg, Timo Heister

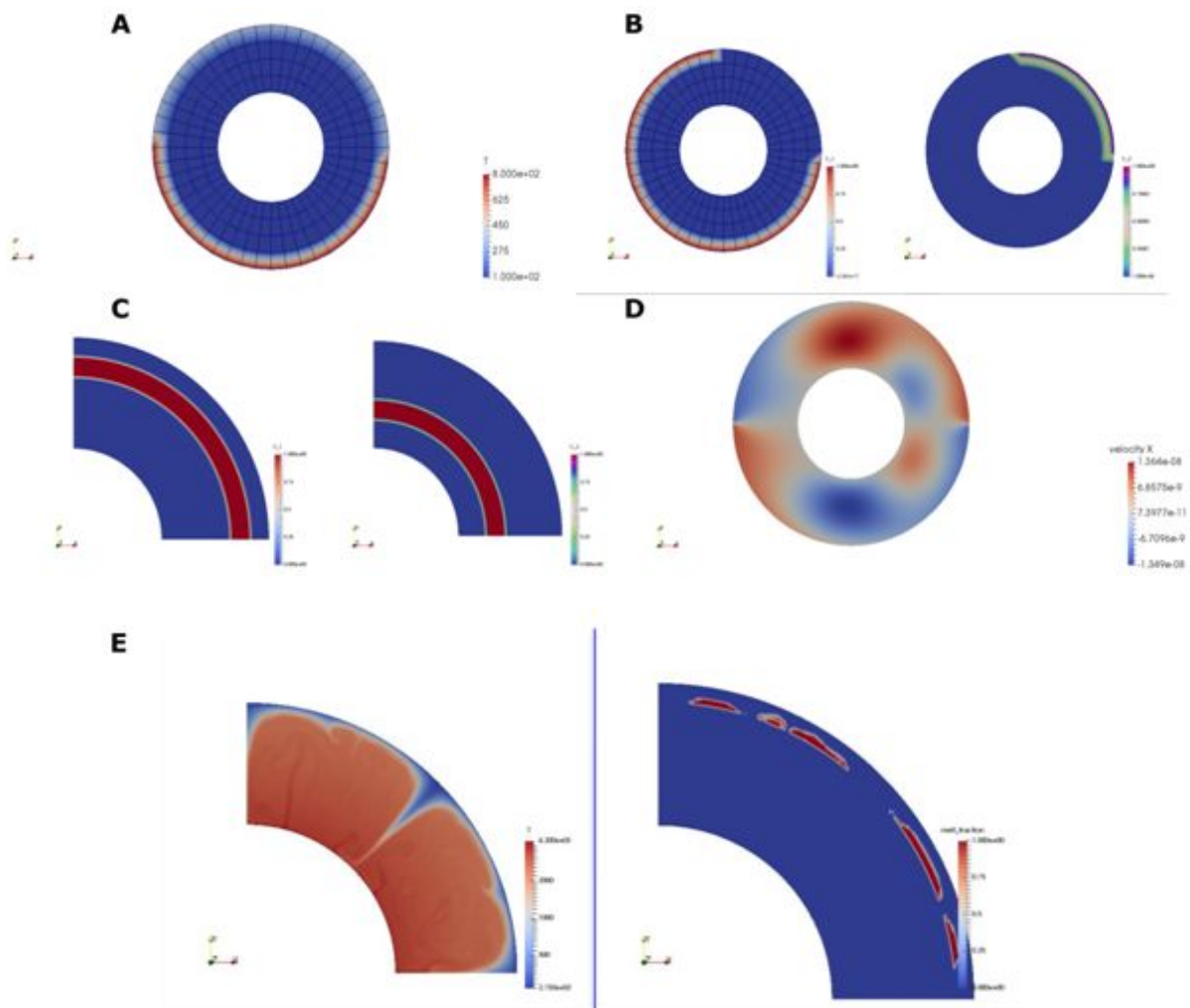
We worked on splitting the time stepping of advection and reaction terms (using operator splitting). This improves the convergence of the nonlinear solver in models with melting and freezing of melt substantially. The picture below shows an example without advection, but reactions of two compositional fields using the Lotka–Volterra equations as reaction terms.



User defined input coordinate systems and conditions

Matt Weller

I added the ability for the user to input spherical, depth, or Cartesian coordinates in the input file for boundary and initial compositions, temperatures, and velocities. Compositions allow for updates in the following time steps. I further tested melt in a 2d quarter annulus. As examples, the image below shows variable surface temperatures in spherical coordinates for a 2d shell (A); variable boundary compositions in spherical coordinates for a 2d shell (B); variable initial compositions in spherical coordinates for a quarter of a 2d shell (C); variable initial compositions in spherical coordinates for a 2d shell; and melting in a 2d annulus (E).

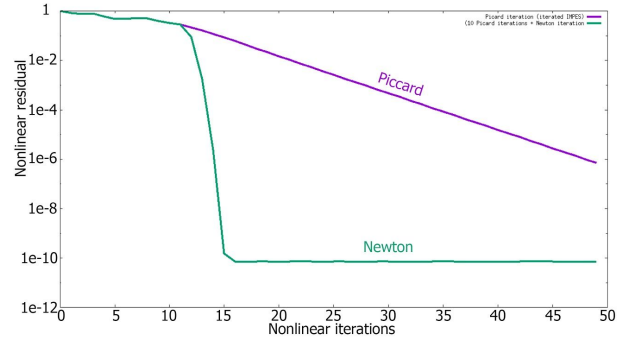
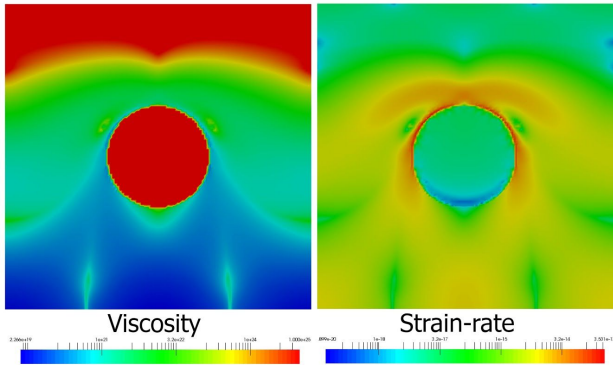


Newton Solver

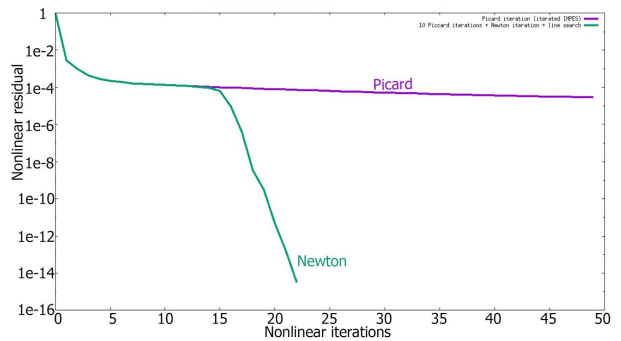
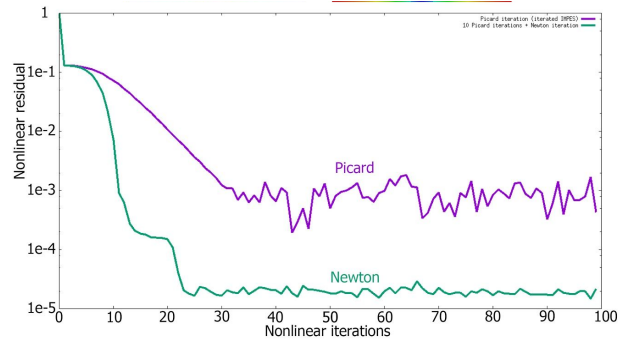
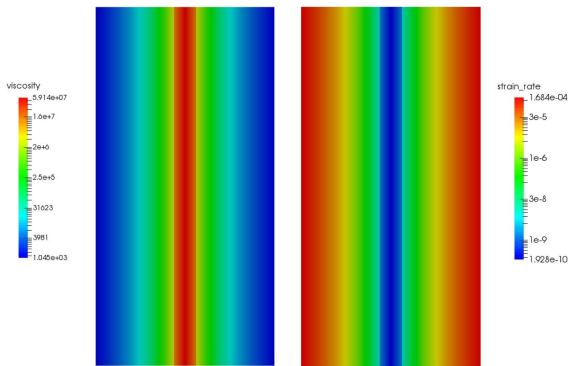
Menno Fraters, Wolfgang Bangerth

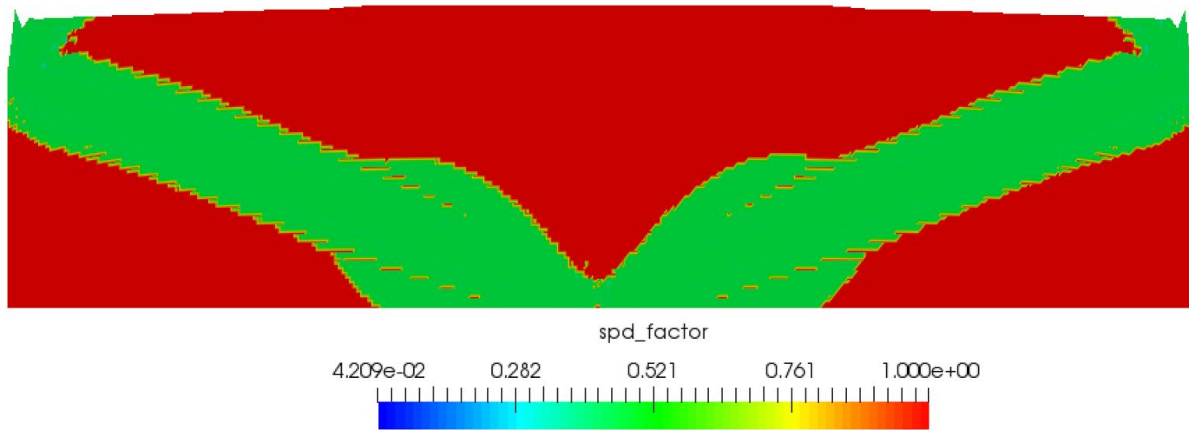
We worked on merging the Newton solver into the main repository. The following figures show results of the Newton solver for different benchmarks. A very large part of the Newton solver was merged into ASPECT during this hackathon.

Nonlinear sinker



Nonlinear channel flow





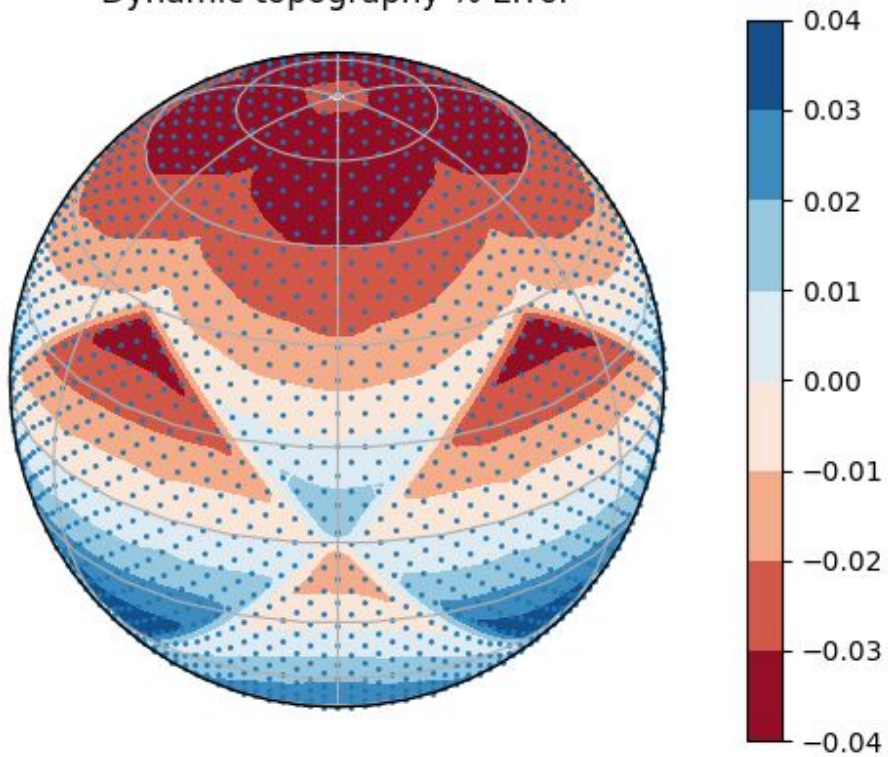
The figure above shows a visualization of the spd factor in the crustal model cookbook after one million years of model evolution, using the spd factor postprocessor, which was added this hackathon. This model uses the Drucker Prager material model.

The consistent-boundary-flux method for dynamic topography computation

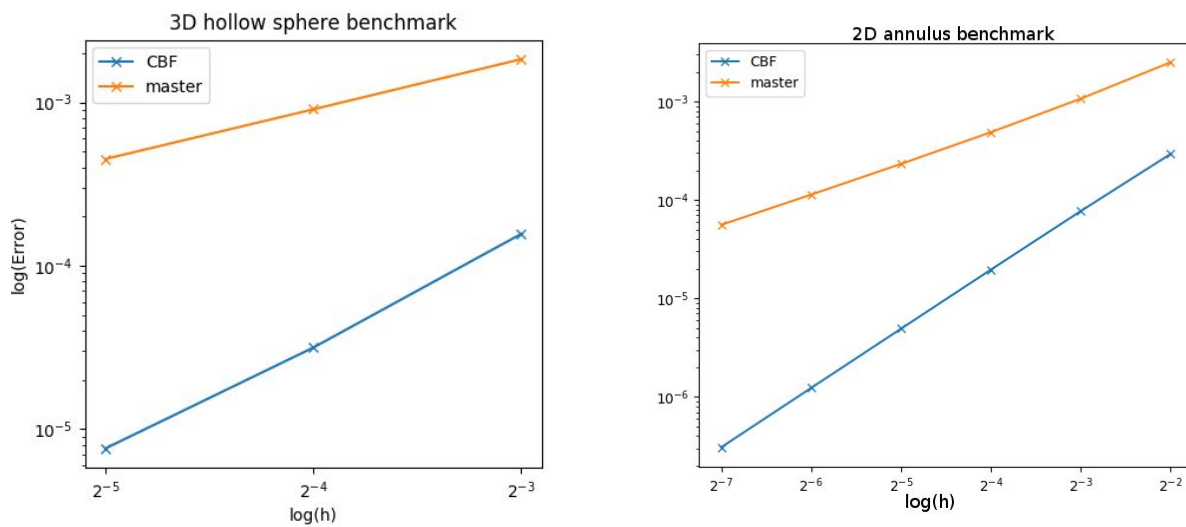
Ian Rose, Jacky Austermann, Cedric Thieulot

Determination of dynamic topography (and the closely associated geoid) depends on an accurate computation of stresses on the simulation surface. The previous method for calculating surface stress (known in the literature as “pressure smoothing”) suffers from poor accuracy and convergence. We implemented a more accurate method known as the “consistent boundary-flux” (CBF) method, which assembles a new Stokes finite element system, the solution to which is the traction at the simulation boundaries. By an appropriate choice of quadrature the solution of the CBF system can be about as fast as the pressure smoothing method.

Dynamic topography % Error



Spatial distribution of % error in the dynamic topography for the "Hollow Sphere" benchmark (with constant viscosity at global refinement level 4).



Convergence tests of error vs cell size (h) for the pressure smoothing vs CBF methods.

Particles and Melting Processes

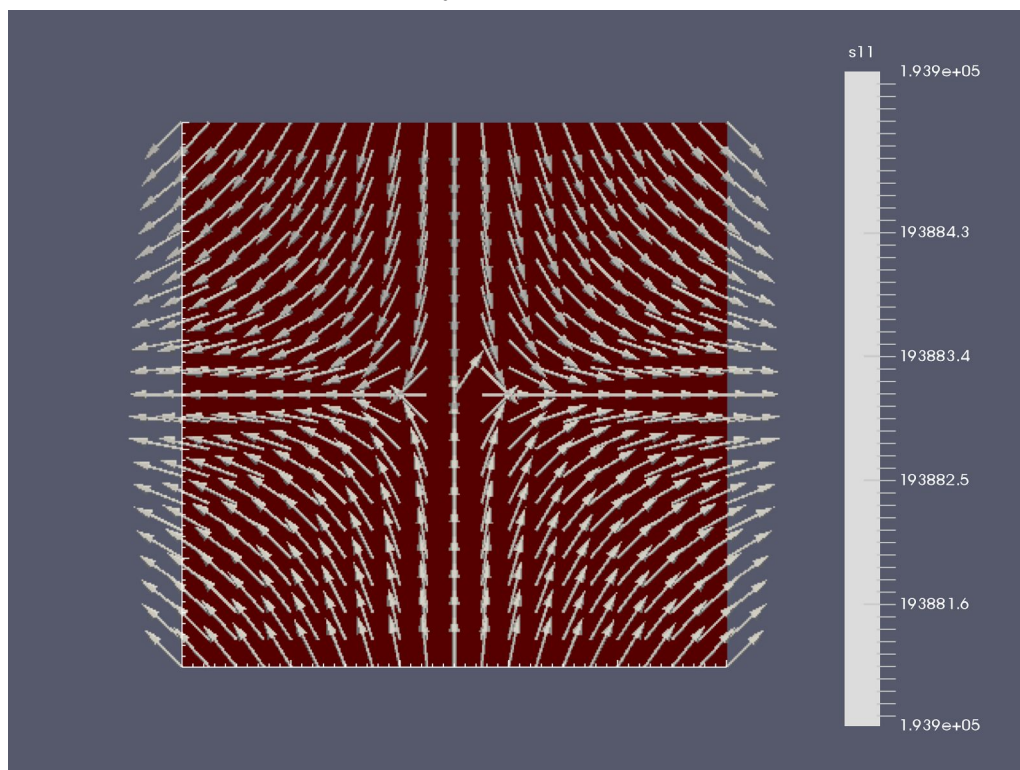
Rene Gassmoeller, Joe Schools

Particles now have the option to follow melt velocity rather than solid velocity using the property melt particle. Additionally, solid composition can be tracked in a rudimentary way by the particle property solid comp which records the peridotite depletion at the solid particle's position. In conjunction with the pT path property these new properties relating to melting can be used to create comparisons between melting in ASPECT and in other thermodynamic software/databases. Specifically a MATLAB script and associated files can be found at github.com/joeschools which convert the particle output files into a format which can be read by the alphaMELTS software.

Viscoelastic Rheology

John Naliboff, Cedric Theiulot

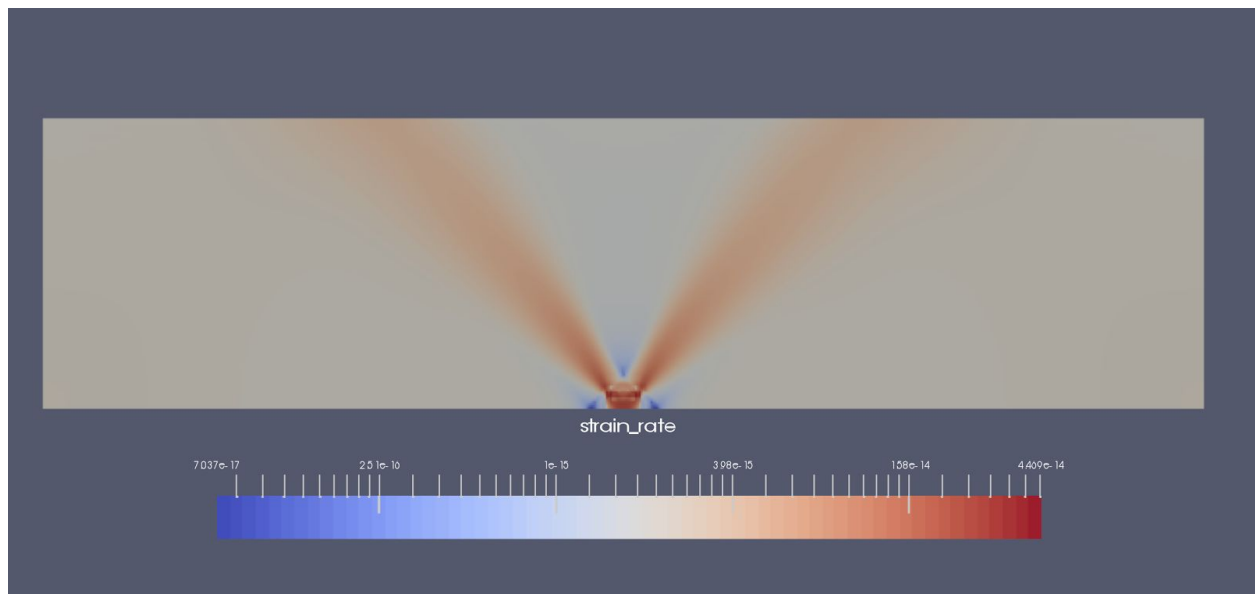
We worked on adding a viscoelastic material model to ASPECT. The implementation can currently match the analytical solution for viscoelastic stress build during pure shear for a few time steps. The horizontal stress and velocity fields associated with this problem (100 x 100 km domain, 0.5 cm/yr tangential velocity applied on each boundary) after the first time step (model time = 100 years) are shown below. However, instabilities eventually develop in the solution leading to rapid stress build-up that feeds into the stokes RHS. The next steps will be to isolate the source of these instabilities and identify solution(s).



Strain weakening using the full finite strain tensor

John Naliboff

I added the option to track the full finite strain tensor within the viscoplastic material model and use invariants derived from this quantity in the strain weakening section of the material model. The image below shows an extensional “brick experiment” (40 x 10 km, 0.5 km grid spacing, 0.2 cm/yr total extension rate) with shear bands localizing above a weak viscous seed. Strain accumulation along the shear bands promotes localization.



Incorporation of operators into Initial Temperature and Initial Composition Interface

Bob Myhill, Rene Gassmoeller

We modified the plugin interface for initial temperature and initial composition construction to accept operators (add, subtract, minimum, maximum) by which the temperature/composition fields can be modified.

This is very useful in cases where one wants to add, for example, a perturbation to an adiabatic background temperature field, because the individual plugins only have to implement one specific perturbation, or one specific background, and those can be mixed and matched.

New Material Model Constructor

Paul Bremner, Ian Rose

We created a new Material Model array constructor to initialize and populate all the relevant MaterialModel arrays. This constructor works in conjunction with a re-initialization function which simply re-initializes the same arrays. Along with the previously existing Material Model constructor, which only sizes the arrays, these new implementations allow simple, and standard, constructor and function calls to create and populate all the necessary arrays, and reduce overall lines of code. These constructors are now available and implemented within material model and postprocessing visualization plugins.

Multicomponent Material Model Clarification

Paul Bremner, Rene Gassmoeller, Ian Rose

We began work on clarifying the list of input parameters for the Multicomponent material model. Some parameters accept a list of values, and those parameters will have explicit labels within the parameter file to associate values to specific compositional fields. These changes make it clear which values are assigned to which compositional fields.

Seismic Model Input into Initial Temperature Overhaul

Paul Bremner, Jacky Austermann

We began to work on restructuring how seismic models are read in and processed into temperature and density distributions. The initial template was the plugin for S40RTS, and from there we have been changing it to be more general so that it is capable of reading in other seismic tomography models, as well as tomographies produced from joint inversions of gravity and seismic velocities.

3D subduction example

Bob Myhill, Anne Glerum, Juliane Dannberg, Rene Gassmoeller

We began work on a new 3D subduction example incorporating visco-plasticity, real slab geometries and plate motions demonstrating the means by which comparisons can be made with seismic observations.

Dynamic Friction Material Model

John Naliboff, Arushi Saxena, Cedric Thieulot

We have proposed a new material model that incorporates a strain rate-dependent friction coefficient into a Drucker Prager yield criterion. This type of material is similar in nature to rate-and-state friction models commonly applied to the strength of rocks during seismogenic events.

Geoid Postprocessor and Benchmark

Shangxin Liu, Ian Rose

We got a geoid post processor for 3D spherical shell geometry merged into ASPECT during the Hackathon, with Shangxin Liu's original version and the following improvement by using CBF (Consistent Boundary Flux) method to calculate dynamic topography contribution and also the geoid visualization plugin by Ian Rose. The geoid postprocessor benchmark was done in this way:

1. The density integral part has been benchmarked against an analytic solution of a single degree 2 order 0 density perturbation field as the current geoid test prm file;
2. The transfer of spherical harmonics of dynamic topography into spatial domain yields the same results as the dynamic topography computed from dynamic topography postprocessor;
3. The CBF method used to calculate dynamic topography has been benchmarked against the analytic solution from "Hollow Sphere" benchmark (See Section "The consistent-boundary-flux method for dynamic topography computation").

The geoid cookbooks of both a single harmonic perturbation and an Earth-like initial temperature condition have been added into ASPECT manual (See Section "Dynamic topography and geoid cookbook").

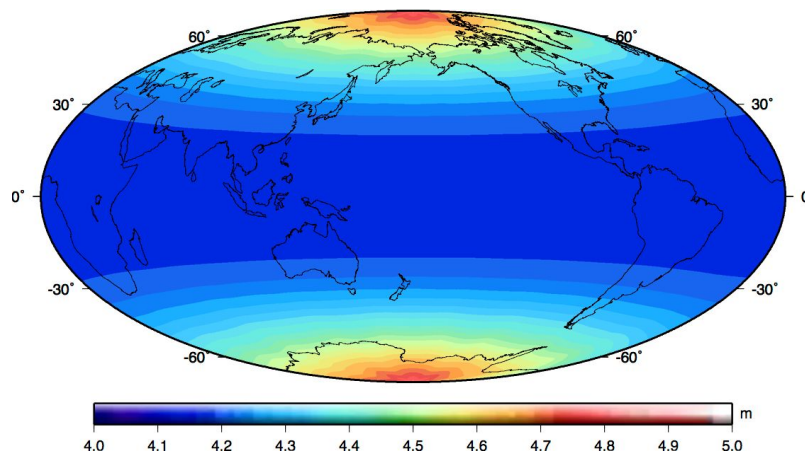


Figure above: Single degree 2 order 0 geoid from pure mathematical benchmark test (degree 0 order 0 part is not removed for benchmarking analytical solution purpose).

Jupyter Notebook for ASPECT

Lorraine J. Hwang, Timo Heister, and Ian Rose

We began developing a Jupyter Notebook for ASPECT. The goal of the project is to provide an introduction to ASPECT for new users without having to install the code. ASPECT and Jupyter notebooks are installed within a Docker container (tjhei/aspect-jupyter). The actual notebook can be run outside of the container or can be uploaded into the container. Major functionality was completed and tested during the hackathon. It is anticipated this will be completed and tested by the end of the summer.

Statistics about ASPECT's growth during the hackathon

The following contains a number of statistics about how much ASPECT has grown during the hackathon:

- Number of source files in ASPECT before/after: 449 -> 466 +17
- Lines of code in ASPECT before/after: 97,055 -> 102,826 +5,771
- Number of merged pull requests before/after: 1023 -> 1210 +187
- Commits in github before/after: 4,545 -> 4,893+338
- Number of tests before/after: 436 -> 476 +40

These numbers are a significant increase over the previous hackathon. (The added number of source lines of code is depressed by the merge of a number of patches that reduce the size of ASPECT significantly by replacing code blocks that have been repeated throughout the code base many times, by a single function call that refactors this code.) For comparison, these were the statistics for last year's (2016) hackathon:

- Number of source files in ASPECT before/after: 384 -> 427 +83
- Lines of code in ASPECT before/after: 83,274 -> 91,462 +8,188
- Number of merged pull requests before/after: 663 -> 747 +84
- Commits in github before/after: 3,721 -> 3,957+236
- Number of tests before/after: 333 -> 380 +47

These statistics were generated through the following commands:

- `find include/ source/ | egrep '\.(h|cc)$' | wc -l`
- `cat `find include/ source/ | egrep '\.(h|cc)$` | wc -l`
- `git log --format=oneline | grep "Merge pull request" | wc -l`
- `git log --format=oneline | grep -v "Merge pull request" | wc -l`
- `ls -l tests/*prm | wc -l`