ETH Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

European Research Council erc

DERDW Departement Erdwissenschaften

Numerical techniques for thermo-mechanical-fluid-flow modelling

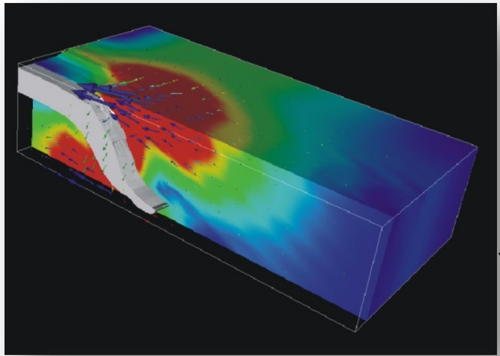
Boris Kaus[1,*], Dave May[1]
Tobias Keller[1], Marcel Thielmann[1], Naiara Fernandez[1,*], Tobias Baumann[1,*], Sarah Lechmann[1], Jonas Ruh[1]

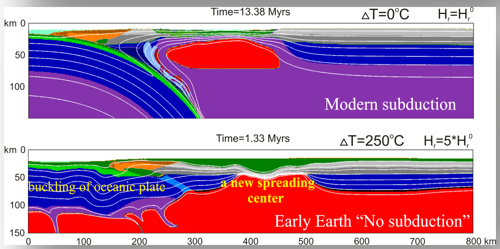
[1]ETH Zürich.
*Johannes-Gutenberg University Mainz (Germany) from next week onwards.

Outline

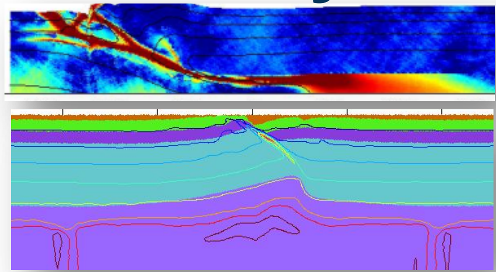
- Modelling of lithospheric deformation
  - Techniques
  - Progress
  - Challenges
- FEM or Finite Differences: what is better?
  - Accuracy, memory usage
  - Effect of element types
- Two-phase flow formulations coupled to lithospheric problems.

The Earth is a dynamical system

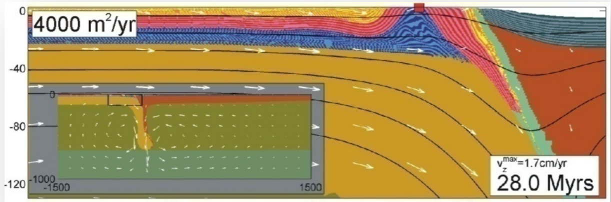




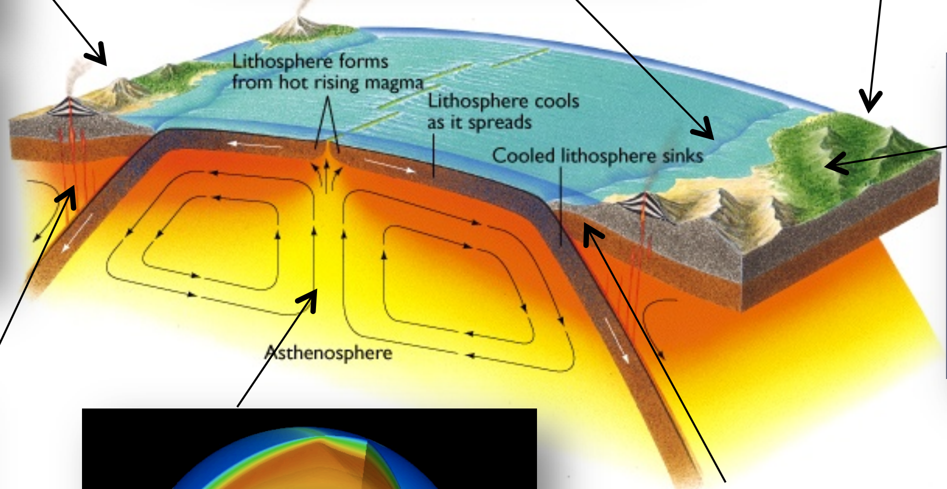
Subduction on present day & early Earth

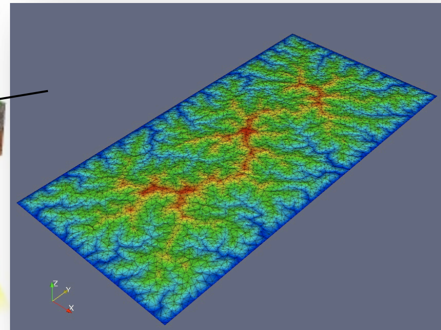


Subduction initiation & lithospheric failure

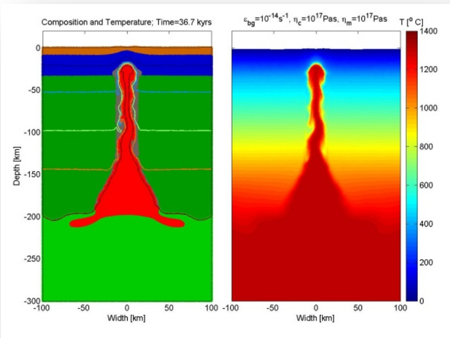


Interaction mantle, lithosphere & surface processes.

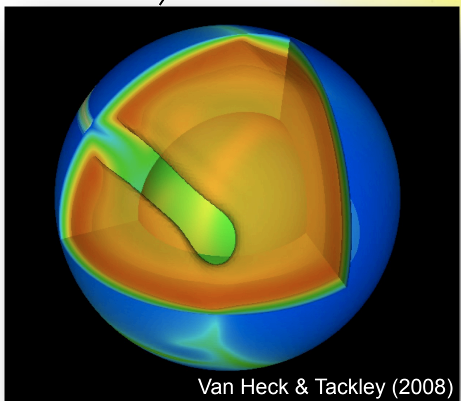




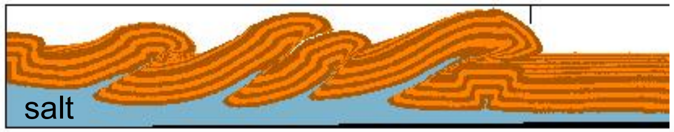
Surface processes

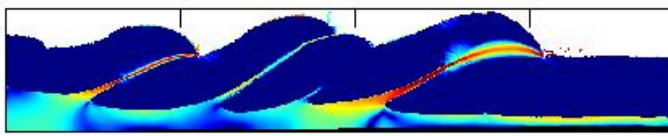


Magma migration & emplacement



Mantle convection





Brittle crustal deformation

Equations for mantle & lithosphere deformation

$$\frac{\partial u_i}{\partial x_i} = -\frac{1}{K} \cdot \frac{dP}{dt}$$ conservation of mass

$$-\rho g_i = -\frac{\partial P}{\partial x_i} + \frac{\partial \tau_{ij}}{\partial x_j}$$ conservation of momentum

$$\dot{\varepsilon}_{ij} = \frac{1}{2\eta_{eff}} \tau_{ij} + \frac{1}{2G} \frac{D\tau_{ij}}{Dt} + \dot{\lambda} \frac{\partial Q}{\partial \sigma_{ij}}$$ visco-elasto-plastic rheology

$$\rho c \frac{DT}{Dt} = \frac{\partial}{\partial x_i} \left(k \frac{\partial T}{\partial x_i} \right) + H_{sources}$$ conservation of energy

Stokes flow

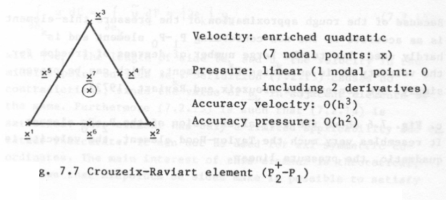
$$\frac{\partial u_i}{\partial x_i} = 0$$

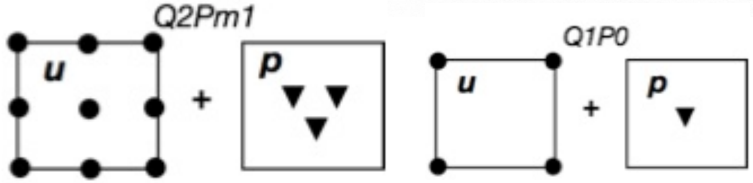
$$-\frac{\partial P}{\partial x_i} + \frac{\partial}{\partial x_j} \left(\eta_{eff} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \right) = -\rho g_i$$

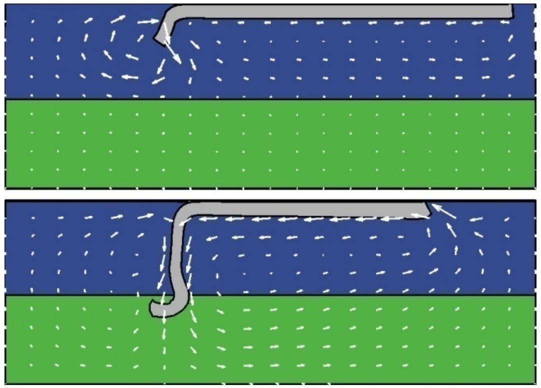
$$\longrightarrow \begin{pmatrix} \mathbf{K} & \mathbf{G} \\ \mathbf{G}^{\mathrm{T}} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{u} \\ \mathbf{p} \end{pmatrix} = \begin{pmatrix} \mathbf{f} \\ \mathbf{g} \end{pmatrix}$$

- Equations well established
- Typically need to be solved numerically
- Often reduced to variable viscosity Stokes problem

2D Lithospheric deformation: FEM - MILAMIN_VEP







- Uses MILAMIN technology (Dabrowski et al. 2008) to speed up matrix assembly phase.
- Visco-elasto-plastic rheology.
- Unstructured OR structured finite elements.
- Q_1P_0, Q_2P_{-1}, T_2P_{-1}.
- Free surface.
- Thermo-mechanical coupling.
- Tracer-based or contour-based material properties
- Lagrangian with remeshing for large deformations
- Phase transitions.
- Set of MATLAB functions.
- Different setups have different needs -> easy to 'build' your own code using existing functions.
- Routinely being used by various people over the last few years.

- **Disadvantages:** 2D only, serial, uses direct solvers.

MILAMIN_VEP solver

$$\begin{pmatrix} \mathbf{K} & \mathbf{G} \\ \mathbf{G}^{\mathrm{T}} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{u} \\ \mathbf{p} \end{pmatrix} = \begin{pmatrix} \mathbf{f} \\ \mathbf{g} \end{pmatrix}$$

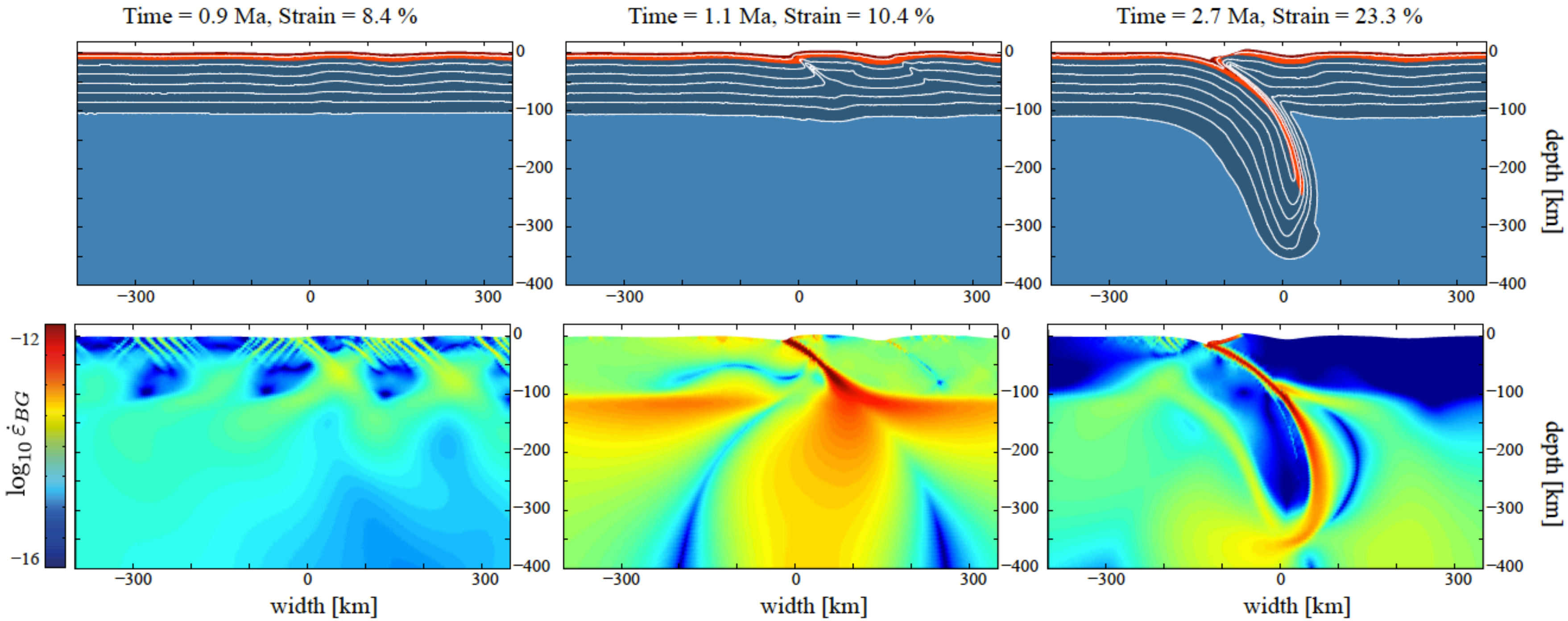
↓

$$\begin{pmatrix} \mathbf{K} & \mathbf{G} \\ \mathbf{G}^{\mathrm{T}} & -\frac{1}{\kappa \Delta t}\mathbf{M}_{\mathrm{p}} \end{pmatrix} \begin{pmatrix} \mathbf{u} \\ \mathbf{p}^{n+1} \end{pmatrix} = \begin{pmatrix} \mathbf{f} \\ \mathbf{g} - \frac{1}{\kappa \Delta t}\mathbf{M}_{\mathrm{p}}\mathbf{p}^{n} \end{pmatrix}$$

Iterated penalty method / Powell Hesteness iterations

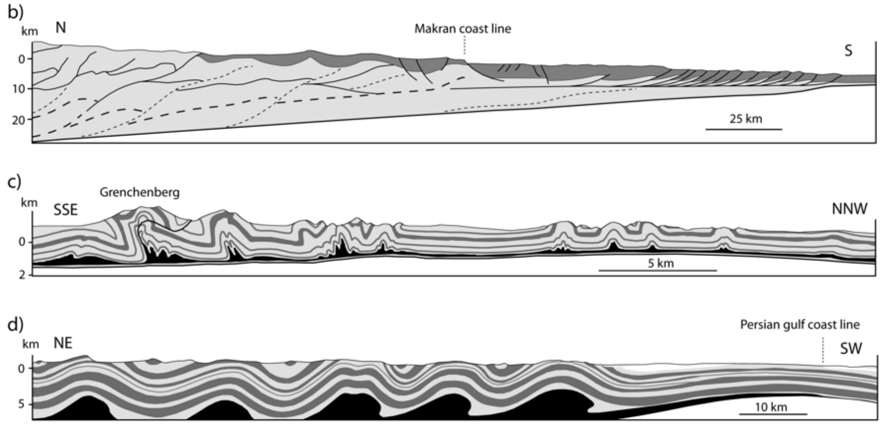
- Makes the system of equations slightly compressible
- Iterations are performed until $\mathbf{G}^{\mathrm{T}}\mathbf{u} < eps$
- Must be used in combination with a direct solver.
- Can deal with large viscosity contrasts *O(10^7).*

Subduction initiation – Marcel Thielmann



MILAMIN_VEP

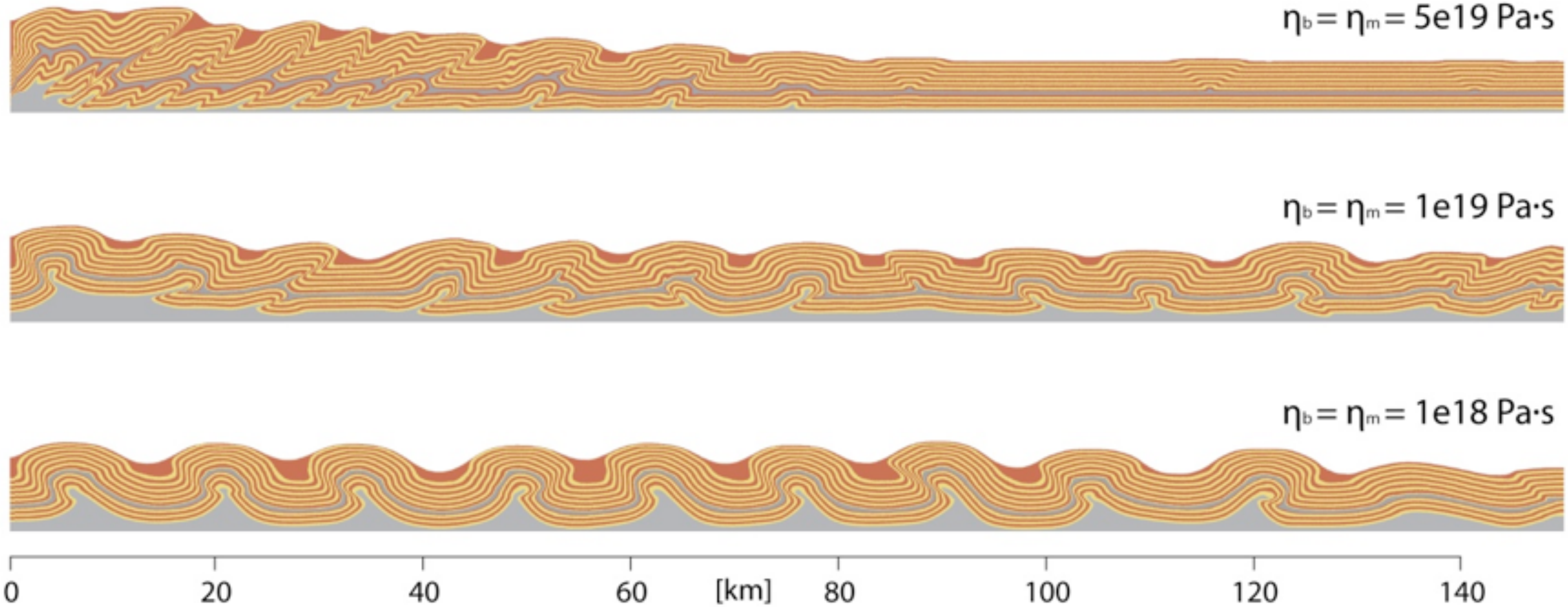
2D modelling of fold and thrust belts – Jonas Ruh



Makran (Iran)

Jura

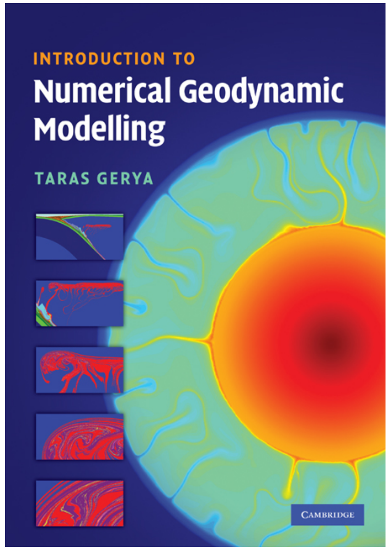
Zagros (Iran)

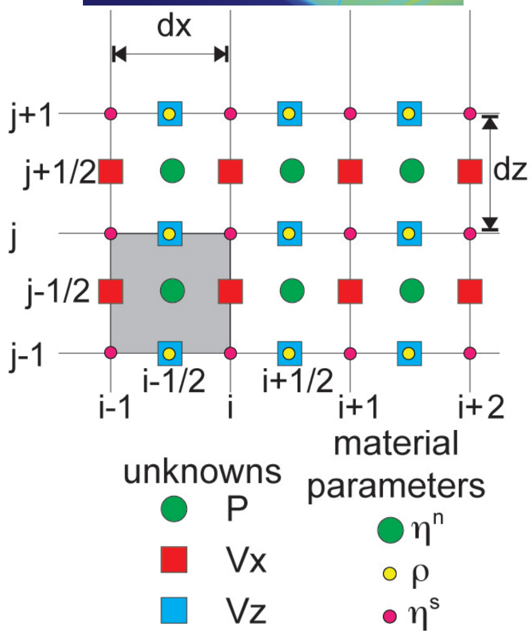


MILAMIN_VEP

2D Lithospheric Deformation: Finite difference methods

- Easy for teaching.
- Various formulations:
  - Primitive variable staggered grid Velocity-Pressure formulation (Gerya, Tackley)
  - Streamfunction approach (Schmeling).
  - Rotated finite difference stencil.
- Which one is better?
- Doesn't really matter.
  - Interpolation of properties is more important (Deubelbeiss & Kaus, 2008)
  - Staggered grid is the easiest to implement.
- Optimal interpolation scheme:
  - Duretz, May, Gerya & Tackley (G^3, 2011)
- Sticky air layer approximates free surface
  - Crameri et al. (submitted)





What are the challenges?

- Rheological/mechanical complexities (multiphase flow, melt migration etc.)

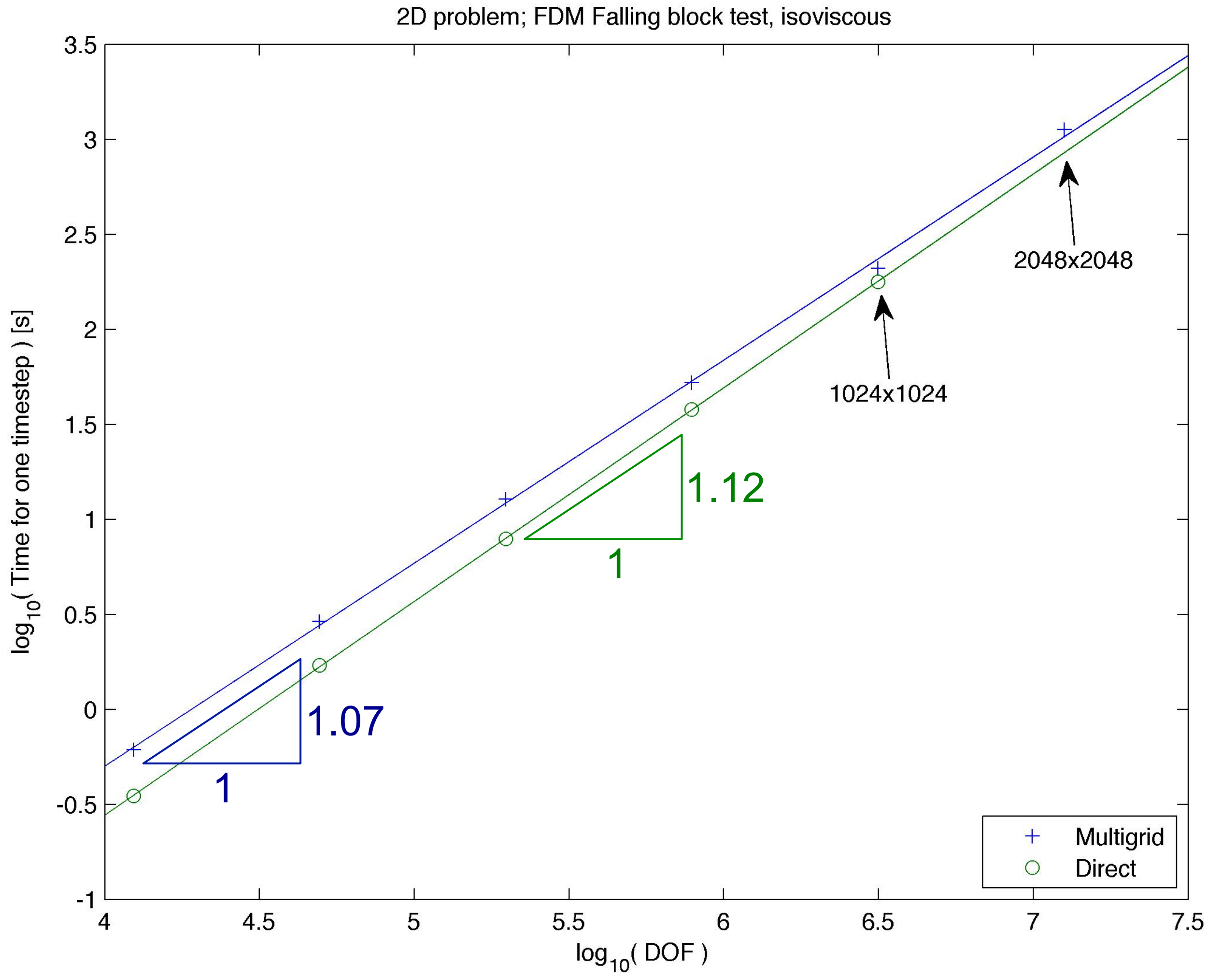
- 3D!!

  Stokes solver should be:
  - Robust
  - Fast
  - Use little memory

Direct vs. iterative solvers – 2D

$$\begin{pmatrix} K & G \\ G^T & 0 \end{pmatrix} \begin{pmatrix} u \\ p \end{pmatrix} = \begin{pmatrix} f \\ h \end{pmatrix}$$

$$Ax = b$$

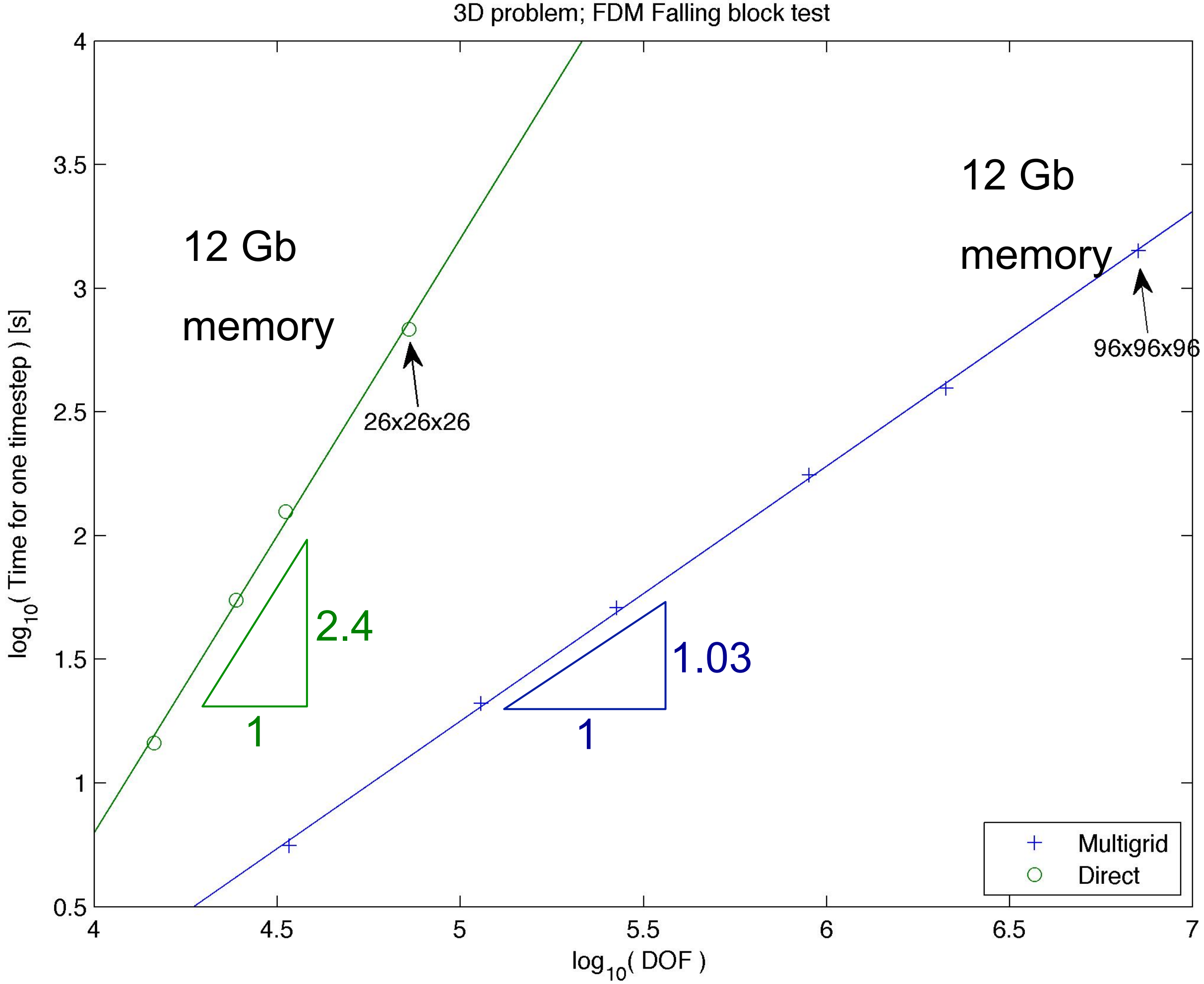


- 2D: direct solvers are quite fast and robust.

Direct vs. iterative solvers – 3D

$$\begin{pmatrix} K & G \\ G^T & 0 \end{pmatrix} \begin{pmatrix} u \\ p \end{pmatrix} = \begin{pmatrix} f \\ h \end{pmatrix}$$

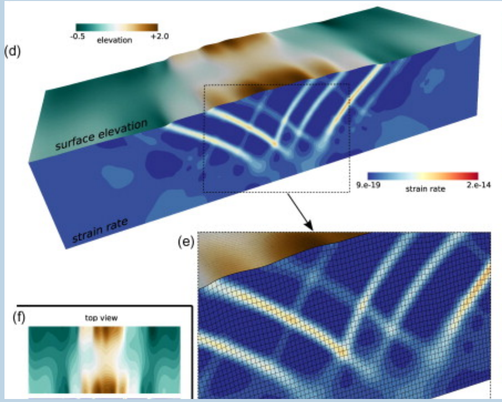
$$\mathcal{A}x = b$$



- 3D: multigrid is the only option for large resolutions.
- But: multigrid convergence deteriorates in presence of viscosity jumps.

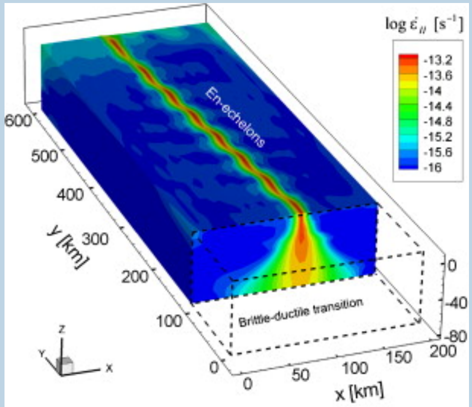
Some 3D lithospheric deformation codes:

Finite Element Models



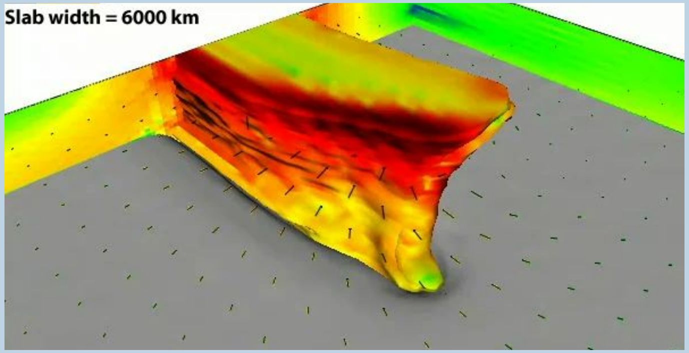
Fantom – C.Thieulot

Direct solver, Q1P0 elements



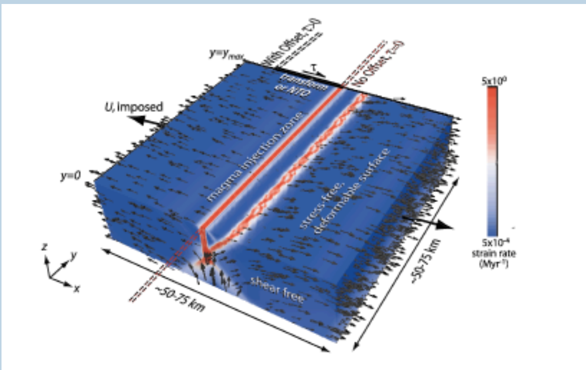
SLIM3D – A.Popov

Direct solver – Q1P0



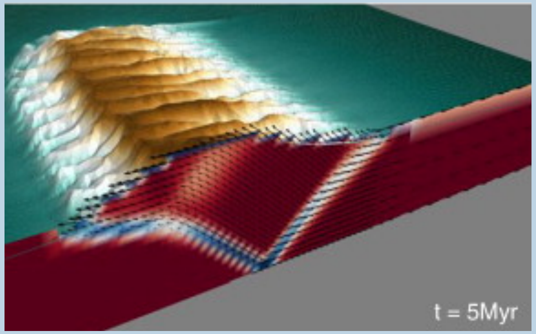
Underworld – L.Moresi & co

iterative solvers/ Q1P0–Q1Q1–[Q2Pm1]



GALE– W.Laundry & CIG

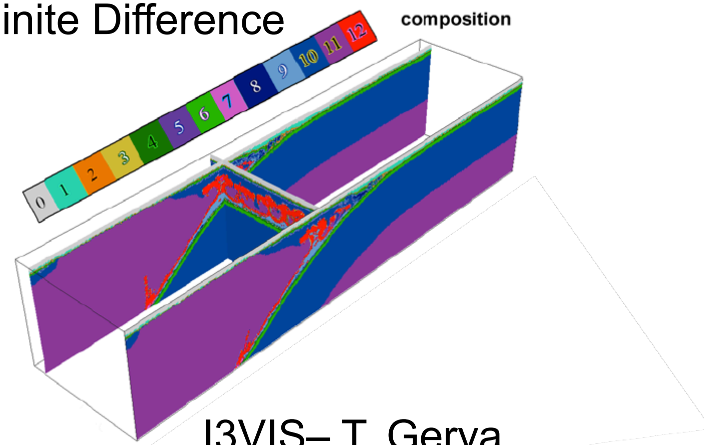
Iterative solvers – Q1Q1



Douar – J. Braun

Direct solver – Q1P0

Finite Difference



I3VIS– T. Gerya

Iterative (multigrid)

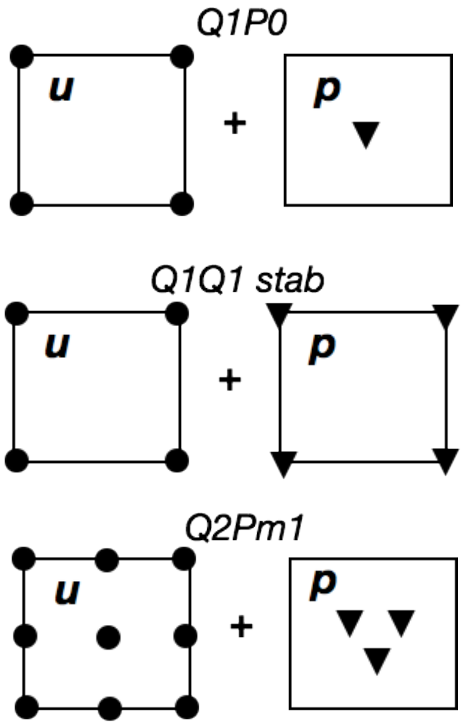
FEM – LaMEM (Lithosphere and Mantle Evolution Model)

- 3D only, written in C, uses PETSc (fully MPI parallel).
- Main idea: we don't really know which iterative/multigrid solvers work well for 3D geodynamics problems.
- Use either a finite element OR a finite difference discretization.
- Particles to trace material properties.
- Most options (solvers etc.) configurable from command-line.
- Change element-type from the command-line.
  - ./LaMEM –vpt_element Q1P0/Q2Pm1_global/Q1Q1/FDSTAG

$$\begin{pmatrix} \mathbf{K} & \mathbf{G} \\ \mathbf{G}^T & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{u} \\ \mathbf{p} \end{pmatrix} = \begin{pmatrix} \mathbf{f} \\ \mathbf{g} \end{pmatrix}$$

LaMEM – elements & discretization

Finite Element Method (FEM)



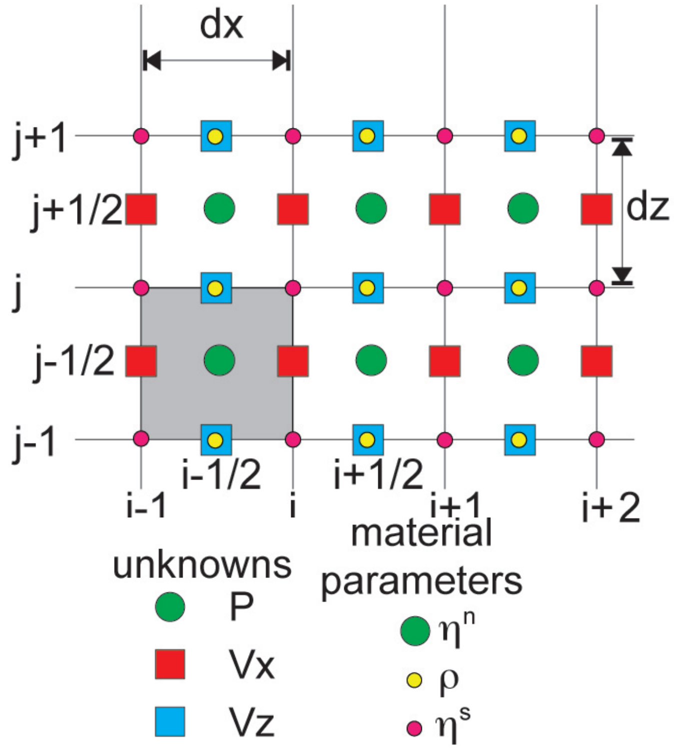
Not LBB stable

Stabilized but not fully incompressible

LBB stable

- Only LBB stable ones are very reliable.
- Viscosity should be constant or smoothly varying within an element

Finite Difference Method



- Staggered grid.
- Viscosity defined at two locations.

LaMEM – solver strategies

$$\begin{pmatrix} K & G \\ G^T & 0 \end{pmatrix}\begin{pmatrix} u \\ p \end{pmatrix} = \begin{pmatrix} f \\ h \end{pmatrix}$$

(1) Schur complement reduction

$$\begin{pmatrix} K & G \\ 0 & S \end{pmatrix}\begin{pmatrix} u \\ p \end{pmatrix} = \begin{pmatrix} f \\ \hat{h} \end{pmatrix}$$

$$S = G^T K^{-1} G$$

$$\hat{h} = G^T K^{-1} f - h$$

1) solve for p: $Sp = \hat{h}$

2) solve for u: $Ku = f - Gp$

$y = Sx$ is computed as:

$$f^* = Gx$$

$$Ku^* = f^*$$

$$y = G^T u^*$$

(2) Fully coupled

$$\begin{pmatrix} K & G \\ G^T & 0 \end{pmatrix}\begin{pmatrix} u \\ p \end{pmatrix} = \begin{pmatrix} f \\ h \end{pmatrix}$$

is solved iteratively, with as preconditioner:

$$\begin{pmatrix} \hat{K} & G \\ 0 & -\hat{S} \end{pmatrix}$$

where $\hat{S} = \frac{1}{\eta^e} M_p$

(3) Powell-Hesteness iterations (penalty method)

$$\begin{pmatrix} K & G \\ G^T & -\frac{1}{\kappa\Delta t}M_p \end{pmatrix}\begin{pmatrix} u \\ p^n \end{pmatrix} = \begin{pmatrix} f \\ -\frac{1}{\kappa\Delta t}M_p p^n \end{pmatrix}$$

Iterations are perfo... $\frac{\ldots^{n+1} - p^n}{\ldots\Delta t} <$ eps

LaMEM & multigrid – FC solver

$$\begin{pmatrix} \mathbf{K} & \mathbf{G} \\ \mathbf{G}^T & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{u} \\ \mathbf{p} \end{pmatrix} = \begin{pmatrix} \mathbf{f} \\ \mathbf{g} \end{pmatrix}$$

1. Solve iteratively (e.g. using FGMRES)

with as preconditioner:

$$\begin{pmatrix} \mathbf{K} & \mathbf{G} \\ \mathbf{0} & -\frac{1}{\overline{\eta}^e}\mathbf{M_p} \end{pmatrix}$$

2. Solve $\mathbf{Ku}$ iteratively (e.g. using FGMRES)

with as preconditioner:

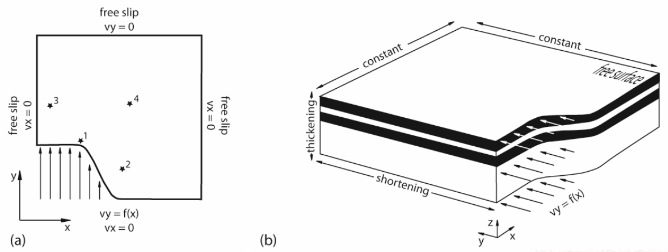
$$\mathbf{K} \sim \begin{pmatrix} \mathbf{K}_{x,x} & \mathbf{K}_{x,y} & \mathbf{K}_{x,z} \\ \mathbf{K}_{y,x} & \mathbf{K}_{y,y} & \mathbf{K}_{y,z} \\ \mathbf{K}_{z,x} & \mathbf{K}_{z,y} & \mathbf{K}_{z,z} \end{pmatrix} \begin{pmatrix} \mathbf{u}_x \\ \mathbf{u}_y \\ \mathbf{u}_z \end{pmatrix}$$

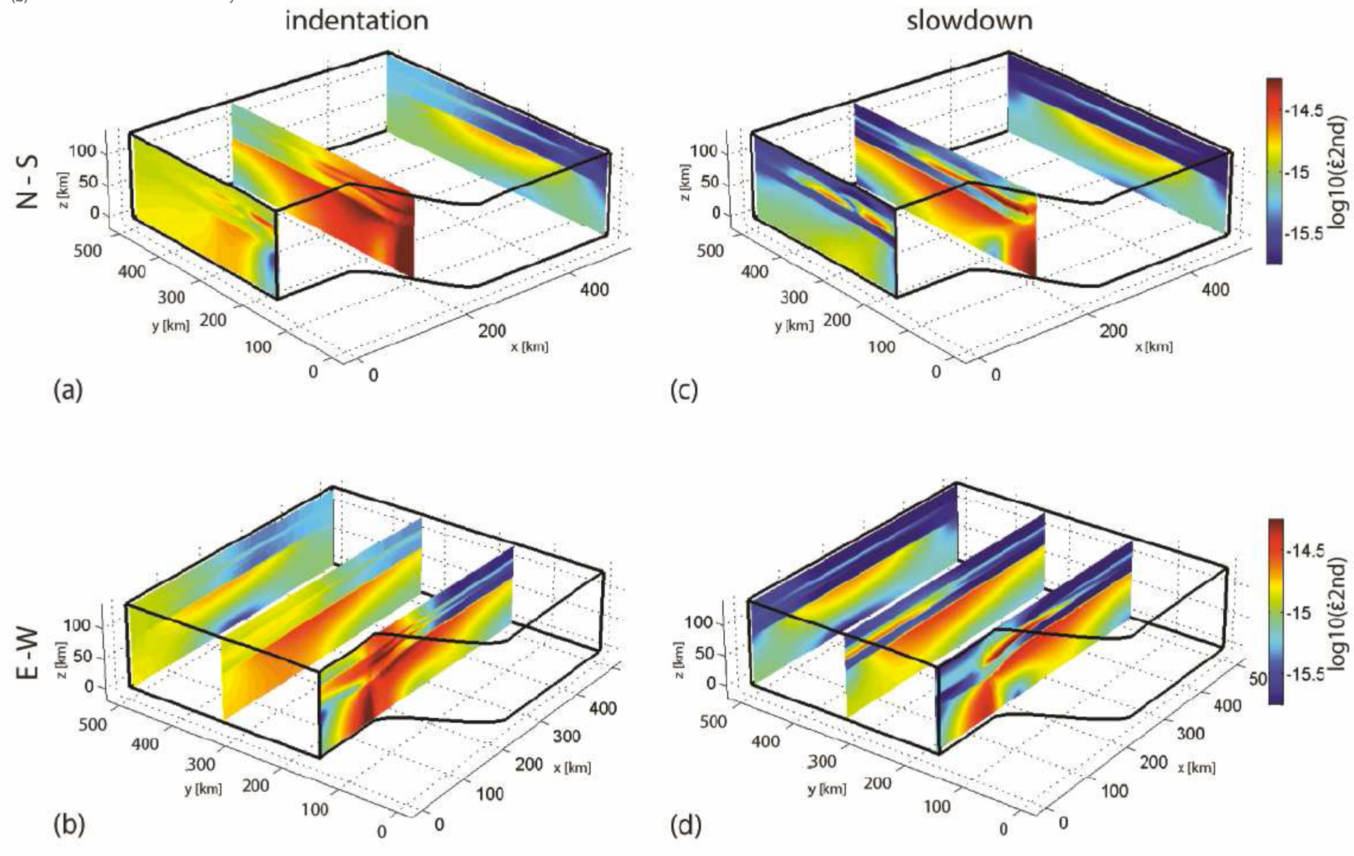
'fieldsplit'

$$\sim \begin{pmatrix} \mathbf{K}_{x,x} & 0 & 0 \\ 0 & \mathbf{K}_{y,y} & 0 \\ 0 & 0 & \mathbf{K}_{z,z} \end{pmatrix} \begin{pmatrix} \mathbf{u}_x \\ \mathbf{u}_y \\ \mathbf{u}_z \end{pmatrix}$$

Solve with iterative method (GMRES or CG) with an Algebraic or Geometric Multigrid method as preconditioner.

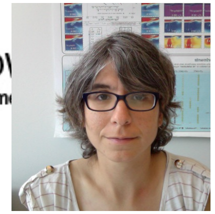
Thin sheet vs. 3D models – Sarah Lechmann



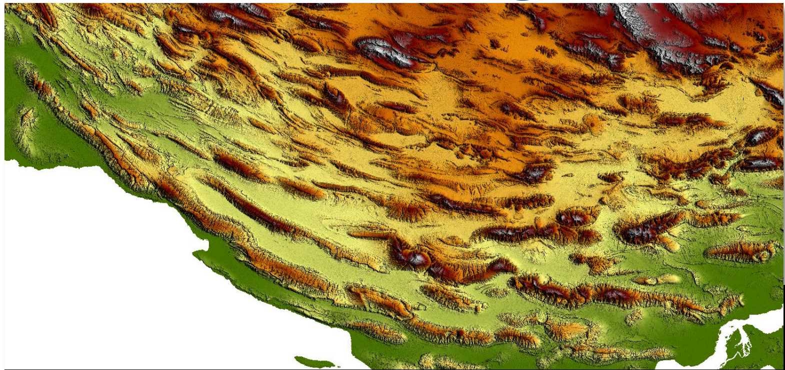


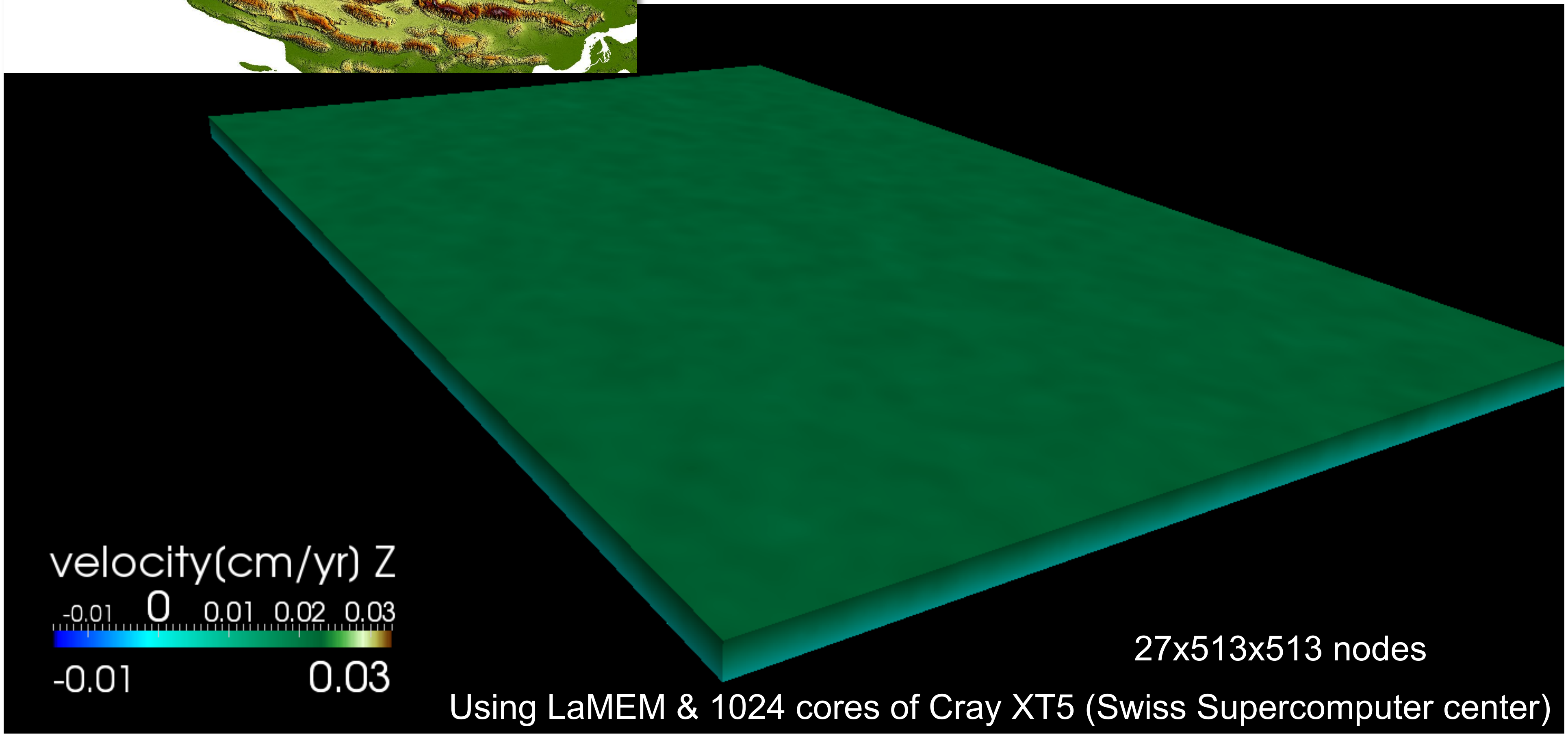
Lechmann, May, Kaus & Schmalholz (in press), GJI.

LaMEM



3D modelling of detachment folding – Naiara Fernandez

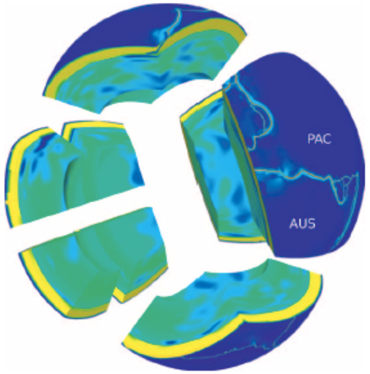


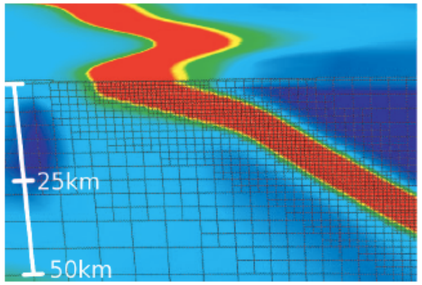


FEM vs. FDM

The Dynamics of Plate Tectonics and Mantle Flow: From Local to Global Scales

Georg Stadler,[1] Michael Gurnis,[2]* Carsten Burstedde,[1] Lucas C. Wilcox,[1]†
Laura Alisic,[2] Omar Ghattas[1,3,4]

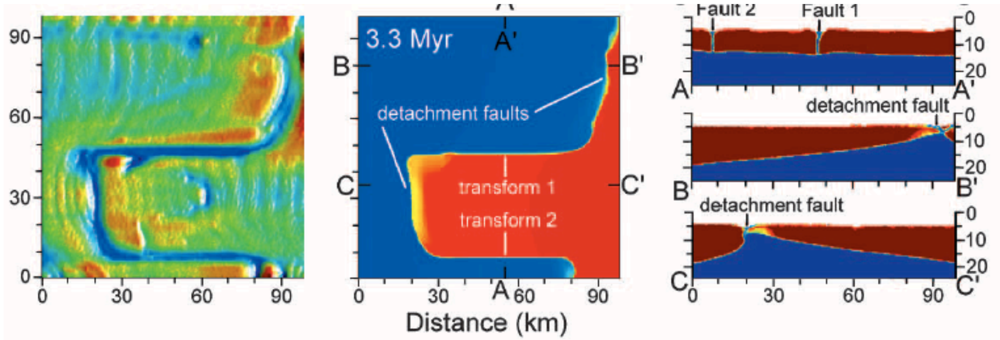




- RHEA
- FEM method, Q1Q1stab
- Adaptive Mesh Refinement (AMR)
- 1.2 billion DOF's on ~8000 processors
- ~150'000 DOF/processor
- 1 timestep: ~9 hours CPU time

Dynamical Instability Produces Transform Faults at Mid-Ocean Ridges

Taras Gerya



- I3VIS
- Staggered grid FDM.
- Uniform mesh
- 197x197x96 nodes; ~15x10^6 DOF's on 1 processor
- ~15 million DOF/processor
- 1 timestep:~5 hours CPU time (1th)
  - 2-3 minutes (subsequent)

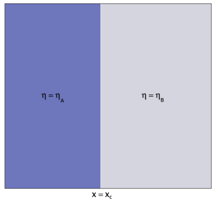
FEM or Finite Difference?

- (1) Accuracy
- (2) Memory usage
- (3) Speed
- (4) How well do they work with iterative solvers?

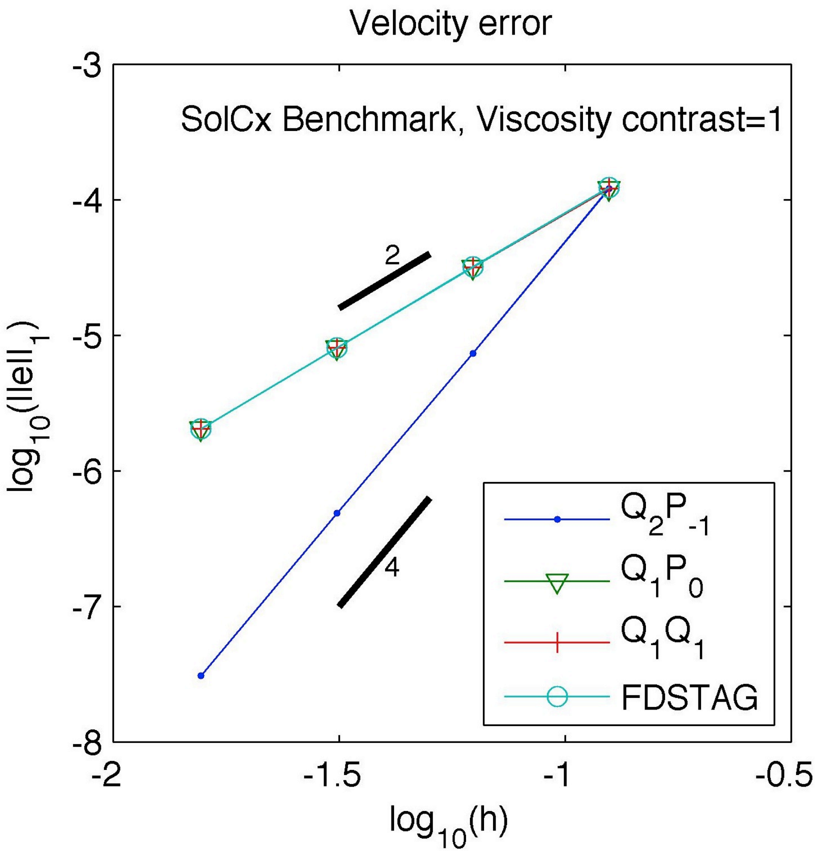
FEM vs. Finite difference - accuracy benchmark

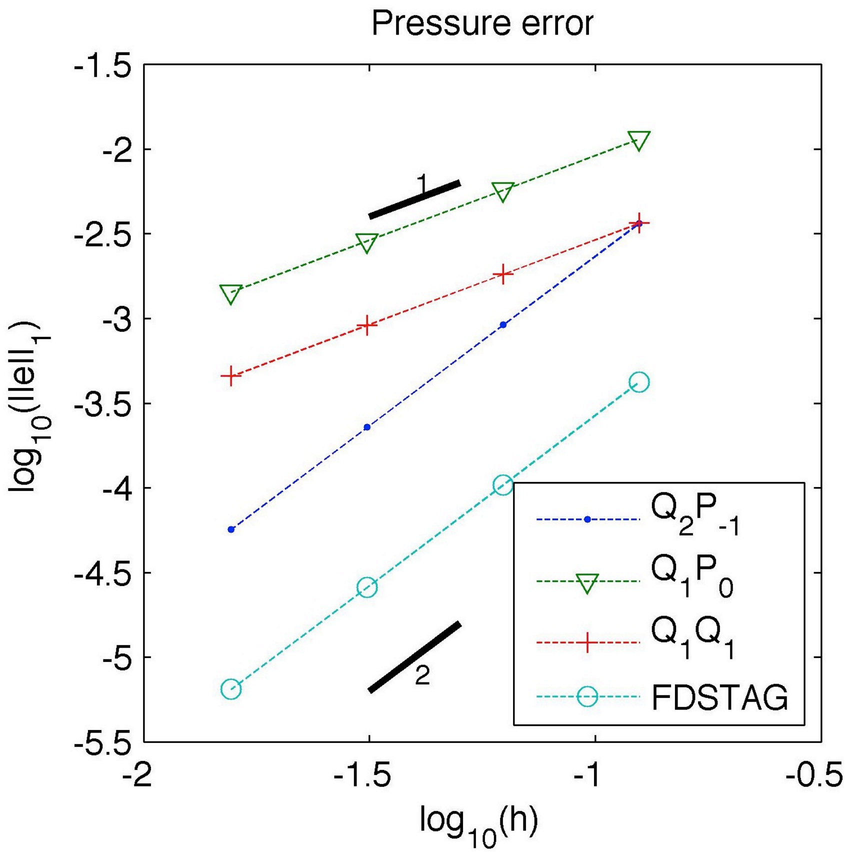
$\rho = -\sigma \sin(n_z \pi z) \cos(n_x \pi x)$

Figure 5: Solution (**SolCx**): This solution has a box of density $\rho = -\sigma \sin(n_z \pi z) \cos(n_x \pi x)$. It is has a viscosity jump at $x = x_c$. The boundary conditions are free slip everywhere on the surfaces of the unit box.



Isoviscous





High-order element wins for velocity

FEM vs. Finite difference - accuracy benchmark

$\rho = -\sigma \sin(n_z \pi z) \cos(n_x \pi x)$

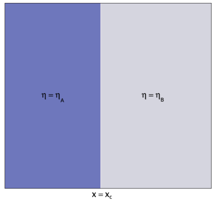
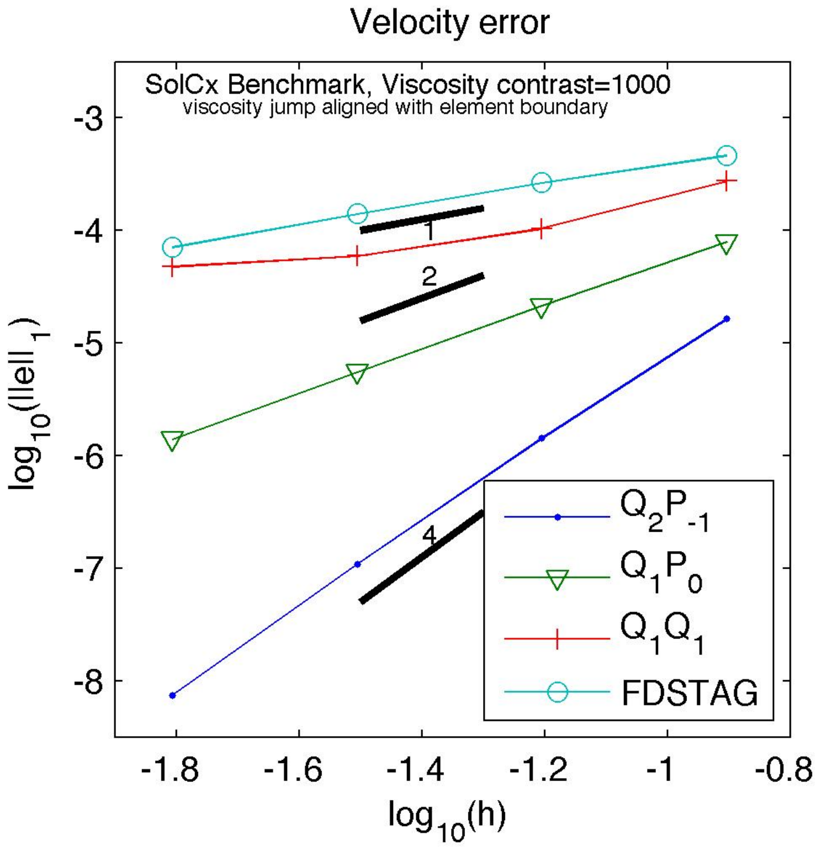
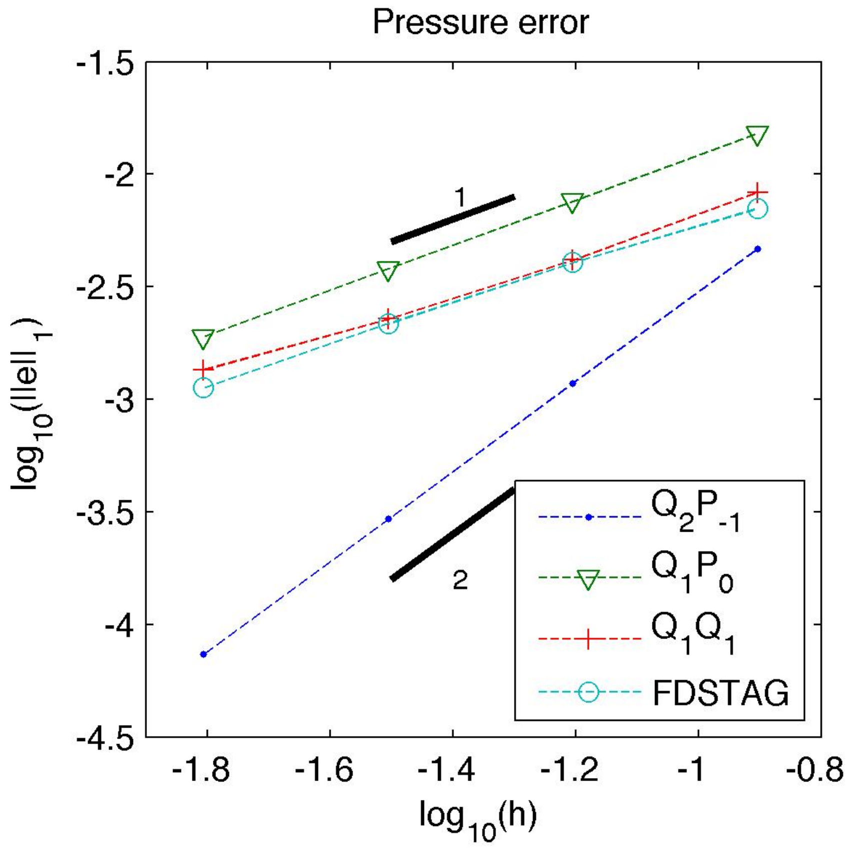


Figure 5: Solution (**SolCx**): This solution has a box of density $\rho = -\sigma \sin(n_z \pi z) \cos(n_x \pi x)$. It is has a viscosity jump at $x = x_c$. The boundary conditions are free slip everywhere on the surfaces of the unit box.

Viscosity contrast 1000, element boundary aligned with jump



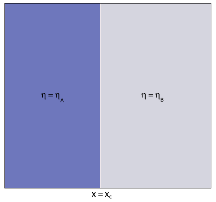


High-order element wins.

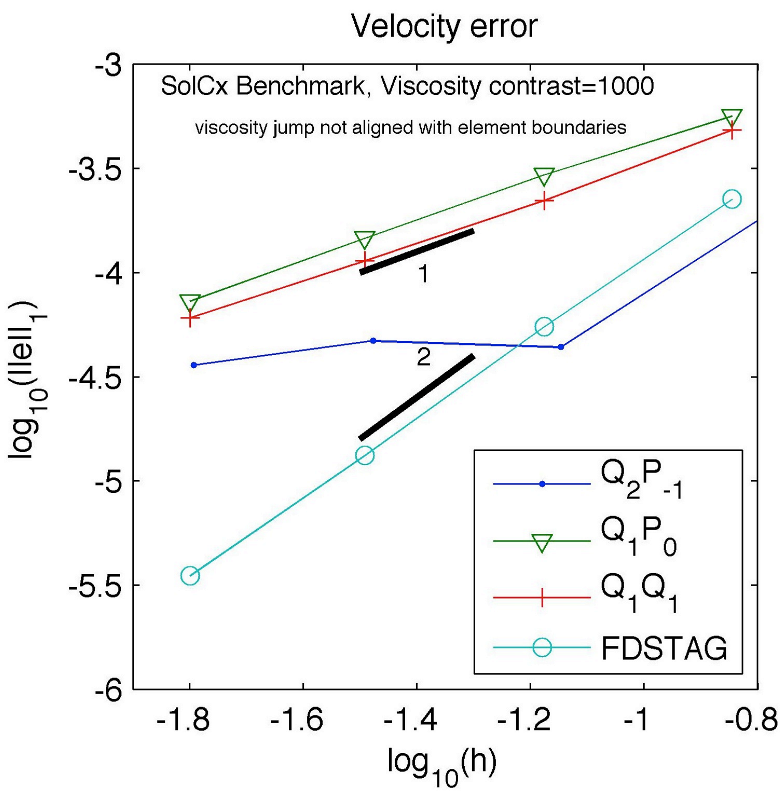
FEM vs. Finite difference - accuracy benchmark

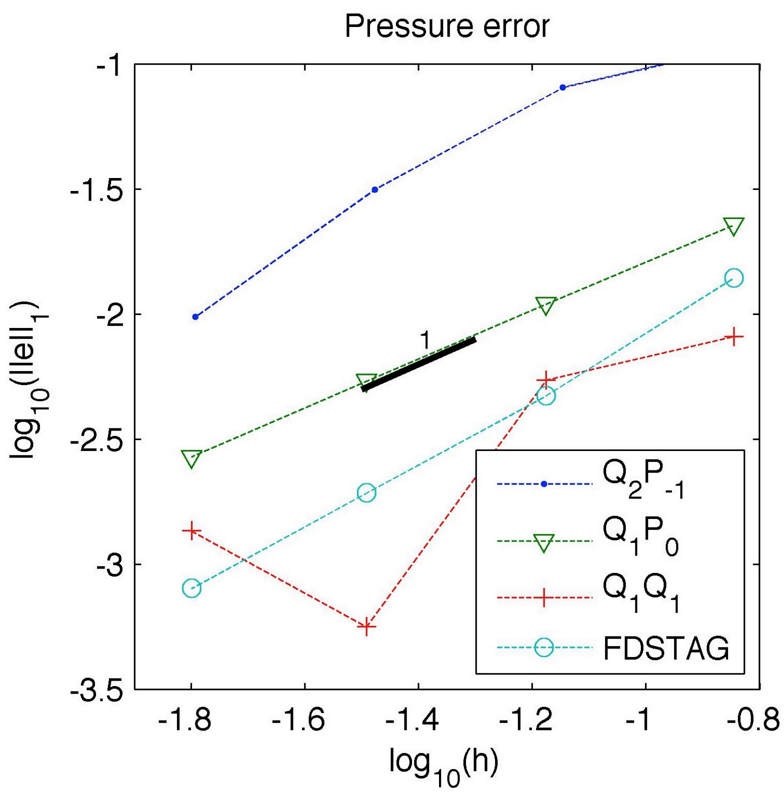
$\rho = -\sigma \sin(n_z \pi z) \cos(n_x \pi x)$

Figure 5: Solution (**SolCx**): This solution has a box of density $\rho = -\sigma \sin(n_z \pi z) \cos(n_x \pi x)$. It is has a viscosity jump at $x = x_c$. The boundary conditions are free slip everywhere on the surfaces of the unit box.



Viscosity contrast 1000, element boundary NOT aligned with jump

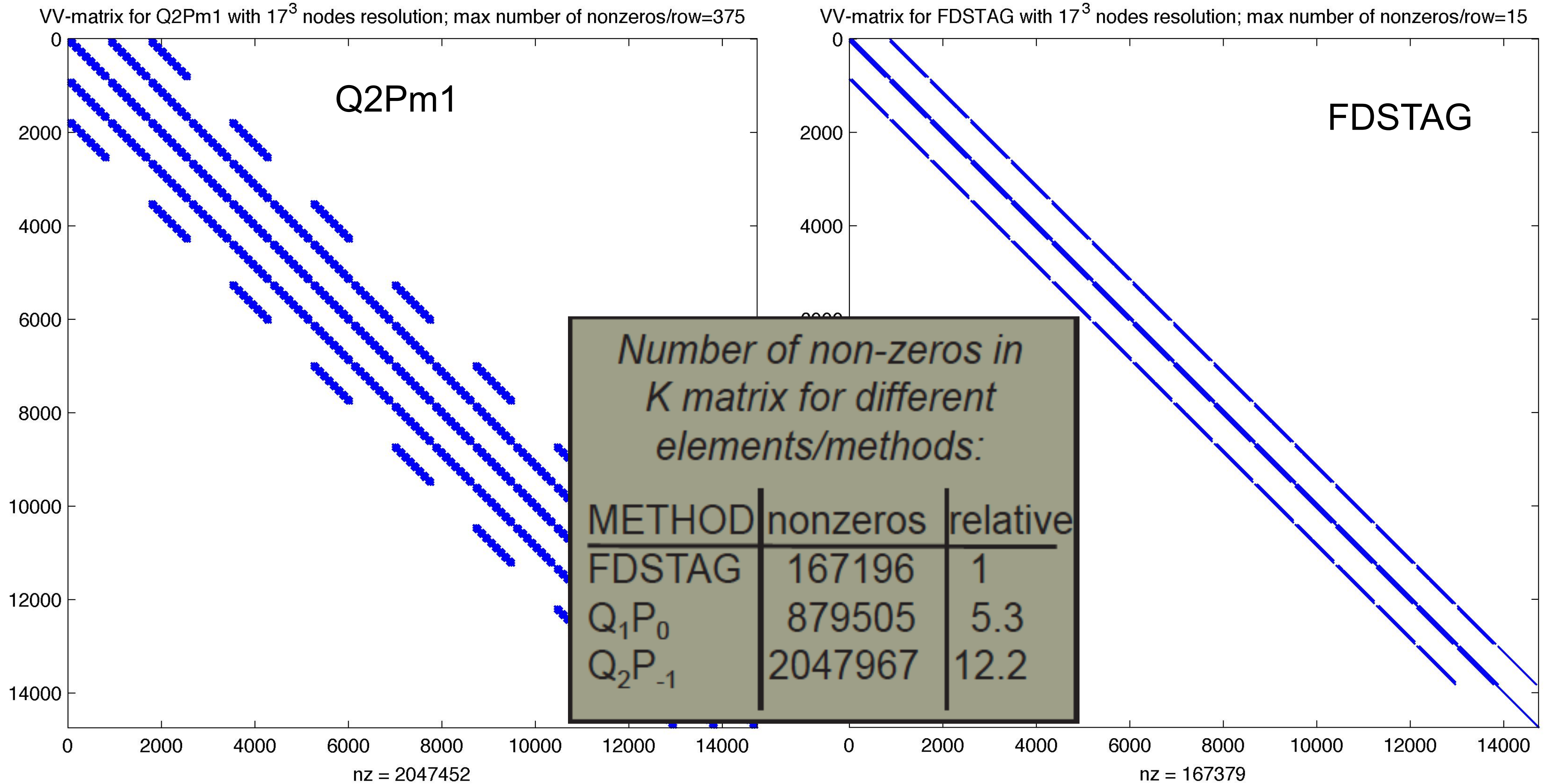




FDSTAG not so bad

High-order element sucks.

FEM vs. FDSTAG – memory usage



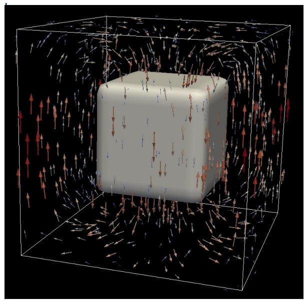
Number of non-zeros in K matrix for different elements/methods:

METHOD	nonzeros	relative
FDSTAG	167196	1
Q_1P_0	879505	5.3
Q_2P_{-1}	2047967	12.2

FDSTAG requires *significantly* less memory!

Matrix-vector multiplications much faster!

FEM vs. FDSTAG – iterative performance



SCR method, Falling block test with viscosity contrast 10^3

# nodes	Q_2P_{-1}	Q_1P_0	Q_1Q_1	FDSTAG
17^3	8/33.0 (46.1s)	24/24.0 (36.4s)	9/25.4 (44.7s)	13/53.4 (29.1s)
33^3	8/33.5 (408s)	31/24.8 (297.0s)	9/26.6 (337.1s)	12/67.5 (162.0s)
65^3	8/47.9 (4630.7s)	44/23.8 (2787.8s)	10/24.5 (2817.1s)	11/78.2(1233.8s)

Reported numbers: # of Schur iterations/ Average number of inner iterations (time in s). In all simulations, 2-3 ML solves are used.

```
./LaMEM -restart 0 -levels 1 -mumax 1e3 -UzawaSolver 1 -SolverType 2 -schur_ksp_type fgmres  -schur_ksp_converged_reason -schur_ksp_monitor -schur_ksp_rtol 1e-5 -mat_schur_ksp_type fgmres -mat_schur_ksp_rtol 1.0e-6 -mat_schur_pc_type fieldsplit
-mat_schur_fieldsplit_pc_type ml -mat_schur_ksp_converged_reason -mat_schur_pc_fieldsplit_type ADDITIVE  -mat_schur_pc_fieldsplit_block_size 3 -vpt_element Q1P0 -nnode_x 17 -nnode_y 17 -nnode_z 17
```

FC method, Falling block test with viscosity contrast 10^3

# nodes	Q_2P_{-1} (LBB stable)	Q_1P_0 (Not LBB stable)	Q_1Q_1 (Stabilized)	FDSTAG
17^3	13/8.5 (17.7s)	38/8.8 (22.0s)	13/9.7 (19.7s)	22/10.2 (20.0s)
33^3	14/10.3 (154.7s)	47/7.4 (123.9s)	15/8.8 (139.9s)	22/10.0 (48.3s)
65^3	14/10.9 (1367.6s)	58/5.7 (793.8s)	16/9.1 (1284.2s)	23/11.7 (318.8s)

Reported numbers: # of FC iterations/ Average number of inner iterations (time in s).

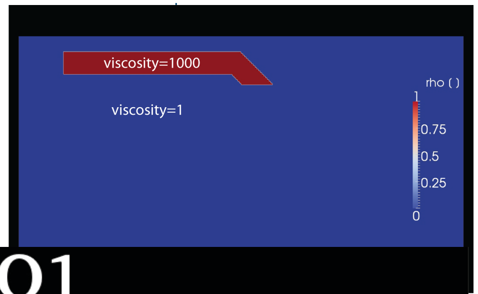
```
../LaMEM -restart 0 -mumax 1e3 -SolverType 2 -A11_ksp_type fgmres -A11_ksp_rtol 1e-2  -A11_pc_type fieldsplit -A11_fieldsplit_pc_type ml -A11_fieldsplit_ksp_rtol 1e-2 -A11_pc_fieldsplit_type ADDITIVE -levels 1  -A11_ksp_converged_reason -A11_pc_fieldsplit_block_size 3
-A11_fieldsplit_ksp_type preonly  -fc_ksp_atol 1.0e-13 -fc_ksp_rtol 1.0e-5 -use_stokes_relative_norm  -use_stokes_norm_L2 -vpt_element FDSTAG -nnode_x 65 -nnode_y 65 -nnode_z 65
```

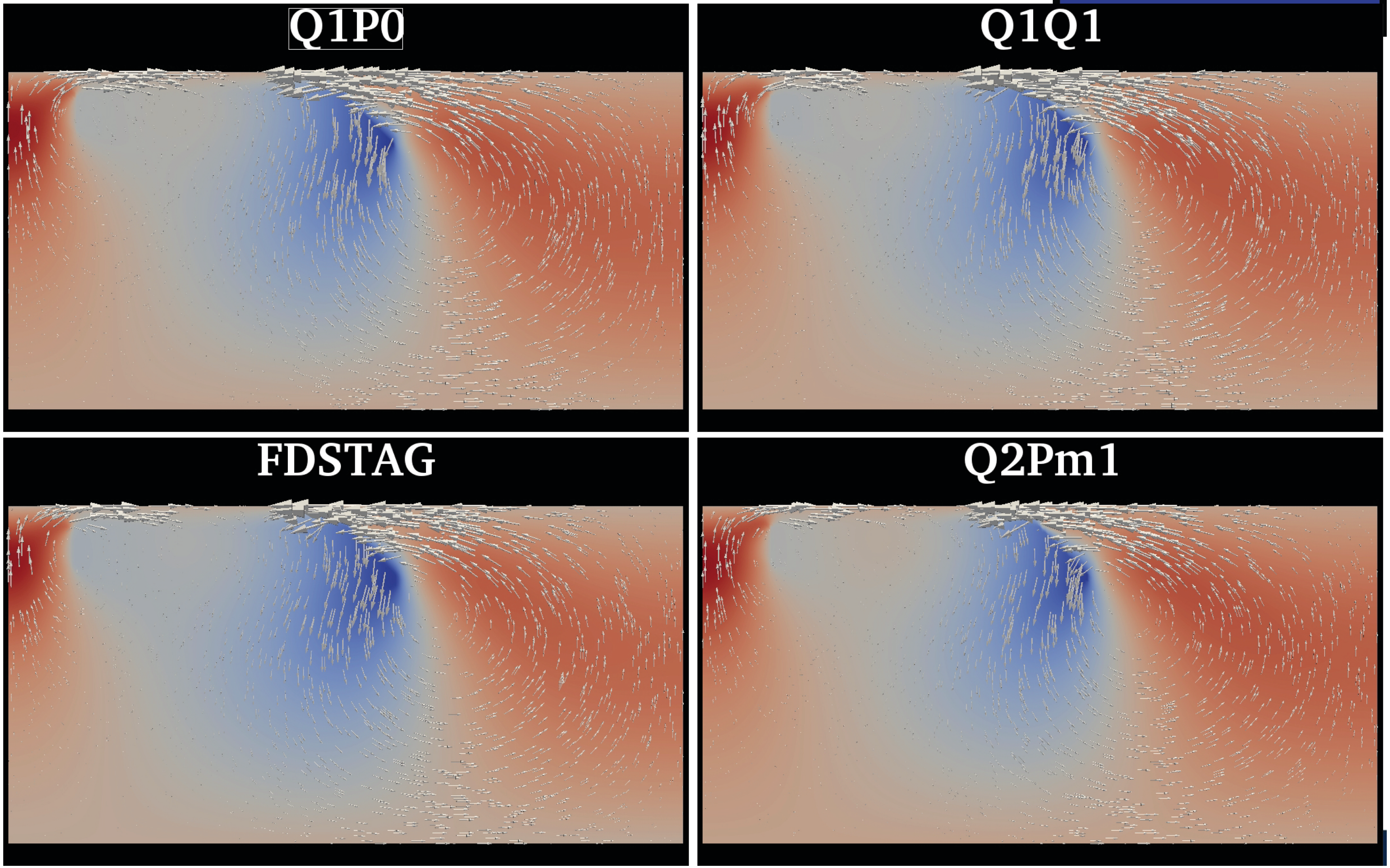
Non-stable elements are bad for iterative solvers

FDSTAG behaves like a **stable** element.

ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

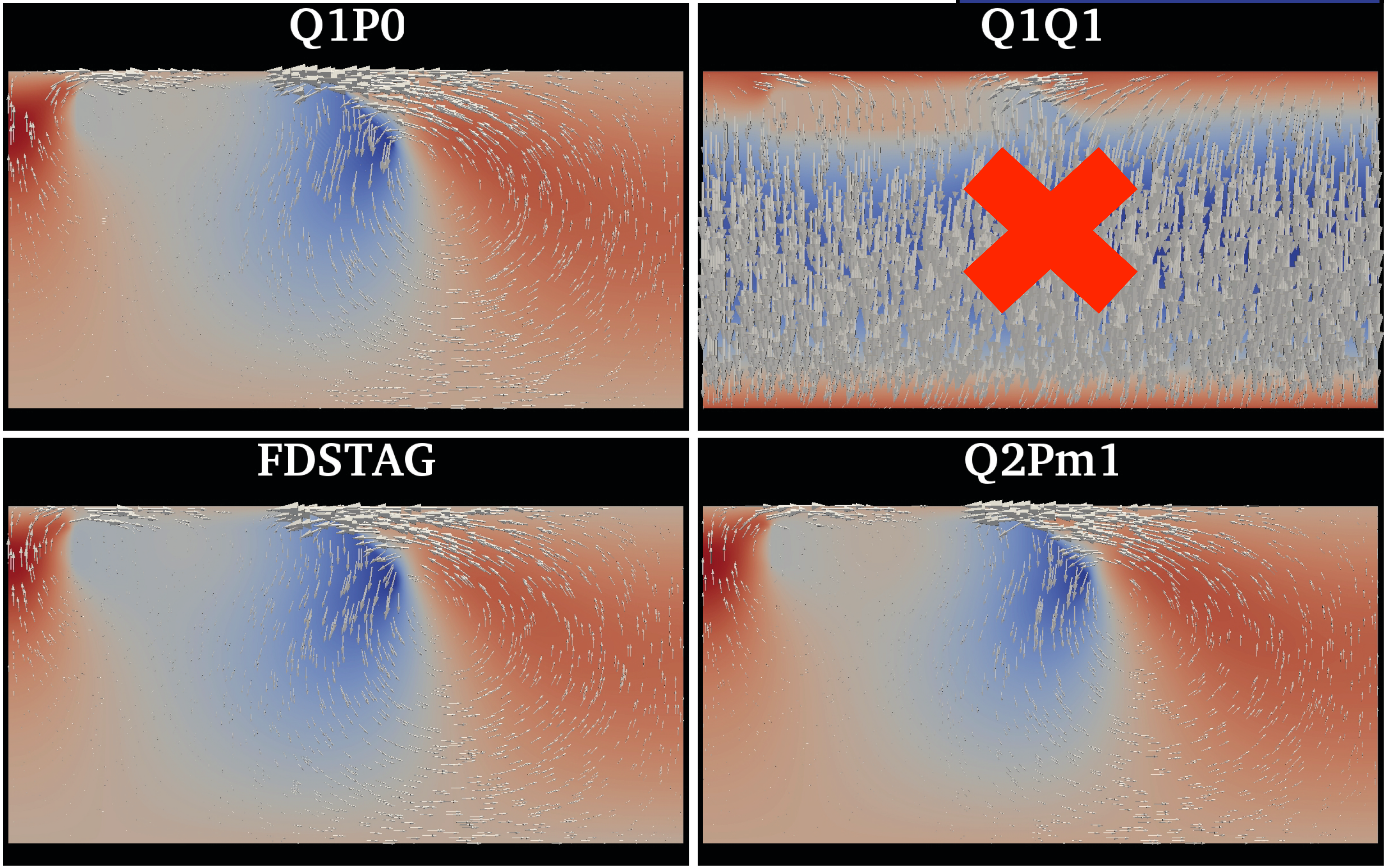
FEM vs. FDSTAG – subduction setup





Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

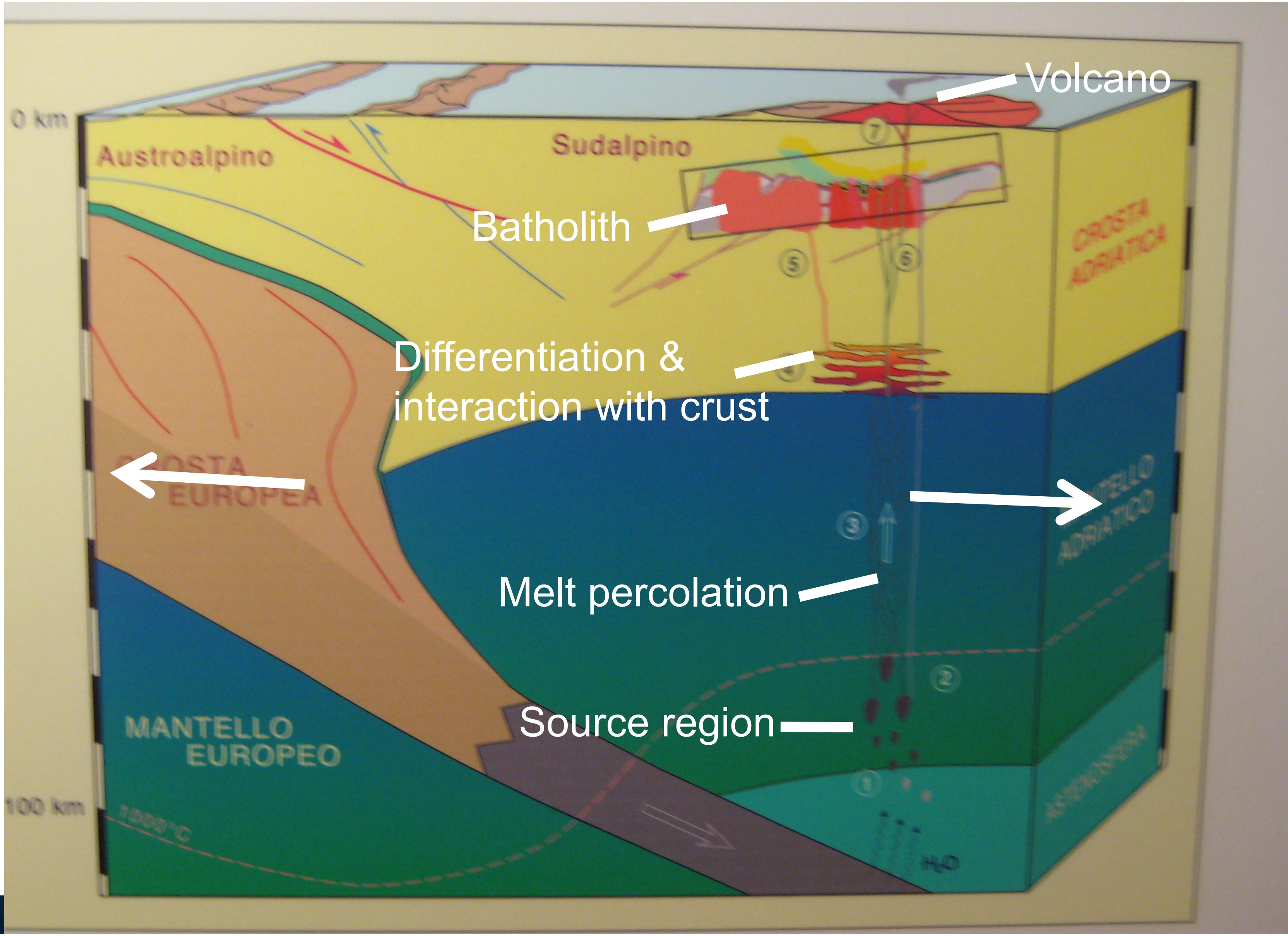
FEM vs. FDSTAG – subduction setup



Summary part 1

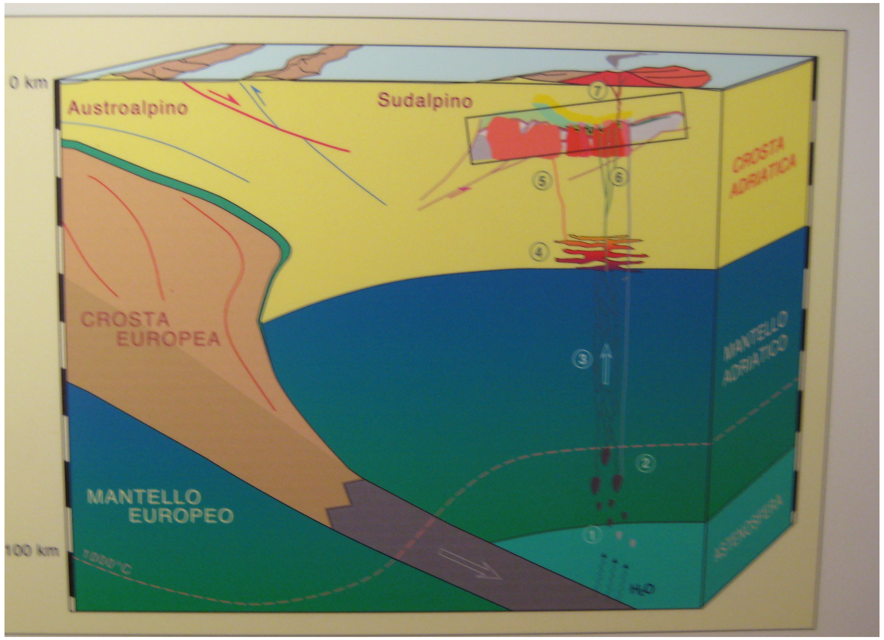
- Modelling high-resolution 3D lithospheric deformation is challenging.
- Need to use iterative solvers.
- Better use a stable element.
- The staggered grid finite difference method behaves like an ideal (small) stable finite element
  - Small
  - Cheap
  - Equally accurate as other linear elements

Geological cartoon of magmatic systems



Challenges:

- Two phase flow formulation required.
- Dikes, melt channels or diapirs?
  - Requires elasto-plastic formulation.
- Magmatic evolution and emplacement of batholiths
- Take (2D) lithospheric deformation into account .



Modeling of two-phase flow – Tobias Keller

- Overcoming limitations
  - Possible to treat regions of magma accumulation as *Stokes flow at lower cutoff viscosity*?
  - Lithosphere deforms as visco-elasto-plastic medium,
    -> use full visco-elasto-plastic compaction rheology

- Overcoming challenges
  - Use implementation style of standard Stokes codes
  - Use primitive variables without flow decomposition

  - > Code in progress: FEM2PHAST
    (**F**inite **E**lement **M**odel of **2-PH**ase and **ST**okes flow)

Two-phase flow equations

- ***Bulk momentum conservation***

$$\nabla \cdot \left[(1-\phi) \cdot \boldsymbol{\sigma}_s^d \right] - \nabla P_e - \nabla P_f - \rho_b g \hat{\mathbf{z}} = 0 \qquad \mathbf{v}_s$$

- ***Solid mass conservation***

$$\nabla \cdot \mathbf{v}_s + \frac{1}{\eta_{s,vep}^v} P_e - \frac{\chi^v}{\eta_{s,vep}^v} P_e^{old} - \dot{\lambda} \sin\psi = 0 \qquad P_e$$

- ***Bulk mass conservation***

$$\nabla \cdot \mathbf{v}_s - \nabla \cdot \left[\frac{k_\phi}{\eta_f} \cdot \nabla P_f + \rho_f g \hat{\mathbf{z}} \right] - \dot{M} \cdot \frac{\Delta\rho}{\rho_s \rho_f} = 0 \qquad P_f$$

- ***Porosity conservation***

$$\frac{D_s \phi}{Dt} + (1-\phi) \nabla \cdot \mathbf{v}_s - \frac{\dot{M}}{\rho_s} = 0 \qquad \phi$$

- ***Energy Conservation***

$$\phi \rho_f C_f \frac{D_f T}{Dt} + (1-\phi) \rho_s C_s \frac{D_s T}{Dt} = -\nabla \cdot \left[k_b \nabla T \right] + H_r + L \cdot \dot{M} + (1-\phi) \cdot \left(\boldsymbol{\sigma}_s^d \dot{\boldsymbol{\varepsilon}}_{s,vp}^d + \eta_{s,vp}^v \left[\nabla \cdot \mathbf{v}_s \right]_{vp}^2 \right) + \frac{\eta_f}{k_\phi} \mathbf{q}_D^2 \qquad T$$

Compaction rheology

- Effective mean stress / volumetric strain rate

$$\sigma_s^* = P_e = P_b - P_f \qquad \dot{\varepsilon}_s^v = -\frac{1}{3}(\nabla \cdot \mathbf{v}_s)$$

- Maxwell visco-elasto-plastic rheology (sum strain rates)

$$\dot{\varepsilon}_{s,tot}^v = \dot{\varepsilon}_{s,vis}^v + \dot{\varepsilon}_{s,ela}^v + \dot{\varepsilon}_{s,pla}^v = \frac{1}{3\eta_s^v}\sigma_s^* + \frac{1}{3K_\phi}\frac{D_s\sigma_s^*}{Dt} + \frac{\dot{\lambda}}{3}tr\left(\frac{\partial Q^t}{\partial \sigma_s^*}\right)$$

- Visco-elasto-plasticity (effective viscosity approach)

$$\eta_{s,ve}^v = \frac{1}{\frac{1}{\eta_s^v} + \frac{1}{K_\phi \Delta t}}$$

$$\chi^v = \frac{1}{1 + \frac{K_\phi \Delta t}{\eta_s^v}}$$

$$P_e = 3\eta_{s,ve}^v \dot{\varepsilon}_{s,ve}^v + \chi^v \tilde{P}_e^{old}$$

$$\longrightarrow$$

$$\sigma_y^t = \sigma_s^t + \sigma_s^*$$

$$Q^t = \sigma_{II}^d - \sigma_s^*$$

Griffith criterion

$$\longrightarrow$$

$$\eta_{s,vep}^v = \frac{\sigma_y^t - \chi^v \tilde{P}_e^{old}}{3\left|\dot{\varepsilon}_{s,vep}^v\right|}$$

$$P_e = 3\eta_{s,vep}^v \dot{\varepsilon}_{s,vep}^v + \chi^v \tilde{P}_e^{old}$$

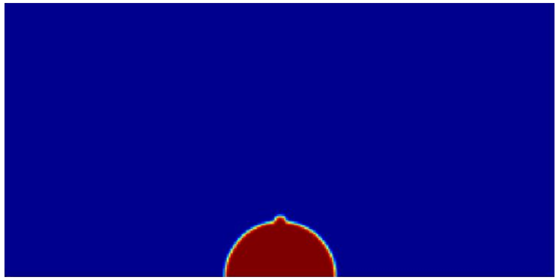
Final constitutive law

First results in progress

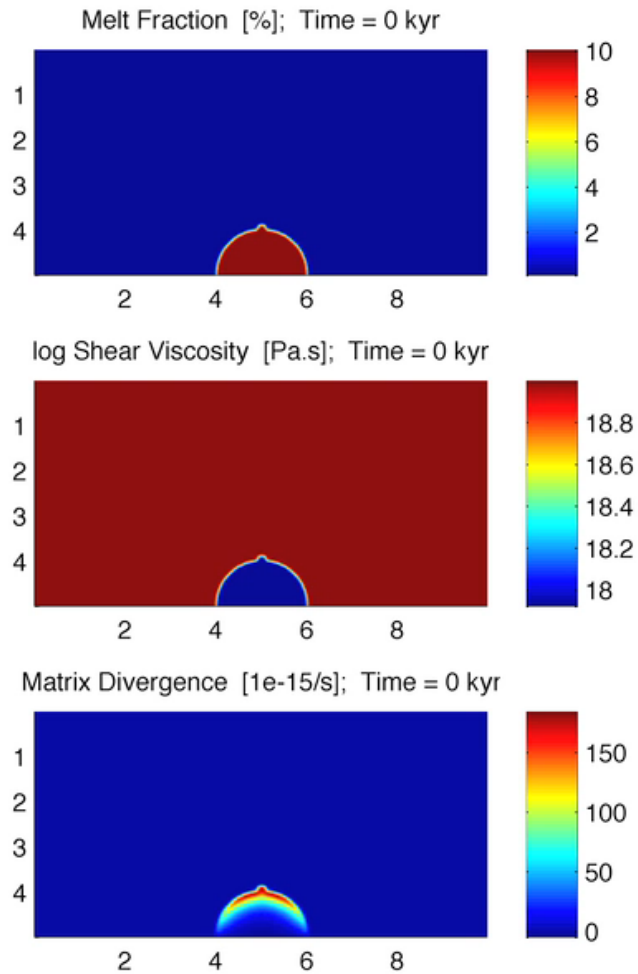
- Simple setup to test rheological regimes
  - Kinematic boundaries: extension *2.e-15 s-1*
  - 10% melt region in 0.1% background material
  - Inhomogeneity at tip of melt region
  - Hydrostatic fluid pressure lower boundary
  - Porosity-weakening of viscosities

$$\eta_s^d = \eta_s^{ref} \cdot \exp(-25 \cdot \phi) \qquad \eta_s^v = \frac{\eta_s^{ref}}{\phi}$$

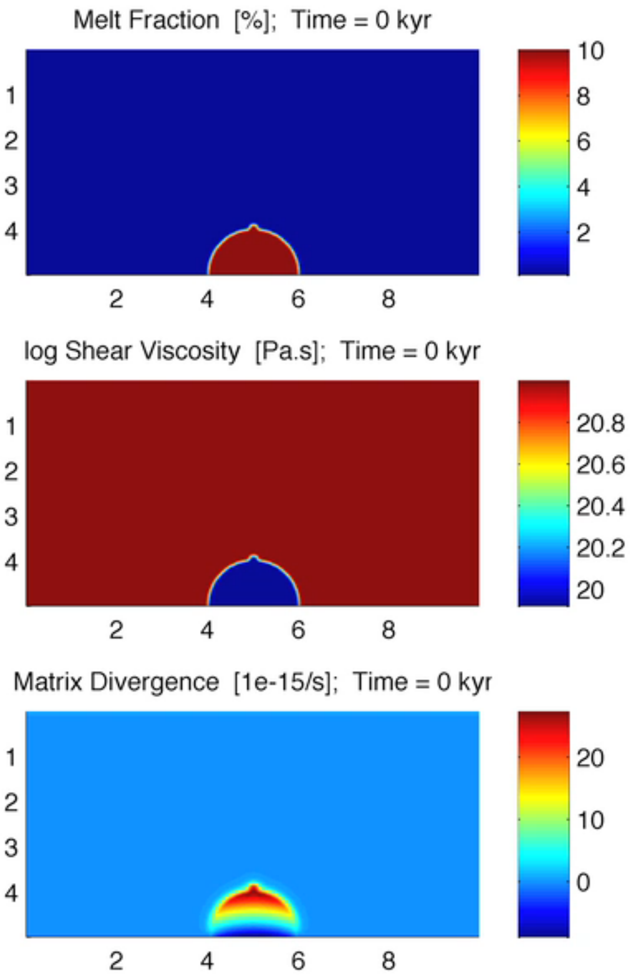
  - > Gradually increase background viscosity from *1.e19* to *1.e24 Pa.s*



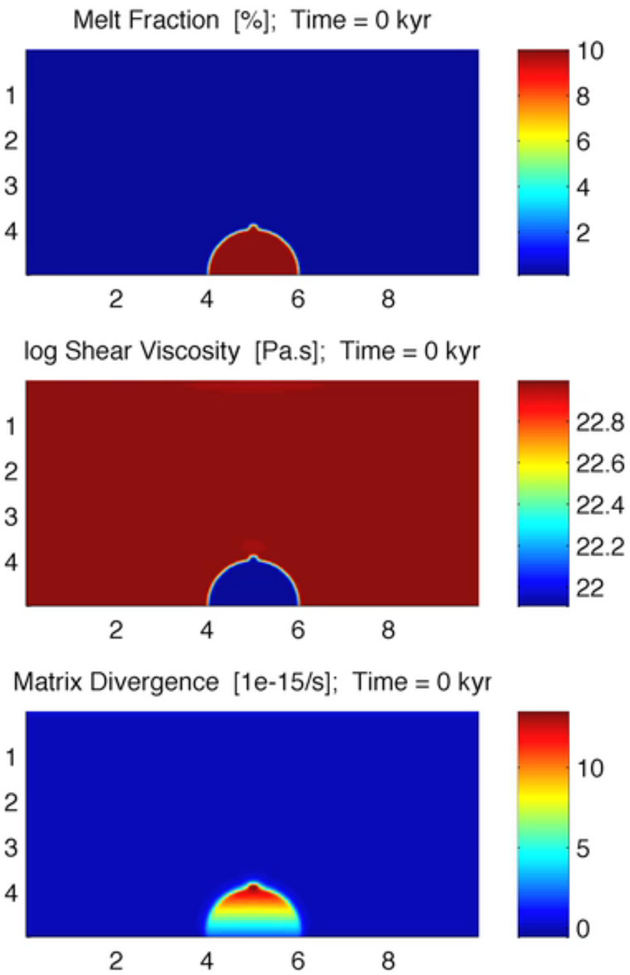
eta_ref = 1.e19 Pa.s



eta_ref = 1.e21 Pa.s



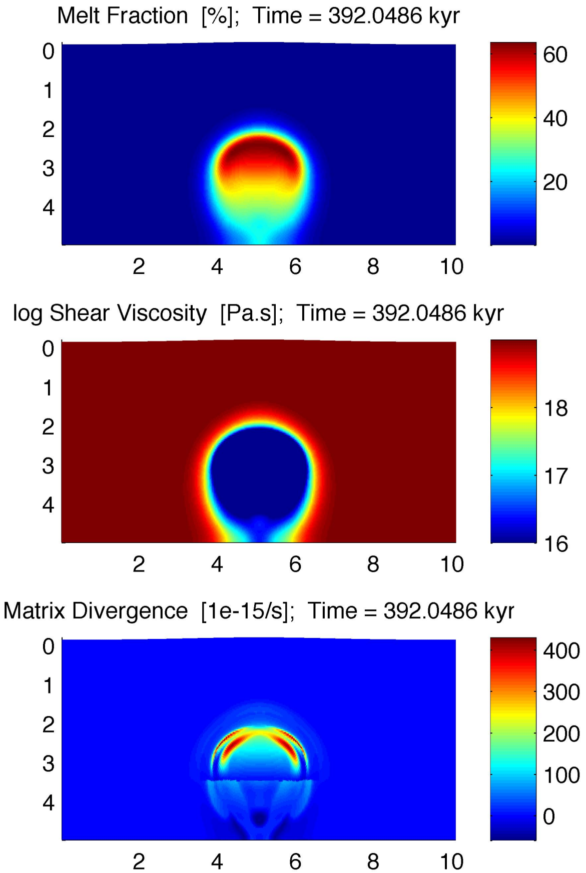
eta_ref = 1.e23 Pa.s



Three (preliminary) regimes

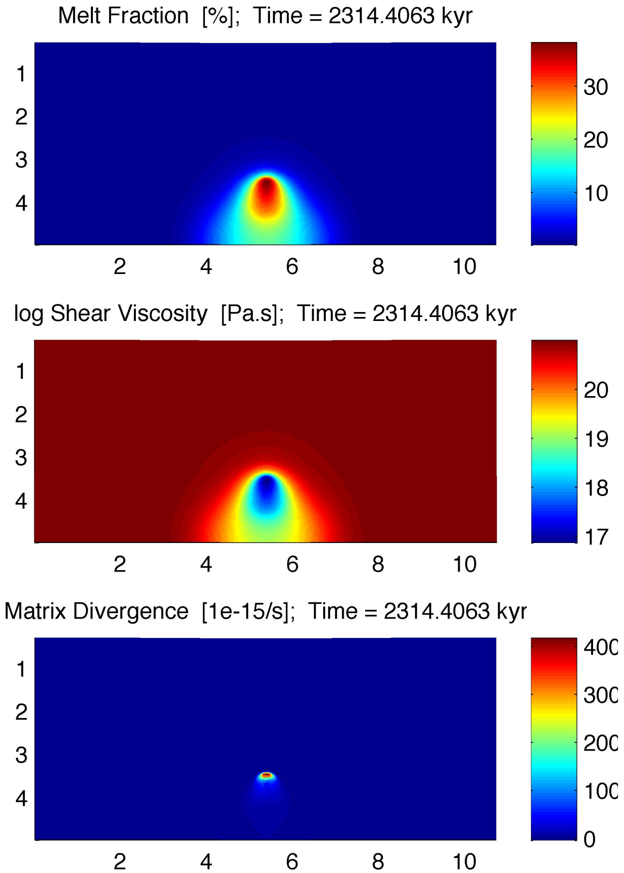
diapirs

porosity weakening



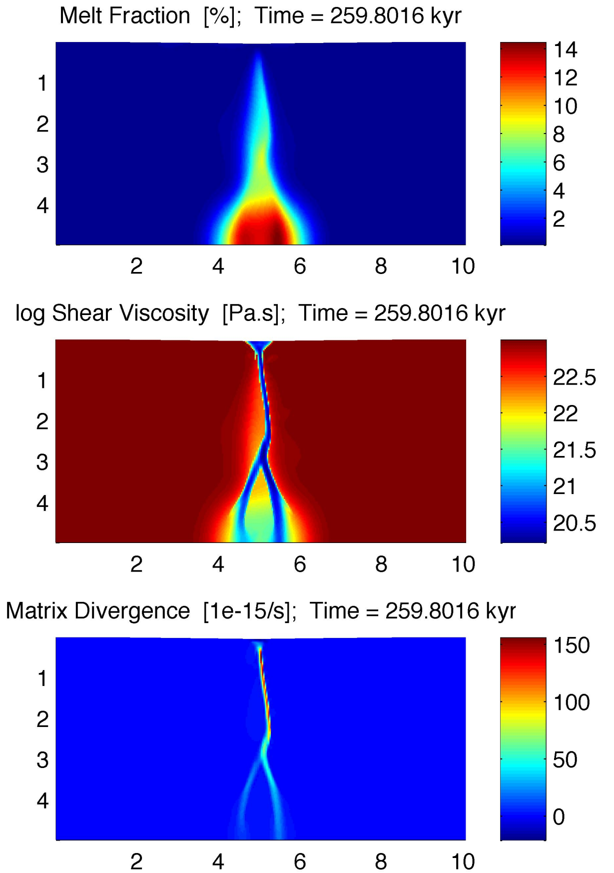
channels

decompaction weakening



dikes

mode-1 plasticity



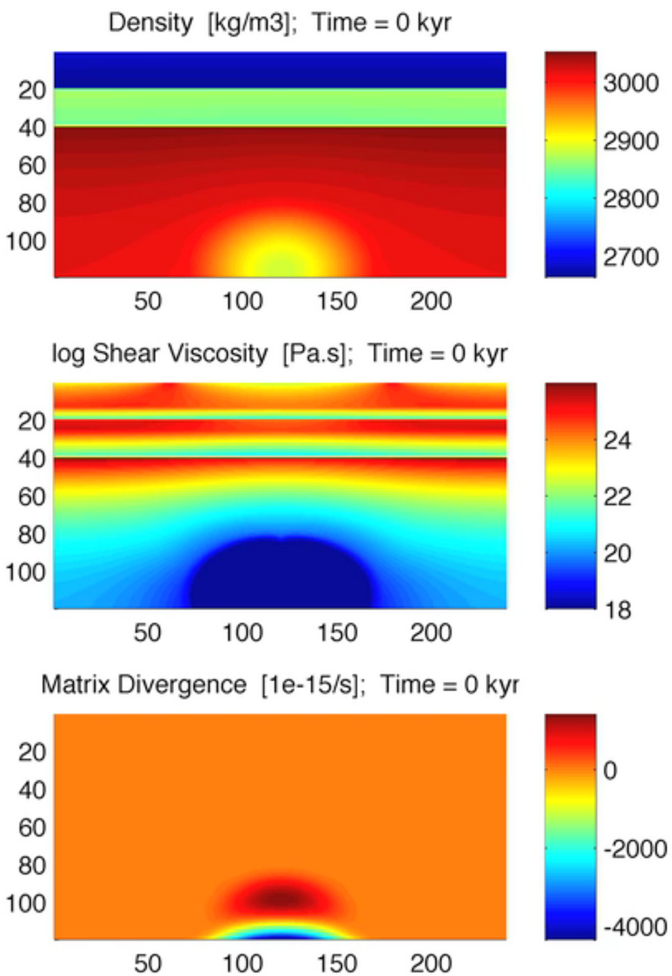
Magmatic evolution model

- Melting model
  - Katz (2003) plus some additions
  - Melt fraction depends on P, T, H_2O and composition

- Magmatic evolution of melt and solid
  - Melt evolution index: 0% (primordial) to 100% (evolved)
  - Solid evolution index: 0% (ultramafic) to 100% (felsic)

  - > compute melting rate / solve energy equation during each iteration of non-linear solver

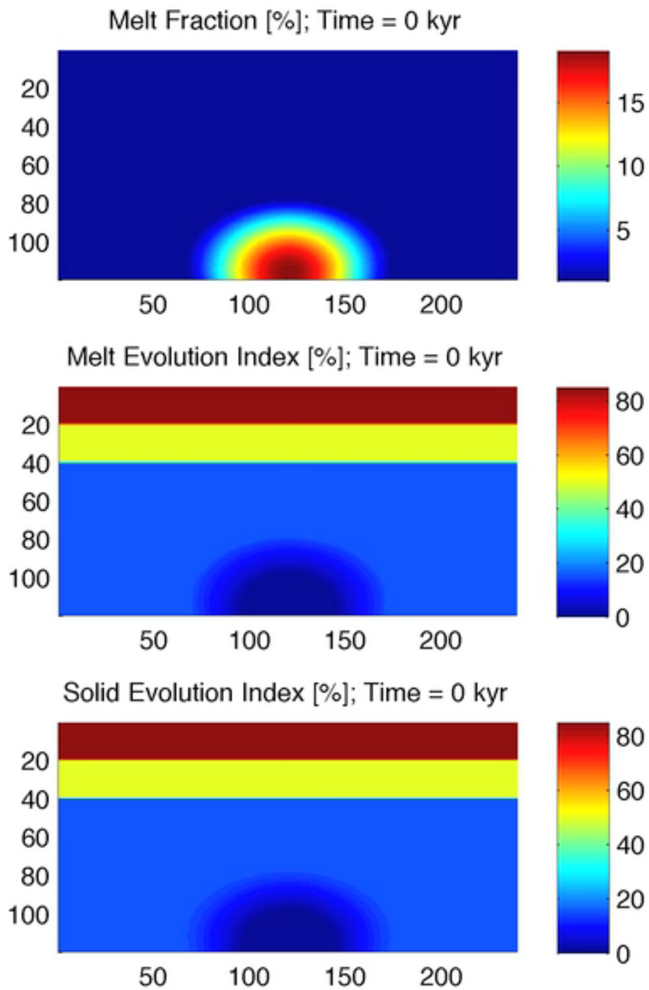
First results…



continental crust and lithosphere

extensional boundaries

initiate melting by 250 K excess temperature



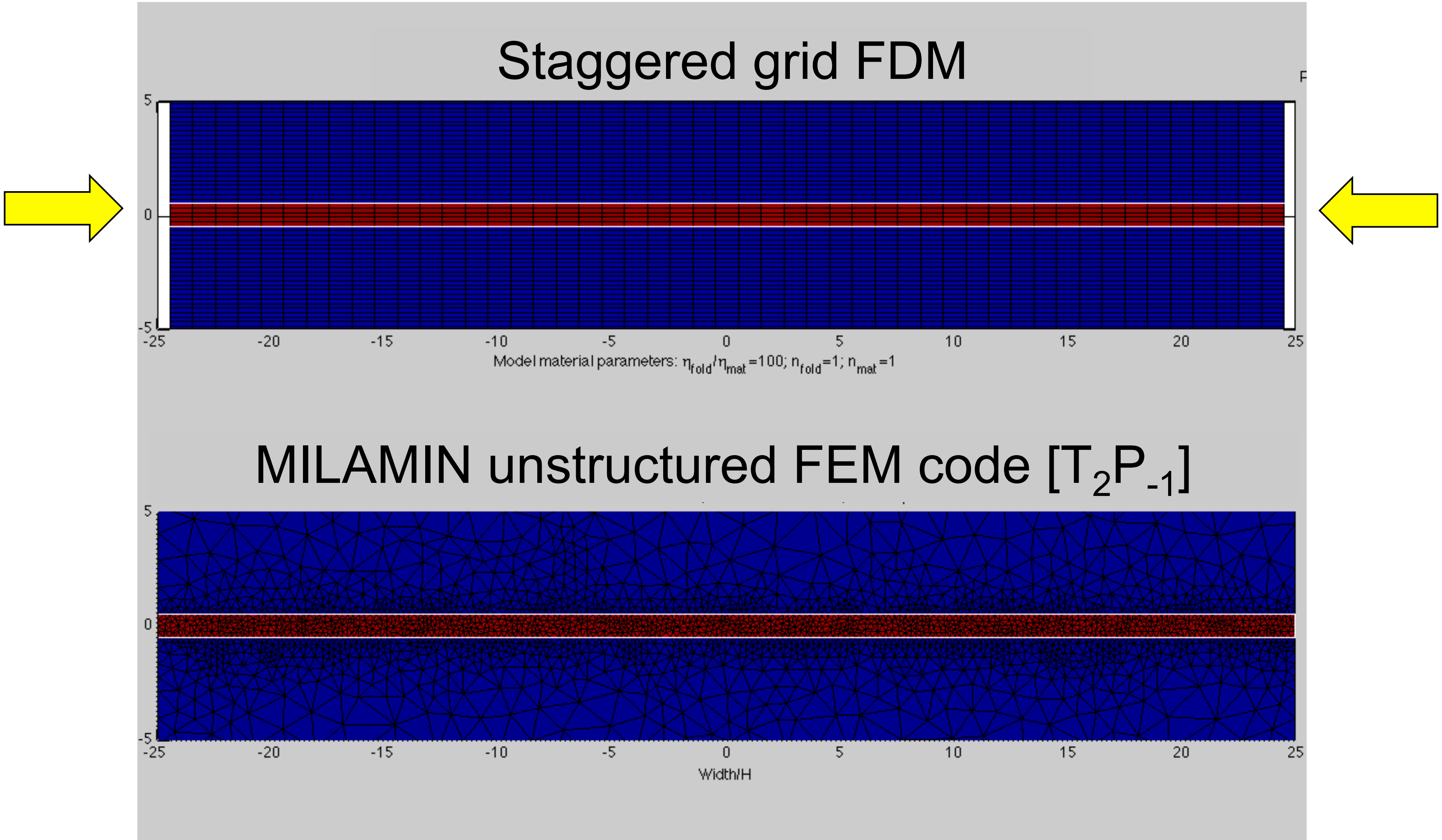
Conclusions

- Code development was/is and remains important in geodynamics.
- 3D is challenging particularly for lithospheric dynamics.
- The staggered finite difference method is (surprisingly) competitive.
- Magmatic systems require including two-phase flow formulations within lithospheric codes.
- Preliminary results produce diapirs, dikes and channelized flow.

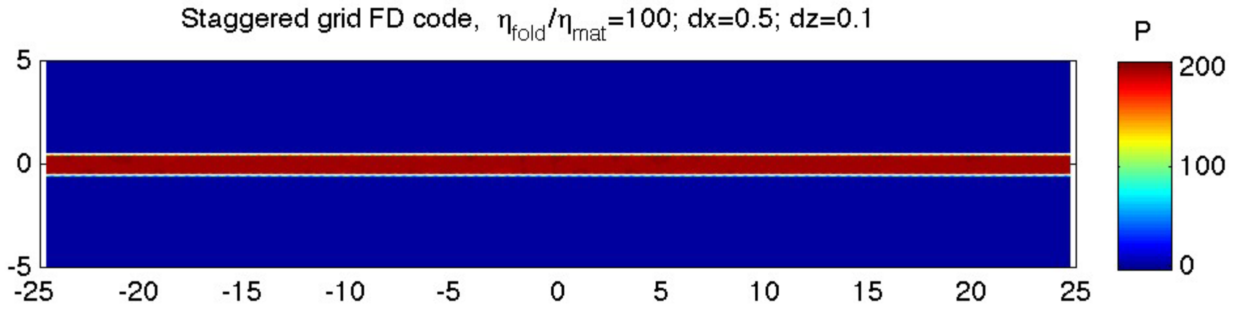
Additional slides

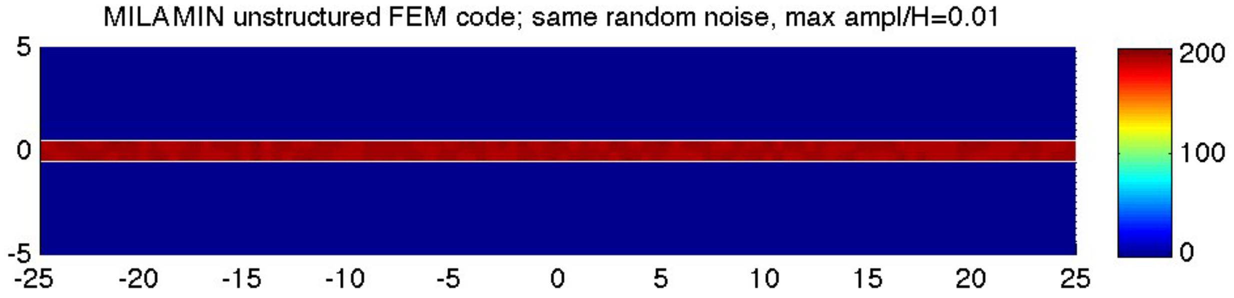
FEM vs. Finite Differences – Model setup

- Folding setup, viscosity contrast 100.



FEM vs. Finite Differences





FDSTAG is doing pretty well.