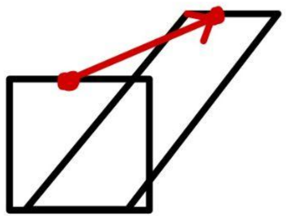
Linear Elasticity

Michael Christoffersen
18 April 2024
Finite Element Seminar

Deformation



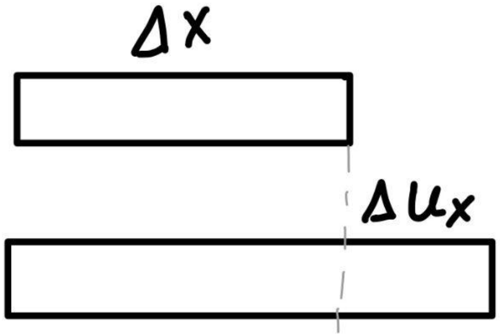
Vector quantity

(u_x, u_y, u_z)

Displacement

Strain

In one dimension



$$\varepsilon = \frac{\Delta u_x}{\Delta x} \rightarrow \frac{\partial u_x}{\partial x}$$

In two or three dimensions

$$\varepsilon = \frac{1}{2}\left(\nabla u + (\nabla u)^T\right)$$

Tensor quantity

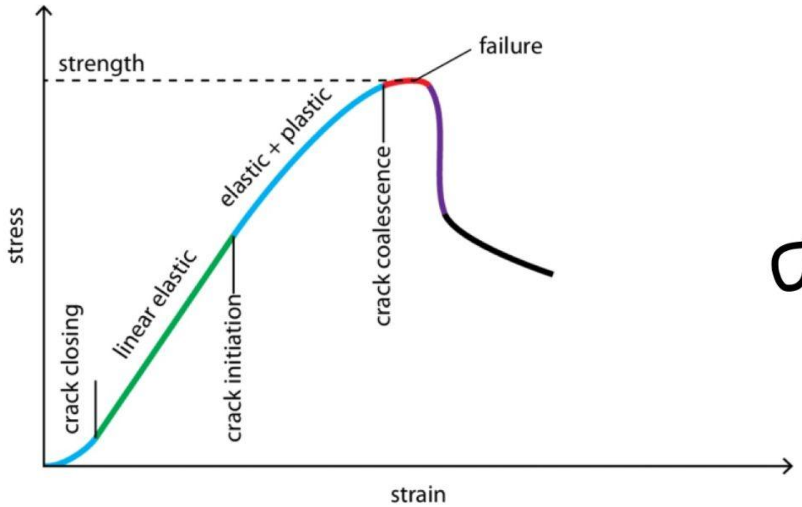
$$\varepsilon = \begin{bmatrix} \varepsilon_{xx} & \varepsilon_{xy} & \varepsilon_{xz} \\ \varepsilon_{yx} & \varepsilon_{yy} & \varepsilon_{yz} \\ \varepsilon_{zx} & \varepsilon_{zy} & \varepsilon_{zz} \end{bmatrix}$$

Stress

Force per unit area

Related to strain by a constitutive relationship

Today - linear elasticity



Hooke's law

$$\sigma = \lambda \, tr(\varepsilon) I + 2\mu\varepsilon$$

$\lambda, \mu \rightarrow$ Lamé parameters

$tr(\varepsilon) = \varepsilon_{xx} + \varepsilon_{yy} + \varepsilon_{zz}$

I = Identity matrix

Tensor quantity

$$\sigma = \begin{bmatrix} \sigma_{xx} & \sigma_{xy} & \sigma_{xz} \\ \sigma_{yx} & \sigma_{yy} & \sigma_{yz} \\ \sigma_{zx} & \sigma_{zy} & \sigma_{zz} \end{bmatrix}$$

Governing equation

From Cauchy momentum equation:

$$-\nabla \cdot \sigma = f$$

f $\Rightarrow$ body forces (e.g., gravity)

To express as a function of u, first plug in Hooke's law:

$$\sigma = \lambda \mathrm{tr}(\epsilon) I + 2\mu\epsilon$$

and strain definition:

$$\epsilon = \frac{1}{2}(\nabla u + (\nabla u)^\top)$$

Weak Form

Multiply by test function and integrate:

$$-\int_{\Omega} (\nabla \cdot \sigma) \cdot v \, \mathrm{dx} = \int_{\Omega} f \cdot v \, \mathrm{dx}$$

Integration by parts to get rid of $\nabla \cdot \sigma$:

$$-\left(\int_{\partial\Omega} (\sigma \cdot \hat{n}) \cdot v \, \mathrm{ds} - \int_{\Omega} \sigma : \nabla v \, \mathrm{dx} \right) = \int_{\Omega} f \cdot v \, \mathrm{dx}$$

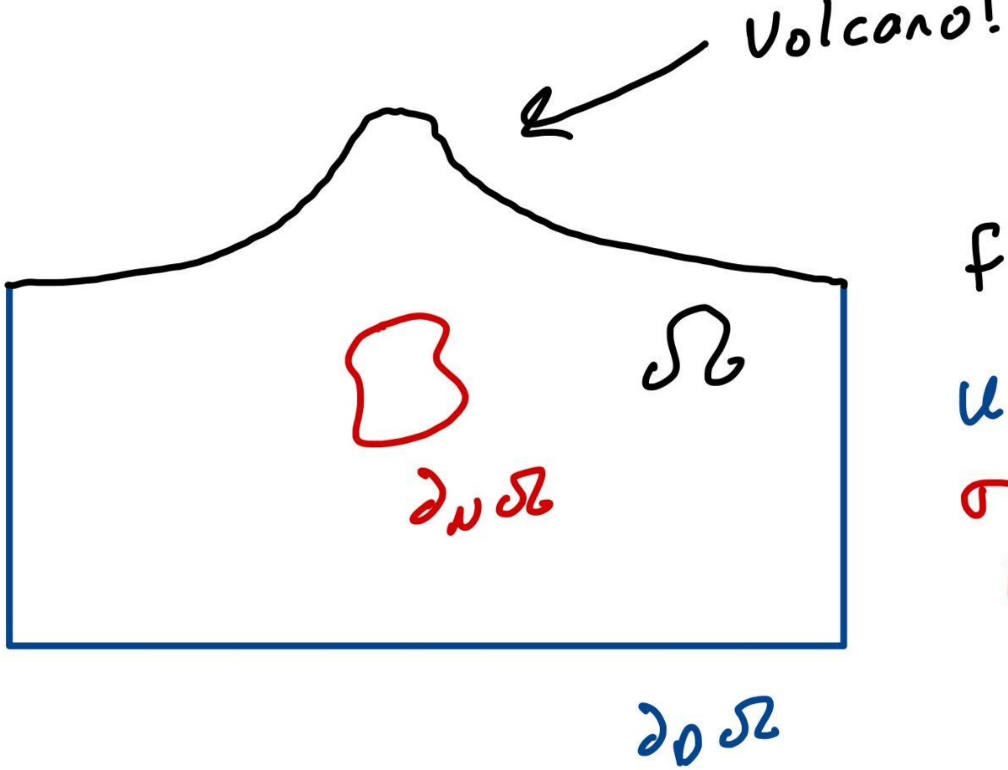
$\hat{n} \Rightarrow$ outward unit normal vector

$: \Rightarrow$ inner product of tensors $(A : B = \Sigma_i \Sigma_j A_{ij} B_{ij})$

$$-\left(\int_{\partial\Omega}(\sigma\cdot\hat{n})\cdot v\,\mathrm{ds}-\int_{\Omega}\sigma:\nabla v\,\mathrm{dx}\right)=\int_{\Omega}f\cdot v\,\mathrm{dx}$$

↓

$$\boxed{\int_{\Omega}\sigma:\nabla v\,\mathrm{dx}=\int_{\Omega}f\cdot v\,\mathrm{dx}+\int_{\partial\Omega}(\sigma\cdot\hat{n})\cdot v\,\mathrm{ds}}$$



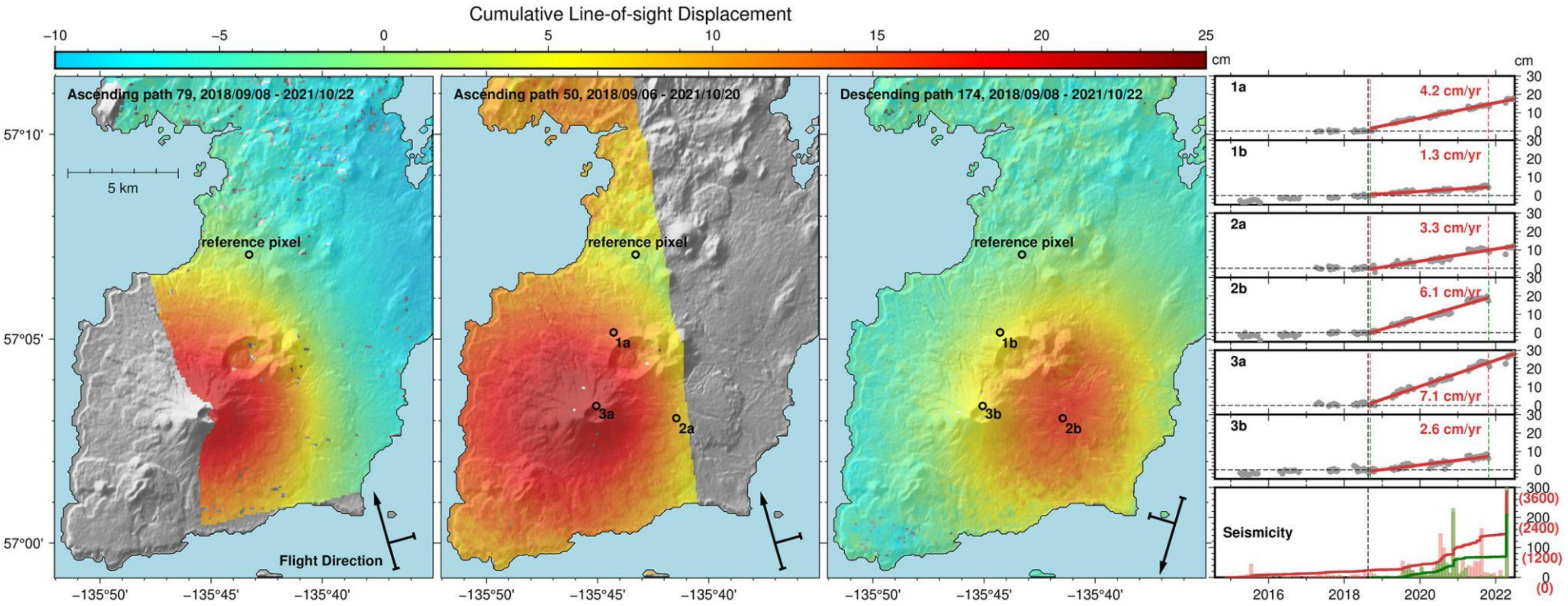
$f = 0$ on $\partial \Omega$

$u = (0, 0, 0)$ on $\partial_D \Omega$

$\sigma \cdot \hat{n} = p \hat{n}$ on $\partial_N \Omega$

where p is scalar

Motivation

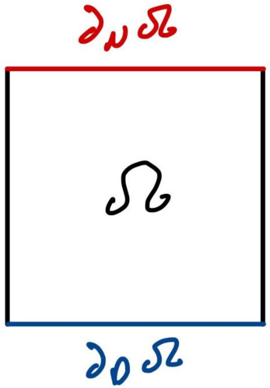


Lots of other applications…

Firedrake Implementation

marshmallow.py

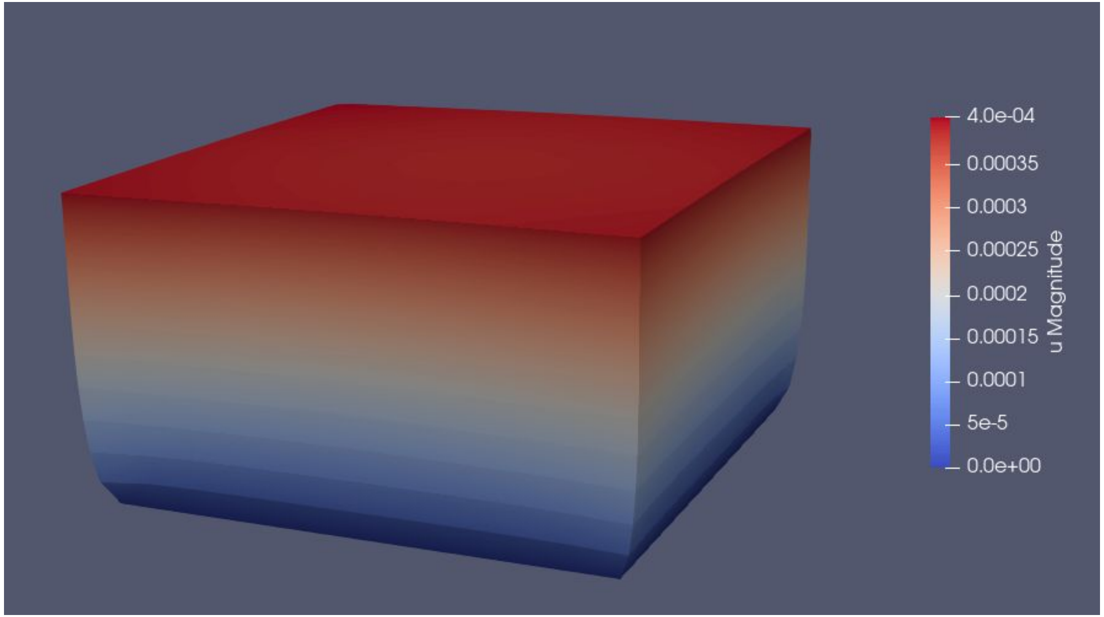
Solve with LU decomposition



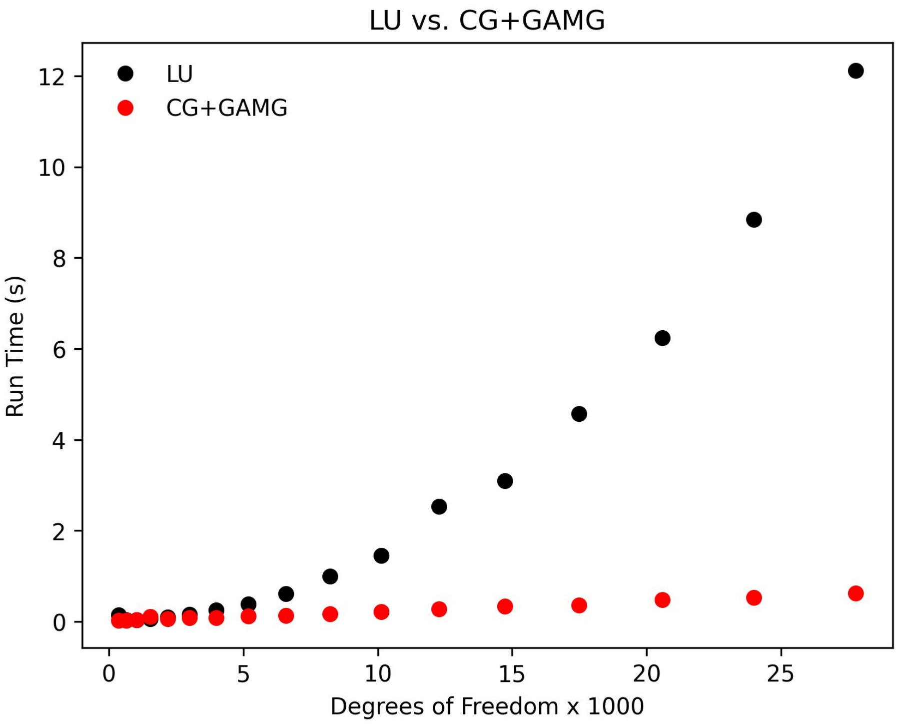
$f = 0$ on Ω

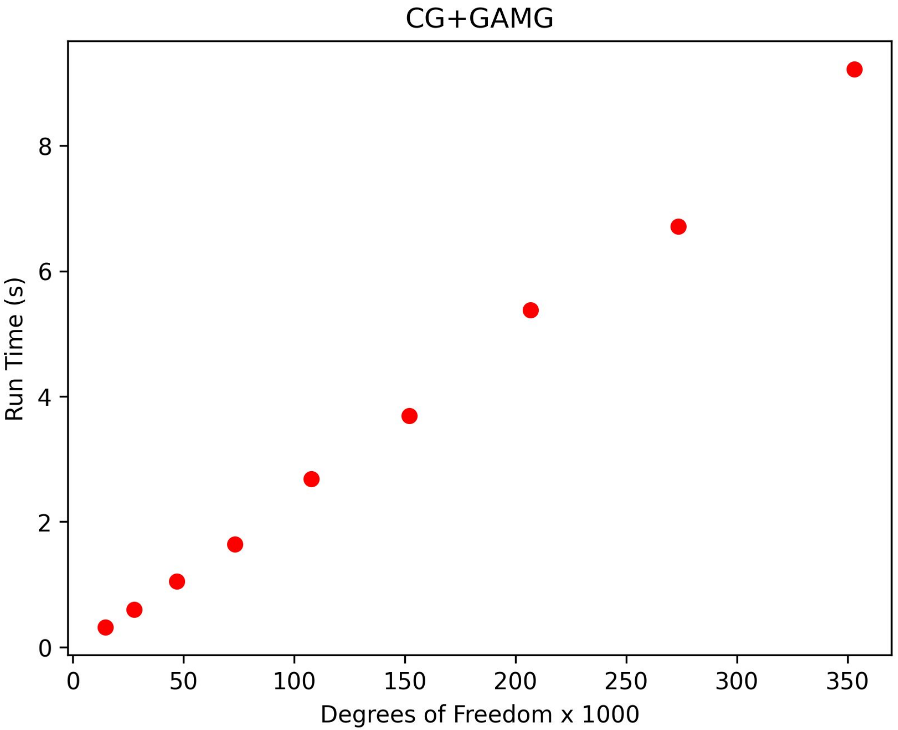
$u = (0,0,0)$ on $\partial_D \Omega$

$\sigma \cdot \hat{n} = p\hat{n}$ on $\partial_N \Omega$
where p is scalar

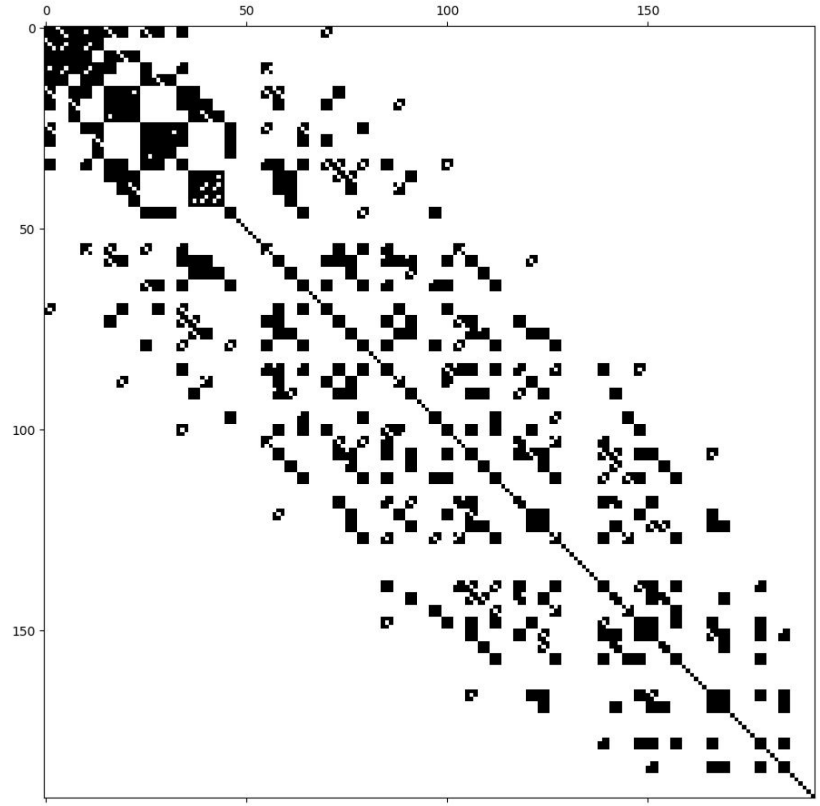


Marshmallow performance (single core)



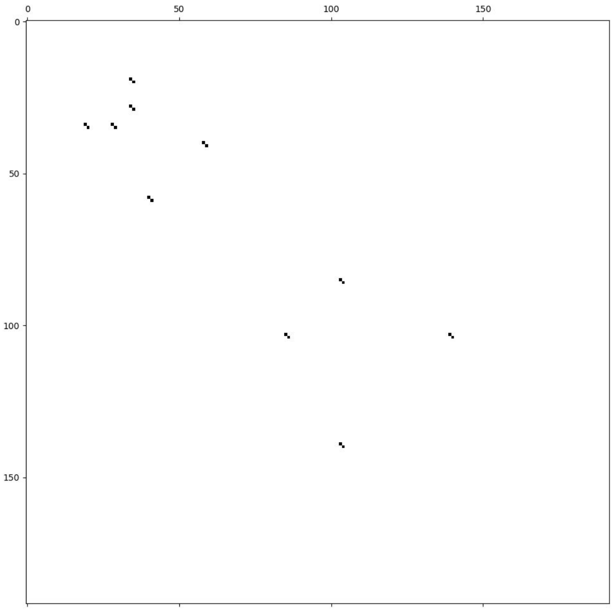


Spy of stiffness matrix for 3x3x3 marshmallow



Almost symmetric

10 non symmetric index pairs



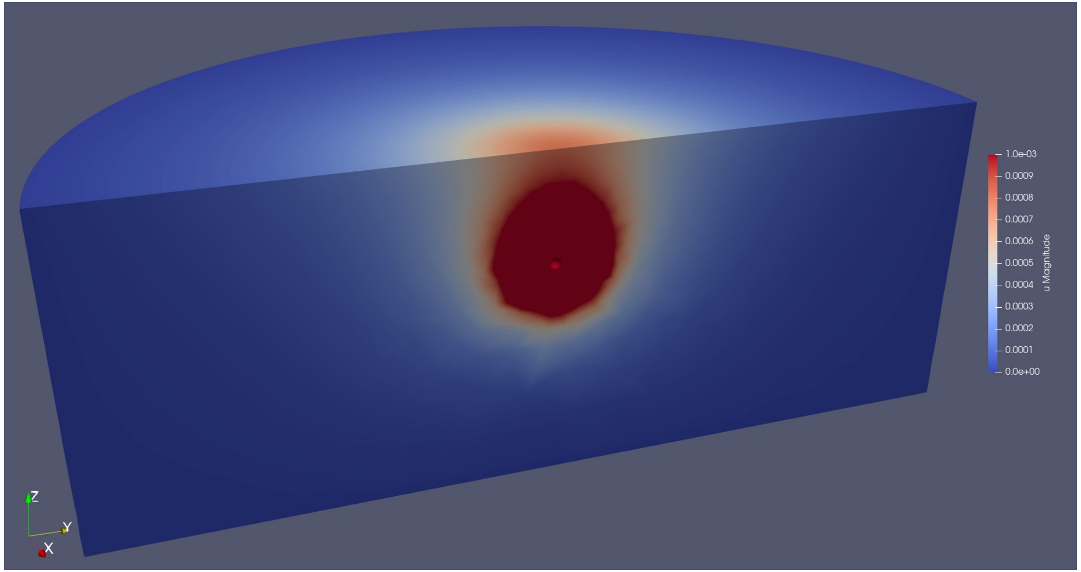
More interesting - Mogi comparison

mesh_mogi.py

Uses Gmsh OCC kernel

mogi.py

LU starts to get slow with this mesh… need CG+GAMG

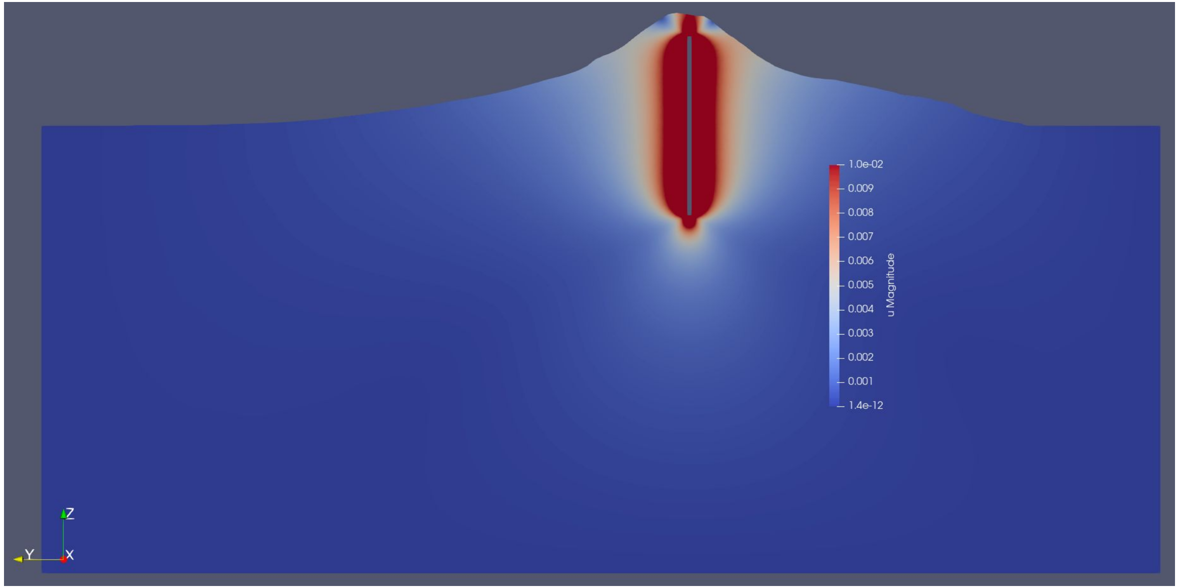


Even more interesting - use topography

mesh_conduit.py

Brute force approach to meshing an elevation model

conduit.py



References

Stress-strain figure from here:

https://hss-opus.ub.ruhr-uni-bochum.de/opus4/frontdoor/index/index/docId/4383

Volcano deformation figure from here:

https://doi.org/10.1029/2022GL099464

These slides pull a lot from the FENICS linear elasticity tutorial:

https://fenicsproject.org/pub/tutorial/html/._ftut1008.html#ftut:elast