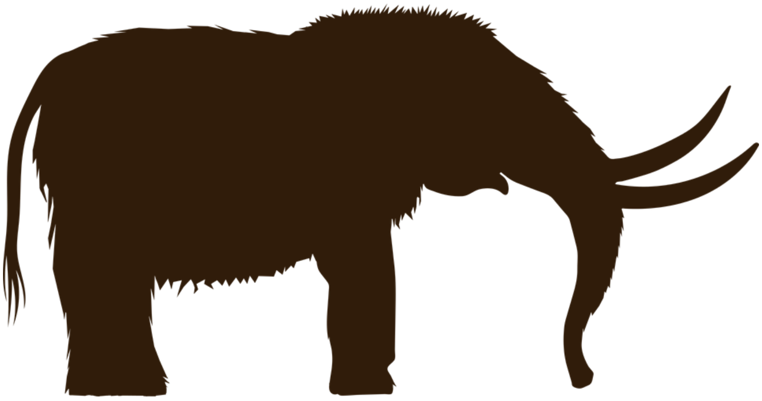


MAMMOTH

C. Thieulot
Utrecht Universiteit

1 Install & run procedure

To compile the code, simply open a terminal and make sure you have the gfortran compiler installed. Then simply do

```
make
```

Unless there is an error, this should generate the executable file *mammoth*. To run the executable, simply do

```
./mammoth
```

If you wish to redirect the standard output to a file (e.g. *opla*), then:

```
./mammoth > opla
```

If you wish to clean all the data files after a run, use the *cleandata* script as follows:

```
./cleandata
```

To clean the code off data and other compilations files:

```
make clean
```

When the code runs it produces ascii data (**.dat* files) and paraview files (**.vtu* files) in the *OUT* folder. The former can be visualised with gnuplot, while the latter should be opened with paraview (or VisIt or MayaVi).

2 Physical equations

The code solves the heat transport equation in the absence of advective processes:

$$\rho c_P \frac{\partial T}{\partial t} = \boldsymbol{\nabla} \cdot k \boldsymbol{\nabla} T + H$$

where ρ is the mass density (kg/m^3), c_P is the heat capacity (), k is the heat conductivity, and H is a heat production term. $T(x,y,z,t)$ is the temperature field. The domain is designated by Ω and its boundary by Γ. This PDE must be supplied with boundary conditions: at a given point of the boundary Γ either a temperature can be prescribed or a heat flux.

3 Discretisation

The heat transport equation is solved by means of the Finite Element Method, following the methodology presented in [**?**]. The FE equation corresponding to this PDE is

$$\boldsymbol{M} \cdot \frac{\partial \boldsymbol{T}}{\partial t} + \boldsymbol{K}_d \cdot \boldsymbol{T} = \boldsymbol{F}$$

where

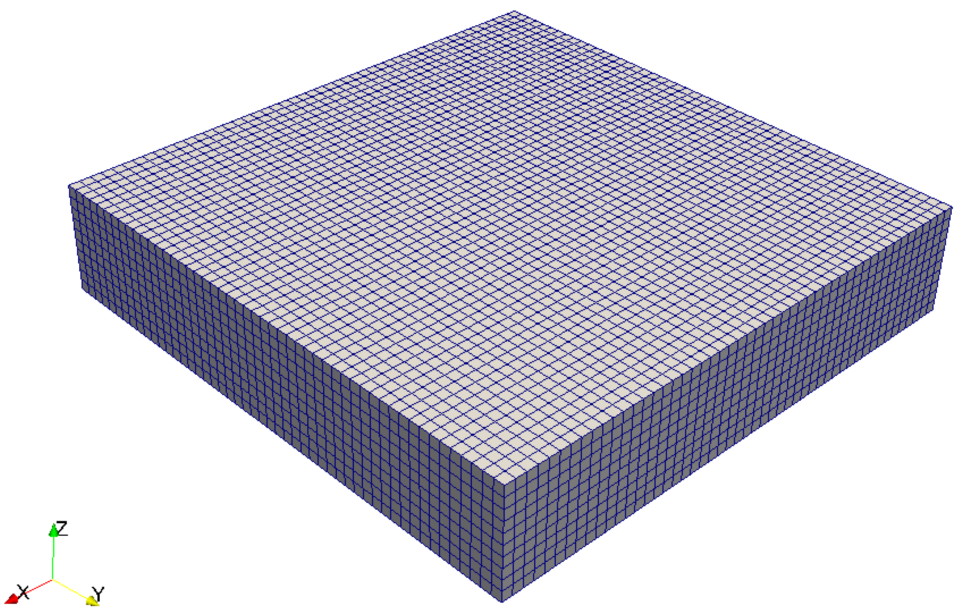
$$\boldsymbol{M} = \int_\Omega \boldsymbol{N}^T \rho c_P \boldsymbol{N} dV$$

$$\boldsymbol{K}_d = \int_\Omega \boldsymbol{B}^T k \boldsymbol{B} dV$$

$$\boldsymbol{F} = \int_\Omega \boldsymbol{N}^T H dV$$

where $\boldsymbol{T}$ is the vector of nodal temperatures.

The domain Ω of size $L_x \times L_y \times L_z$ is divided into $nel = nelx \times nely \times nelz$ elements as shown hereunder. The computational grid therefore counts $np = (nelx+1)(nely+1)(nelz+1)$ nodes and as many degrees of freedom. A Cartesian coordinate system is used with its origin at the bottom corner of the domain.



4 Setting up an experiment

4.1 Initial temperature field

4.2 Magma chambers

The code can model as many magma chambers as needed. This number is controlled by the `nchamber` parameter. Each chamber is parametrised by:

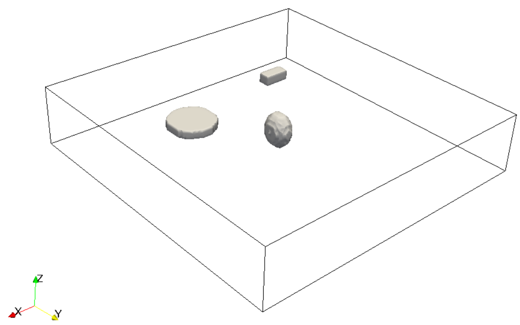
- its `shape`: 1 is a spheroid, 2 is a cuboid and 3 is a cylinder;
- its center coordinates `xc`, `yc`, `zc`;
- its spatial extent in the x, y, z directions: `a`, `b` and `c`;
- its initial temperature `T`.

Example The following figure is obtained by setting three magma chambers as follows:

```
mc(1)%shape=1
mc(1)%xc=0.5*Lx
mc(1)%yc=0.5*Ly
mc(1)%zc=0.5*Lz
mc(1)%a=4d3
mc(1)%b=5d3
mc(1)%c=6d3
mc(1)%T=500

mc(2)%shape=2
mc(2)%xc=0.25*Lx
mc(2)%yc=0.2*Ly
mc(2)%zc=0.4*Lz
mc(2)%a=10d3
mc(2)%b=5d3
mc(2)%c=4d3
mc(2)%T=500

mc(3)%shape=3
mc(3)%xc=0.7*Lx
mc(3)%yc=0.33*Ly
mc(3)%zc=0.6*Lz
mc(3)%a=8d3
mc(3)%b=8d3
mc(3)%c=3d3
mc(3)%T=500
```



4.3 Boundary conditions

A Steady state solutions

A.1 One layer

At steady state: $\partial T/\partial t = 0$ so that one has to solve

$$\Delta T = -\frac{H}{k}$$

Assuming the domain to be infinite in the x and y direction this equation becomes:

$$\frac{d^2T}{dz^2} = -\frac{H}{k}$$

The generic solution to this ODE is given by:

$$T(z) = -\frac{H}{2k}z^2 + Az + B \tag{1}$$

where A and B are integration constants to be determined by means of the specified boundary conditions.

At the surface a constant temperature is maintained, i.e. $T(z = L) = T_s$, so that

$$T(L) = -\frac{H}{2k}L^2 + AL + B = T_s$$

which allows us to express B as a function of the rest and insert back into Eq. (1):

$$T(z) = -\frac{H}{2k}(z^2 - L^2) + A(z - L) + T_s$$

Temperature prescribed at the bottom In this case $T(z = 0) = T_b$ so that

$$T(z) = -\frac{H}{2k}L^2 + AL + T_s = T_b$$

which yields

$$A = \frac{T_b - T_s}{L} + \frac{HL}{2k}$$

Heat flux prescribed at the bottom In this case $kdT/dz(z = 0) = \phi_b$ which yields $kA = \phi_b$ so that

$$T(z) = -\frac{H}{2k}(z^2 - L^2) + \frac{\phi_b}{k}(z - L) + T_s$$

A.2 Two layers

In each layer i the solution is given by:

$$T(z) = -\frac{H_i}{2k_i}z^2 + A_iz + B_i \tag{2}$$

Aside from the two previous boundary conditions we need to enforce that the temperature and the heat flux are continuous fields, .i.e. $T(z = l^-) = T(z = l^+)$ and $\phi(z = l^-) = \phi(z = l^+)$.

B todo