Heterogeneous 3D model of concrete

April 22, 2013

Abstract

Documentation of heterogeneous 3D model of concrete written in MATLAB by Filip Nilenius. The model implementation solves the heat/diffusion equation both for stationary and transient conditions. All MATLAB code required to use the model is attached to this PDF. Just click the pins to open each `*.m`-file.

Contents

1 MATLAB files

1.1 `SVEGenerator.m`

- generates random structured SVE.
- saves topology data for input to `PreProcessor.m`.

1.2 `computeDiffusivity.m`

- works as a wrapper around LinStatSolver.m and StatPostProcessor.m to compute the 9 components of the homogenized diffusivity tensor.
- the implementation is parallelized for speed.
- requires nothing.

1.3 `PreProcessor.m`

- discretizes SVE to structured grid.
- creates and saves topology matrices for input to the processor files.
- requires input data generated by `SVEGenerator.m`.

1.4 `findITZ.m`

- computes $A_{\mathrm{ITZ}}, V_{\mathrm{a}}$ and V_{cp} in fig. 2.
- called by `PreProcessor.m`.

1.5 `LinStatSolver.m`

- Solves the linear system of equations, $\boldsymbol{K}\boldsymbol{a} = \boldsymbol{f}$.
- Dirichlet and convective boundary conditions types are implemented.
- requires input data generated by `PreProcessor.m`.

1.6 `LinTransSolver.m`

- Solves the linear system of equations, $\boldsymbol{C}\dot{\boldsymbol{a}} + \boldsymbol{K}\boldsymbol{a} = \boldsymbol{f}$.
- convective boundary conditions types are implemented.
- requires input data generated by `PreProcessor.m`.

1.7 `StatPostProcessor.m`

- post processor routine for `LinStatSolver.m`
- generates `.vtk` file which can be imported to eg Paraview for visualization.

1.8 `TransPostProcessor.m`

- post processor routine for `TransStatSolver.m`
- generates `.vtk` file which can be imported to eg Paraview for visualization.

2 Element node numbering

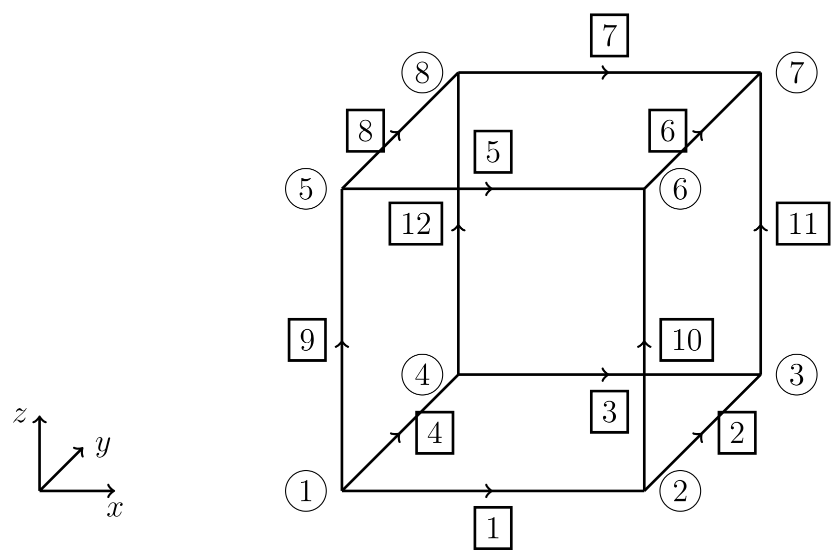

Figure 1: Node numbers in circles and line segments in rectangles.

3 ITZ implementation

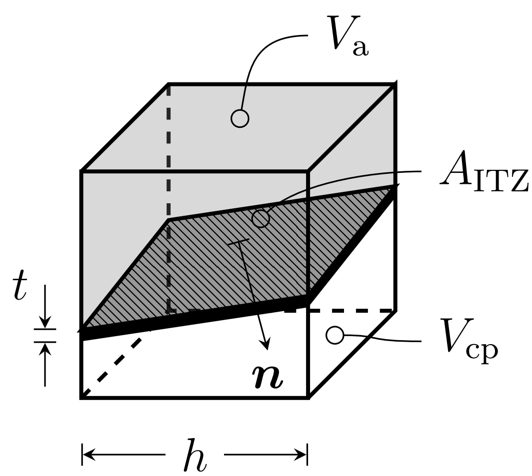

Figure 2: Interface voxel.

3.1 Line-sphere intersection

$$d = -(\boldsymbol{l} \cdot (\boldsymbol{o} - \boldsymbol{c})) \pm \sqrt{(\boldsymbol{l} \cdot (\boldsymbol{o} - \boldsymbol{c}))^2 - (\boldsymbol{o} - \boldsymbol{c})^2 + r^2} \tag{1}$$

3.2 Voigt assumption

3.2.1 Isotropic

$$\bar{D} = \frac{1}{2}(D_\mathrm{a} + D_\mathrm{c}) + \frac{t}{h} D_\mathrm{ITZ} \tag{2}$$

3.2.2 Anisotropic

$$\bar{\boldsymbol{D}} = \frac{V_\mathrm{a} D_\mathrm{a} + V_\mathrm{c} D_\mathrm{c}}{V_\mathrm{a} + V_\mathrm{c}} \boldsymbol{I} + \frac{A_\mathrm{ITZ} D_\mathrm{ITZ} t}{V_\mathrm{a} + V_\mathrm{c}} \left(\boldsymbol{I} - \boldsymbol{n} \otimes \boldsymbol{n}\right) \tag{3}$$