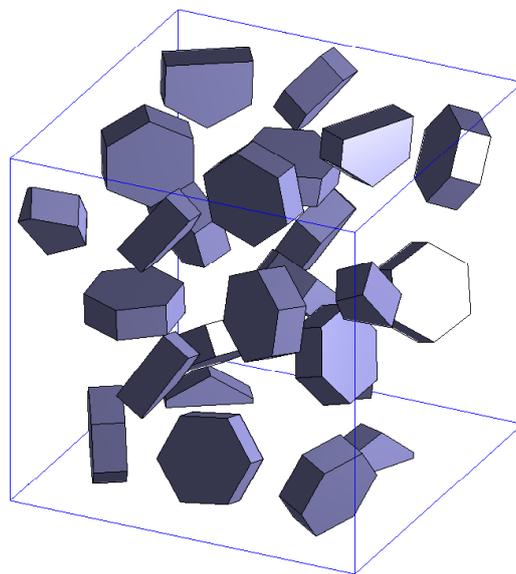


USER'S GUIDE

Periodic Cell Generation and Meshing: Inclusions in Shape of Hexagonal Prisms

Generation of periodic cell and input file for Ansys to construct a periodic mesh



Authors:

Václav NEŽERKA,
Petr KABELE

October 7, 2013

The program was developed for fast and user-friendly preparation of periodic unit cell containing the predefined inclusions of hexagonal prismatic shape, and generation of input file for Ansys software package to build a periodic mesh of predefined parameters. If you encounter any problems or bugs when using the program, please don't hesitate and contact the author at nezerka.v@seznam.cz. The author would like to thank his supervisor, Prof. Petr Kabele, who provided him with fruitful consultations during the software development. This work was supported by the grant no. GAP210/11/1168.

1 Program Structure

The program is started using the file `geometryGeneration.m` in the main folder. The program structure is following (see Fig.1):

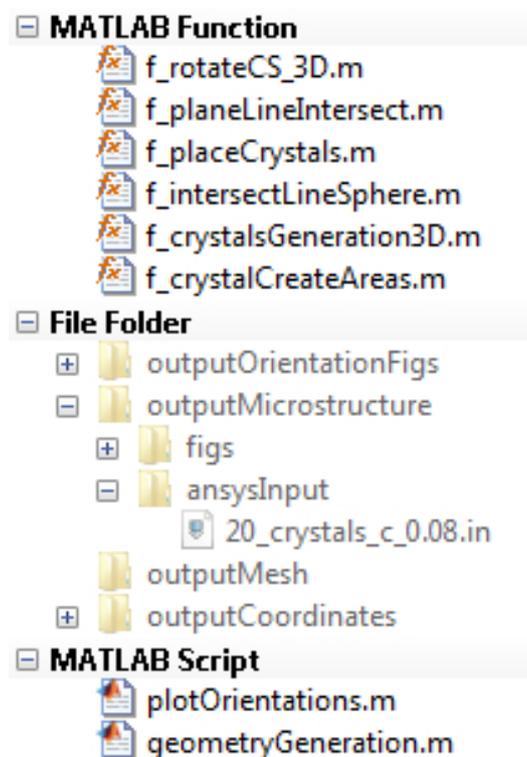


Figure 1: Program data structure

2 Input File Format

The input for the program is defined by the user in the main file `geometryGeneration.m`, the explanation of individual items (see Fig.2) follows:

- *cAgg* – desired volume fraction of inclusions
- *numberOfCrystals* – desired number of inclusions
- *particleRadius* – radius of circumscribed circle around the modeled hexagonal prisms
- *particleHeight* – height of the modeled prisms

- *minDistBetweenGrains* – minimum distance between grains (if not controlled, mesh can be unacceptable)
- *controlBoundaryMesh* – to control the element size on the boundary
- *controlMeshSizeInMatrix* – to control the element size within the matrix
- *elSize* – element size within the matrix
- *elSizeBoundary* – element size at the boundary of the sample
- *elSizeGrain* – element size inside grains

```

%% Input

cAgg = 0.08; % volume fraction of aggregates in the mix
numberOfCrystals = 20;
particleRadius = 1.5e-3; % radius of circumscribed circle around the modeled hexagonal prisms.
particleHeight = 1e-3; % height of the modeled prisms

sphereRadius = sqrt((particleHeight/2)^2+particleRadius^2); % radius of circumscribed sphere to check colisions
cubeSizeRVE = (numberOfCrystals*1.5*sqrt(3)*particleRadius^2*particleHeight/cAgg)^(1/3);
minDistBetweenGrains = 4e-4; % minimum distance between grains

figureOut = true; % true = plots the generated microstructure, false = no plotting
coordinatesOfVertieeesOut = true; % true = prints the coordinates of individual vertieees out

%% ANSYS and meshing options

controlBoundaryMesh = true; % whether to control the elementsize on the boundary
controlMeshSizeInMatrix = true; % whether to control the elementsize manually
elSize = particleRadius/3.3; % element size
elSizeBoundary = particleRadius/3.3; % element size at the boundary of the sample
elSizeGrain = particleRadius/3.3; % element size inside grains

```

Figure 2: Settings available in main.m

After the calculation is complete, the results are printed in the MATLAB command window (console). The user should be provided with the data as depicted in Figure 3.

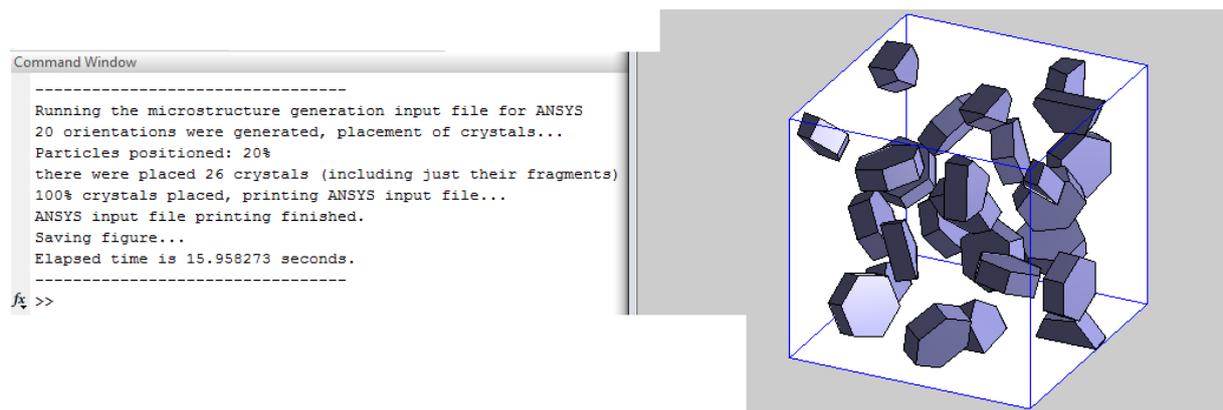


Figure 3: Program output

3 Mesh Generation

The mesh starts to generate after loading the generated file, located in the folder outputMicrostructure/AnsysInput/*.*in* (see Figure 1). The Ansys starts to generate the grains, by creating the keypoints, areas and volumes. After the generation is finished, the boolean operations

subtract the inclusion from the matrix and the meshing takes place. In order to produce truly periodic cell, the mesh is copied from the opposite sides of the produced RVE.

Then the model is ready for exportation of the mesh or calculation using the Ansys package. The examples of Ansys mesh are illustrated in Figures 4a and 4.

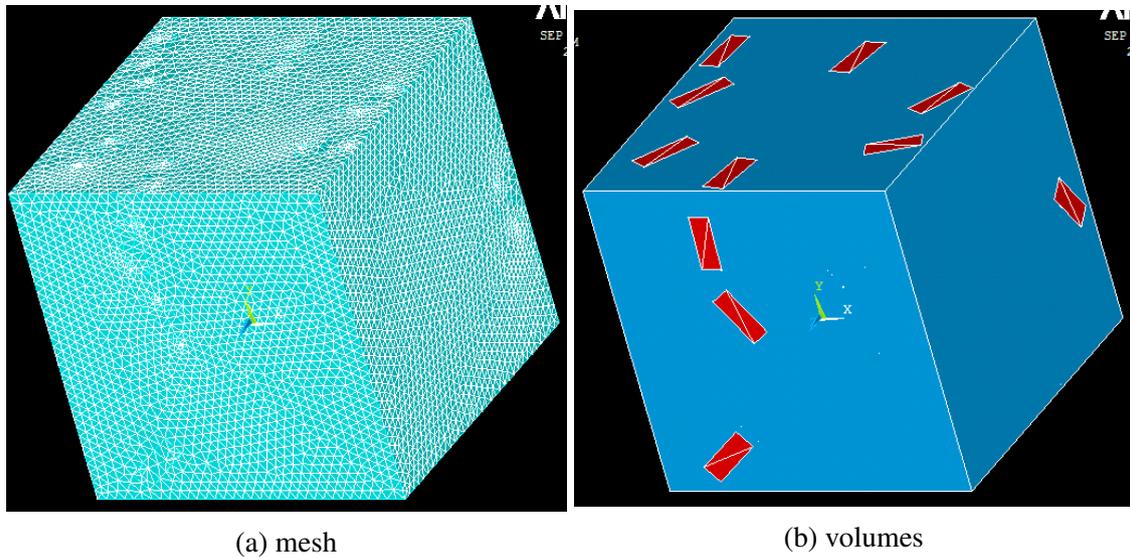


Figure 4: Ansys output — result of the commands generated by the MATLAB software.