

USER'S GUIDE

Homogenizator MT

**Effective stiffness estimation according to Mori-Tanaka scheme,
containing coated and/or uncoated spherical inclusions and voids +
calculation of the deviatoric stress in matrix**

The program was developed for fast and user-friendly homogenization using Mori-Tanaka scheme, considering spherical inclusions, voids and spherical inclusions with coating. It also evaluates deviatoric stress in the matrix (for unit macroscopic loading $\Sigma_{xx} = 1$), which can be helpful in estimation of strength. 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, Jan Zeman, PhD., who provided him with fruitful consultations during the software development. This work was supported by the grant no. DF11P01OVV008.

1 Program Structure

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

- Input file processing script can be found in the folder "files" and the user needn't open this folder.
- The functions needed for a calculation can be found in the folder "functions".
- Folder "input" is of the biggest importance for the user, since all the input data should be stored there. The program starts searching for the input files right in this folder.

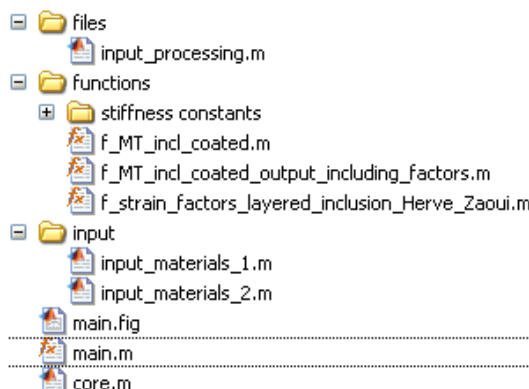


Figure 1: Program data structure

2 Input File Format

The input file must be in a format as in the sample file input_materials_1.m, Every phase is characterized by several data, according to its type: matrix, coated inclusion, coating, uncoated inclusion, and voids (see Fig.2).

3 Calculation of Effective Material Properties

The effective material properties are calculated according to Mori-Tanaka scheme. The inclusions can be spherical or spherical coated. The deviatoric stress in matrix can serve for

```

input_name = 'composite material 1';

%% Input parameters for individual phases

% matrix:
materials(1).name = 'matrix';
materials(1).code = 'matrix';

materials(1).m = 4; % mass portion in the mixture [kg]
materials(1).rho = 1900; % density [kg/m^3]
materials(1).E = 2000e6; % Young's modulus [Pa]
materials(1).nu = 0.25; % Poisson's ratio [-]

% brick:
materials(2).name = 'coated inclusion 1';
materials(2).code = 'coated';
materials(2).ID = 1; % it must be in agreement with a corresponding coating

materials(2).m = 4; % mass portion in the mixture [kg]
materials(2).rho = 1600; % density [kg/m^3]
materials(2).E = 2400e6; % Young's modulus [Pa]
materials(2).nu = 0.17; % Poisson's ratio [-]

materials(2).GC.frac = [0.063,0.125;
                       0.125,0.25;
                       0.25,0.5;
                       0.5,1;
                       1,2;
                       2,4]*1e-3; % grading curve - fractions (diameter) [mm]
materials(2).GC.amount = [6;
                          24;
                          45;
                          60;
                          66;
                          150]; % number of grains from each interval (mass, volume) [-]

materials(2).n_in_fraction = 2; % number of fractions within one fraction interval

% sand:
materials(3).name = 'uncoated inclusion';
materials(3).code = 'uncoated';

materials(3).m = 4; % mass portion in the mixture [kg]
materials(3).rho = 2600; % density [kg/m^3]
materials(3).E = 70000e6; % Young's modulus [Pa]
materials(3).nu = 0.17; % Poisson's ratio [-]

% voids:
materials(4).name = 'voids';
materials(4).code = 'voids';

materials(4).volume = 35; % volume [%]
materials(4).E = 1e-9; % Young's modulus [Pa]
materials(4).nu = 0.25; % Poisson's ratio [-]

% C-S-H gel:
materials(5).name = 'coating of coated inclusion';
materials(5).code = 'coating';
materials(5).ID = 1; % it must be in agreement with a corresponding coated particle

materials(5).rho = 2000; % density [kg/m^3]
materials(5).E = 22000e6; % Young's modulus [Pa]
materials(5).nu = 0.2; % Poisson's ratio [-]
materials(5).thickness = 20e-6; % thickness of coating [m]
materials(5).p_i = 20; % percentage of inclusion consumed by coating when coating is created;
materials(5).p_m = 50; % percentage of matrix consumed by coating when coating is created;
materials(5).p_v = 30; % percentage of voids consumed by coating when coating is created;

```

Figure 2: Format of an input file

estimation of the material strength σ and its derivation is based on the equality between average energy density if expressed by microscopic and macroscopic quantities.

After opening the program, a user is asked to choose an input file. The program starts searching in the folder "input".



Figure 3: Main menu of the program

After the file is chosen, the program immediately calculates the effective stiffness, Poisson's ratio and deviatoric stress in the matrix (for unit macroscopic loading $\Sigma_{xx} = 1$).



Figure 4: Input menu