`fibergen`: An introductory tool for FFT-based material homogenization

Felix Ospald[1]

1 Research Group Numerical Mathematics, TU Chemnitz, Chemnitz, Germany

Summary

Many engineering applications require material properties of heterogeneous media. For instance elasticity constants and thermal conductivity of fiber reinforced plastics, the viscosity of a suspension, permeability of porous media, electrical conductivity of sintered metals, … Laboratory experiments for determining these properties are often difficult, expensive and error-prone. Therefore, increasingly computer simulations are used instead, both for the computation of effective properties and the generation of the geometry of such materials in order to avoid the use of expensive computer tomography (CT) scans.

`fibergen` is a easy to use tool (command-line/GUI) for the homogenization (i.e. obtaining effective material parameters of an equivalent homogeneous media) and generation of periodic materials, especially but not limited to fiber reinforced composites produced by injection molding. The voxel-based spectral method relies on the fast Fourier transform (FFT) and OpenMP parallelization for highly efficient computations, which can be carried out on an average desktop computer. The tool also includes a geometry generator, but is also capable to process raw data (e.g. CT data).

Relation to similar projects

There exists a number of similar projects for the homogenization of materials, developed by well-known people in this research area:

- AMITEX, Lionel Gélébart, Atomic Energy and Alternative Energies Commission, France, (Fortran)
- morphhom, François Willot, Université Paris, France, (Fortran/C)
- DAMASK, Martin Diehl, Max-Planck-Institut für Eisenforschung, Germany, (Fortran, C, C++)
- janus, Sébastien Brisard, Université Paris, France, (Cython)
- FFTHomPy, Jaroslav Vondřejc, TU Braunschweig, Germany (Python)

All these projects are written in different programming languages and implement different solver methods and provide different features. Most of these tools provide only the pure solver without a geometry generator and without post-processing (visualization) of the results. `fibergen` on the other hand provides a solver but also an GUI/IDE for editing project files, running the solver and post-processing. It is also written in object-oriented C++ and provides a simple Python interface for the automation of computations.
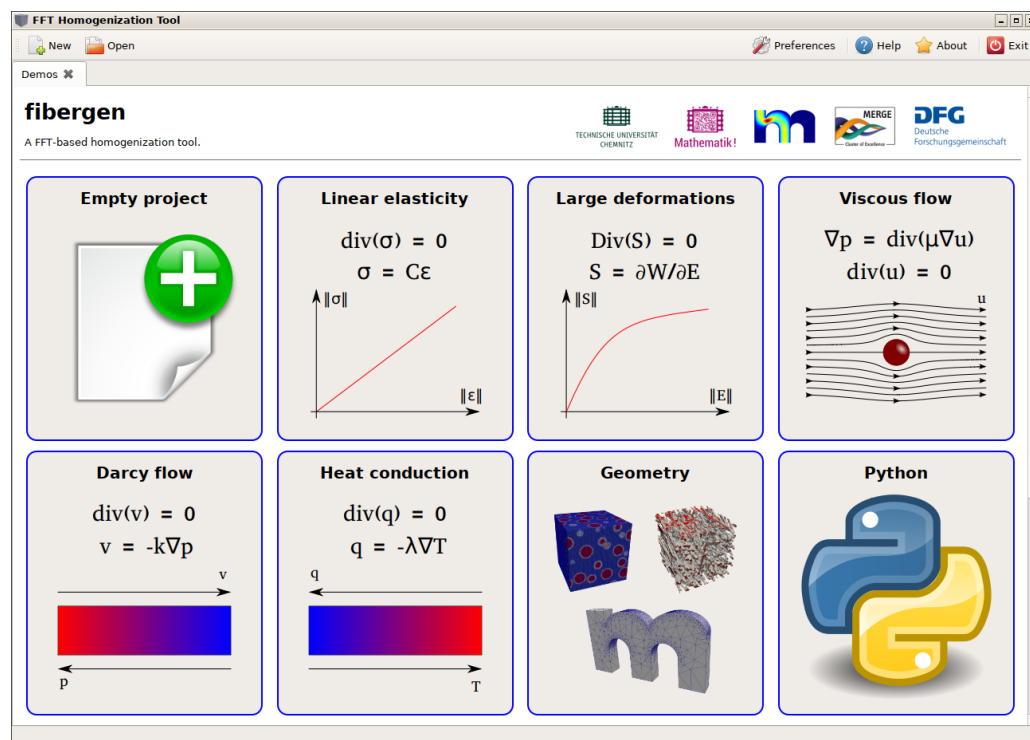
Figure 1: `fibergen` GUI, showing the main screen for selecing one of the predefined demos.

Features

`fibergen` implements FFT-based homogenization based on the Lippmann-Schwinger equation. It includes the original method by (Moulinec & Suquet, 1994) and the more recent discretizations (Willot, 2015) and (M. Schneider, Ospald, & Kabel, 2016). Especially the staggered grid method is not implemented by any other of the listed tools, although it is the superior method. `fibergen` also implements efficient fixed point and Newton-Krylov solvers for FFT-based homogenization of elasticity at large deformations as described in (Kabel, Böhlke, & Schneider, 2014). Further it implements so called composite voxels, for the linear (M. Kabel, Merkert, & Schneider, 2014) and nonlinear case (M. Schneider et al., 2016), which increases the accuracy and/or speed of the method enormously. Also mixed boundary conditions (Kabel, Fliegener, & Schneider, 2016) are supported. Further homogenization solvers for the effective thermal conductivity, porosity and viscosity are provided. Geometry can be loaded from (gzip compressed) raw data (e.g. CT scans) or also be generated from simple sphere, cylinder and capsule-like objects, but also tetrahedral meshes can be used as input geometry. For cylinder and capsule objects a orientation distribution can be selected for the generation, as well as distributions for the spatial position, length- and diameter. Here we provide the angular central Gaussian distribution as employed by (Montgomery-Smith, Jack, & Smith, 2011) beside others.

Projects are stored as XML configuration files, which may be executed over the command-line or using the GUI. Furthermore `fibergen` provides a Python library, which can be used to create solver objects, modify settings, run the solver and extract results in an automated fashion.
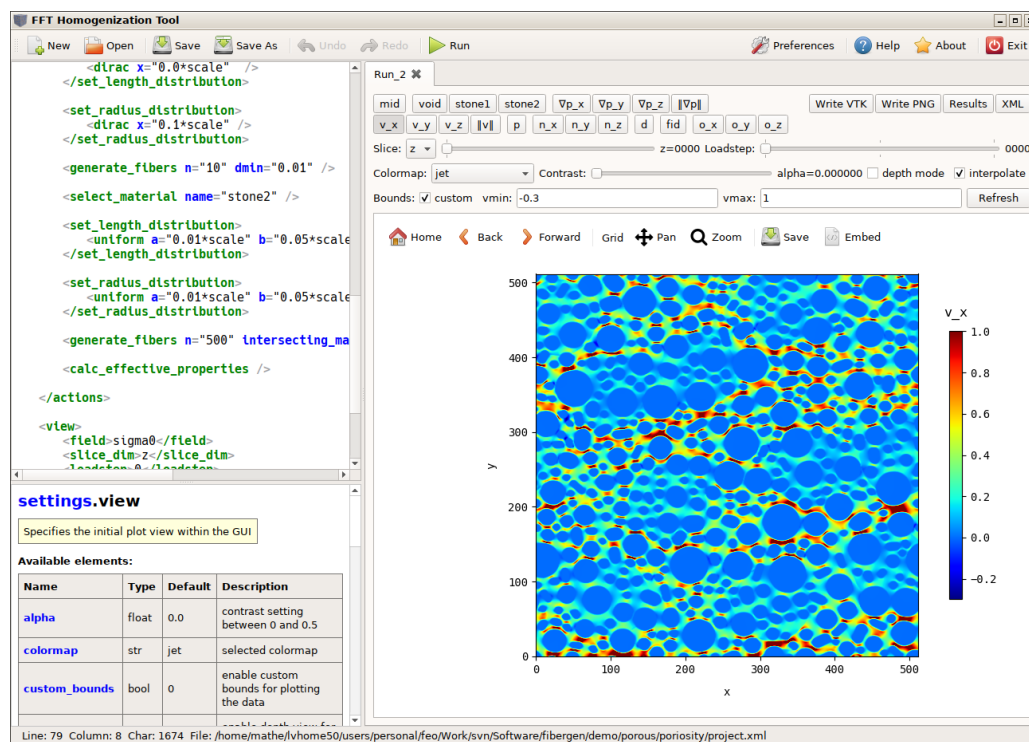
Figure 2: `fibergen` GUI, showing the project editor on the left and the postprocessing view on the right.

Development

The `fibergen` core was developed in C++ heavily using the [Boost](#) and [boost-numeric-bindings](#) for Boost uBLAS library. Parallelization is realized through [OpenMP](#). The Python 2/3 compatible interface is provided using [Boost.Python](#), where computed results are returned as [numpy](#)-array. The graphical user interface was developed using [PyQt5](#), the Python bindings of [Qt5](#), including the [QWebEngine](#) (or QWebKit) for browsing demos as well as the Qt back-end of [matplotlib](#) for embedding plots. A small amount of [scipy](#) functionality is used to write images.

Limitations

`fibergen` does not implement advanced methods for crystal-plasticity or even cracking. It should be seen as an simple tool for FFT-based homogenization, with a simple-to-use GUI where different methods and parameters can be selected and compared easily. The geometry generation tool is definitely limited to simple arrangements of objects. The nonlinear solvers also have some limitations/convergence issues for certain problems. Nevertheless it can solve problems with several 100 million unknowns easily, provided enough memory is available. The computed results are without any liability and warranty for their correctness.

Acknowledgements

References

Kabel, M., Böhlke, T., & Schneider, M. (2014). Efficient fixed point and newton-krylov solvers for fft-based homogenization of elasticity at large deformations. *Computational Mechanics*, *54*(6), 1497–1514. doi:10.1007/s00466-014-1071-8

Kabel, M., Fliegener, S., & Schneider, M. (2016). Mixed boundary conditions for fft-based homogenization at finite strains. *Computational Mechanics*, *57*(2), 193–210. doi:10.1007/s00466-015-1227-1

Kabel, M., Merkert, D., & Schneider, M. (2014). FFT-based homogenization using composite voxels. *Computer Methods in Applied Mechanics and Engineering*.

Montgomery-Smith, S., Jack, D., & Smith, D. E. (2011). The fast exact closure for jeffery's equation with diffusion. *Journal of Non-Newtonian Fluid Mechanics*, *166*(7-8), 343–353. doi:http://dx.doi.org/10.1016/j.jnnfm.2010.12.010

Moulinec, H., & Suquet, P. (1994). A fast numerical method for computing the linear and nonlinear mechanical properties of composites. *Comptes rendus de l'Académie des sciences. Série II, Mécanique, physique, chimie, astronomie*, *318*(11), 1417–1423.

Schneider, M., Ospald, F., & Kabel, M. (2016). Computational homogenization of elasticity on a staggered grid. *International Journal for Numerical Methods in Engineering*, *105*(9), 693–720. doi:10.1002/nme.5008

Willot, F. (2015). Fourier-based schemes for computing the mechanical response of composites with accurate local fields. *Comptes Rendus Mécanique*, *343*(3), 232–245. doi:http://dx.doi.org/10.1016/j.crme.2014.12.005