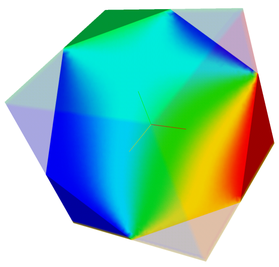

SfePy: finite element analysis software in Python

Robert Cimrman, Eduard Rohan, Vladimír Lukeš

Department of Mechanics & New Technologies Research Centre
University of West Bohemia, Univerzitní 8
306 14 Plzeň, Czech Republic
cimrman3@ntc.zcu.cz, rohan@kme.zcu.cz, lukes@kme.zcu.cz

1. Introduction

MADE publicly available at `http://sfepy.kme.zcu.cz` under the BSD open-source license, SfePy is a general finite element analysis software. The latest sources can be obtained at the developer's pages currently at `http://code.google.com/p/sfepy/`, containing also mailing lists and the issue (bug) tracker. We encourage and support everyone who joins! Current applications include

- homogenization of porous media (parallel flows in a deformable porous medium),
- acoustic band gaps (homogenization of a strongly heterogenous elastic structure: phononic materials),
- shape optimization in incompressible flow problems.

2. Appetizer I: shape optimization of channels

THE incompressible flow problem is defined in an open bounded domain $\Omega \subset \mathbb{R}^3$ with two (possibly overlapping) subdomains defined as

$$\overline{\Omega} = \overline{\Omega_D \cup \Omega_C} \quad \text{with} \quad \Gamma_C = \partial\Omega_D \cap \partial\Omega_C \,, \tag{1}$$

where Ω_C is the control domain and Ω_D is the design domain, see Fig. 1. The shape of Ω_D is modified exclusively through the design boundary, $\Gamma_D \subset \partial\Omega_D \setminus \Gamma_{\mathrm{in-out}}$ where $\Gamma_{\mathrm{in-out}} \subset \partial\Omega$ is the union of the "inlet-outlet" boundary of the channel; in general $\Gamma_{\mathrm{in-out}}$ consists of two disjoint parts, $\Gamma_{\mathrm{in-out}} = \Gamma_{\mathrm{in}} \cup \Gamma_{\mathrm{out}}$.

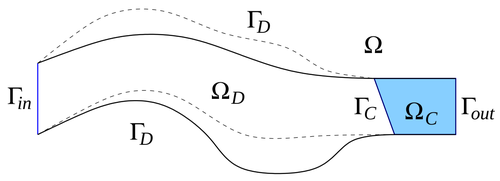

Figure 1: *The decomposition of domain Ω, control domain Ω_C at the outlet sector of the channel.*

We wish to enhance flow uniformity by reducing the gradients of flow velocities $\underline{u}$ in Ω_C. Our objective is thus to minimize

$$\Psi(\underline{u}) = \frac{\nu}{2} \int_{\Omega_C} |\nabla \underline{u}|^2 \tag{2}$$

by moving the design boundary Γ_D. The design changes are performed by means of the free-form deformation (FFD), cf. [4]. Example results can be seen in Fig. 2.

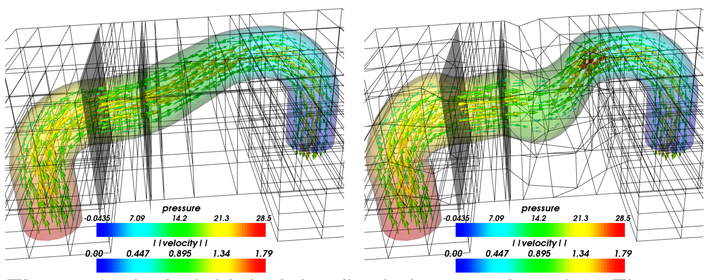

Figure 2: *Left: initial, right: final shapes of a tube. Flow and domain control boxes; Ω_C between two grey planes.*

For more information, see [2] or [5].

3. Which language?

WITH the advent of very high level dynamic, or scripting, programming languages new possibilities of code development emerged, accelerated by easily available computing power we have at our hands presently. Nowadays a very popular way of application development relies upon expressing the logic of a code in a high-level scripting language while, following a careful profiling, the real bottlenecks are implemented in a traditional language like fortran, C, or C++. The scripting language then serves as a glue, providing high-level interface to both (numerical) fortran legacy codes and newer libraries. The language of our choice is Python (`http://python.org`) — a remarkably powerful dynamic programming language that is used in a wide variety of application domains.

BY mixing programming languages we get best of both worlds: the speed of C in relevant parts of the code, and the flexibility, power and maintainability of Python.

- low level code (C or fortran): element matrix evaluations, costly mesh-related functions, ...
- high level code (Python): logic of the code, particular applications, configuration files, problem description files

We conclude this section with

SfePy = Python + C (+ fortran)

4. Dependencies

TO install and use SfePy, several other packages or libraries are needed:

- NumPy and SciPy: free (BSD license) collection of numerical computing libraries for Python
  - enables Matlab-like array/matrix manipulations and indexing
- other: UMFPACK, Pyparsing, Matplotlib, Pytables (+ HDF5), swig
- visualization of results: ParaView, Mayavi2, or any other VTK-capable viewer

5. Appetizer II: homogenization of porous media

TO tackle strong heterogeneities we use a multiscale approach based on the theory of homogenization, cf. [3]. In Fig. 3 (left) a periodic microstructure with a matrix and two systems of highly permeable channels (→strong heterogeneity in permeability) is shown ($3 \times 3 \times 3$ repetition of the reference volume element). Our model of parallel flows in a deformable porous medium is defined in terms of displacements $\underline{u}$ and two pressures p_1, p_2 in each channel on the macroscale, and by so-called corrector shape functions (displacement- and pressure-like) on the microscale, see Fig. 3 (middle, right).

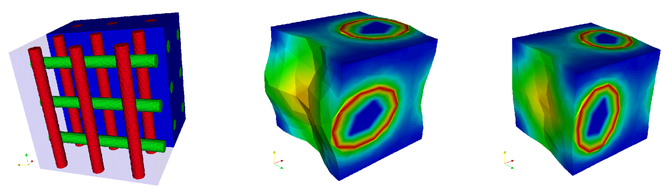

Figure 3: *Microstructure (left), example correctors (middle, right), color = pressure.*

The homogenized constitutive coefficients (elasticity, fading memory effects, Biot-like coefficients, permeability, etc.) computed for such a microstructure are then used within the macroscopic homogenized model, see Fig. 4.

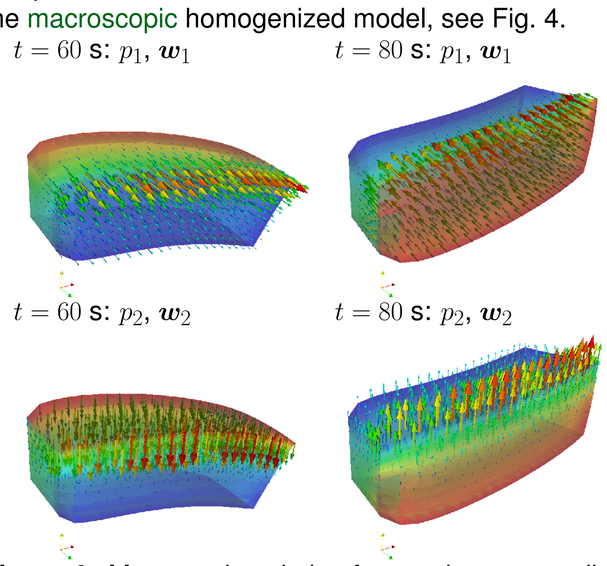

Figure 4: *Macroscale solution for two time steps: displacements 10× magnified, color = pressures p_1, p_2, arrows = diffusion velocities $\boldsymbol{w}_1$, $\boldsymbol{w}_2$ (5000× magnified).*

Finally let us show corrector shape functions of an idealized 2D microstructure of an osteon in Fig. 5, a structure quasi-periodically repeating in bones.

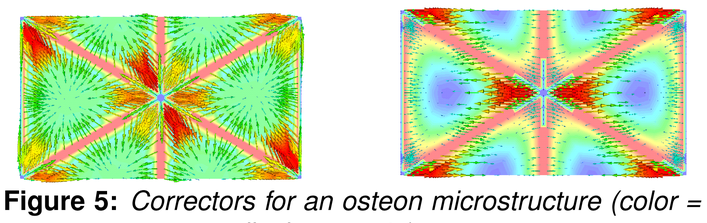

Figure 5: *Correctors for an osteon microstructure (color = pressure, arrows = displacements).*

6. Describing problems to solve

IN order to solve PDEs, these must be translated to a form that SfePy can deal with. Fortunately, this form is similar to a "paper" version of the problem. The problem description file is a regular Python module, i.e. all Python syntax and power is accessible, and consists of entities defining:

- fields of various FE approximations, variables,
- equations in the weak form, quadratures,
- boundary conditions (Dirichlet, periodic, "rigid body"),
- FE mesh file name, options, solvers, ...

Let us illustrate this using a trivial example: the Laplace equation

$$c\Delta u = 0 \text{ in } \Omega, \quad u = \bar{u} \text{ on } \Gamma \,. \tag{3}$$

EQUATIONS in SfePy are built using terms, which correspond directly to the integral forms of weak formulation of a problem to be solved. The weak formulation of (3) is: Find $u \in V$, such that

$$\int_\Omega c\, \nabla u : \nabla v = 0, \quad \forall v \in V_0 \,. \tag{4}$$

In SfePy input files, this can be written as

```
dw_laplace.i1.Omega( c, v, u ) = 0 ,
```
(5)

which directly corresponds to the discrete version of (4): Find $\boldsymbol{u} \in V_h$, such that

$$\boldsymbol{v}^T (\int_{\Omega_h} c\, \boldsymbol{G}^T \boldsymbol{G}) \boldsymbol{u} = 0, \quad \forall \boldsymbol{v} \in V_{h0} \,, \tag{6}$$

where $\nabla u \approx \boldsymbol{G}\boldsymbol{u}$. The integral over the discrete domain Ω_h is approximated by a numerical quadrature, that is named `i1` in our case.

7. Appetizer III: acoustic band gaps

AC oustic band gaps appear in strongly heterogeneous media composed of a matrix (mtx) and a periodic pattern of inclusions (inc). The strong heterogeneities in the elasticity can lead to negative eigenvalues of an effective mass tensor A^* for certain frequency ranges, resulting either in a strong band gap (no waves at all) or a weak band gap (waves in a particular direction only). The effective mass tensor $A^* = \sum_{j \in J} A^{*,j}(\omega) + r^* I$, $r^* = \int_{\mathrm{inc}} r_i + \int_{\mathrm{mtx}} r_m(y)$, r_i is the inclusion density, r_m the matrix density, has the following form:

$$A^{*,j}_{pq} = \frac{-\omega^2}{\omega^2 - \lambda^2} \int_{\mathrm{inc}} r_i \varphi^j_p \int_{\mathrm{inc}} r_i \varphi^j_q \,, \tag{7}$$

where ω is the frequency and $\{\varphi^j, \lambda^j\}_{j \geq 1}$ are the eigenelements associated to the elasticity operator in the inclusion domain, see [1].

Example results for the spherical inclusion are in Fig. 6 and and for the elliptical inclusion in Fig. 7.

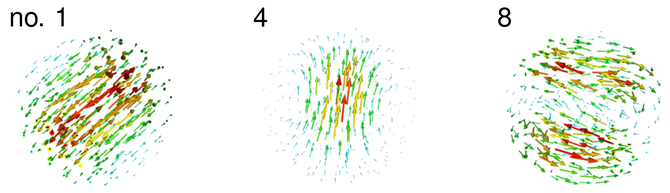

Figure 6: *Displacement eigenvectors of elasticity operator.*

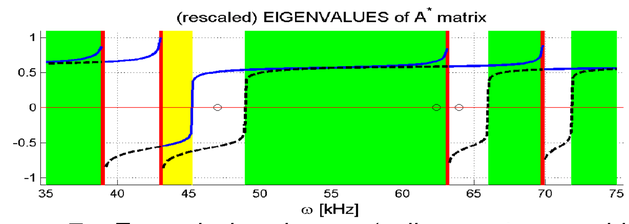

Figure 7: *Example band gaps (yellow = strong, white = weak), the largest (solid) and smallest (dashed) eigenvalues of A^* and resonance frequencies $\sqrt{\lambda^j}$ (vertical lines).*

Acknowledgement: This work has been supported by the grant project GAČR 101/07/1471 of the Grant Agency of the Czech Republic, entitled "Finite element modelling of linear, non-linear and multiscale effects in wave propagation in solids and heterogeneous media".

References

[1] A. Ávila, G. Griso, B. Miara, and E. Rohan. Multi-scale modelling of elastic waves – theoretical justification and numerical simulation of band gaps. *Multiscale Modeling and Simulation, SIAM*, 2007.

[2] R. Cimrman and E. Rohan. On shape optimization of conduits with incompressible flow. *Applied and Computational Mechanics*, 1(2):393–400, 2007. ISSN: 1802-680X, Nečtiny.

[3] G. Griso and E. Rohan. On the homogenization of a diffusion-deformation problem in strongly heterogeneous media. *Ricerche di Matematica*, 56(2):161–188, 2007.

[4] S. Menzel, M. Olhofer, and B. Sendhoff. Application of free form deformation techniques in evolutionary design optimization. In *Proceedings of 6th World Congress on Structural and Multidisciplinary Optimization*, Rio de Janeiro, Brazil, 2005.

[5] E. Rohan and R. Cimrman. Shape sensitivity analysis for flow optimization in closed channels. In *Proceedings of the conference Engineering Mechanics 2006*, Svratka, 2006. Full version on CD-ROM.