1   General information

This software is a script compilation for FreeFem++ to solve interior and interface cell problems to arrive with coefficients in governing equations of poroelastic material and boundary condition (BC) between poroelastic material medium and free fluid. This software gives the possibility to compute these coefficients in three-dimensions (3D) for fully connected poroelastic skeleton. If you find this software useful, please cite the corresponding publication <> (arXiv preprint).

The contents of this software are listed below.

```
src/0_setup/*                   - scripts to set-up microscale geometry
src/1_Interior/*                - scripts to solve for interior properties
src/2_Interface/*               - scripts to solve for interface coefficients
src/3_effective_solver/*        - solver for the averaged fields in the problem
src/4_DNS_solver/*              - used DNS solver and obtained fields
doc/Poroelastic_full_strCont.pdf - this document
```

In this document we explain the key elements of the software, parameters of the simulation, and implemented variational formulation. In the following sections, we describe each of the software directory in more details.

2   Setting up geometry for model parameter computation

The contents in the set-up directory are

```
src/0_setup/3D_MSE_parameters.in
src/0_setup/3D_MSE_gen_geom.py
```

The first file is an example parameter file, containing:

1. Size of the interface cell ($ztop - z$ coordinate of upper boundary, $Nbot$ – number of structures below the interface, resulting in $z = -Nbot$ coordinate of the lower boundary). The interface location is always at $z = 0$;

2. Definition of the one solid microstructure with $a_1 = eA$, $a_2 = eB$, $a_3 = eC$, $r = cR$ and $\phi = phi$. See the main paper for explanations of geometrical parameters;

3. Mesh spacings $\Delta s_1 = dsMin$ and $\Delta s_2 = dsMax$.

Given the required parameters (the example input file contains parameters for monoclynic symmetric geometry from the main paper with intermediate resolution), the geometry definition can be generated executing the Phython script

```
./3D_MSE_gen_geom.py
```

which generates 5 geometry definition files and 1 FEM space definition file, used later by FreeFem++ solver. Before proceeding, all mesh files must be generated using GMSH software,

which in the command line can be done by typing

```
gmsh -3 3D_MSE_gen_geomX.geo
```

where X should be replace by I, Is, B, L and T. The generated meshes are later used to obtain needed coefficients for the effective/averaged model equations.

3   Obtaining interior properties

The equations, which are solved to obtain the effective interior properties, are described in the main paper. Standard weak forms for Stokes equations and linear elasticity equations are used. The final results for the interior can be obtained by running the FreeFem++ software in folder "1_Interior" in the command line

```
FreeFem++ -nw 3D_interiorC_solver.edp
FreeFem++ -nw 3D_interiorK_solver.edp
```

which would produce a text file "3D_interiorK_matrix.txt" containing the permeability tensor, and a text file "3D_interiorC_matrices.txt" containing the effective elasticity tensor in Voigt notation and other elasticity related parameters. The tensor D_{ij} is outputted in order to make sure that relationship

$$\alpha_{kl} = \theta\delta_{kl} - D_{kl}$$

holds. The other parameters will be used later in the effective/averaged effective equation solver.

4   Obtaining interface properties

The equations, which are solved to obtain the effective interior properties, are described in the main paper as well as in the related paper <arXiv:1604.02880>. The results for the interface can be obtained by executing in folder "2_Interface" the help script

```
./3D_MSE_bc_do_all.sh
```

which executes the FreeFem++ scripts "3D_MSE_bc_Ksolver.edp" and "3D_MSE_bc_Lsolver.edp" for all (forced) problem indices. This execution step outputs both plane averaged field profiles over vertical distance as well as components of the K_{ij} and L_{ij3} tensors.

5   Running the effective equation solver

The effective equation solver is located in folder "3_effective_solver" and can be executed in command line by entering

```
FreeFem++ 2DfluidPEmedium_cavity_UnsteadyNStokes.edp
```

This execution produces visual representation of generated mesh for both free fluid and poroelastic domain, as well as 2D isocontour plots for velocity and deformation fields. In addition, the script outputs 4 line probes, 2 for fluid velocity and deformation fields, respectively. The problem solved (and the parameters given) is the steady Stokes cavity flow as described in the main paper, but the solver is capable of solving time-dependent Navier-Stokes problems. The parameters for the microstructure should be entered from simulations carried out over sections 2 to 4.

Parameters in the solver should be self-explanatory with the accompanying comments. One non-trivial aspect is transferring the interface results from the interface problem in "2_Interface" to solver script. One has to be careful by observing the tensor indices. For example, a possible result file from L_{ijk} solver might look like

```
cat 3D_MSE_bc_Li23_intf.txt
# zintf, L1kl, L2kl, L3kl
0 0.000321191 0.186116 1.3653e-05
```

from where we find non-zero element L_{223}. The value of this tensor component then must be correspondingly entered in position $(2, 3)$ in L_{2ij} matrix, as shown below

```
97  real[int,int] intfL2 = [ [0, 0, 0 ],
98                           [0, 0, 1.86e-1],
99                           [0, 0, 0  ] ]; // Interface slip length matrix L2kl
```

Other non-zero elements (after observing convergence) must be treated in the same way, that is, analysed element by element and entered in the correct positions of the solver matrices.

Using the information provided above, one can use this tool to predict material properties for different geometries, and test such materials when exposed to the free fluid.

In the following section we quickly present the weak form, which the presented software solves for readers interested in more details.

5.1   Weak form

In the presented solver, we discretise the equations presented in main paper but rewritten in dimensionless form in the microscale coordinate. Equations in the free fluid region are

$$\epsilon^2 Re_m \left(\partial_t u_i + u_j u_{i,j}\right) = \Sigma_{ij,j}, \qquad \Sigma_{ij} = -p\delta_{ij} + 2\epsilon\varepsilon_{ij}\left(u\right) \qquad \text{in } \Omega_f, \qquad (1)$$

$$u_{i,i} = 0, \qquad \text{in } \Omega_f, \qquad (2)$$

whereas equations in poro-elastic region are

$$-\left(\mathcal{K}_{ij}\frac{1}{\epsilon}\tilde{p}_{,j}\right)_{,i} = -\alpha_{kl}\varepsilon_{kl}\left(\partial_t v\right) - \epsilon\mathcal{E}\partial_t\tilde{p} \qquad \text{in } \Omega_p, \qquad (3)$$

$$\epsilon\left(1-\theta\right)\bar{\rho}\partial_t^2 v_i = \left[\frac{\bar{\bar{E}}}{\epsilon}\mathcal{C}_{ijmn}\varepsilon_{mn}\left(v\right)\right]_{,j} - \alpha_{ij}\tilde{p}_{,j} \qquad \text{in } \Omega_p, \qquad (4)$$

and boundary conditions between free fluid and poroelastic medium reads as

$$u_i - \partial_t v_i = -\bar{\mathcal{K}}_{ij}\frac{1}{\epsilon}\tilde{p}_{,j} + \bar{\mathcal{L}}_{ijk}\left(u_{j,k} + u_{k,j}\right) \qquad \text{at} \qquad \Gamma, \qquad (5)$$

$$p - \tilde{p} = 0 \qquad \text{at} \qquad \Gamma, \qquad (6)$$

$$\Sigma_{ij}n_j = \left[\frac{\bar{\bar{E}}}{\epsilon}\mathcal{C}_{ijmn}\varepsilon_{mn}\left(v\right) - \alpha_{ij}\tilde{p}\right]n_j \qquad \text{at} \qquad \Gamma. \qquad (7)$$

Here we write the equations in the weak form. The resulting weak formulation is regrouped to pull to the left all unknowns, and to the right all the knowns. The implicit Euler scheme for fluid viscous stress is used, and explicit Adams-Bashforth 2nd order scheme is used for the non-linear terms. Equations in the free fluid region are

$$\epsilon^2 Re_m \int_\Omega \frac{1}{\Delta t}u_i^{n+1}\hat{u}_i \, d\Omega + \int_\Omega \Sigma_{ij}^{n+1}\epsilon_{ij}\left(\hat{u}\right) \, d\Omega - \int_{\partial\Omega} \Sigma_{ij}^{n+1}n_j\hat{u}_i \, dS = \qquad (8)$$

$$= \epsilon^2 Re_m \int_\Omega \left(\frac{1}{\Delta t}u_i^n - \frac{3}{2}u_j^n u_{i,j}^n + \frac{1}{2}u_j^{n-1}u_{i,j}^{n-1}\right)\hat{u}_i \, d\Omega, \qquad (9)$$

$$\int_\Omega u_{i,i}^{n+1}\hat{p} \, d\Omega = 0, \qquad (10)$$

whereas equations to be solved in poro-elastic region are

$$\int_\Omega \frac{1}{\Delta t} \left(\alpha_{kl} \varepsilon_{kl}^{n+1}(v) + \epsilon \mathcal{E} \tilde{p}^{n+1} \right) \hat{p}\, d\Omega + \frac{1}{\epsilon} \int_\Omega \mathcal{K}_{ij} \tilde{p}_{,j}^{n+1} \hat{p}_{,i}\, d\Omega - \frac{1}{\epsilon} \int_{\partial\Omega} \mathcal{K}_{ij} \tilde{p}_{,j}^{n+1} n_i \hat{p}\, dS =$$

$$= \int_\Omega \frac{1}{\Delta t} \left(\alpha_{kl} \varepsilon_{kl}^n(v) + \epsilon \mathcal{E} \tilde{p}^n \right) \hat{p}\, d\Omega, \tag{11}$$

$$\epsilon (1 - \theta) \bar{\rho} \int_\Omega \frac{1}{\Delta t^2} v_i^{n+1} \hat{v}_i\, d\Omega + \frac{\bar{\bar{E}}}{\epsilon} \int_\Omega \mathcal{C}_{ijmn} \varepsilon_{mn}\left(v^{n+1}\right) \varepsilon_{ij}(\hat{v})\, d\Omega - \tag{12}$$

$$- \frac{\bar{\bar{E}}}{\epsilon} \int_{\partial\Omega} \mathcal{C}_{ijmn} \varepsilon_{mn}\left(v^{n+1}\right) n_j \hat{v}_i\, dS + \int_\Omega \alpha_{ij} \tilde{p}_{,j}^{n+1} \hat{v}_i\, d\Omega = \epsilon (1 - \theta) \bar{\rho} \int_\Omega \frac{1}{\Delta t^2} \left(2 v_i^n - v_i^{n-1} \right) \hat{v}_i\, d\Omega.$$

Boundary conditions between free fluid and poroelastic medium reads as

$$u_i^{n+1} - \frac{v_i^{n+1}}{\Delta t} + \bar{\mathcal{K}}_{ij} \frac{1}{\epsilon} \tilde{p}_{,j}^{n+1} - \bar{\mathcal{L}}_{ijk} \left(u_{j,k}^{n+1} + u_{k,j}^{n+1} \right) = -\frac{v_i^n}{\Delta t} \qquad \text{at} \qquad \Gamma, \tag{13}$$

$$p^{n+1} - \tilde{p}^{n+1} = 0 \qquad \text{at} \qquad \Gamma, \tag{14}$$

$$\Sigma_{ij}^{n+1} n_j - \left[\frac{\bar{\bar{E}}}{\epsilon} \mathcal{C}_{ijmn} \varepsilon_{mn}\left(v^{n+1}\right) - \alpha_{ij} \tilde{p}^{n+1} \right] n_j = 0 \qquad \text{at} \qquad \Gamma. \tag{15}$$

This weak formulation is implemented in the FreeFem++ solver using a set of macros, which group the different parts of the weak form (based on different combinations between the variable and test-function). The addition of Lagrangian multipliers can be understood from the solver code or documentation of software "Porous_full_bc2ifScales" module "2D_MSE_bc_velocity".

6   DNS solver and post-processing

The DNS solver, post-processing routines and final flow and displacement field can be found in directory "4_DNS_solver". In this section we briefly explain, how the solution fields were obtained and also how to post-process the field data.

The DNS solver includes a separate parameter file "3D_DNS_parameters.in", in which the geometry is defined similarly as in previous sections. Recall from the main paper, that the geometry is one-structure thick cavity slice. Therefore number of microsctructures in the cavity has to bee defined. It is done by setting parameters $NbotX$ and $NbotD$. The example parameter file corresponds to the finest results reported in the main paper. Given the parameter file, geometry definition files and meshes can be obtained by executing

```
./3D_DNS_gen_geom.py
gmsh -3 3D_DNS_gen_geomF.geo
gmsh -3 3D_DNS_gen_geomS.geo
```

After that, one can execute the solver "3D_DNS_solver.edp" and wait until GMRES solver produces a solution. Alternatively (what we have done), one can start with a coarser mesh, and use the solution as an initial guess for the finer simulation. The published data-set is obtained in this way.

In order to post-process the fields used to produce the plots in the main paper, one has to extract the archive containing the fields and meshes

```
tar -xvf 3D_DNS_solution.tar.gz
```

After extracting the needed files, one can use two provided FreeFem++ post-processing tools. The first tool "3D_DNS_post_line.edp" provides output of raw line probes of specified coordinates, while the second tool "3D_DNS_post.edp" provides output of y-averaged line probe values of specified coordinates in a two-dimensional (x, z) plane. The averaging is carried out as defined in the main paper. We have also found that introducing a discontinuous FE space (piecewise constant P0) is useful in certain cases.