DIN - DEPARTMENT OF INDUSTRIAL ENGINEERING

PhD Course in Mechanics and Advanced Engineering Sciences (DIMSAI)

PHD STUDENT POSTER - AY 2016/2017 Curriculum number 3 Cycle 31

Candidate: Roberto Da Vià Supervisor: Sandro Manservisi

Development of a computational platform for the simulation of turbulent low Prandtl number fluids in the condition of natural convection

Goal of the project

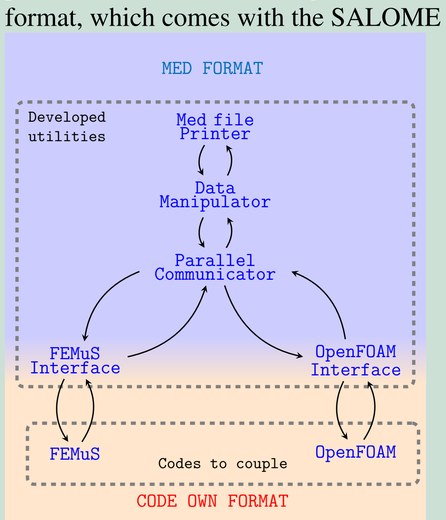
With the present PhD work we focus our attention on the numerical simulation of liquid metals, e.g. fluids with a low value of molecular Prandtl number. These particular fluids represent a promising future technology as they have a high thermal conductivity and they can be controlled with volumetric forces (electromagnetic forces). Low Prandtl number fluids require proper turbulence models to study turbulent heat transfer as common turbulence models lead to overestimations of heat transfer. Moreover, in the condition of natural convection, turbulence models that takes into account the anisotropic behavior of the fluid motion should be used. **The purpose of the work is to provide suitable means for the numerical study of liquid metals, in the condition of turbulent natural convection. In particular we aim at integrating our K-Ω model with an anisotropic formulation of turbulent heat transfer.** The overall work is structured as follows:

- Development and improvement of an isotropic K-Ω turbulence model
- Development of an anisotropic turbulence model for heat transfer
  - Use external codes for calculation of Reynolds stresses, required for anisotropic model
  - Development of routines for numerical code coupling

As the improvement of the K-Ω turbulence model was investigated during the first year of PhD study, here we focus our attention on the realization of numerical code coupling, in order to extend the capabilities of our code.

Code coupling between FEMuS and OpenFOAM

`FEMuS` is our inhouse developed finite element code while `OpenFOAM` is an opensource finite volume code. As each code works with its own format for storing numerical fields and meshes, we need to provide methods for handling the data exchange between the two different codes. We use the MED data format, which comes with the SALOME platform, as an intermediate data format.



Interfaces

The coupling is performed by building specific *interfaces* for each code. An interface consists of a mesh that is a MED duplicate of a part of the original code mesh, on whose support we can store a numerical field. The main features of an interface are:

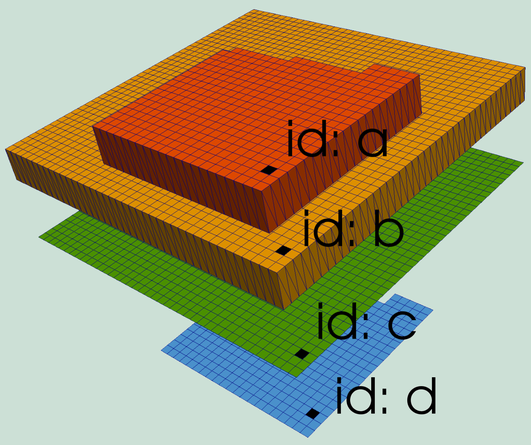
- extract a code solution and store it in MED format;
- set a MED field as part of a code solution.

Once the solution is stored in the MED format, we can perform a data manipulation in order to satisfy the receiver code requirements.

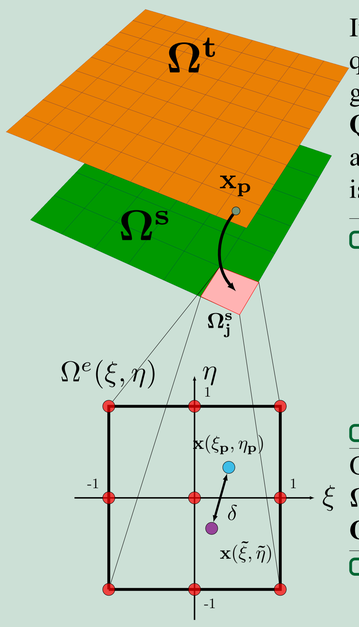
Huge simulation are generally solved in parallel. For this reason we also built a parallel communicator that is used to gather and scatter MED fields across all the different processors used for running the simulation.

The interfaces contain also maps that associates the node/cell numbering of the original code mesh to the relative numbering of the interface MED mesh duplicate. OpenFOAM requires 3D meshes even for 2D simulations. In order to couple the code with FEMuS we use a 3D MED mesh (yellow) and also a 2D equivalent grid (green). The 2D equivalent grid is used to perform data manipulation on the numerical fields.

When extracting a field from OpenFOAM solution, the field value of cell a will be stored in cell d of the 2D equivalent grid of the single processor mesh. When gathering the solution from all the processors, the particular value will be stored in cell c of the global 2D equivalent grid.



Data Manipulator



It is a class that contains routines for performing Gauss integration of quadratic fields, interpolation of quadratic fields on different meshes, generation of piecewise numerical fields.

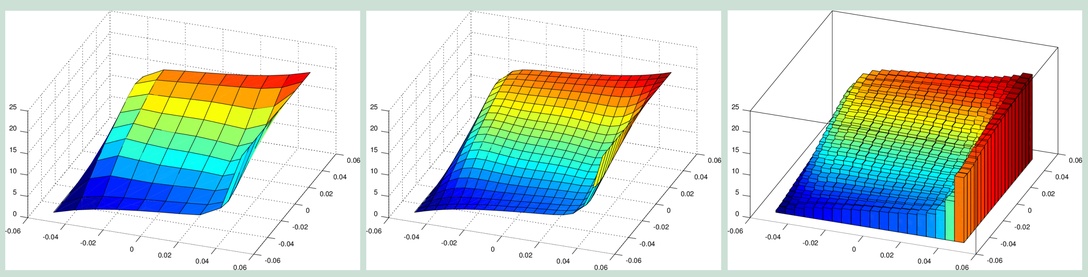
Quadratic field interpolation of a field Ψ^s from a source domain Ω^s to a target one Ω^t. The interpolated field is written as $\Psi^t = \mathbf{P}\Psi^s$, where $\mathbf{P}$ is an operator we build. Interpolation algorithm:

- $\forall\, \mathbf{x_p} \in \Omega^t$:
  - find $\Omega^s_j : \mathbf{x_p} \in \Omega^s_j$ + transformation of Ω^s_j into canonical $\Omega^e(\xi,\eta)$
  - $\boldsymbol{\xi_p} : min\left(\delta^2(\mathbf{x_p},\tilde{\boldsymbol{\xi}})\right) \rightarrow \delta^2(\mathbf{x_p},\tilde{\boldsymbol{\xi}}) = \sum_{i=1}^{dim}\left(x^i_p - \sum_{l=1}^{n_n} x^i_l \phi_l(\tilde{\boldsymbol{\xi}})\right)^2$
  - $\Psi^t(\mathbf{x_p}) = \sum_{l=1}^{n_n} \Psi^s(\mathbf{x_l})\phi_l(\boldsymbol{\xi_p})\,, \qquad \mathbf{x_l} \in \Omega^s_j\,, \mathbf{x_p} \in \Omega^s_j$
  - fill $\mathbf{P}$ with values of $\phi_l(\boldsymbol{\xi_p})$
- $\boldsymbol{\Psi}^t = \mathbf{P}\boldsymbol{\Psi}^s$

Operator P must be updated at every simulation time step if $\boldsymbol{\Omega^s}$ and/or $\boldsymbol{\Omega^t}$ change with time.

Generation of piecewise field Ψ^p, from source field Ψ, on domain Ω.

- $\forall\, \boldsymbol{\Omega_j} \in \Omega$:
  - $\Psi^p_j = \int_{\Omega_j} \Psi d\mathbf{x} / \int_{\Omega_j} d\mathbf{x}$



Simulation of a natural convection case

Some of the described coupling routines were succesfully tested with the realization of multiscale simulations by coupling our inhouse cfd code `FEMuS` with a system code[a,b], while here we report a study performed by coupling our code with OpenFOAM[c].

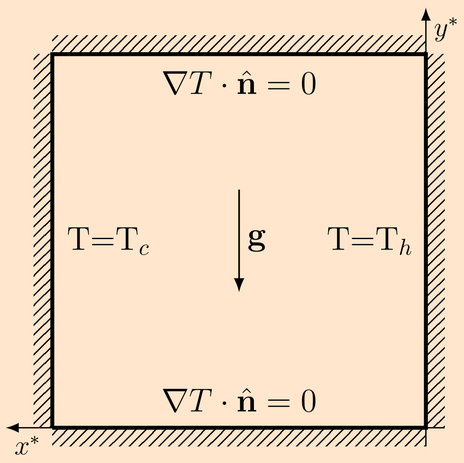
Test case: natural convection in a squared cavity, $Ra = 10^5$.

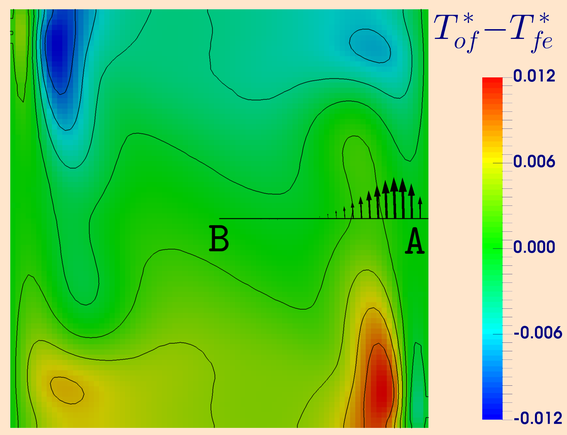
$$\boldsymbol{\nabla}\cdot\mathbf{u} = \mathbf{0}\,, \tag{1}$$

$$\rho\left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u}\cdot\boldsymbol{\nabla}\mathbf{u}\right) = -\boldsymbol{\nabla}P + \boldsymbol{\nabla}\cdot\left[\mu\left(\boldsymbol{\nabla}\mathbf{u} + \boldsymbol{\nabla}\mathbf{u}^{\mathbf{T}}\right)\right] + \underbrace{\rho\mathbf{g}\beta(\tilde{T} - T_{ref})}_{coupling\,term}\,, \tag{2}$$

$$\rho C_p\left(\frac{\partial T}{\partial t} + \mathbf{u}\cdot\boldsymbol{\nabla}T\right) = k\nabla^2 T\,. \tag{3}$$

The system of equations (1-3) is solved with both OpenFOAM and FEMuS in order to see how each code solution is affected by the coupling procedure, with respect to the uncoupled case. We simulate two different coupling situations, that are from FEMuS to OpenFOAM and from OpenFOAM to FEMuS.

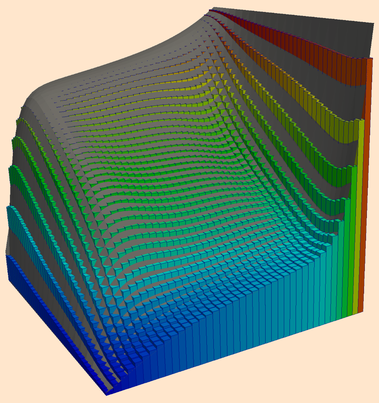




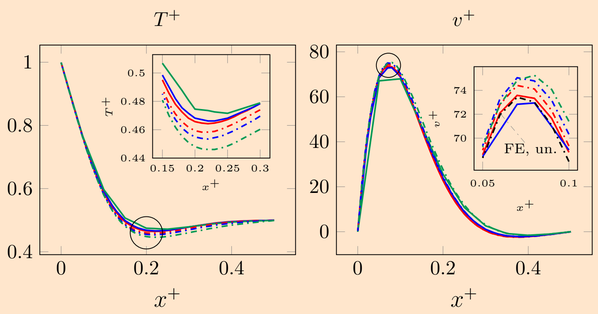
	v^*_{max} on A - B	
Grid size	FEMuS	OpenFOAM
20×20	73.51	67.99
40×40	73.48	73.93
80×80	73.48	73.98

Uncoupled case: maximum value of OpenFOAM non dimensional velocity component v is higher than the FEMuS one. Temperature field is higher near hot wall and colder near cold wall. This behavior leads to an increased velocity field obtained from OpenFOAM solution.

Coupling case A: FEMuS temperature field in OpenFOAM buoyancy source term. For this case we report the results obtained with a FEMuS grid 20×20 and a 40×40 OpenFOAM grid. *What we expect: to obtain a smaller value of v^*_{max}, as the new OpenFOAM buoyancy source term should be smaller than the one of the uncoupled case.* Two different strategies: A1) interpolation of FEMuS T on OpenFOAM grid and then generation of piecewise T field and A2) generation of piecewise T field on FEMuS grid and then interpolation on OpenFOAM grid.



v^*_{max} on A-B	FE	OF un.	OF A1	OF A2
	73.51	73.93	73.24	75.76



Coupling case B: OpenFOAM temperature field in FEMuS buoyancy source term. Here we report the results obtained using three different grids, for each code, namely 20×20, 40×40 and 80×80. *We expect to obtain a velocity field, from FEMuS solution, higher, in magnitude, than the one obtained for the uncoupled case.* The coupling procedure involves a piecewise field interpolation from OpenFOAM 2D equivalent grid to FEMuS mesh. In the plotted results the FEMuS grid is 80×80. It can be seen that by increasing the OF grid resolution the solutions converge.

L^2 norm of velocity field

Grid	OF	FE un.	FE grid, coupled 20×20	40×40	80×80
20×20	0.06761	0.06470	0.06897	0.06898	0.06898
40×40	0.06650	0.06470	0.07057	0.06729	0.06729
80×80	0.06572	0.06470	0.07019	0.06692	0.06610

[a] D. Cerroni, R. Da Vià, S. Manservisi and F. Menghini *Multiscale simulation of a power-cooling system*, AIP Conference Proceedings, 2017, 1863

[b] A. Chierici, L. Chirco, R. Da Vià, S. Manservisi and R. Scardovelli *A multiscale numerical algorithm for heat transfer simulation between multidimensional CFD and monodimensional system codes*, 35-th UIT conference, 2017

[c] L. Chirco, R. Da Vià and S. Manservisi *Natural convection in a squared cavity via a numerical coupling between a FEM code and OpenFOAM*, 15-th ICNAAM conference, 2017

Conclusions

In the present work we realized a numerical code coupling between our inhouse finite element code FEMuS and the opensource finite volume code OpenFOAM. The work led to the development of coupling routines that are not code-specific, as they use the MED data format. These routines can then be used in general, when the MED libraries are implemented in a numerical code. We tested the code coupling in both directions, showing, in each case, which can be the best configuration, in terms of grid resolution and data manipulation.

In future work we will use OpenFOAM as a mean for the calculation of Reynolds stresses, as they are required in the formulation of an anisotropic turbulence model for heat transfer.

IL PRESENTE MATERIALE È RISERVATO AL PERSONALE DELL'UNIVERSITÀ E NON PUÒ ESSERE UTILIZZATO AI TERMINI DI LEGGE DA ALTRE PERSONE O PER FINI NON ISTITUZIONALI