14th World Congress on Computational Mechanics (WCCM)
ECCOMAS Congress 2020)
Virtual Congress: 11-15 January 2021
F. Chinesta, R. Abgrall, O. Allix and M. Kaliske (Eds)

A FINITE ELEMENT PENALIZED DIRECT FORCING METHOD TO TAKE INTO ACCOUNT INFINITELY THIN IMMERSED BOUNDARIES IN A DILATABLE FLOW

GEORIS BILLO[1]*, MICHEL BELLIARD[1] AND PIERRE SAGAUT[2]

[1] French Alternative Energies and Atomic Energy Commission (CEA), DES, IRESNE, DER, SESI, LEMS
Cadarache
F-13118 Saint-Paul-Lez-Durance, France

[2] Aix-Marseille University, CNRS, Centale Marseille
38 rue Joliot-Curie
13451 Marseille, France

* Presenting and corresponding author, georis.billo@cea.fr

Key words: Immersed Boundary Method, Finite Elements Method, Thermalhydraulics, Computational Fluid Dynamics, Projection methods for solving Navier-Stokes equations

Abstract. In the framework of the new passive safety systems developed by the French Atomic Energy Commission (CEA) for the second and third generations of nuclear reactors, a numerical simulation tool capable of modeling thin inflow obstacles is needed [1]. Considering its future use in shape optimization and thermalhydraulics safety studies, the tool must be the fastest, the most accurate and the most robust possible.

The aforementioned context has lead to the Computational Fluid Dynamics (CFD) modeling we are currently developing. For now, it involves a projection scheme to solve the dilatable Navier-Stokes equations and, to take into account obstacles, an adaptation of the Penalized Direct Forcing (PDF) method [2] – a technique whose characteristics inherit from both penalty [3] and Immersed Boundary Method (IBM) [4] – to a Finite Element (FE) formulation. This first modeling offers two variants : one in which the velocity imposed at the vicinity of an obstacle is constant and another in which it is linearly interpolated using properties of the considered immersed boundary (normal vector, barycenter, characteristic function) and the FE basis functions. The results obtained via those two variants, for laminar flow, are in good agreement with analytical and experimental data. However, when compared to each other, it appears that the interpolation of the velocity imposed at the vicinity of the immersed boundary increases the mesh convergence order – which is very interesting, in term of accuracy/computation time ratio.

Some enhancements of the tool are also considered, mainly related to turbulence modeling. Indeed, the interpolating process, instead of being linear, could follow a turbulent wall law.

1 INTRODUCTION

The increasing performance and safety requirements for the third generation of nuclear reactors led to new research and development studies. In this context, new innovative passive safety systems are designed to prevent or soften potential accidental or incidental situations. Some of those systems involve thin rigid motionless fins to guide the flow (*cf.* Figure 1.a). The pressure drop, a quantity of interest for higher scales (*i.e.* reactor design), induced by those devices is greatly influenced by the shape of the fins. Thus, in order to carry out shape optimization and safety studies, a fast and robust numerical tool, able to faithfully model the new passive safety systems, is needed. For more information about the background, the interested reader can refer to [1]. The aimed applications involve a subsonic compressible two-phase flow in non-equilibrium state. As a first approach, the Homogeneous Equilibrium Model could be considered to preserve low computational times while taking into account two-phase aspects. However, for purpose of clarity, only one-phase Navier-Stokes equations are considered in this paper. From this point of view, projection schemes are often used to deal with those equations [5] at low Mach number. To take into account the effect of the complex geometry of the fins on the flow, while preserving simple grids and short computation times, an immersed boundary approach has been chosen : the Penalized Direct Forcing (PDF) method, developed by M. BELLIARD *et al.* [2] in a Finite Difference context, whose characteristics inherits from both penalty [3] and direct forcing [6] techniques. We adapted this method to a Galerkin Finite Element (FE) formulation in order to interpolate the velocity imposed by the immersed boundaries in its vicinity using the FE discrete basis. At last, the flow regime, together with the complex geometry of the devices, also indicate that turbulence will be a major phenomenon. However, this immersed boundary approach falls outside the scope of standard applications of usual turbulence models, requiring a deep investigation. Thus, it will not be treated in this paper. The first section of this document briefly describes the model we are currently developing while the second shows and explain some simulation results.

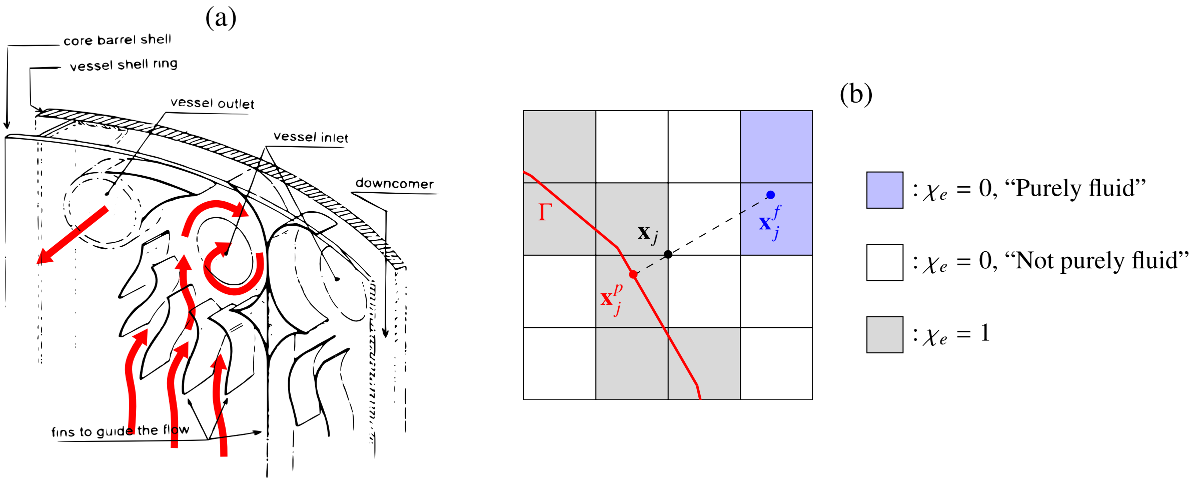

Figure 1: (a): Schematic representation of the flow limiter in an accidental situation. (b): Schematic representation of the interpolation stencil with $\mathbf{x}_j$ a grid node, $\mathbf{x}_j^p$ its approximate projection on the immersed obstacle Γ and $\mathbf{x}_j^f$ a point which is not directly influenced by the immersed obstacle.

2 DESCRIPTION OF THE PROPOSED MODEL

2.1 Governing equations

In the method we developed, the one-phase Navier-Stokes equations are considered:

$$\begin{cases} \partial_t(\rho\mathbf{u}) + \nabla\cdot(\rho\mathbf{u}\otimes\mathbf{u} - \bar{\bar{\sigma}} + \bar{\bar{I}}p) = \rho\mathbf{g} & \text{on } \Omega \\ \partial_t(\rho) + \nabla\cdot(\rho\mathbf{u}) = 0 & \text{on } \Omega \\ +\text{Boundary Conditions} & \text{on } \partial\Omega \\ +\text{Initial Conditions} & \text{on } \Omega \end{cases} \tag{1}$$

with $\bar{\bar{I}} \in \mathbb{R}^{3\times 3}$ the identity matrix, ρ the fluid density, p the pressure, $\mathbf{u}$ the fluid velocity, $\mathbf{g}$ the gravity vector and $\bar{\bar{\sigma}}$ the viscous stress tensor.

2.2 Time discretization

Our approach involves a quasi-implicit Euler scheme coupled with a three-step projection algorithm [5]. Those three steps are the following (with δt the time step):

1. *Prediction*: only time, inertia, mechanical stress and source terms are considered. An intermediate velocity, called the predicted velocity and denoted $\mathbf{u}^*$, is computed:

$$\frac{\rho^n}{\delta t}\mathbf{u}^* + \nabla\cdot\left(\rho^n\mathbf{u}^n\otimes\mathbf{u}^* - \bar{\bar{\sigma}}^* + \bar{\bar{I}}p^n\right) = \rho^n\mathbf{g} + \frac{\rho^n}{\delta t}\mathbf{u}^n \tag{2}$$

2. *Projection*: the pressure field is computed using the predicted velocity and mass balance equation – as we mainly consider steady regimes and use a standard projection scheme, the time derivative of ρ is neglected:

$$\Delta\phi^{n+1} = \frac{1}{\delta t}\nabla\cdot(\rho^n\mathbf{u}^*) \quad \text{with} \quad \phi^{n+1} = p^{n+1} - p^n \tag{3}$$

3. *Correction*: the velocity is computed using the pressure corrector gradient:

$$\mathbf{u}^{n+1} = \mathbf{u}^* - \frac{\delta t}{\rho^n}\nabla\phi^{n+1} \tag{4}$$

A Picard algorithm is actually implemented to make the advective term quasi-implicit. ρ^n is computed using the variables taken at the previous step of the Picard algorithm to fill an equation of state.

2.3 Penalized Direct Forcing method

To take into account infinitely thin in-flow obstacles, denoted Γ, we use the PDF method. It inherits from the Immersed Boundary Method developed by C.S. PESKIN [7] (reviewed by J. MOHD-YUSOF [6] in the Direct Forcing technique) because it consists in adding a new source term – which can be seen as a backmoving force – in the governing equations. However, it also involves a penalty parameter (*i.e.* a numerically infinite stiffness), inspired by the work of P. ANGOT *et al.* [3]. Finally, as shown by many ([8] and references within), this term, when used in a fractional-step algorithm, needs to be split between the prediction and projection equations to ensure consistency:

$$\begin{aligned} \frac{\rho^n}{\delta t}(\mathbf{u}^* - \mathbf{u}^n) + \nabla \cdot \left(\rho^n \mathbf{u}^n \otimes \mathbf{u}^* - \bar{\bar{\sigma}}^* + \bar{\bar{I}} p^n\right) &= \rho^n \mathbf{g} + \mathbf{f}_1^{n+1} && (5) \\ \frac{\rho^n}{\delta t}\left(\mathbf{u}^{n+1} - \mathbf{u}^*\right) + \nabla \phi^{n+1} &= \mathbf{f}_2^{n+1} && (6) \end{aligned}$$

with:

$$\mathbf{f}_1^{n+1} := \chi \frac{\rho^n}{\eta \delta t}\left(\mathbf{u}_\Gamma^{n+1} - \mathbf{u}^*\right) \tag{7}$$

$$\mathbf{f}_2^{n+1} := \chi \frac{\rho^n}{\eta \delta t}\left(\mathbf{u}^* - \mathbf{u}^{n+1}\right) \tag{8}$$

where $0 < \eta \ll 1$ is the penalty parameter, χ is the characteristic function of the obstacles and $\mathbf{u}_\Gamma^{n+1}$ is the velocity imposed by the obstacles (the value of the Dirichlet immersed boundary condition). When gathering terms together, we obtain:

$$\frac{\rho^n}{\delta t}\left(1 + \frac{\chi}{\eta}\right)\mathbf{u}^* + \nabla \cdot \bar{\bar{\Sigma}}^{n,*} = \rho^n \mathbf{g} + \frac{\rho^n}{\delta t}\left(\mathbf{u}^n + \frac{\chi}{\eta}\mathbf{u}_\Gamma\right) \tag{9}$$

$$\frac{\rho^n}{\delta t}\mathbf{u}^{n+1} + \frac{\eta}{\eta + \chi}\nabla \phi^{n+1} = \frac{\rho^n}{\delta t}\mathbf{u}^* \tag{10}$$

Thus, our method can be seen as a contribution to the implicit part of the time term as well as an added source term in the prediction equation and an added diffusivity in the projection equation. One can note that those forcing terms are discontinuous among the immersed boundary Γ. This implies that the one associated to the projection equation has to be added before applying the divergence and substituting the mass equation – the actual substitution occurs at a discrete level.

2.4 Finite Element discretization

A Galerkin Finite Element Method (FEM) is used to achieve the space discretization of the problem [9]. This choice is motivated by the possibility of using the FE basis (potentially enriched) to interpolate the fluid velocity at the vicinity of an obstacle (*cf.* following section) and, also, by the existence of a house-code [10] including immersed-boundary-like features. It is also worth noting that the general idea of the PDF method does not depend strongly on the space discretization.

In short, the computation domain is divided in hexahedral elements. For the velocity, a $\mathbb{Q}_1$ basis (*i.e.* trilinear decomposition at nodes) is used while, for pressure, a $\mathbb{Q}_0$ basis (*i.e.* the discrete pressure field is constant by element) is used. This pair of elements is known to be unstable but we kept it for two main reasons: it counts very few degrees of freedom per element (so it is quite fast) and the pressure instabilities are soften when the diffusivity is high enough – *i.e.* in laminar cases or when the turbulent viscosity is high, which is a case of interest for nuclear passive safety systems. The discrete version of the characteristic function of Γ is denoted χ_e and is considered as $\mathbb{Q}_0$ (*i.e.* constant by element). If an element numbered e is crossed by the immersed obstacle $\chi_e = 1$, else $\chi_e = 0$.

Using this pair of elements, we can write the discrete versions of equations (9) and (10) in an element numbered $e \in \mathbb{N}$:

$$\left(\frac{1}{\delta t}\mathbf{M}_e\rho_e^n\left[1+\frac{\chi_e}{\eta}\right]-\mathbf{D}_e\mu_e^n+\mathbf{N}_e\rho_e^n\lambda^n\right)\lambda^* = \frac{1}{\delta t}\mathbf{M}_e\rho_e^n\left(\lambda^n+\frac{\chi_e}{\eta}\lambda_\Gamma^{n+1}\right)+\mathbf{B}_e p_e^n+\mathbf{M}_e\rho_e^n\lambda_g \quad (11)$$

$$\mathbf{B}_e^T\mathbf{M}_e^{-1}\frac{\eta}{\eta+\chi_e}\mathbf{B}_e\phi_e^{n+1} = -\frac{1}{\delta t}\mathbf{B}_e^T\rho_e^n\lambda^* \quad (12)$$

with $\mathbf{M}_e$ (resp. $\mathbf{D}_e$, $\mathbf{N}_e$ and $\mathbf{B}_e$) the lumped mass [11] (resp. diffusivity, advective and gradient-divergence) elemental matrix, λ^n (resp. λ^* and λ_g) the components of the velocity (resp. predicted velocity and gravity vector) in the $\mathbb{Q}_1$ finite elements basis, ϕ_e^{n+1} (resp. ρ_e^n) the discrete pressure corrector (resp. fluid density) and, finally, λ_Γ^{n+1} the decomposition of the imposed velocity $\mathbf{u}_\Gamma^{n+1}$ in the $\mathbb{Q}_1$ finite elements basis – those values are interpolated as explained in the following section.

2.5 Velocity interpolation at the vicinity of the immersed boundary

For each node numbered j and belonging to an element numbered e such as $\chi_e = 1$ (*i.e.* crossed by an obstacle), an approximate geometric projection on the obstacle $\mathbf{x}_j^p$ is computed. This also gives an approximate normal vector $\mathbf{n}_j$ which is used to determine a purely fluid point $\mathbf{x}_j^f$ – *i.e.* a point belonging to an element whose nodes are not directly impacted by the immersed boundary (*cf.* Figure 1.b).

The velocity at $\mathbf{x}_j^p$ is directly given by the immersed Dirichlet boundary condition and the velocity at $\mathbf{x}_j^f$ is computed using the finite element basis functions. With those values and points, we are able to get an approximate value of the velocity at node j by linear interpolation and to use it as the imposed velocity $\mathbf{u}_\Gamma$ presented in equation (5) – in this section, the notation of time indices is omitted for purpose of readability keeping in mind that this process occurs at each time step.

The velocity at $\mathbf{x}_j^p$, denoted $\mathbf{u}_j^p$, is directly given by the immersed Dirichlet boundary condition and the velocity at $\mathbf{x}_j^f$, denoted $\mathbf{u}_j^f$, is computed using the finite elements basis functions. With those value and points, we are able to create an approximate value of the velocity at node j by linear interpolation and use it as the imposed velocity presented in equation (9):

$$\mathbf{u}(\mathbf{x}=\mathbf{x}_j) \approx \mathbf{u}_j = \mathbf{u}_j^p + \frac{\mathbf{u}_j^f-\mathbf{u}_j^p}{|\mathbf{x}_j^f-\mathbf{x}_j^p|}|\mathbf{x}_j-\mathbf{x}_j^p| = \mathbf{u}_\Gamma \quad (13)$$

3 NUMERICAL RESULTS

3.1 Convergence order studies

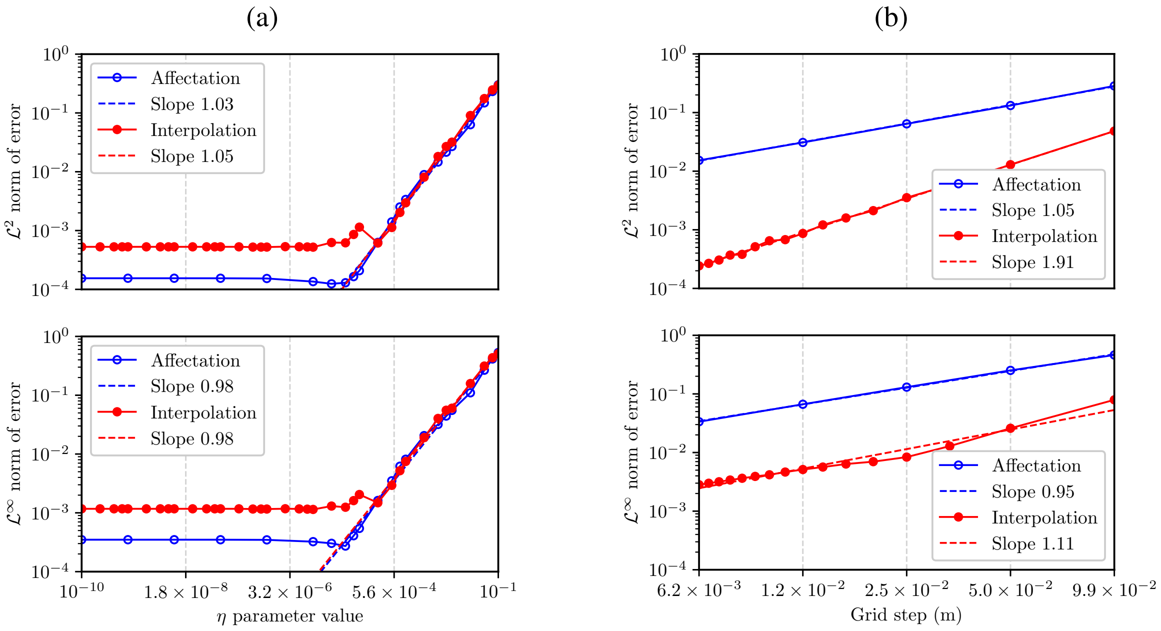

Figure 2: (a): the evolution of relative norm of error between the computed velocity along the channel (aligned with the mesh) and its analytical value with respect to the penalty parameter η in the case of a 2D Poiseuille flow. (b): the evolution of the relative norm of error between the computed velocity magnitude and its analytical value with respect to the grid step in the case of a 2D Poiseuille flow (channel tilted of a 30 degrees angle).

Table 1: Aerodynamic coefficients related to the steady laminar flow around a static circular cylinder ($Re = 20$).

		$2r/h$		References						
		10	20	[12]	[13]	[14]	[15]	[16]	[17]	[18]
C_d	Ass.	2.245	2.154	2.059	2.03	2.02	2.06	2.06	2.00	2.09
	Int.	2.142	2.075	2.054						
L_w	Ass.	1.297	1.166	0.925	0.92	0.90	0.94	0.93	0.91	–
	Int.	0.749	0.901	0.900						

The behavior of our method with respect to the penalty parameter η has been studied in the case of a 2D Poiseuille flow – the channel is aligned with the grid and the immersed boundary are conforming to the element faces in order to minimize the spatial error and focus on the penalty error. Figure 2.a) shows the $\mathcal{L}^2$ and $\mathcal{L}^\infty$ relative norms of error of the component of the velocity oriented along the channel obtained from several values of the penalty parameter. The blue curve corresponds to results obtained with a variant of the method in which the immersed Dirichlet boundary condition value is directly assigned to the imposed velocity $\mathbf{u}_\Gamma^{n+1}$. The red curve correspond to the results obtained by using the velocity

Table 2: Aerodynamic coefficients related to the steady laminar flow around a rotating circular cylinder ($Re = 20$).

		$2r/h$		References				
		10	20	[12]	[19]	[20]	[21]	[22]
C_d	Ass.	2.057	1.984	1.8608	1.888	1.85	1.925	2.000
	Int.	2.002	1.913	1.8679				
C_l	Ass.	3.181	3.032	2.9419	2.629	2.75	2.617	2.740
	Int.	2.970	2.868	2.7745				
$\alpha(°)$	Ass.	57.11	56.79	57.68	54.31	56	53.66	53.87
	Int.	56.02	56.29	56.05				

Table 3: Aerodynamic coefficients related to the unsteady laminar flow around a static circular cylinder ($Re = 100$).

	$2r/h$		References						
	20	40	[12]	[20]	[16]	[14]	[23]	[24]	[25]
$\overline{C_d}$	1.338	1.319	1.347	1.337	1.340	1.340	1.350	1.317	1.376
$C_d{}'$	0.010	0.012	0.009	0.009	0.009	0.011	0.012	0.009	0.010
$C_l{}'$	0.276	0.285	0.326	0.326	0.333	0.315	0.303	0.349	0.339
St	0.145	0.161	0.165	0.165	0.166	0.164	0.167	0.170	0.170

interpolation technique described just before. In both cases, the numerical order of convergence is close to 1, which is consistent with the conclusion presented by P. ANGOT *et al.* in [3]. Mesh convergence studies have also been carried out: one involving the 2D tilted Poiseuille flow (*i.e.* the channel is not aligned with the mesh, it is tilted of an angle θ), one involving the Taylor-Couette flow and, finally, another one involving the flow around a circular cylinder – for purpose of concision, only the results obtained from the Poiseuille flow are shown but the conclusions apply to all cases. Figure 2.b) shows the evolution of the relative norm of error with respect to the grid in the case of 2D Poiseuille flow with a channel tilted of a 30° angle. The same variants of the methods are compared in terms $\mathcal{L}^2$ and $\mathcal{L}^\infty$ norm. As expected from the theory and the literature, a spatial order of convergence of almost 2 is reached when the imposed velocity is interpolated.

3.2 Global quantities studies

The flow around static and rotating circular cylinders have been studied at different Reynolds numbers – Figure 3 shows an example of the computed streamlines and isobars in the case of a static cylinder at Reynolds $Re = 20$. Global quantities such as drag and lift coefficients – C_d and C_l with, in unsteady cases (*i.e.* $Re = 100$), their mean values denoted with a bar and their fluctuations denoted with an apostrophe. The angle between the aerodynamic force and the horizontal axis (α) and the Strouhal number (St) have been computed and compiled in Tables 1, 2, 3 and 4 – r refers to the cylinder radius and h to the space step so $2r/h$ is an estimation of the number of elements included within the diameter of the cylinder. Those

Table 4: Aerodynamic coefficients related to the unsteady laminar flow around a rotating circular cylinder ($Re =$ 100).

	$2r/h$		References			
	20	40	[12]	[20]	[19]	[26]
$\overline{C_d}$	1.143	1.097	1.12	1.1080	1.1890	1.0979
$C_d{}'$	0.098	0.091	0.11	0.0986	0.1195	0.0988
$\overline{C_l}$	2.634	2.573	2.51	2.5040	2.4050	2.4833
$C_l{}'$	0.285	0.291	0.37	0.3616	0.4427	0.3603
St	0.151	0.165	0.165	0.1658	0.1732	0.1650

Table 5: Aerodynamic coefficients related to the laminar flow past a NACA airfoil. Steady regimes are not recovered for lines colored in gray.

θ	C_d		C_l	
	Interpolation	[27]	Interpolation	[27]
0	0.0563	0.0555	0.0094	0.0000
1	0.0571	0.0559	0.0092	0.0184

values are in good agreement with the literature, as shows the reference values, even if some minor discrepancies are noticed (lift-related coefficients and Strouhal number) in unsteady cases ($Re = 100$). Those discrepancies might be explained by the lack of the particular derivative of the velocity in the approximation of the aerodynamic force – this assumption will be tested in upcoming works.

The laminar flow past a NACA0012 airfoil has also been studied and the results have been compared to data coming from a different simulation [27]. Figure 4 gives a representation of the streamlines and velocity magnitude of the flow past a NACA0012 airfoil with an angle of attack of 0°. Shapes and recirculation length are really similar to the one presented in [27] (about 0.17 m in both cases). Concerning aerodynamic coefficients, we gathered the steady regime results obtained for angles of attack

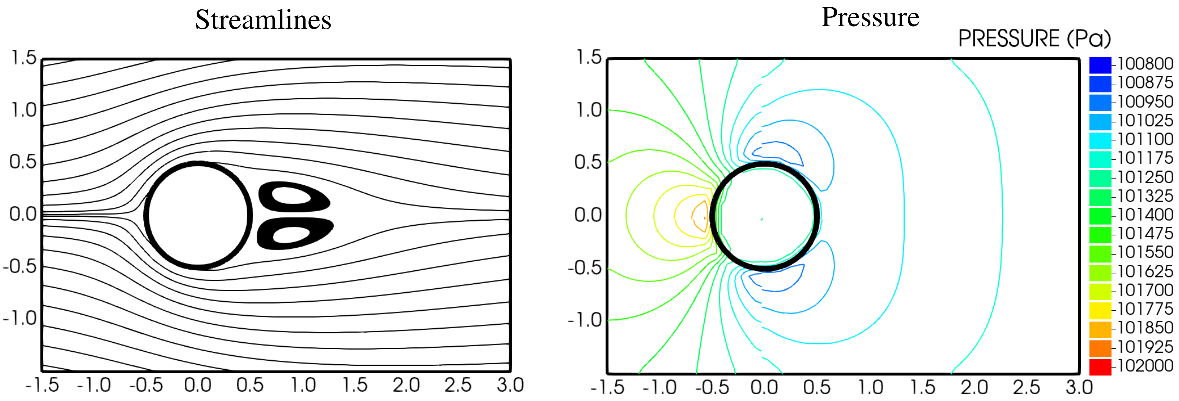

Figure 3: Streamlines and isobars of the steady flow around a static cylinder ($Re = 20$)

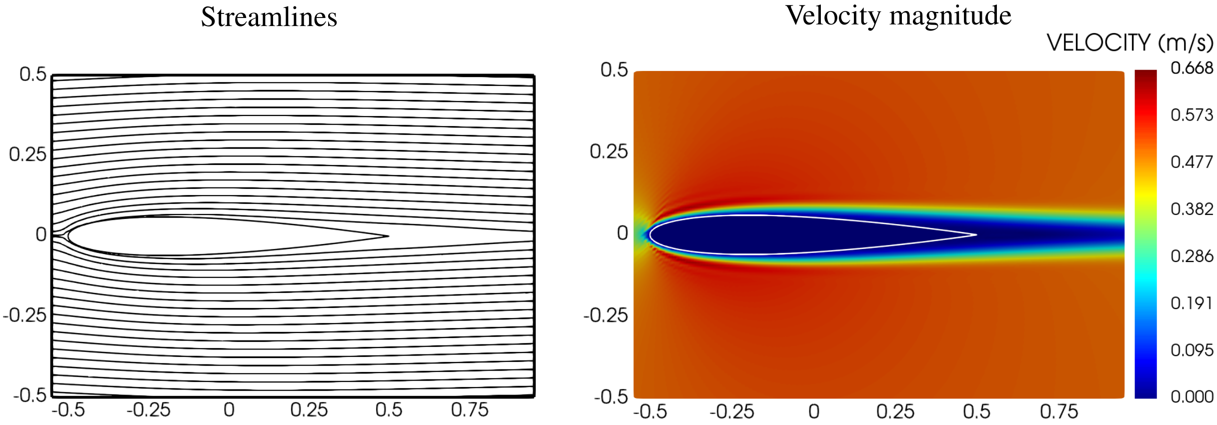

Figure 4: Computed streamlines and velocity magnitude map in the case of the flow past a NACA0012 airfoil with an angle of attack of 0°.

of 0 and 1° in Table 5. Taking into account that an angle difference of 1° can not be exactly reproduced using our immersed boundary geometrical model, we conclude that the aerodynamic coefficients are in a good agreement. However, for cases involving an angle of attack superior to 1°, our method was not able to recover steady state solutions so only the mean values of the aerodynamic coefficients are shown in Table 5. This can be induced by several phenomena such as:

- Residuals being too large because of high convergence criteria (as suggested by R.C. SWANSON),
- Boundary effects (the size of the used domain is smaller than the one used for the circular cylinder cases),
- Grid being too coarse and not adapted to the problem (*i.e.* staircase description of the obstacle which could decrease the transition between steady / unsteady regimes in terms of Reynolds number),
- The use of super-convergence algorithm in the work of R.C. SWANSON.

4 CONCLUSION AND PERSPECTIVES

In this document, the main theoretical aspects of the Penalized Direct Forcing method (an IBM whose characteristics inherit from both penalty and direct forcing methods) are briefly explained: it involves solving the Navier-Stokes equations using a fractional-step algorithm and an additional source term representative of the modeled obstacles (some kind of backmoving force). It was initially developed and tested in the context of a space-time discretization based on a finite difference scheme. Here, we specify and test the method using a Galerkin finite element discretization with a lumping of the mass matrix. Moreover, an enhancement of the method is also proposed : it consists in a directional interpolation of the velocity in the vicinity of the obstacles. Validation test cases are presented, as well as numerical results. The steady laminar Poiseuille and Taylor-Couette flows, for which analytical solutions are known, allowed us to carry out numerical convergence studies, with respect to the penalty parameter and the grid step. It is worth noting that the new linear interpolation technique reduces the difference between the computation results and the analytical solution while increasing the spatial order of convergence (almost reaching two for the velocity in $\mathcal{L}^2$ norm). Another validation case presented in this document is the laminar flow around a circular cylinder. It is declined in four configurations: steady regime with static cylinder, steady regime with rotating cylinder, unsteady regime with static cylinder and unsteady regime with rotating cylinder. Even if no analytical solution is available for the flow around a cylinder, a mesh convergence study is presented (the numerical results computed with the finest grid are considered as a reference) and shows, once again, that the interpolation technique increases the spatial order of the method. The results obtained with our approach is also compared to experimental data and other simulations via quantities such as aerodynamic coefficients and the Strouhal number on the circular cylinder test case and a laminar flow past a NACA0012 airfoil test case. Globally, the values obtained are in good agreement with the literature which provides valuable validation data.

In the near future, two developments are considered. The first one consists in interpolating the normal component of the pressure gradient in the vicinity of the immersed boundaries. Indeed, the velocity is interpolated using the presented methodology but not the pressure gradient (nor pressure corrector). Yet, the pressure gradient appears in the Navier-Stokes prediction equation. An idea to deal with this issue could be to interpolate the normal pressure gradient or corrector (as we consider $\nabla p^{n+1} \cdot \mathbf{n} = 0$ as immersed boundary condition for the pressure) using the same methodology as the one used of the velocity. The second one involves turbulence modeling: the idea is to extend the interpolation process to turbulent wall laws (*i.e.* power wall law [28]). Thus, we will be able to carry out turbulent simulations of the flow limiter involving well-established turbulent models as the wall laws, RANS and LES models.

REFERENCES

[1] M. Belliard. Numerical Modeling of an in-vessel flow limiter using an Immersed Boundary Approach. *Nuclear Engineering and Design*, 330:437–449, 2018.

[2] M. Belliard and C. Fournier. Penalized Direct Forcing and projection schemes for Navier-Stokes. *C.R. Acad. Sci. Paris, Ser. I*, 348:1133–1136, 2010.

[3] P. Angot, C.H. Bruneau, and P. Fabrie. A penalization method to take into account obstacles in incompressible viscous flows. *Numerische Mathematik*, 81:497–520, 1999.

[4] C.S. Peskin. *Flow Patterns around heart vavles: A digital computer method for solving the equations of motion*. PhD thesis, Albert Einstein College of Medecine, 1972.

[5] D. L. Brown, R. Cortez, and M. L. Minion. Accurate projection methods for the incompressible Navier-Stokes equations. *Journal of Computational Physics*, 168:464–499, 2001.

[6] J. Mohd-Yusof. Combined Immersed-Boundary/B-spline methods for simulations of flow in complex geometries. *Annual Research Briefs*, pages 317–327, 1997.

[7] C.S. Peskin. The immersed boundary method. *Acta Numerica*, 11:479–517, 2002.

[8] R.D. Guy and D.A. Hertenstgine. On the accuracy of direct forcing immersed boundary methods with projection methods. *Journal of Computational Physics*, 229:2479–2496, 2010.

[9] V. Girault and P.A. Raviart. *Finite Element Methods for Navier-Stokes Equations*. Springer-Verlag, 1986.

[10] M. Grandotto and P. Obry. Calcul des coulements diphasiques dans les échangeurs par une méthode

aux éléments finis. *Revue européenne des éléments finis*, 5:53–74, 1996.

[11] R. L. T. Bevan and P. Nithiarasu. Accelerating incompressible flow calculations using a quasi-implicit scheme: local and dual time stepping approaches. *Computational Mechanics*, 50:687–693, 2012.

[12] C. Introïni, M. Belliard, and C. Fournier. A second order penalized direct forcing for hybrid Cartesian/immersed boundary flow simulations. *Computers & Fluids*, 90:21–41, February 2014.

[13] T. Ye, R. Mittal, H.S. Udaykumar, and W. Shyy. An accurate cartesian grid method for viscous incompressible flows with complex immersed boundaries. *Journal of Computational Physics*, 156(2):209 – 240, 1999.

[14] Jung-Il Choi, Roshan C. Oberoi, Jack R. Edwards, and Jacky A. Rosati. An immersed boundary method for complex incompressible flows. *Journal of Computational Physics*, 224(2):757 – 784, 2007.

[15] Kunihiko Taira and Tim Colonius. The immersed boundary method: A projection approach. *Journal of Computational Physics*, 225(2):2118 – 2137, 2007.

[16] Mark N. Linnick and Hermann F. Fasel. A high-order immersed interface method for simulating unsteady incompressible flows on irregular domains. *Journal of Computational Physics*, 204(1):157 – 192, 2005.

[17] Bengt Fornberg. A numerical study of steady viscous flow past a circular cylinder. *Journal of Fluid Mechanics*, 98(4):819–855, 1980.

[18] D. J. Tritton. Experiments on the flow past a circular cylinder at low reynolds numbers. *Journal of Fluid Mechanics*, 6(4):547–567, 1959.

[19] Meng-Hsuan Chung. Cartesian cut cell approach for simulating incompressible flows with rigid bodies of arbitrary shape. *Computers & Fluids*, 35(6):607 – 623, 2006.

[20] D. Stojkovi, M. Breuer, and F. Durst. Effect of high rotation rates on the laminar flow around a circular cylinder. *Physics of Fluids*, 14(9):3160–3178, 2002.

[21] D.B. Ingham and T. Tang. A numerical investigation into the steady flow past a rotating circular cylinder at low and intermediate reynolds numbers. *Journal of Computational Physics*, 87(1):91 – 107, 1990.

[22] H.M. Badr, S.C.R. Dennis, and P.J.S. Young. Steady and unsteady flow past a rotating circular cylinder at low reynolds numbers. *Computers & Fluids*, 17(4):579–609, 1989.

[23] P.H. Chiu, R.K. Lin, and Tony W.H. Sheu. A differentially interpolated direct forcing immersed boundary method for predicting incompressible navier-stokes equations in time-varying complex geometries. *Journal of Computational Physics*, 229(12):4476 – 4500, 2010.

[24] Yoann Cheny and Olivier Botella. The ls-stag method: A new immersed boundary/level-set method for the computation of incompressible viscous flows in complex moving geometries with good conservation properties. *Journal of Computational Physics*, 229(4):1043 – 1076, 2010.

[25] C. Ji, A. Munjiza, and J.J.R. Williams. A novel iterative direct-forcing immersed boundary method and its finite volume applications. *Journal of Computational Physics*, 231(4):1797 – 1821, 2012.

[26] S. Kang, H. Choi, and S. Lee. Laminar flow past a rotating circular cylinder. *Physics of Fluids*, 11(11):3312–3321, 1999.

[27] R.C. Swanson and S. Langer. Steady-state laminar flow solutions for NACA 0012 airfoil. *Computers and Fluids*, 126:102–128, 2016.

[28] S. Wilhelm, J. Jacob, and P. Sagaut. An explicit power-law-based wall model for lattice boltzmann method-reynolds-averaged numerical simulations of the flow around airfoils. *Physics of Fluids*, 30:065111, 2018.