20th Australasian Fluid Mechanics Conference
Perth, Australia
5-8 December 2016

# Flows in metal foams using Immersed Boundary Method

**L. Manueco[1], T. Sarfati[1] E. Sauret[2] and Y. D'Angelo[3]**

[1]Institut National des Sciences Appliqués de Rouen
Normandie University, 76800 Saint-Etienne-du-Rouvray, France

[2]School of Chemistry, Physics and Mechanical Engineering,
Queensland University of Technology, Brisbane , Queensland, 4000, Australia

[3]Laboratoire de Mathématiques J.A. Dieudonné,
Université de Nice Sophia Antipolis, CNRS UMR 7351, Nice, France

## Abstract

Porous media are nowadays common in many thermal or mechanical engineering applications such as heat exchangers. A recent example of such porous media used in engineering is metal foams. These foams increase thermal transfer thanks to their enhanced exchange surface (high surface to volume ratio) while staying light-weighted. However, due to the complexity of the geometry of these foams the flow characterisation at the pore-level is challenging. In addition, turbulence might also play an important role inside the porous metal foams and affect both the fluid flow and particle transport. The aim of this paper is to investigate the suitability of Immersed Boundary (IB) method for the simulation of flows in metal foams. For that purpose, the IB approach is compared against a standard CFD solver based on a Body-Fitted (BF) grid and against experimental data. First a flow around a cylinder from Reynolds numbers (Re) = 20 up to 700 is considered to compare present BF and IB results against experimental results. The simulations on different pore shapes are also performed, as the geometry gets closer to a real metal foam implying a better prediction of the different velocity profiles and fluctuations.

## Introduction

Metal foams are complex metallic porous structures that can be used in heat exchangers. Metal foams enable increased thermal transfer thanks to their enhanced exchange surface (high surface to volume ratio) while staying light-weighted[5]. However, the simulations on those complex structures can be challenging notably due to mesh requirements and the complexity of creating body-fitted grids. In order to overcome this issue, Immersed Boundary method can be applied. The IB approach appeared for the first time in a publication of Peskin [8] in 1972. The main purpose of this first method was to develop an easy way to manage moving boundary in the case of heart valve modelling. Since then many other IB methods [4][6][10] were developed in order to use uniform grids instead of adapted mesh. Immersed Boundary can be generated using different approach which are gathered in two main categories. The first is called continuous forcing approach which imposes a force at the interface fluid-solid in the Navier-Stokes equation. The second is called discrete forcing approach and consists of the direct implementation of the boundary condition around the interface. Due to the relative simplicity of the implementation of moving IB this method is currently used for biological applications at low Reynolds numbers[6]. Porous media were also investigated using immersed methods [11]. Despite the attractiveness of implementation, as it does not require to significantly modify the flow solver, the standard cartesian grid used in IB method, leads to a lack of precision in the boundary layer at medium to high Reynolds numbers that could be solved with a substantial reduction of the cell size [4]. In this study, an IB method based on a discrete forcing is applied to idealized geometries in order to demonstrate the feasibility of this approach for porous metal foams. The IB method is implemented in an incompressible open source solver. The accuracy of the IB method is compared against results from the literature and from simulations using standard OpenFOAM solvers, based on BF (body-fitted) grid. Turbulence in metal foams is expected to appear for Re, based on the pore size of the foams around 300 [3]. The present study is thus limited to Reynolds numbers up to 700. The purpose of the paper is to demonstrate the feasibility of the Immersed Boundary method for porous media. The mesh requirements for both IB and BF approaches are also discussed.

## Numerical Method

The IB method is implemented in OpenFoam (extension 3.2), in which a PISO (Pressure Implicit with Splitting of Operators) incompressible solver is used for the general flow solution. The results of the IB solver are compared against a standard PISO solver also implemented in OpenFOAM using a BF grid. The PISO algorithm has been widely used and validated in the literature [7,10]. All the simulations for the standard and the IB methods were done using exactly the same parameters. The only difference between the two approaches reside in the management of the boundaries. The IB mesh consists of a Cartesian grid, in which three types of cells are used in the computational domain to define the interface between solid and fluid as illustrated in figure 1. The fluid cells correspond to standard cells in which the finite volume PISO solver is applied, while the solid cells represent the solids. The Immersed cell correspond to the cells in which the interface between the solid and the fluid is managed. In the immersed cells, the IB method imposes the

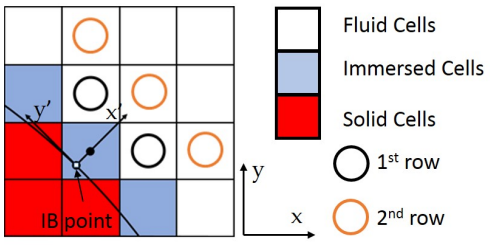


Figure 1: Cells nomenclature in the IB approach

boundary conditions at the solid walls. At the interface solid-fluid, the IB method sets a Dirichlet condition for the velocity and a Neumann condition for the pressure. In order to simulate a real boundary the IB method calculate at each time step a velocity and a pressure for each Immersed cell and then the PISO algorithm uses those values to update the values of the fluid cells. The velocity for the immersed cell is obtained thanks to

the following polynomial interpolation:

$$V_p = C_0(x_p - x_i b) + C_1(y_p - y_i b) + C_2(x_p - x_i b)(y_p - y_i b) + C_3(x_p - x_i b)^2 + C_4(y_p - y_i b)^2 \quad (1)$$

And the pressure for the immersed cell is obtained using a second polynomial interpolation:

$$P_p = A_0 + A_1(n_i b \cdot (\bigtriangledown \phi)_i b)x'_p + A_2(y'_p) + A_3 y'_p x'_p + A_4(x'_p)^2 + A_5(y'_p)^2 \quad (2)$$

The pressure interpolation involves local coordinates where $y'$ is tangent to the immersed surface at the IB point and $x'$ is normal to $y'$ and by the middle of the immersed cell as shown in figure 1. The velocity interpolation keeps the Cartesian coordinates (x,y) as shown in figure 1. For both interpolations, first and second row cells as in figure 1 are used to increase the accuracy of the interpolation. The first row corresponds to the fluid cells sharing an edge with the current immersed cell, while the second row of cells is defined as the fluid cells sharing an edge with the cells from the first row. The number of fluid cells used for the interpolation is consequently related to the geometry near the immersed cell. To determine the $C_i$ and $A_i$ coefficients for the velocity and the pressure (equations (1) & (2)), the two following sets of equations need to be solved, respectively:

$$L = M \cdot C \quad (3)$$

$$E = S \cdot A \quad (4)$$

Where $L$ is the relative velocity of the fluid cells to the immersed boundary and $E$ is the pressure of the fluid cells for the second interpolation. $C$ is the matrix containing the velocity coefficients $C_i$ from equation (1) and $A$ is the matrix containing the pressure coefficients $A_i$ from equation (2). $M$ and $S$ are the linear operators and are easily computed with the relative coordinate of the fluid cells using a weighted least square method[12]. The implementation of the weighted factors allow for a better interpolation of the parameters for the immersed cell. The weighted factors are determined using a cosine weight function for the velocity given as follow:

$$W_i = \frac{1}{2} \cdot (1 + cos(\pi \frac{r_i}{1.1 \cdot r_{max}})) \quad (5)$$

with $r_i$ is the distance between the fluid cell $i$ and the immersed cell, and $r_{max}$ is the larger distance between the immersed cell and the fluid cells. The pressure weighted factors are obtained using the following equation:

$$G_i = 1 - \frac{r_i}{1.1 \cdot r_{max}} \quad (6)$$

Finally, in each immersed cell, the $A_i$ ad $C_i$ coefficients are determined using the weighted factors of each cell:

$$C = (M^T W M)^{-1} \cdot M^T W \cdot L \quad (7)$$

$$A = (S^T G S)^{-1} \cdot S^T G \cdot E \quad (8)$$

All those calculations are computationally inexpensive and can be done rapidly at each time step in order to create moving boundary. However, for a fixed boundary, only $E$ and $L$ change over time and have to be updated at each time step while the other parameters, $W_i$, $G_i$, M and S are only computed once at the beginning of the simulation. Ultimately, the velocity in the immersed cells is scaled to prevent the flux to penetrate the solid boundary. The interpolated pressure in the immersed cells imposes a Neumann condition and is used in the momentum equation to comply with the conservation of momentum.

#### Flow around a cylinder: validation of the IB method

In order to validate the IB approach, a flow around a cylinder is considered. The computational domain was made large enough as seen in figure 2 so that the boundary conditions do not impact on the flow around the cylinder. Zero gradient conditions were imposed at the top and bottom for the pressure and the velocity. At the inlet, a zero pressure gradient and a constant velocity input were imposed. At the outlet, a zero pressure and a zero gradient velocity were imposed. Two different

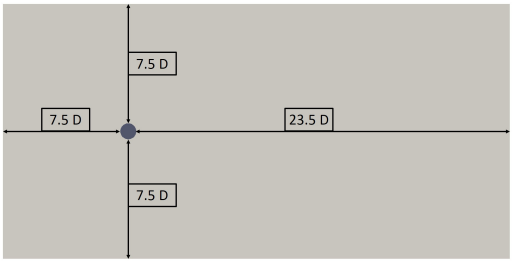


Figure 2: Computational domain with the distance expressed in function of the diameter $D$ of the cylinder with D=0.01 m

regimes of the flow around the cylinder were investigated to validate the model. First, a study on the attached vortices flow regime was realised for Re between 10 and 40. This flow is well documented and has parameters which can be compared easily against the literature. Secondly an unsteady flow around a cylinder was investigated from $100 \leq Re \leq 700$ with the well-known Karman vortex street. The frequency of the vortex shedding is also a well documented parameter in the literature.

##### Attached vortices flow regime

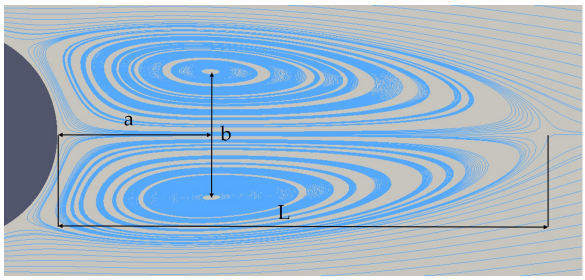


Figure 3: Visualisation of the main vortex parameters

The parameters showed in figure 3 and the angle of separation of the streamline from the contour of the cylinder θ were investigated experimentally by Coutanceau and Bouard[2] but also numerically by Xu[9]. The results of the standard, and immersed solvers were compared against experimental and numerical results. The results presented in table 1 show that the IB and standard solvers are in good agreement with each other and match the published data very well.

##### Von Karman vortex street regime

The Reynolds number was then increased from 100 up to 700 and the results compared against the experimental work of Berger and Wille [1]. The results figure 4 clearly show that the IB solver has mostly the same trend as the standard solver for low Re. However, when the Re increases above 500, the differ-

| Reynolds 20 | L | a | b | θ |
|---|---|---|---|---|
| Coutanceau [2] | 0.93 | 0.33 | 0.46 | 45 |
| Xu [9] | 0.93 | 0.36 | 0.43 | 44 |
| IB solver | 0.96 | 0.36 | 0.45 | 44 |
| Standard solver | 0.94 | 0.34 | 0.44 | 44 |
| Reynolds 40 | L | a | b | θ |
| Coutanceau [2] | 2.13 | 0.76 | 0.59 | 53.8 |
| Xu [9] | 2.24 | 0.72 | 0.6 | 53.8 |
| IB solver | 2.19 | 0.77 | 0.6 | 51 |
| Standard solver | 2.27 | 0.76 | 0.59 | 52 |

Table 1: Comparison of the characteristic parameters with both IB and BF solvers for attached vortices regime

ence between the IB and standard solvers and the experimental results become clearly non-negligible anymore, with about 5% maximum difference between the simulations and the experiments at Re=700. Those difference between the solvers and the experimental data can be explained by the transition from a grid with a mesh size $\triangle x$ equals to $D \cdot Re^{-3/4}$ down to $D \cdot Re^{-1/2}$ at Re above 500. This increase of mesh size avoided the prohibitive simulation time at higher Re. As stated in the introduction, it is expected to reach a turbulent regime in the foam from Re=200-300[3]. This region will thus be correctly captured by the IB approach. It is also important to note that, for the same mesh size at $Re > 500$, the IB results diverge from the standard solver due to the reduction of the stability of the forcing approach in the IB method. This can be explained by the current interpolations implemented in the IB approach in OpenFOAM which require accuracy improvements. For example, the weightings could be adapted in the current version in order to get higher accuracy at higher Reynolds numbers. However, the IB method and standard solver are in almost perfect agreement at low Re as shown in figures 4 & 7 and is thus acceptable for porous media.

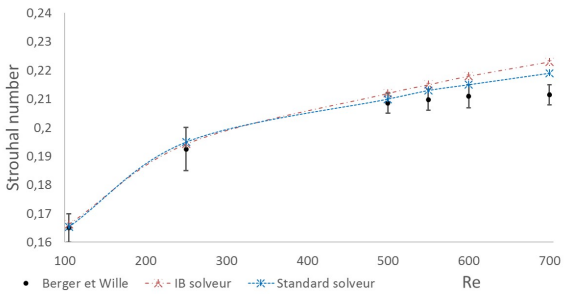


Figure 4: Comparison of the Strouhal value between the standard and IB solver and the experimental results [1]

Extended study at Re=250

As turbulence appears around Re=200-300 in metal foams a detailed study at Re=250 is carried out. The influence of the mesh size (Table 2) on the IB results was evaluated. Figure 5 shows that mesh 1 notably differs from mesh 2 and 3 which are really similar to each other with less than 1% difference. Despite the overall good agreement between mesh 2 and 3 there are some large changes around the IB interface, as shown in figure 6 which presents the relative non-dimensional velocity at the centreline after the cylinder between mesh 1,2 and mesh 3. The error remains above 5% for Mesh 1 until x/D=2 while at only x/D=1, the difference between Mesh 2 and 3 is around 2%. This phenomena could be a significant source of error in confined geometry as porous media. Highly refined mesh (Mesh 3) with $\triangle x$ equal to $D \cdot Re^{-3/4}$ is thus recommended in order to limit this zone of influence of the IB method. Mesh 3 is also adequate to ensure the grid independence of the results. However, for really low Re as in the attached vortices flow regime, a $\triangle x$ equal to $D \cdot Re^{-3/4}$ is not enough to properly capture the IB interface. It is thus recommended to use a minimal mesh size equivalent to the characteristic dimension of the shape divided by 25 in order to have enough Immersed cells to properly emulate the boundary. A mesh study was also conducted for the BF grids with inflation. The same $\triangle x$ as for the IB grid was adopted in order to have a mesh independence of the results.

| Mesh | $\triangle x$ (m) |
|---|---|
| 1 | $3.2 \cdot 10^{-4}$ |
| 2 | $2.14 \cdot 10^{-4}$ |
| 3 | $1.59 \cdot 10^{-4}$ |

Table 2: Mesh sizes for the IB method

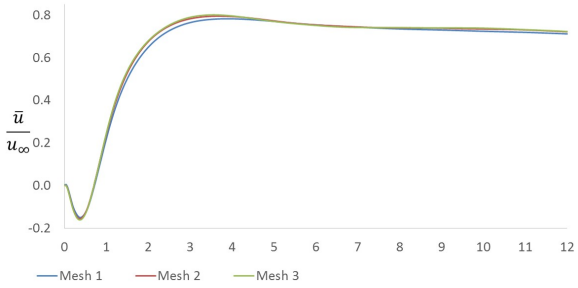


Figure 5: Mean streamwise velocity along the centreline of the cylinder for the different mesh using IB solver at Re=250

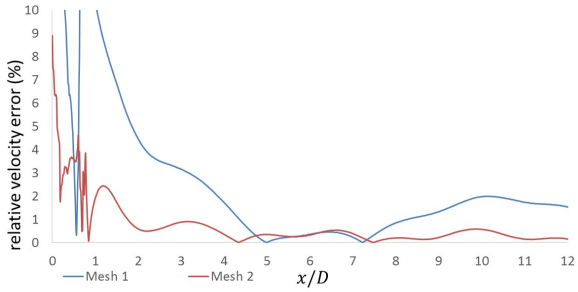


Figure 6: Relative error for the velocity of mesh 1 and 2 along the centreline against mesh 3 at Re=250

The comparison between standard and IB was also extended at Re=250 on more sensitive parameters. Figure 7 shows that the velocity for the IB and the standard solvers match. The difference for the different Reynolds stress components remains at an acceptable level considering the sensitivity of those parameters. Overall, the IB and standard solvers are in close agreement, highlighting the suitability of the IB approach at Re=250.

### Pores shape comparisons

The IB method is then assessed on an idealised pore configuration at Re=300, shown in figure 8. The velocity and pressure profiles are plotted along the white line seen in figure 8 and compared using both the IB and the standard solver. The results figure 9 show that the overall behaviour of the flow is comparable between the two solvers demonstrating the ability of the IB method to accurately simulate flow in idealised pore structures without any adaption requirements.

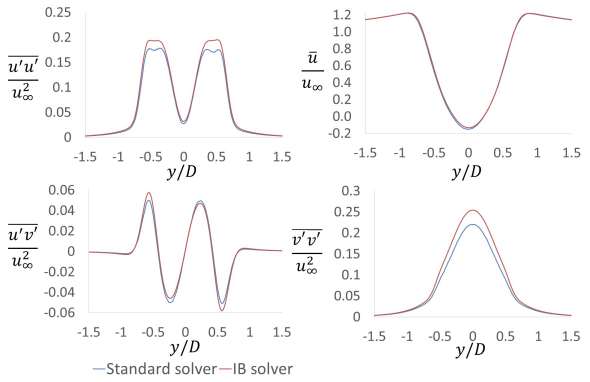


Figure 7: Vertical profiles of velocity and Reynolds stresses at x/D = 0.5 behind the cylinder at Reynolds 250

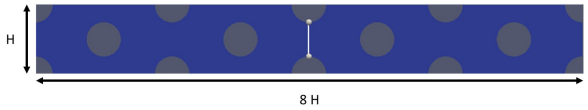


Figure 8: Computational domain: Porosity = 0.8, H = $0.1m$

### Computational cost

The computational cost of both the IB and standard solvers are also investigated. For the flow around the cylinder, the CPU time is much lower for the IB compared to the standard solver as seen in table 3. This is mostly due to the reduced number of cells for the IB induced by the non-adaption of the grid to the geometry. However, in the pore configuration, the number of immersed cells is largely increased with a ratio of IB cells over total number of cells increased by over 100 times. The simulation time was then approximately 1.5 times larger than for the standard solver with equivalent $\triangle x$. The computational cost of the IB method seems directly correlated to the number of immersed cells. Further investigations are required to evaluate and improve the cost competitiveness of the IB solver compared to the standard solver on more complex geometries. It is also expected that the increased number of cells due to inflation is going to impact the overall CPU time of the standard solver, thus a non-trivial evaluation of both methods.

| | Cylinder case Mesh 3 Re=250 | Pore configuration Re=300 |
|---|---|---|
| IB | | |
| CPU Time | 7 | 4.25 |
| IB Cells/Total Cells | $8 \times 10^{-5}$ | $1 \times 10^{-2}$ |
| Total Cells | 2.25 millions | 166,032 |
| Standard | | |
| CPU Time | 17 | 3.1 |
| Total Cells | 2.5 millions | 181,743 |

Table 3: Comparison of CPU time (minutes of calculation for 1 second of simulation) and number of cells

### Conclusion

The IB method was successfully applied to two idealized configurations. For Re below 500, it provides results comparable to a standard solver based on BF grids. Mesh requirements for the IB showed that a minimal number of Immersed cell is required to capture the flow physics around the cylinder. Finally, there is a net gain of CPU time on the cylinder configuration with the IB method. However, the standard solver becomes more advantageous for the pore configuration due to the increased number of IB cells. The IB method will next be applied to more realistic structures, such as geometries obtained from $\mu - CT$ scans. Improvements of the currenlty implemented IB method will be also investigated in order to better optimise the management of the IB cells for cost-effectiveness and accuracy of the approach.

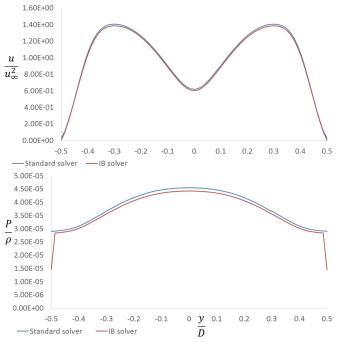


Figure 9: Velocity and pressure profiles between 2 pores.

### References


[1] Berger, R., Wille, E., Periodic flow phenomena. *Annual Review in Fluids Mechanics*, 4:313-340, 1972.

[2] Coutanceau, R., Bouard, M., Experimental determination of main features of the viscous flow in the wake of a circular cylinder in uniform translation, part 1 steady flow, *Journal of Fluids Mechanics*, 79:4018-4037, 1977.

[3] De Lemos, M.J.S., Turbulence in porous media , Second edition, Elsevier, 2012.

[4] Mittal, R., Iaccarino, G., Immersed Boundary Methods, Annual Review of Fluid Mechanics, 37:239-61, 2005.

[5] Sauret, E., Hooman, K., Particle size distribution effects on preferential deposition areas in metal foam wrapper tube bundle,*Int. J. Heat & Mass Trans.*, 79:905-915, 2014.

[6] Sedaghat, M.H., Shahmardan, M.M., Norouzi, M., Jayathilake, P.G., Nazari, M., Numerical simulation of mucociliary clearance: immersed boundary lattice Boltzmann method *Computers and Fluids*, 131:91-101, 2016.

[7] Seif, M.S., Asnaghi, A., Jahanbakhsh, E., Implementation of PISO algorithm for simulating unsteady cavitating flows , *Ocean Engineering*, 37:1321-1336, 2010.

[8] Peskin, C.S., Flow patterns around heart valves: A numerical method, *Journal of Comp. Physics*, 10:252-271, 1972.

[9] Xu, S., The immersed interface method for simulating prescribed motion of rigid objects in an incompressible viscous flow, *Journal of Computational Physics*, 227:5045-5071, 2008.

[10] Wanik, A., Schnell, U., some remarks on the PISO and SIMPLE algorithms for steady Turbulent flow problems, *Computers & Fluids*, 17:555-570, 1989.

[11] Wei, Z.A., Zheng, Z.C., Yang, X., Computation of Flow trough a tree-dimensional periodic array of porous structures by parallel immersed boundary method, *Journal of Fluids Engineering*, 136(4):040905-040905-10, 2014.

[12] Zhdanov, M.S., *Inverse Theory and Applications in Geophysics*, Second edition, Elsevier, 2015.