



## NRC Publications Archive Archives des publications du CNRC

### **A finite element immersed boundary method for fluid flow around rigid objects**

Ilinca, F.; Hétu, J. -F.

This publication could be one of several versions: author's original, accepted manuscript or the publisher's version. / La version de cette publication peut être l'une des suivantes : la version prépublication de l'auteur, la version acceptée du manuscrit ou la version de l'éditeur.

For the publisher's version, please access the DOI link below. / Pour consulter la version de l'éditeur, utilisez le lien DOI ci-dessous.

#### **Publisher's version / Version de l'éditeur:**

<https://doi.org/10.1002/flid.2222>

*International Journal for Numerical Methods in Fluids*, 65, 7, pp. 856-875, 2010-01-28

#### **NRC Publications Record / Notice d'Archives des publications de CNRC:**

<https://nrc-publications.canada.ca/eng/view/object/?id=e0263d14-a654-46d1-ab4d-092620326c35>

<https://publications-cnrc.canada.ca/fra/voir/objet/?id=e0263d14-a654-46d1-ab4d-092620326c35>

Access and use of this website and the material on it are subject to the Terms and Conditions set forth at

<https://nrc-publications.canada.ca/eng/copyright>

READ THESE TERMS AND CONDITIONS CAREFULLY BEFORE USING THIS WEBSITE.

L'accès à ce site Web et l'utilisation de son contenu sont assujettis aux conditions présentées dans le site

<https://publications-cnrc.canada.ca/fra/droits>

LISEZ CES CONDITIONS ATTENTIVEMENT AVANT D'UTILISER CE SITE WEB.

**Questions?** Contact the NRC Publications Archive team at

PublicationsArchive-ArchivesPublications@nrc-cnrc.gc.ca. If you wish to email the authors directly, please see the first page of the publication for their contact information.

**Vous avez des questions?** Nous pouvons vous aider. Pour communiquer directement avec un auteur, consultez la première page de la revue dans laquelle son article a été publié afin de trouver ses coordonnées. Si vous n'arrivez pas à les repérer, communiquez avec nous à PublicationsArchive-ArchivesPublications@nrc-cnrc.gc.ca.



# A finite element immersed boundary method for fluid flow around rigid objects<sup>‡</sup>

F. Ilinca<sup>\*, †, §</sup> and J.-F. Héту<sup>§</sup>

*National Research Council, 75 de Mortagne, Boucherville, Qc, Canada J4B 6Y4*

## SUMMARY

This paper proposes a new immersed boundary (IB) method for solving fluid flow problems in the presence of rigid objects which are not represented by the mesh. Solving the flow around objects with complex shapes may involve extensive meshing work that has to be repeated each time a change in the geometry is needed. Important benefit would be reached if we are able to solve the flow without the need of generating a mesh that fits the shape of the immersed objects. This work presents a finite element IB method using a discretization covering the entire domain of interest, including the volume occupied by immersed objects, and which produces solutions of the flow satisfying accurately the boundary conditions at the surface of immersed bodies. In other words the finite element solution represents accurately the presence of immersed bodies while the mesh does not. This is done by including additional degrees of freedom on interface cut elements which are then eliminated at element level. The boundary of immersed objects is defined using a level set function. Solutions are shown for various flow problems and the accuracy of the present approach is measured with respect to solutions obtained on body-fitted meshes. Copyright © 2010 Crown in the right of Canada.

Received 2 July 2009; Revised 23 September 2009; Accepted 24 September 2009

KEY WORDS: immersed boundary method; finite elements; immersed objects; non-body-fitted mesh; pressure enrichment

## 1. INTRODUCTION

The numerical solution of partial differential equations describing the fluid flow is usually done by performing spatial and temporal discretization. While the temporal discretization has only to deal with a first-order derivative that can be easily discretized, the spatial representation must take into account for the boundaries of the computational domain and then make use of a discrete representation via meshing. Most flow solvers are based on body-conforming grids (i.e. the external boundary and surfaces of immersed bodies are represented by the mesh faces), but there is an increased interest in solution algorithms for non-body-conforming grids. Such methods are presented under a variety of names: immersed boundary (IB), immersed interface, embedded mesh, fictitious domain, all having in common the fact that the spatial discretization is done over a single domain containing both fluid and solid regions and where mesh points are not necessarily located on the fluid–solid interface. For simplicity, we will use in the present work the IB term to identify a non-body-fitted method. IB methods have the main advantage of avoiding costly and sometimes extremely difficult meshing work on body-fitted geometries. Generally, a regular parallelepiped is meshed with

\*Correspondence to: F. Ilinca, National Research Council, 75 de Mortagne, Boucherville, Qc, Canada J4B 6Y4.

†E-mail: florin.ilinca@nrc-nrc.gc.ca

‡This article was published online on [28 January 2010]. An error was subsequently identified. This notice is included in the online and print versions to indicate that both have been corrected [11 February 2010].

§Research Officer, Industrial Materials Institute.

an uniform grid. The IB method can also result in important algorithmic simplifications when immersed moving bodies are considered. One important drawback of such a method is that the boundary that has an important influence on the solution is also a place where distorted elements may be found once regular mesh elements are cut by the solid boundary. The imposition of the boundary conditions on the IB is also a point of concern as it is either costly in both computational time and memory if mesh nodes are added/removed dynamically or it is a possible source of errors if imposed approximately on the closest grid nodes.

The IB method was initially introduced by Peskin [1] for finite differences applied to fluid–structure interactions. The method received particular attention in recent years and his application goes from flow in the presence of heart devices [1, 2] to interaction of body movement and fluid flow [3], modeling anguilliform swimming [4] and 3-D parachute simulation [5]. In the original IB method, the structure is considered as a region embedded into the fluid and where additional source terms are imposed in such a way as to take into account for the force applied by the structure to the fluid. A review of the developments of the IB method can be found in [3]. Two classes of methods are discussed depending on how the source terms describing the presence of the boundary are imposed: directly in the continuous equations describing the flow (continuous forcing approach) or in the resulting system of linear equations (discrete forcing approach). A study on the behavior of an IB method in unsteady flows over sharp-edged bodies is presented in [6]. Special emphasis is put on how the no-slip boundary condition at the fictitious boundary wall is recovered in the startup flow.

A finite element implementation of the IB method which makes use of a Dirac function to impose the interaction force at the fluid–structure interface was presented by Boffi and Gastaldi [7]. A discussion on the stability of the space–time discretization can be found in [8]. Zhang *et al.* [9] present the immersed finite element method for the solution of fluid–structure interactions with deformable solids. In their method, a Lagrangian solid mesh moves on top of a background Eulerian fluid mesh which spans over the entire computational domain and a discretized delta function is used to describe the coupling between the fluid and solid domains.

Glowinski *et al.* [10] proposed a Lagrange multiplier-based fictitious domain method which was further extended to obtain finite element solutions around moving rigid bodies using meshes which are not boundary fitted [11]. The method consists of filling the moving bodies by the surrounding fluid and impose rigid body motions to the fluid occupying the regions originally occupied by the rigid bodies. The rigid body motion constraint is then relaxed by using distributed Lagrange multipliers and a flow problem over the entire domain is solved.

Löhner *et al.* [12] present an IB method implemented inside an edge-based finite element solver. The embedded surfaces are represented by a triangulation, while immersed bodies are given by a tetrahedral mesh independent of the computational mesh. A discussion on first- and higher-order treatment for imposing kinematic conditions on the surface of the immersed bodies is made. For first-order schemes the elements cut by the immersed surface are deactivated and special ‘boundary contributions’ are added to account for the presence of the boundary. A higher-order treatment of the embedded surfaces may be achieved by using ghost points or mirrored points to compute the contribution of elements cut by the boundary. Such a method does not need extensive modification of the fluid flow solver but on the other hand it requires dynamic structures handling and more CPU time to perform renumbering/reordering of the degrees of freedom. Various applications are shown, notably the flow around a shuttle in ascent configuration, the dispersion of a contaminant cloud in a city and flow past a car. A comparison of body-fitted, embedded and IB techniques using the same flow solver is presented in [13]. All methods recover the same grid converged solution and it is observed that there is a need for accurate boundary conditions when the mesh does not fit on the immersed bodies.

The finite element solution of the flow around immersed objects using a fixed mesh arbitrary Lagrangian–Eulerian (ALE) approach is presented by Codina *et al.* [14]. In their work the solid boundary is represented using a level set function defined on a background fixed mesh, while the moving boundary is treated by an ALE technique in the region close to the immersed surface. Boundary conditions on the IB are imposed by a least-squares approximation.

In this work the flow around immersed bodies is solved using a 3-D finite element IB method. The solution is interpolated using linear elements and time integration is done by an implicit Euler scheme. The IB is represented using a level set function. Therefore, the boundary is represented in the discrete form by the same interpolation functions that are used to solve the flow problem. Special emphasis is put on the accurate imposition of boundary conditions on the solid boundary. The proposed approach recovers the accuracy of methods based on the addition of nodes at the intersection between the mesh edges and the boundary, but without any of the drawbacks of such methods in terms of increased computational time and dynamic data structure management. The proposed approach is verified on simple cases against the solution on a body-fitted grid and then applied to various flow problems. In this paper we consider only problems for which the immersed bodies are not moving. However, the procedure of imposing the boundary solution on the interface is general and the extension to moving immersed solids will be the subject of future work.

The paper is organized as follows. First the model problem and the associated finite element formulation are presented. The IB formulation is discussed in Section 3 and the procedure to impose the boundary conditions is detailed. Section 4 illustrates the performance of the present IB method for a selection of 2-D and 3-D test problems. The paper ends with conclusions.

## 2. THE MODEL PROBLEM

We consider the incompressible fluid flow problem on a bounded computational domain  $\Omega$  formed by the fluid region  $\Omega_f$  and the solid volume  $\Omega_s$  as shown in Figure 1. The immersed interface  $\Gamma_i = \partial\Omega_f \cap \partial\Omega_s$  represents a boundary for the fluid flow and therefore exerts a force on the fluid contained in  $\Omega_f$ .

### 2.1. Model equations and boundary conditions

The equations of motion are the incompressible Navier–Stokes equations:

$$\rho \left( \frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} \right) = -\nabla p + \nabla \cdot [\mu(\nabla \mathbf{u} + (\nabla \mathbf{u})^T)] + \mathbf{f}, \quad (1)$$

$$\nabla \cdot \mathbf{u} = 0, \quad (2)$$

where  $\rho$  is the density,  $\mathbf{u}$  the velocity vector,  $p$  the pressure,  $\mu$  the dynamic viscosity and  $\mathbf{f}$  a volumetric force vector. The interface  $\Gamma_i$  between the fluid and solid regions is specified using a level set function  $\psi$  which is defined as a signed distance function from the immersed interface:

$$\psi(\mathbf{x}) = \begin{cases} d(\mathbf{x}, \mathbf{x}_i), & \mathbf{x} \text{ in the fluid region,} \\ 0, & \mathbf{x} \text{ on the fluid/solid interface,} \\ -d(\mathbf{x}, \mathbf{x}_i), & \mathbf{x} \text{ in the solid region,} \end{cases} \quad (3)$$

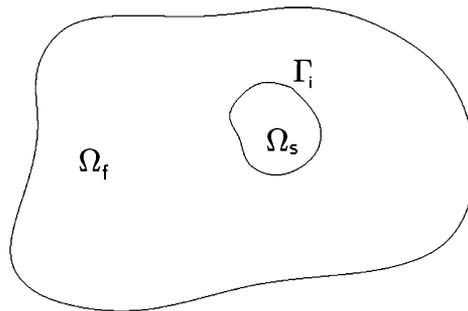


Figure 1. Computational domain formed by fluid region  $\Omega_f$  and solid region  $\Omega_s$ .

where  $d(\mathbf{x}, \mathbf{x}_i)$  is the distance between the point  $P(\mathbf{x})$  and the fluid/solid interface. Hence, points in the fluid region will take on positive values of  $\psi$ , whereas  $\psi$  for points in the solid region will be negative. The definition of the level set function may be more complicated for complex 3-D geometries. In such a case we may consider that the IB surface is provided in the form of a CAD file from which we generate a surface mesh with a mesh size sufficiently small to have a correct representation of the surface. Then, the level set function is simply computed from the shortest distance between the nodes of the 3-D mesh and those of the surface mesh.

The boundary conditions associated with the momentum–continuity equations are

$$\mathbf{u} = \mathbf{U}_D(\mathbf{x}), \quad \text{for } \mathbf{x} \in \Gamma_D, \quad (4)$$

$$\mu(\nabla \mathbf{u} + \nabla \mathbf{u}^T) \cdot \hat{\mathbf{n}} - p \hat{\mathbf{n}} = \mathbf{t}(\mathbf{x}), \quad \text{for } \mathbf{x} \in \Gamma_t, \quad (5)$$

where  $\Gamma_D$  is the portion of the fluid boundary  $\partial\Omega_f$  where Dirichlet conditions are imposed, and  $\mathbf{t}$  is the traction imposed on the remaining fluid boundary  $\Gamma_t = \partial\Omega_f \setminus \Gamma_D$ . Dirichlet boundary conditions are imposed at the interface between the fluid and solid regions, i.e.  $\Gamma_i \subset \Gamma_D$ . Because  $\Gamma_i$  is not represented by the finite element discretization, a special procedure is used to enforce velocity boundary conditions on this surface. This approach will be described in details in Section 3.

### 2.2. The finite element formulation

In this work we discretize both velocity and pressure using linear continuous interpolants. It is well known that the P1-P1 element used in conjunction with the Galerkin finite element formulation produces spurious oscillations and some form of stabilization is needed. We use the Galerkin least-squares (GLS) method that leads to the following weak formulation:

$$\int_{\Omega} \rho \left( \frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} \right) N_i^u \, d\Omega + \int_{\Omega} \mu(\nabla \mathbf{u} + \nabla \mathbf{u}^T) \cdot \nabla N_i^u \, d\Omega - \int_{\Omega} p \nabla N_i^u \, d\Omega - \int_{\Omega} \mathbf{f} N_i^u \, d\Omega + \sum_K \int_{\Omega_K} \left\{ \rho \left( \frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} \right) + \nabla p - \nabla \cdot [\mu(\nabla \mathbf{u} + \nabla \mathbf{u}^T)] - \mathbf{f} \right\} \tau \mathbf{u} \cdot \nabla N_i^u \, d\Omega_K = \int_{\Gamma_t} \mathbf{t} N_i^u \, d\Gamma, \quad (6)$$

$$\int_{\Omega} \nabla \cdot \mathbf{u} N_i^p \, d\Omega + \sum_K \int_{\Omega_K} \left\{ \rho \left( \frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} \right) + \nabla p - \nabla \cdot [\mu(\nabla \mathbf{u} + \nabla \mathbf{u}^T)] - \mathbf{f} \right\} \tau \nabla N_i^p \, d\Omega_K = 0, \quad (7)$$

where  $N_i^u, N_i^p$  are continuous, piecewise linear test functions associated with the velocity and pressure equations. The first four integrals in the right-hand side of Equation (6) and the first integral in Equation (7) correspond to the Galerkin formulation whereas the integrals over the elements interior are the GLS stabilization terms. The stabilization parameter  $\tau$  is computed as from Reference [15, 16]:

$$\tau = \left[ \left( \frac{2\rho|\mathbf{u}|}{h_K} \right)^2 + \left( \frac{4\mu}{m_k h_K^2} \right)^2 \right]^{-1/2} \quad (8)$$

Here  $h_K$  is the size of the element  $K$  and  $m_k$  is a coefficient set to  $\frac{1}{3}$  for linear elements (see [15, 17]).

The velocity  $(u, v, w)$  and pressure  $p$  are discretized as

$$u \approx u_h = \sum_{j=1, N_n} u_j N_j^u(\mathbf{x}), \quad (9)$$

$$v \approx v_h = \sum_{j=1, N_n} v_j N_j^u(\mathbf{x}), \quad (10)$$

$$w \approx w_h = \sum_{j=1, N_n} w_j N_j^u(\mathbf{x}), \quad (11)$$

$$p \approx p_h = \sum_{j=1, N_n} p_j N_j^p(\mathbf{x}), \quad (12)$$

where  $N_j^u, N_j^p$  are nodal shape functions that, in the present case, are the same as the test functions and  $N_n$  is the number of nodes in the mesh.

To alleviate the writing we regroup the four nodal unknowns into the vector  $s_j=(u_j, v_j, w_j, p_j)^T \equiv (\mathbf{u}_j, p_j)^T$ . Using this notation Equations (6) and (7) lead to the following system of equations:

$$\sum_{j=1, N_n} \mathbf{A}_{ij} s_j = \mathbf{b}_i \quad \text{for } i=1, N_n, \tag{13}$$

$$\mathbf{A}_{ij} = \begin{pmatrix} A_{ij}^{uu} & A_{ij}^{up} \\ A_{ij}^{pu} & A_{ij}^{pp} \end{pmatrix}, \quad s_j = \begin{pmatrix} \mathbf{u}_j \\ p_j \end{pmatrix}, \quad \mathbf{b}_i = \begin{pmatrix} b_i^u \\ b_i^p \end{pmatrix}, \tag{14}$$

whose unknowns are the nodal values of the velocity and pressure.

### 3. THE IB METHOD

In the present work we discretize the entire domain  $\Omega = \Omega_f \cup \Omega_s$ , but a special treatment is applied when solving for nodes in the solid region. The mesh is intersected by the interface  $\Gamma_i$  at points located along element edges and we consider those points as additional degrees of freedom in the finite element formulation. In Figure 2 the mesh is shown by the black continuous lines, the interface is indicated by the discontinuous line and the additional interface nodes are indicated by the filled square symbols. The proposed technique was implemented for 3-D meshes but for the ease of the presentation only a 2-D case is illustrated in Figure 2. The fluid/solid interface is defined by a level set function and therefore the additional nodes are easily determined as being those for which  $\psi = 0$ . Moreover, because the level set function is interpolated using linear shape functions, the intersection between  $\Gamma_i$  and a tetrahedral element is a plane, either a triangle or a quadrilateral (some exceptions are cases in which the interface intersects with one or more grid nodes). Elements cut by  $\Gamma_i$  are therefore decomposed into sub-elements which are either entirely in the fluid or entirely in the solid regions. The addition of degrees of freedom associated with the nodes on the interface would normally result in a modification of the global matrix resulting from the finite element equations. In such a case, the implementation is more challenging and the increase in computational cost is inherent as dynamic data structures and renumbering are needed. The present approach does not need the explicit addition of the degrees of freedom associated with those nodes. The procedure is described in detail in this section.

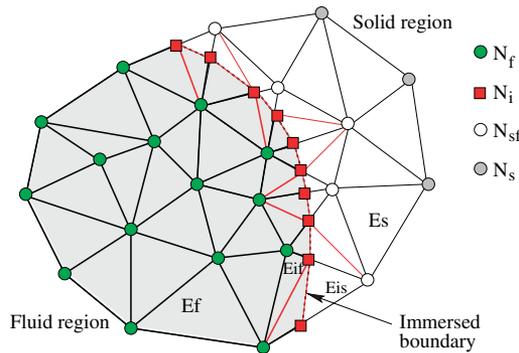


Figure 2. Treatment of elements cut by the immersed boundary.

### 3.1. Classification of grid nodes and elements

As a first step we introduce a nomenclature that will ease the understanding of the proposed approach. Mesh nodes are separated depending on their relative location with respect to the IB, as follows:

- nodes in the fluid region (Nf): identified by the filled circles in Figure 2;
- additional nodes on the interface (Ni): filled squares in Figure 2;
- nodes in the solid region, but connected with nodes in the fluid region (Nsf): open circles in Figure 2;
- nodes in the solid region which are not connected with nodes in the fluid region (Ns): gray-filled circles in Figure 2.

Mesh elements are also separated depending on their relative location with respect to the IB:

- elements in the fluid region (Ef);
- elements cut by the IB (Ei). Those elements have nodes in both fluid and solid regions. However, the addition of nodes on the interface and the decomposition of elements cut by  $\Gamma_i$  yields to the formation of elements which are either entirely in the fluid region (Eif) or in the solid region (Eis);
- elements in the solid region (Es).

### 3.2. Imposition of boundary conditions at the interface

We consider here that the solid embedded in the mesh has a prescribed velocity  $\mathbf{u}_s$ . In the present work only applications for  $\mathbf{u}_s = 0$  are considered and therefore the finite element formulation does not take into account for moving immersed interfaces. Velocity boundary conditions are imposed as follows:

- the velocity at nodes (Nf) is an unknown of the global system of equations (no boundary condition), except for nodes located on  $\Gamma_D$  for which a standard Dirichlet boundary condition is imposed;
- the velocity  $\mathbf{u}_s$  is imposed on nodes in the solid, i.e. for nodes (Ns), (Nsf) and (Ni).

By construction the additional nodes (Ni) have a prescribed velocity. Therefore, there are no additional equations associated with the velocity of those nodes. Only the right-hand side of the fluid nodes connected to them will change and by these means take into account for the boundary as being located on  $\Gamma_i$ .

The pressure degrees of freedom are associated with the continuity equations. In order to enforce mass conservation in the entire fluid region the continuity equations are solved on all fluid elements, i.e. in (Ef) and (Eif). The continuity equations are not solved in the solid elements (Es) and (Eis) and the pressure is set to a constant (say zero for simplicity) on solid nodes (Ns). On nodes (Nsf) a mean pressure from the fluid nodes to which the solid node is connected is determined in order to improve the visual representation of the solution. Those values are not affecting the solution in the fluid region.

This approach has the advantage of enforcing mass conservation on the entire fluid region because the continuity equation is solved inside all (Ef) and (Eif) elements. One drawback is that it introduces additional pressure degrees of freedom associated with the nodes on the interface (Ni). These nodal pressures are unknown and should be included in the global system matrix. One way to avoid this issue is to consider that the pressure discretization is discontinuous on element faces containing interface nodes. The pressure at a node (Ni) is therefore different on the interface elements containing the respective node and the additional degree of freedom can be eliminated by static condensation. Figure 3 illustrates the 2-D case where the interface elements  $e1$  (formed by the fluid node  $P_{f1}$  and the solid nodes  $P_{s1}$ ,  $P_{s2}$ ) and  $e2$  (formed by the fluid nodes  $P_{f1}$ ,  $P_{f2}$  and the solid node  $P_{s2}$ ) are decomposed into sub-elements. The pressure on the interface nodes is different on different elements of the original mesh cut by the interface. In the example shown in Figure 3 the pressure on the interface node  $P_i$  is different when the node is considered inside the

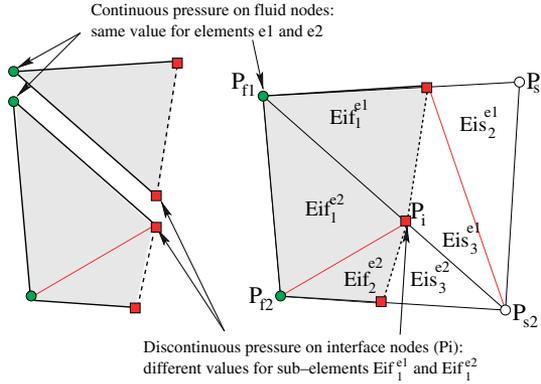


Figure 3. Local enrichment of the pressure discretization on interface elements.

sub-element  $Eif_1^{e1}$  and inside the sub-element  $Eif_1^{e2}$ , respectively. However, the pressure at  $P_i$  for sub-elements  $Eif_1^{e2}$  and  $Eif_2^{e2}$  is the same as both sub-elements are subdivisions of the element  $e2$ .

Elementary systems are therefore constructed using the following algorithm:

- Identify elements (Ei) that are cut by the interface. Elements entirely in the solid region do not contribute to the global system, while elements entirely in the fluid region are computed as usual, ignoring the presence of the interface;
- Create the additional nodes (Ni) on the interface;
- Decompose interface elements (Ei) into sub-elements which are either entirely in the fluid region (Eif) or in the solid region (Eis);
- For interface elements, only fluid sub-elements (Eif) contribute to the elementary system. For each interface element, the sub-element matrices are assembled into an augmented elementary system that also contains the velocity and pressure degrees of freedom associated with the interface nodes (Ni);
- The degrees of freedom associated with the interface nodes are eliminated at the element level by using the Dirichlet boundary conditions for the velocity and by static condensation for the pressure. The pressure degrees of freedom on interface nodes can also be computed using the flow solution at the previous iteration.

Hereafter, the elimination of interface degrees of freedom is presented. The contribution of an interface element including both the fluid and interface nodes can be written as follow:

$$A_e s_e = b_e, \quad A_e = \begin{pmatrix} A_{ff}^{uu} & A_{ff}^{up} & A_{fi}^{uu} & A_{fi}^{up} \\ A_{ff}^{pu} & A_{ff}^{pp} & A_{fi}^{pu} & A_{fi}^{pp} \\ A_{if}^{uu} & A_{if}^{up} & A_{ii}^{uu} & A_{ii}^{up} \\ A_{if}^{pu} & A_{if}^{pp} & A_{ii}^{pu} & A_{ii}^{pp} \end{pmatrix}, \quad s_e = \begin{pmatrix} u_f \\ p_f \\ u_i \\ p_i \end{pmatrix}, \quad b_e = \begin{pmatrix} b_f^u \\ b_f^p \\ b_i^u \\ b_i^p \end{pmatrix}, \quad (15)$$

where indice  $f$  denotes degrees of freedom associated with fluid nodes (Nf) and indice  $i$  denotes degrees of freedom associated with interface nodes (Ni). The velocity at interface nodes is known  $u_i = u_s$  and can therefore be eliminated from the unknowns to be determined leading to the following system:

$$A_e s_e = b_e, \quad A_e = \begin{pmatrix} A_{ff}^{uu} & A_{ff}^{up} & A_{fi}^{up} \\ A_{ff}^{pu} & A_{ff}^{pp} & A_{fi}^{pp} \\ A_{if}^{pu} & A_{if}^{pp} & A_{ii}^{pp} \end{pmatrix}, \quad s_e = \begin{pmatrix} u_f \\ p_f \\ p_i \end{pmatrix}, \quad b_e = \begin{pmatrix} b_f^u - A_{fi}^{uu} u_s \\ b_f^p - A_{fi}^{pu} u_s \\ b_i^p - A_{ii}^{pu} u_s \end{pmatrix}. \quad (16)$$

The last remaining step before assembling the elementary system into the global one is the static condensation of the pressure degrees of freedom  $p_i$  associated with the interface nodes. Those degrees of freedom are local on each interface element and therefore can be determined as a function of the other degrees of freedom of the element:

$$p_i = (A_{ii}^{pp})^{-1} [(b_i^p - A_{ii}^{pu} \mathbf{u}_s) - A_{if}^{pu} \mathbf{u}_f - A_{if}^{pp} p_f] \quad (17)$$

The resulting equations contain only the degrees of freedom corresponding to the fluid nodes:

$$\mathbf{A}_e \mathbf{s}_e = \mathbf{b}_e, \quad \mathbf{A}_e = \begin{pmatrix} (A_{ff}^{uu} - A_{fi}^{up} (A_{ii}^{pp})^{-1} A_{if}^{pu}) & (A_{ff}^{up} - A_{fi}^{up} (A_{ii}^{pp})^{-1} A_{if}^{pp}) \\ (A_{ff}^{pu} - A_{fi}^{pp} (A_{ii}^{pp})^{-1} A_{if}^{pu}) & (A_{ff}^{pp} - A_{fi}^{pp} (A_{ii}^{pp})^{-1} A_{if}^{pp}) \end{pmatrix}, \quad (18)$$

$$\mathbf{s}_e = \begin{pmatrix} \mathbf{u}_f \\ p_f \end{pmatrix}, \quad \mathbf{b}_e = \begin{pmatrix} b_f^u - A_{fi}^{uu} \mathbf{u}_s - A_{fi}^{up} (A_{ii}^{pp})^{-1} (b_i^p - A_{ii}^{pu} \mathbf{u}_s) \\ b_f^p - A_{fi}^{pu} \mathbf{u}_s - A_{fi}^{pp} (A_{ii}^{pp})^{-1} (b_i^p - A_{ii}^{pu} \mathbf{u}_s) \end{pmatrix}.$$

The following observations can be made with respect to the proposed method:

*Remark 1*

The velocity and pressure discretization inside interface elements make use of the shape functions defined on the sub-elements determined by the cutting interface. Therefore, the resulting solution is similar to the one obtained on a grid having nodes on the fluid/solid interface.

*Remark 2*

The imposition of the solid velocity is made exactly on the interface nodes, thus resulting in an accurate representation of boundary conditions at this location.

*Remark 3*

The continuity equations are solved for elements covering the entire fluid region, including fluid elements (Ef) and interface/fluid sub-elements (Eif). Therefore, mass conservation is enforced in the discrete form of the equations.

*Remark 4*

The pressure on interface nodes is not an unknown of the global system of equations, but it is determined at element level as function of the solution inside the fluid (see Equation (17)). Note that the size of the matrix  $A_{ii}^{pp}$  is given by the number of interface nodes of the respective element, which means 2 in 2-D and 3 or 4 in 3-D, making the solution of Equation (17) very inexpensive. The pressure interpolation is continuous except for the element faces cut by the interface where it is discontinuous. The procedure results in the local enrichment of the pressure discretization. For other pressure enrichment techniques resulting in local discontinuous pressure interpolations, the reader may consult the work of Minev *et al.* [18] for two-fluid flow with surface tension and that of Coppola-Owen and Codina [19, 20] for two-phase and free surface flows.

*Remark 5*

The procedure can be seen as a local refinement of the mesh such as to have grid nodes on the interface between the fluid and solid regions. However, it does not lead to the burden of changing dynamically the mesh as the additional degrees of freedom are eliminated at element level.

*3.3. Decomposition of interface elements for 3-D tetrahedral grids*

The sub-elements creation is illustrated in Figure 2 for 2-D triangular elements but the procedure is general and applications in this work are using 3-D tetrahedral grids. For 3-D elements the connectivity of sub-elements should be done with care as each triangular face cut by the interface should be decomposed in the same way for the adjacent tetrahedrons. The intersection between interface elements and the fluid/solid interface is either a triangle or a quadrilateral. When the intersection is a triangle the element decomposition yields one tetrahedron on one side and three on the other side, whereas for a quadrilateral we have three tetrahedrons on each side of the interface.

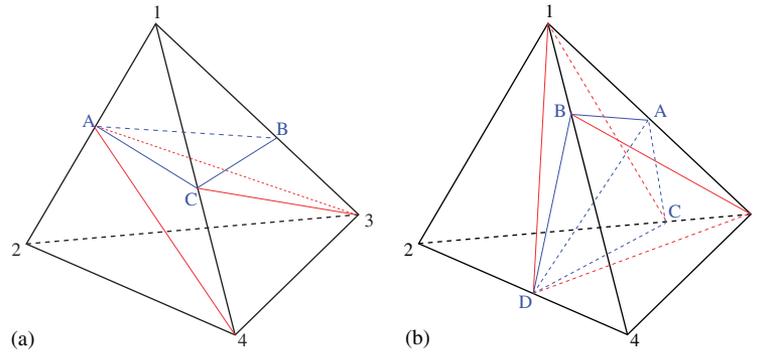


Figure 4. Decomposition of interface elements: (a) one or three nodes in the fluid region and (b) two nodes in the fluid region.

In this section we describe a procedure to perform the subdivision of tetrahedral elements cut by the interface which is coherent for adjacent elements. In doing so, we will use the number associated with each node in the mesh, hereafter identified as the nodal global number. The cases for which the interface is a triangle and a quadrilateral are treated separately.

For an interface element having one node in the fluid region and three in the solid region, we perform a local renumbering of the nodes such as ‘1’ identifies the fluid node, whereas the solid nodes are identified by the labels ‘2’, ‘3’ and ‘4’. The labels to the three solid nodes are allocated depending on the global number of each node: ‘2’ is the node with the highest number, ‘3’ is the node with the lowest number and ‘4’ is the remaining solid node. The intersection between the element 1234 and the fluid/solid interface is the triangle ABC, where  $A \in \overline{12}$ ,  $B \in \overline{13}$  and  $C \in \overline{14}$  (Figure 4(a)). The element is therefore decomposed into four sub-elements as illustrated in Figure 4(a):

- sub-element 1 (fluid): formed by nodes 1, A, B and C
- sub-element 2 (solid): formed by nodes A, 2, 3 and 4
- sub-element 3 (solid): formed by nodes A, 4, 3 and C
- sub-element 4 (solid): formed by nodes A, 3, B and C

In Figure 4(a) the edges of interface triangles are those connecting interface nodes (identified by letters) and the additional edges of the sub-elements are those connecting interface nodes to grid nodes. Visible edges are shown with continuous lines whereas the edges on the background are shown with discontinuous lines.

In the case where the interface element has three nodes in the fluid region the procedure is similar but the solid node is labeled ‘1’ and the fluid nodes are identified by ‘2’, ‘3’ and ‘4’.

For an interface element having two fluid nodes and two solid nodes the local renumbering is done such as the fluid nodes to be labeled ‘1’, ‘2’ and the solid nodes ‘3’ and ‘4’. Moreover, we associate the label ‘1’ with the fluid node with the lower global number and similarly the label ‘3’ to the solid node with the lower global number. In this case the intersection between the element 1234 and the fluid/solid interface is the quadrilateral ABCD where  $A \in \overline{13}$ ,  $B \in \overline{14}$ ,  $C \in \overline{23}$  and  $D \in \overline{24}$  (Figure 4(b)). The element is therefore decomposed into six sub-elements as illustrated in Figure 4(b):

- sub-element 1 (fluid): formed by nodes 1, 2, C and D
- sub-element 2 (fluid): formed by nodes 1, D, A and B
- sub-element 3 (fluid): formed by nodes 1, C, A and D
- sub-element 4 (solid): formed by nodes 3, B, D and 4
- sub-element 5 (solid): formed by nodes 3, A, C and D
- sub-element 6 (solid): formed by nodes 3, A, D and B

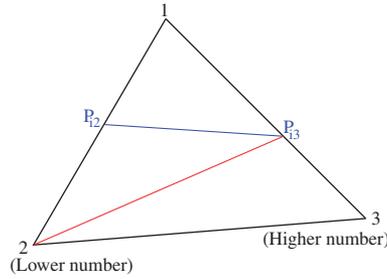


Figure 5. Decomposition of a triangular face.

This procedure ensures that each element face cut by the interface is decomposed following a consistent pattern illustrated in Figure 5. If, for example, ‘1’ is a fluid node, ‘2’ and ‘3’ are solid nodes with ‘2’ being the solid node with the lower global number, then the triangular face 123 is decomposed into the sub-triangles  $1P_{i2}P_{i3}$ ,  $P_{i2}2P_{i3}$  and  $23P_{i3}$  for both tetrahedral elements containing the respective face.

Special cases may be encountered when the interface intersects with one or more grid nodes. In order to avoid treating a large number of exception combinations and to avoid creation of very small or degenerate sub-elements we proceed to the following correction when grid nodes are too close to or on the interface:

$$\text{if } |\psi| < \varepsilon h_e \quad \text{then } \psi = \begin{cases} \varepsilon h_e & \text{if } \psi \geq 0, \\ -\varepsilon h_e & \text{if } \psi < 0, \end{cases} \quad (19)$$

where  $\varepsilon$  is a small positive parameter and  $h_e$  is the element size. Computations for this work were carried out by using  $\varepsilon=0.001$ .

## 4. APPLICATIONS

### 4.1. 2-D flow in an obstructed channel

This first test problem represents the flow in an obstructed channel for which experimental data are available [21]. The geometry, illustrated in Figure 6, consists of a channel of height  $H$  which is partially obstructed by a fence of height  $S$  and thickness  $l$  located on the lower wall. The flow has a recirculation zone behind the fence and for high blockage ratios a second recirculation region is formed at the top wall. The dependence of the reattachment length on the Reynolds number was investigated experimentally by Carvalho *et al.* [21] for a fence having  $l/H=0.1$  and three blockage ratios,  $S/H=0.25, 0.5$  and  $0.75$ .

Computations were first carried out for a blockage ratio  $S/H=0.75$  by using a body-fitted mesh and also the IB method on a mesh covering the entire channel as if the obstruction would not be present. The problem is two-dimensional and therefore only a slab was meshed with 3-D tetrahedral elements. Figure 7 illustrates the two meshes in the region of the top of the obstacle ( $y/H=0.75$ ). For the IB mesh, the region of the mesh in the solid region is shown with less intensity and the mesh was constructed to avoid intersection between interface and mesh nodes. This way the performance of the procedure to impose boundary conditions in the IB method can be evaluated. A mesh convergence study was carried out for both body-fitted and IB methods and it was found that solutions on the final meshes shown in Figure 7 are grid converged. The Reynolds number used for the simulations is defined based on the obstacle height  $S$  and the inlet mean velocity  $\bar{U}$  as  $Re_s = (\rho \bar{U} S) / \mu$ . The  $u$ -velocity profile for the solution at  $Re_s = 82.5$  at various locations along the channel is shown in Figure 8. The IB solution is shown for *Mesh 1* having 11 322 nodes and 27 360 elements, for *Mesh 2* with 44 530 nodes and 109 584 elements and for the final refined *Mesh 3* having 176 610 nodes and 437 760 elements. The solution on

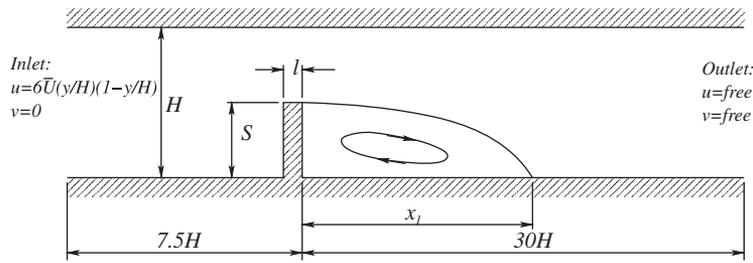


Figure 6. 2-D flow in an obstructed channel.

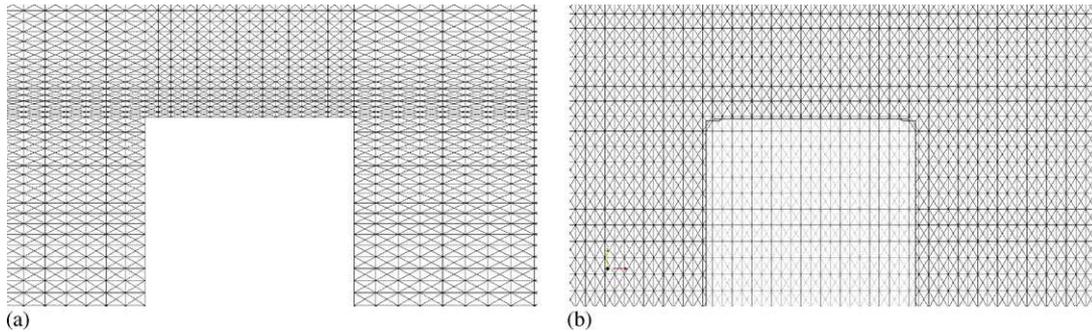


Figure 7. Meshes for the obstructed channel flow at  $S/H = 0.75$ : (a) body-fitted mesh and (b) IB mesh.

the body-fitted mesh is shown only for the final refined mesh having 159 810 nodes and 395 520 elements. The IB solution changes very little between meshes 2 and 3 thus indicating that the solution on mesh 3 is grid converged. Remark also that the solution obtained using the IB method agrees extremely well with the solution on the body-fitted mesh. This indicates that the procedure of imposing the boundary conditions for the IB method works well. The agreement between the numerical solutions and experimental data is good at  $x/S = 0, 2$  and  $8.8$ . At  $x/S = 5$ , location that is very close to the reattachment point, the agreement with the experiment is not as good. The same behavior was reported for other numerical solutions of the obstructed channel problem [22]. The numerical predictions for the length of the primary recirculation zone agree well with the experiment as can be seen in Figure 9. The results shown in Table I indicate that the IB method predicts the same position of the reattachment point as the solution on the body-fitted mesh. One advantage of the IB method is that it can generate solutions for other obstruction ratios without remeshing, thus reducing significantly the total solution time. Only the level set-function should be changed in order to get a solution for a different obstacle height. IB solutions on the same mesh were computed for an obstruction ratio  $S/H = 0.25$  and the resulting position of the reattachment point is compared with the experimental data in Figure 10.

#### 4.2. Flow around an array of spheres

**4.2.1. Problem definition.** This test problem corresponds to the flow around spheres placed on a simple cubic lattice. A non-dimensional solution is obtained by defining reference values for the variables describing both the immersed geometry and the flow. The reference length is taken equal to the distance between the center of adjacent spheres  $L$  and the reference velocity is given by the inlet velocity  $V_i$ . Following the work of Martys *et al.* [23] computations were carried out without considering the inertia in the momentum equations (low Reynolds number approach). The dimensionless viscosity and the inlet velocity are set to 1 and the dimensionless diameter of the spheres  $\tilde{D} = D/L$  is varied between 0.2 and 1.4 resulting in the solid fraction of the bed of spheres varying from  $4.19 \times 10^{-3}$  to 0.959.

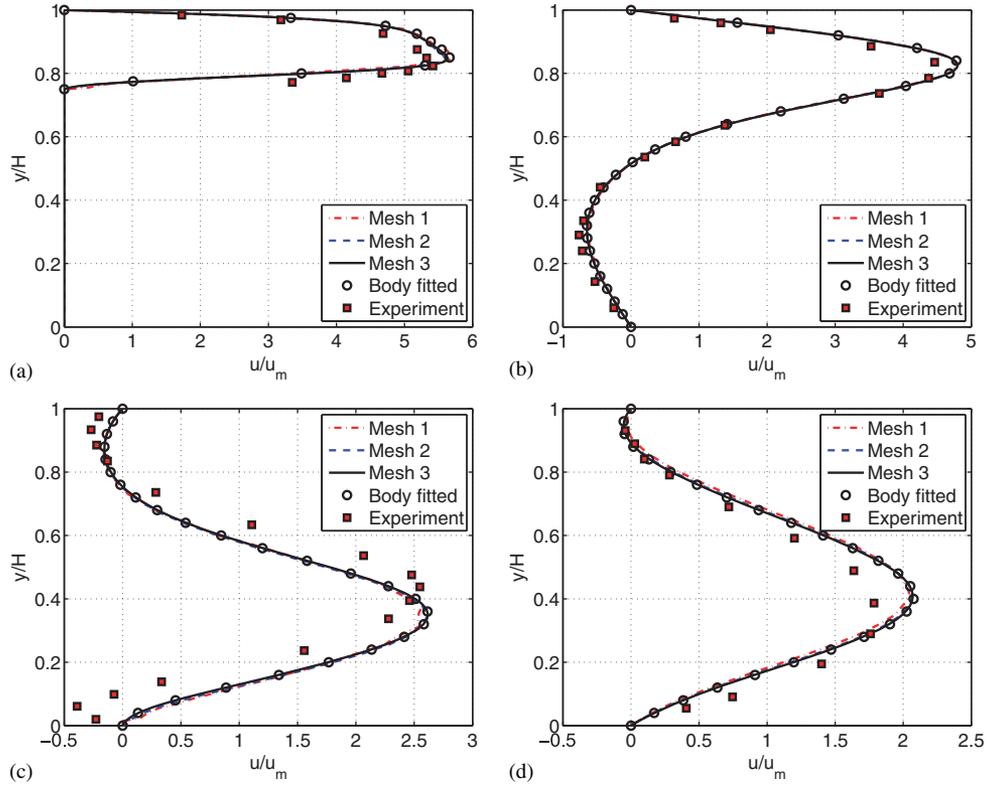


Figure 8. Comparison of predicted and measured  $u$ -velocity profile: (a)  $x/S=0$ ; (b)  $x/S=2$ ; (c)  $x/S=5$ ; and (d)  $x/S=8.8$ .

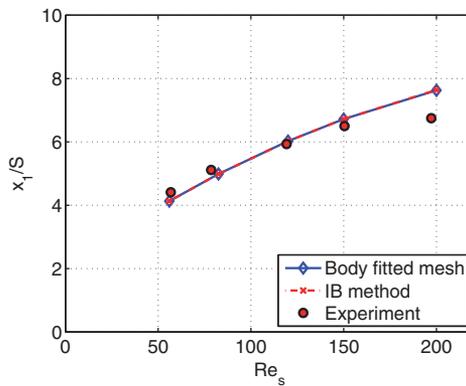


Figure 9. Length of the primary recirculation zone for  $S/H=0.75$ .

Table I. Length of the primary recirculation zone ( $x_1/S$ ) for  $S/H=0.75$ .

$Re$	56	82.5	120	150	200
Experiment [21]	4.41	5.11	5.93	6.5	6.75
Body-fitted grid	4.13	4.99	6.03	6.72	7.63
Present IB method	4.13	4.99	6.03	6.72	7.64

Given the symmetry of the arrangement of spheres, only one array is considered having the size of the transverse section equal to the distance between the centroids of two adjacent spheres. A first series of computations considered one array of 5 spheres in the flow direction. The computational

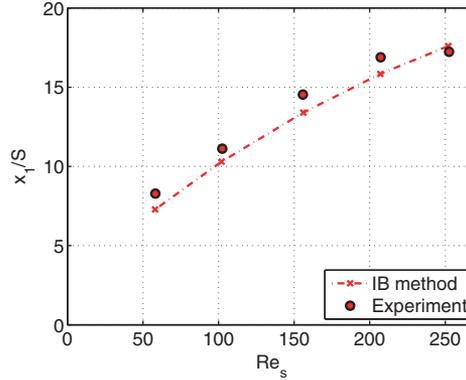


Figure 10. Length of the primary recirculation zone for  $S/H=0.25$ .

domain spans in the flow direction from  $x = -4$  to  $9$  with the five spheres located in the space between  $x=0$  and  $5$  (the centroids are at  $x=0.5, 1.5, 2.5, 3.5, 4.5$ , respectively). The domain has unit length in the  $y$  and  $z$  directions. Computations were also carried out considering an array of nine spheres. In this second case the computational domain spans from  $x = -4$  to  $13$  with the nine spheres located in the space between  $x=0$  and  $9$ . The surface of the spheres on which Dirichlet boundary conditions should be imposed is represented by the level set function:

$$\psi(x, y, z) = \min(\psi_0, ((x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2)^{1/2} - D/2) \quad \text{for } i = 1, N_s, \quad (20)$$

where  $N_s$  is the number of spheres in the array,  $\psi_0$  is an arbitrary positive value larger than the size of the computational domain (say  $\psi_0 = 100$ ) and  $(x_i, y_i, z_i)$  are the centroid coordinates of the sphere  $i$ . The level set  $\psi=0$  will then indicate the location of the boundary. The region having  $\psi>0$  is the flow region inside which the flow equations are solved and the region having  $\psi<0$  represents the volume of the spheres. The solid fraction of the bed of spheres for a given sphere radius  $R = D/2$  is given by

$$f_s = V_s / V_{\text{tot}}, \quad (21)$$

where  $V_s$  is the volume of one sphere and  $V_{\text{tot}} = L^3$  is the volume of the cubic box of size  $L$  within which the sphere is located. In dimensionless variables ( $L = 1$ ) the volume of the box is equal to unity and the volume of the sphere is given by:

$$V_s = \frac{4\pi R^3}{3} \quad \text{for } R \leq 0.5 \quad (22)$$

$$V_s = \frac{4\pi R^3}{3} - 6\frac{2\pi R^3}{3} + 6\frac{\pi L}{24}(12R^2 - L^2) \quad \text{for } R > 0.5 \quad (23)$$

The expression in Equation (23) takes into account for the spheres overlapping when  $R > 0.5$ .

**4.2.2. Numerical results.** The solution for the pressure distribution on the surface of the spheres is shown in Figure 11 for an array of five spheres of diameter 0.7. The pressure distribution and the mesh are also shown in a mid-plane along the flow direction. The mesh has 20 elements in directions transverse to the flow. As can be seen, the solid boundary of the spheres cuts through the elements, as the mesh is not constructed to fit the geometry of the immersed objects. The flow field for the same case is presented in Figure 12 in the plane containing the centers of the spheres indicating that homogeneous velocities are recovered on sphere surface.

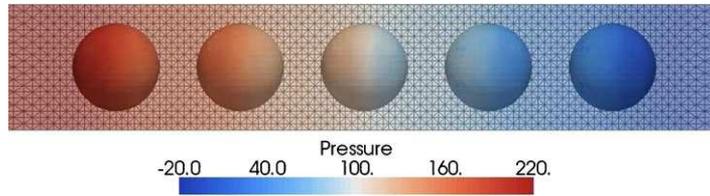


Figure 11. Distribution of the pressure and computational mesh.

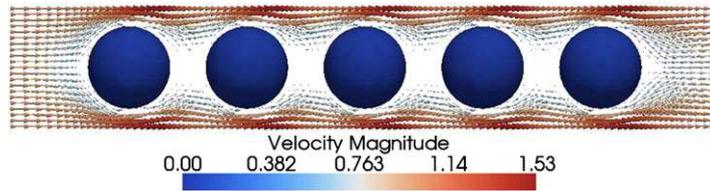


Figure 12. Flow field around the spheres.

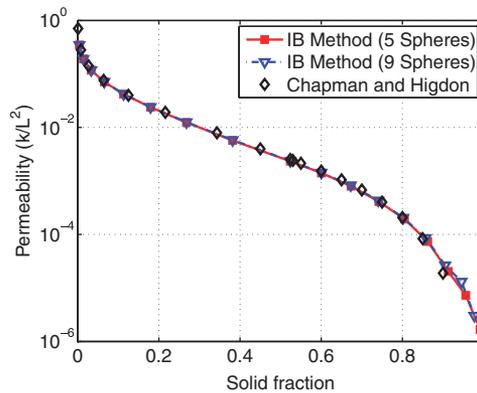


Figure 13. Permeability of an array of spheres.

Hereafter we present the results obtained for the pressure drop in the flow and the resulting permeability coefficient for various diameters of the spheres. The permeability  $k$  of the array of spheres is determined from Darcy's law and is given by

$$k = \frac{\mu \bar{u}}{|\nabla p|}, \quad (24)$$

where  $\bar{u}$  is the average flow rate.

The solution for the permeability of the array of spheres is shown in Figure 13 as a function of the solid fraction. The results obtained by Chapman and Higdon [24] are also presented as a reference. Similar results were reported by Martys *et al.* [23] using a lattice Boltzmann method. As can be seen, the permeability decreases as the solid fraction increases, reaching a value close to zero when the solid fraction approaches the maximum value of 1. The present numerical results agree very well with those reported by Chapman and Higdon [24].

The present FE-IB method is sensitive to the position of the solid boundary inside mesh elements. One limitation lies in the fact that given the linear interpolation functions used by the finite element method, the boundary is represented by planar surfaces inside mesh elements. Thus, the accuracy with which the boundary is represented depends on the local mesh size. In order to assess the sensitivity of the IB method to changes in the location of the solid boundary within a mesh element, computations were carried out by varying the spheres diameter from 0.7 to 0.8 with a constant

increment of 0.01. The results are plotted in Figure 14(a). Furthermore, very small changes in the location of the solid boundary were simulated by considering the diameter in the range between 0.7 and 0.71 with constant increments of 0.001. Recall that the mesh size is 0.05, which means that each change in the radius of the spheres determines a displacement of the interface with a distance one hundred times smaller than the element size. The results from this set of computations are shown in Figure 14(b). As can be seen, the present numerical method is able to represent accurately even extremely small changes in the location of the IB.

4.3. Transient flow past a circular cylinder

4.3.1. Problem statement. The computational domain and boundary conditions for this problem are shown in Figure 15. A steady velocity profile was imposed at the inflow. The cylinder is located at 10 diameters from the inlet and at 20 diameters from the outlet. Because the problem is two-dimensional, only a slab was meshed with 3-D tetrahedral elements. The mesh is shown in Figure 16 and was designed to have smaller elements in the region of the immersed cylinder and in the wake of the cylinder where the solution is expected to exhibit larger time variations. The grid has 300 elements in flow direction, 200 elements in direction normal to the flow and only 2 elements are used in the width of the slab for a total of 600 000 tetrahedral elements.

4.3.2. Flow solution. Computations were carried out for the Reynolds number  $Re = \rho U_0 d / \mu$  equal to 20, 40, 80, 100, 125 and 200. For  $Re = 20$  and 40 steady-state solutions are obtained presenting two symmetric recirculation regions in the wake of the cylinder as shown in Figure 17. The length of the recirculation zone ( $L/d$ ) is compared with the experimental and computational results in

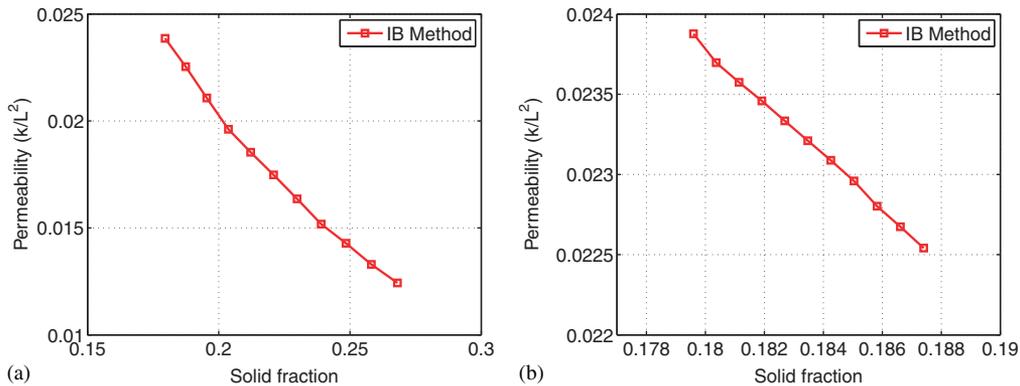


Figure 14. Permeability for small changes in sphere diameter: (a) diameter increments of 0.01 ( $D=0.7$  to 0.8) and (b) diameter increments of 0.001 ( $D=0.7$  to 0.71).

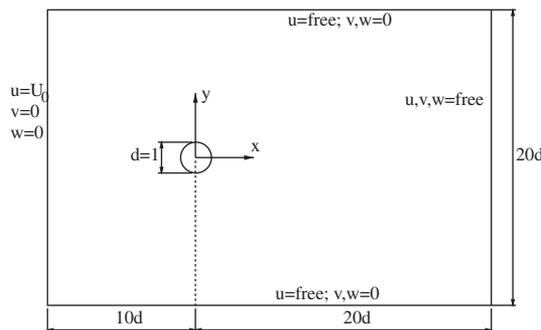


Figure 15. Uniform flow around a circular cylinder: domain and boundary conditions.

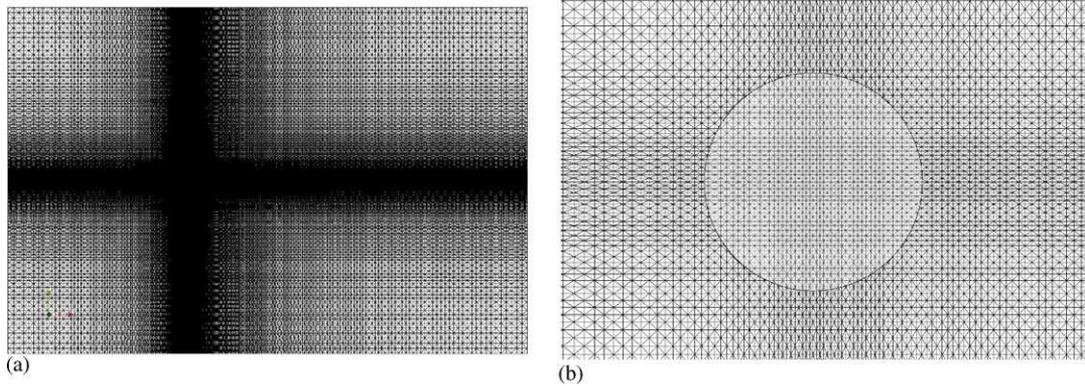


Figure 16. Mesh for the flow around a circular cylinder: (a) global view of the mesh and (b) detail around the immersed cylinder.

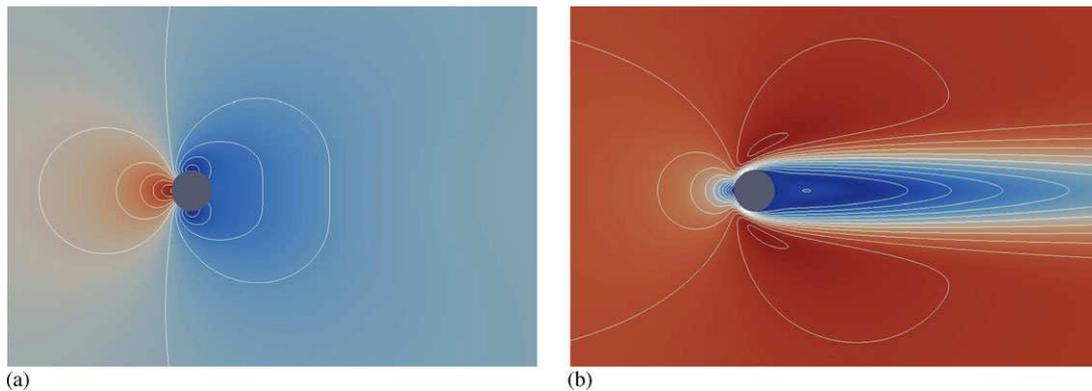


Figure 17. Solution for flow around circular cylinder at  $Re=40$ : (a) pressure and (b) velocity.

Table II. Length of the recirculation zone at low  $Re$ .

	$(L/d)$ at $Re=20$	$(L/d)$ at $Re=40$
Experiment [25]	0.73	1.89
IB method [26]	1.05	2.59
Numerical results referred in [26]	0.91–1.04	2.18–2.55
Present IB method	0.94	2.27

Table II. The present IB solution is close to the lower limit of previous numerical results, which is in good agreement with the experimental data.

For Reynolds numbers higher than the critical Reynolds number  $Re_{cr}=51$  [27] a vortex street develops in the wake of the cylinder. Following the work of Sohankar [27] the time-step is set to  $\Delta t=0.025d/U_0$ . The initial conditions are obtained from steady state solutions at  $Re$  up to 125. For  $Re=200$  the steady state solution at  $Re=125$  was used as an initial solution and the time step was set at  $\Delta t=0.0125d/U_0$ . Figure 18 shows the time variation of the flow velocity at a point located on the symmetry axis two diameters downstream of the center of the cylinder. The signal is shown from time  $t=0$  to  $160d/U_0$ . Notice that during the first part of the simulation the transverse velocity  $v$  is zero because the initial solution is symmetric with respect to the  $x$ -axis. The axial velocity  $u$  is also constant until approximately  $tU_0/d=40$ , after which time perturbations are observed and the solution is no longer symmetrical. The amplitude of these perturbations increases in time and leads to the formation of the well-known Karman vortex street. Vorticity

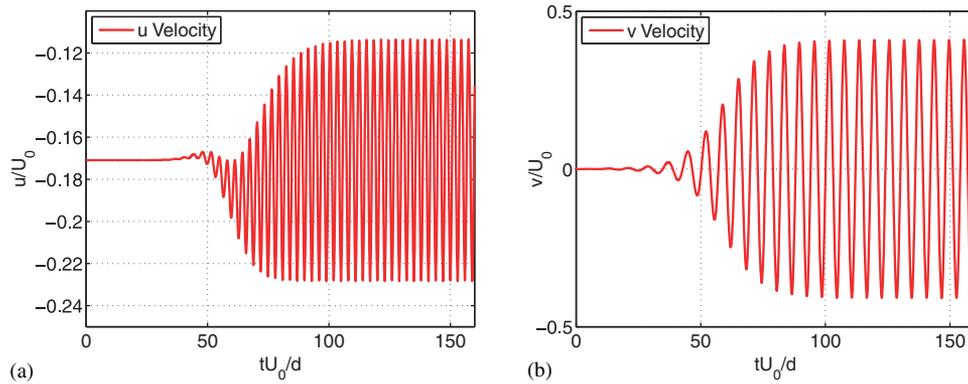


Figure 18. Uniform flow around a circular cylinder: flow response at  $(x=2, y=0)$ : (a)  $u$ : velocity and (b)  $v$ : velocity.

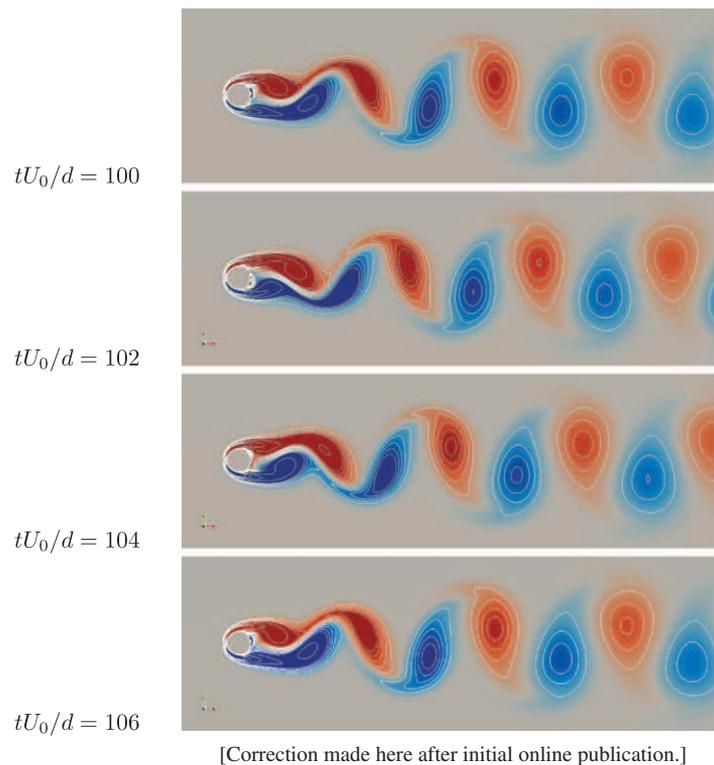


Figure 19. Uniform flow around a circular cylinder: Karman vortex street at  $Re=100$ .

contours are shown in Figure 19 from  $tU_0/d=100$  through  $tU_0/d=106$  clearly illustrating the periodic Karman vortex street.

*4.3.3. Strouhal number of the periodic flow.* The Strouhal number  $St$  is a standard non-dimensional representation of the vortex shedding frequency. It is defined as the ratio of the characteristic frequency of the flow response  $f$  to the inverse of the fluid dynamics time scale  $d/U_0$ , that is the time it takes for a fluid particle moving at velocity  $U_0$  to travel the distance  $d$ :

$$St = \frac{fd}{U_0} \tag{25}$$

It is well known that the Strouhal number depends on the Reynolds number of the flow. The present numerical predictions are compared with the experimental and computed data in Table III. The agreement with the experimental correlation of Williamson [28] is very good. The results reported by Ilinca *et al.* [29] were obtained using the same finite element method as used in the present work but on a body-fitted grid. Remark that the present IB method leads to the predictions closer to the experiment than those obtained on the body-fitted grid. This difference however is marginal and can be explained by the fact that in [29] the computational domain had the lower and upper boundaries closer to the cylinder.

#### 4.4. Flow past an NACA0012 profile

For this test problem we consider the flow past an NACA0012 airfoil at an angle of incidence of  $3.59^\circ$ . Computations were carried out for  $Re=100, 400, 2000$  and  $5000$  and the solutions were compared with those obtained on a very refined body-fitted mesh. The resulting lift coefficient is given in Table IV indicating that the IB solution compares extremely well with the solution from the body-fitted mesh. The difference is very small at low Reynolds number and increases slightly for the largest  $Re$ . The two solutions for  $Re=5000$  are shown in Figure 20 for the pressure and  $u$ -velocity contours. The pressure coefficient on the airfoil is shown in Figure 21. The two solutions are almost superimposed one over each other, thus indicating that the proposed IB method performs well.

Table III. Strouhal number at various flow regimes.

	$Re=80$	$Re=100$	$Re=125$	$Re=200$
Experiment [28]	0.153	0.164	0.175	0.197
IB method [26]	0.15	0.160	—	0.192
Numerical results referred in [26]	0.15	0.160–0.175	—	0.187–0.202
Ilinca <i>et al.</i> [29]	0.157	0.168	0.178	—
Present IB method	0.155	0.166	0.176	0.197

Table IV. Lift coefficient for NACA0012 at  $3.59^\circ$ .

	$Re=100$	$Re=400$	$Re=2000$	$Re=5000$
Body-fitted grid	0.1045	0.1015	0.0807	0.0480
Present IB method	0.1053	0.1016	0.0813	0.0502

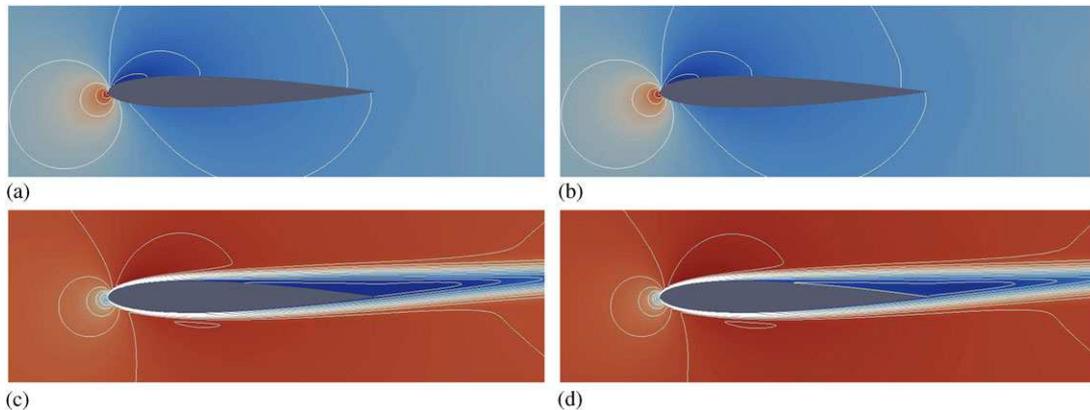


Figure 20. Flow past an NACA0012 airfoil at  $Re=5000$ : (a) pressure: IB method; (b) pressure: body-fitted grid; (c)  $u$  velocity: IB method; and (d)  $u$  velocity: body-fitted grid.

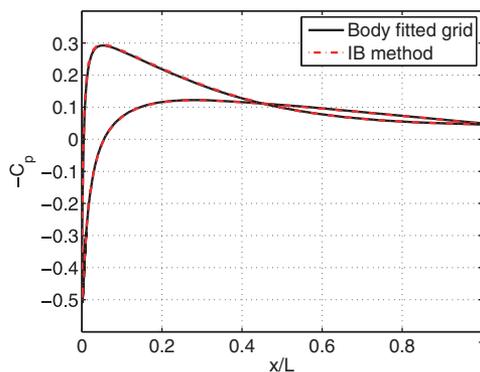


Figure 21. Pressure coefficient for NACA0012 at  $Re = 5000$ .

## 5. CONCLUSIONS

A new IB finite element method that accurately imposes the boundary conditions on the immersed interface is presented. The procedure consist of incorporating into the grid the points where the mesh intersects the boundary. The degrees of freedom associated with the additional grid points are then eliminated either because the velocity is known or by static condensation in the case of the pressure.

Application of the proposed method to the flow in an obstructed channel shows an excellent agreement with the body-fitted grid solution. The three-dimensional IB finite element method was also used to solve the flow around spheres placed on a simple cubic lattice. The resulting permeability was computed as a function of the solid fraction and the results agree very well with previously computed solutions. The method is shown to detect even the smallest changes in the shape of the IB. Computations for different values of the sphere diameter are obtained on the same mesh by simply changing the level set function thus reducing the time of performing this type of simulation.

For the application to transient flow past a circular cylinder the results compare well with experimental data and previous numerical results. The computed Strouhal number for the periodic vortex shedding is in excellent agreement with the experimental correlation of Williamson. Finally, for the flow past an NACA0012 airfoil, the solution from the present IB method compares very well with a solution computed on a body-fitted grid.

The proposed method can be extended to problems involving moving solids and fluid–structure interaction. Those topics will be the subject of future investigations.

## REFERENCES

1. Peskin CS. Flow patterns around heart valves: a numerical method. *Journal of Computational Physics* 1972; **10**:252–271.
2. Peskin CS. Numerical analysis of blood flow in the heart. *Journal of Computational Physics* 1977; **25**:220–252.
3. Mittal R, Iaccarino G. Immersed boundary methods. *Annual Review of Fluid Mechanics* 2005; **37**:239–261.
4. Tyson R, Jordan CE, Hebert J. Modelling anguilliform swimming at intermediate Reynolds number: a review and a novel extension of immersed boundary method applications. *Computer Methods in Applied Mechanics and Engineering* 2008; **197**:2105–2118.
5. Kim Y, Peskin CS. 3-D parachute simulation by the immersed boundary method. *Computers and Fluids* 2009; **38**:1080–1090.
6. Margnat F, Morinière V. Behaviour of an immersed boundary method in unsteady flows over sharp-edged bodies. *Computers and Fluids* 2009; **38**:1065–1079.
7. Boffi D, Gastaldi L. A finite element approach for the immersed boundary method. *Computers and Structures* 2003; **81**:491–501.
8. Boffi D, Gastaldi L, Heltai L. On the CFL condition for the finite element immersed boundary method. *Computers and Structures* 2007; **85**:775–783.
9. Zhang L, Gerstenberger A, Wang X, Liu WK. Immersed finite element method. *Computer Methods in Applied Mechanics and Engineering* 2004; **193**:2051–2067.

10. Glowinski R, Pan T-W, Périaux J. A fictitious domain method for external incompressible viscous flow modeled by Navier–Stokes equations. *Computer Methods in Applied Mechanics and Engineering* 1994; **112**:133–148.
11. Glowinski R, Pan T-W, Hesla TI, Joseph DD, Périaux J. A distributed Lagrange multiplier/fictitious domain method for flows around moving rigid bodies: application to particulate flows. *International Journal for Numerical Methods in Fluids* 1999; **30**:1043–1066.
12. Löhner R, Cezral JR, Camelli FE, Appanaboyina S, Baum JD, Mestreau EL, Soto OA. Adaptive embedded and immersed unstructured grid techniques. *Computer Methods in Applied Mechanics and Engineering* 2008; **197**:2173–2197.
13. Löhner R, Appanaboyina S, Cezral JR. Comparison of body-fitted, embedded and immersed solutions of low Reynolds-number 3-D incompressible flows. *International Journal for Numerical Methods in Fluids* 2008; **57**:13–30.
14. Codina R, Houzeaux G, Coppola-Owen H, Baiges J. The fixed mesh ALE approach for the numerical approximation of flows in moving domains. *Journal of Computational Physics* 2009; **228**:1591–1611.
15. Tezduyar TE, Shih R, Mittal S, Ray SE. Incompressible flow using stabilized bilinear and linear equal-order-interpolation velocity–pressure elements. *Research Report UMSI 90/165*, University of Minnesota/Supercomputer Institute, Minneapolis, 1990.
16. Ilinca F, Pelletier D, Garon A. An adaptive finite element method for a two-equation turbulence model in wall-bounded flows. *International Journal for Numerical Methods in Fluids* 1997; **24**:101–120.
17. Franca LP, Frey SL. Stabilized finite element methods: II. The incompressible Navier–Stokes equations. *Computer Methods in Applied Mechanics and Engineering* 1992; **99**(2–3):209–233.
18. Mineev PD, Chen T, Nandakumar K. A finite element technique for multifluid incompressible flow using Eulerian grids. *Journal of Computational Physics* 2003; **187**:255–273.
19. Coppola-Owen AH, Codina R. Improving Eulerian two-phase flow finite element approximation with discontinuous gradient shape functions. *International Journal for Numerical Methods in Fluids* 2005; **49**:1287–1304.
20. Coppola-Owen AH, Codina R. A finite element model for free surface flows on fixed meshes. *International Journal for Numerical Methods in Fluids* 2007; **54**:1151–1171.
21. Carvalho MG, Durst F, Pereira JCF. Predictions measurements of laminar flow over two-dimensional obstacles. *Applied Mathematical Modelling* 1987; **11**:23–34.
22. Guibault F, Garon A, Ozell B, Camarero R. Analysis and visualisation tools in CFD, Part II: a case study in grid adaptivity. *Finite Elements in Analysis and Design* 1995; **19**:309–324.
23. Marty NS, Hagedorn JG, Goujon D, Devaney JE. Large scale simulations of single and multi-component flow in porous media. *Proceedings of the SPIE Conference on Developments in X-Ray Tomography II*, vol. 3772. SPIE: Bellingham, WA, 1999; 205–213.
24. Chapman AM, Higdon JLL. Oscillatory Stokes flow in periodic porous media. *Physics of Fluids A* 1992; **4**:2099–2116.
25. Coutanceau M, Bouard R. Experimental determination of the main features of the viscous flow in the wake of a circular cylinder in uniform translation. Part 1, steady flow. *Journal of Fluid Mechanics* 1977; **79**:231–256.
26. Le DV, Khoo BC, Lim KM. An implicit-forcing immersed boundary method for simulating viscous flows in irregular domains. *Computer Methods in Applied Mechanics and Engineering* 2008; **197**:2119–2130.
27. Sohankar A, Norberg C, Davidson L. Low-Reynolds-number flow around a square cylinder at incidence: study of blockage, onset of vortex shedding and outlet boundary condition. *International Journal for Numerical Methods in Fluids* 1998; **26**:39–56.
28. Williamson CHK. Defining a universal and continuous Strouhal–Reynolds number relationship for the laminar vortex shedding of a circular cylinder. *Physics of Fluids* 1988; **31**:2742–2744.
29. Ilinca F, Pelletier D, Hay A. First- and second-order sensitivity equation methods for value and shape parameters. *International Journal for Numerical Methods in Fluids* 2008; **57**:1349–1370.