

Immersed Boundary Methods: Historical Perspective and Future Outlook

Roberto Verzicco^{1,2,3}

¹Dipartimento di Ingegneria Industriale (DII), Università di Roma “Tor Vergata,” Rome, Italy;
email: verzicco@uniroma2.it

²Gran Sasso Science Institute, L’Aquila, Italy

³Physics of Fluids Group, University of Twente, Enschede, The Netherlands

Annu. Rev. Fluid Mech. 2023. 55:129–55

First published as a Review in Advance on
September 23, 2022

The *Annual Review of Fluid Mechanics* is online at
fluid.annualreviews.org

<https://doi.org/10.1146/annurev-fluid-120720-022129>

Copyright © 2023 by the author(s). This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited. See credit lines of images or other third-party material in this article for license information.

ANNUAL
REVIEWS **CONNECT**

www.annualreviews.org

- Download figures
- Navigate cited references
- Keyword search
- Explore related articles
- Share via email or social media

Keywords

immersed boundary methods, complex flows, wall models, multiphysics problems

Abstract

Immersed boundary methods (IBMs) are versatile and efficient computational techniques to solve flow problems in complex geometric configurations that retain the simplicity and efficiency of Cartesian structured meshes. Although these methods became known in the 1970s and gained credibility only in the new millennium, they had already been conceived and implemented at the beginning of the 1960s, even if the early computers of those times did not allow researchers to exploit their potential. Nowadays IBMs are established numerical schemes employed for the solution of many complex problems in which fluid mechanics may account for only part of the multiphysics dynamics. Despite the indisputable advantages, these methods also have drawbacks, and each problem should be carefully analyzed before deciding which particular IBM implementation is most suitable and whether additional modeling is necessary. High-Reynolds number flows constitute one of the main limitations of IBMs owing to the resolution of thin wall shear layers, which cannot benefit from anisotropic grid refinement at the boundaries. To alleviate this weakness, researchers have developed IBM-compliant wall models and local grid refinement strategies, although in these cases possible pitfalls must also be considered.

1. INTRODUCTION

Numerical simulations, next to theory and experiments, are considered the third paradigm for the advancement of science and technology thanks to computer models that are routinely used for the analysis of complex systems. One of the branches of computational science that has benefited the most from powerful computers is computational fluid dynamics (CFD) owing to the nonlinear nature of the governing equations, which restrict the derivation of analytical solutions to overly simplified configurations. This fact was clearly recognized by John von Neumann, who already in 1940 had written: “Our present analytical methods seem unsuitable for the solution of important problems arising in connection with nonlinear partial differential equations. . . . The truth of this statement is particularly striking in the field of fluid dynamics” (Goldstine 1980, pp. 179–80).

In fact, the common strategy of any CFD method is to tessellate the fluid domain into several discrete elements to transform the original set of partial differential equations into a system of algebraic equations efficiently tackled by numerical algorithms. For problems of practical interest, however, the presence of complex domains, possibly with elements in relative motion, makes the discretization step an extremely difficult and time-consuming procedure; the resulting body-fitted grids are curvilinear and not guaranteed to be orthogonal, thus requiring complex solution algorithms with significant overhead in the per-cell operation count and reduced precision.

Immersed Boundary Methods (IBMs) have emerged as a strategy to alleviate these issues and to allow for flow simulations in complex geometric configurations without running into the difficulties of body-conformal meshes, thus significantly easing the application of CFD to practical problems. More generally, IBMs encompass a variety of numerical methods aimed at solving systems of partial differential equations on meshes not conforming to the physical boundaries; they have been successfully adopted in fields such as solid mechanics (Rangarajan et al. 2009), electromechanical systems (Andreykiv & Rixen 2009), crack propagation (Möes et al. 1999), 3D elasticity (Schillinger et al. 2016), and electromagnetic scattering (Alkhateeb & Tsukerman 2013).

These methods, however, have become especially popular in fluid mechanics, and in the last two decades the field has grown at an increasing rate without signs of saturation. In **Figure 1** we present the number of scientific publications per year containing the term “immersed boundary” or “immersed boundaries” in the title, and it is evident that there has been explosive growth since the start of the millennium. The empirical data fit yields $N \approx 5t^2$ (with N the number of entries and t the year after 2000), indicating a constant acceleration of 10 papers per year squared. Furthermore, as evident from the inset of **Figure 1**, the interest in IBMs had begun to increase already in the 1990s, with the first paper dating back to the seminal work by Peskin (1972), in which the term “immersed boundaries” was first introduced. Indeed, initially only Peskin’s group was active in the field, with applications to biological flows, and this earned him the title of the Father of Immersed Boundary. However, starting from the 1990s and stimulated by the growth of computing power, the application of IBMs has expanded to many other fields; in 2021 publications in the field include electromagnetism, turbulent combustion, piezoelectric materials, interface chemistry, multibody dynamics, and Brownian motion, in addition to the overwhelming fluid dynamic examples.

The main limitation of the above analysis is that the search is limited to “immersed boundary,” which does not cover all the synonyms referring to analogous methods: embedded boundaries, immersed interfaces, boundary body forces, penalty methods, and distributed Lagrange multipliers, to give just a few examples. Furthermore, there are also many papers whose titles do not hint at IBMs at all, and only reading them unveils their real content. When considering all of this additional literature, it appears that predecessors of IBMs had been proposed already in the early 1960s, although the primordial computers available by that time prevented an effective use of the

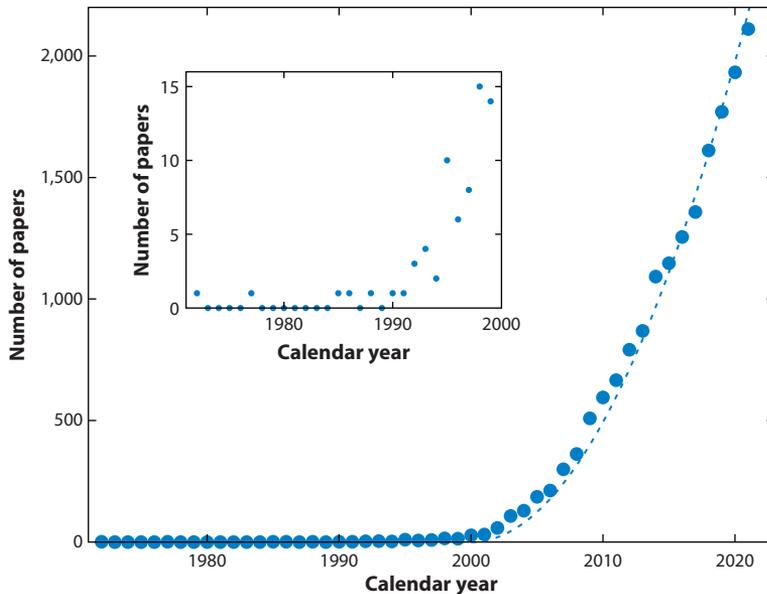


Figure 1

Number of scientific papers containing “immersed boundary” or “immersed boundaries” in the title by publication year. The inset shows the same but with the publication year restricted to 1970–2000. The dashed line is the fit $N \approx 5t^2$, with $t = x - 2000$ and x the publication year. Data from Scopus (accessed January 31, 2022).

methods. A more detailed account of this material and the relations with current techniques are given in Section 3.

As the popularity of IBMs has increased and the range of applications broadened, review papers covering relevant aspects of this methodology have also been published. The paper by Peskin (2002) is concerned with the mathematical structure of the IBM aimed at fluid–structure interactions (FSIs) for biological fluid dynamics. Iaccarino & Verzicco (2003) discussed the use of IBMs for simulations of turbulent flows in industrial problems and also included examples using large eddy simulations (LES) and Reynolds-averaged Navier–Stokes (RANS) turbulence modeling. There are also previous reviews in this journal, such as that by Mittal & Iaccarino (2005), who described the main techniques used for flows around rigid bodies, moving and at rest, or more recently that of Griffith & Patankar (2020), who illustrated the latest developments of IBMs with FSIs for biological systems. Indeed, there is a further review paper in this journal dealing with IBMs (Peskin 1982), although it is devoted to the fluid mechanics of heart valves, and numerical methods are briefly mentioned only as a means to study FSIs in the context of cardiovascular flows.

The present review is an attempt to fill the gaps among the various reviews and to give a complementary account of the subject. By necessity the result is somewhat cross-eyed since it looks at the past to describe the earliest attempts to develop IBMs before Peskin (1972), as well as to the future to anticipate the next developments of the method.

Writing this review has required confronting opposing challenges since, on the one hand, the literature on the origins of IBMs is scarce and access to it is limited. Neither the name “immersed boundary” nor its numerous synonyms had been coined yet in these earliest works; therefore, a laborious reading of old reports and forgotten papers has been necessary to unearth the early history of these ideas.

On the other hand, as shown in **Figure 1**, the current scientific publication on IBMs exceeds 2,000 papers per year, and keeping up with everything is impossible. Within this context, making delimiting choices is unavoidable, and in this review I have decided to focus on the main limitation of the method (i.e., near-wall modeling in high-Reynolds number flows) and on its most promising applications—multiphysics problems with active interfaces occurring at small (or micro) scales. In the next section, we provide a concise description of the method with its main benefits and pitfalls, which is used as a background for the subsequent discussion.

2. PROBLEM FORMULATION AND BACKGROUND

In this section we summarize the main steps needed to construct an IBM. In order to link the discussion to a real problem, we consider a relatively simple flow that could be easily handled by both body-fitted and nonconformal meshes: a sphere of radius R placed at a distance D from a plane wall and subject to a uniform stream with unperturbed velocity \mathbf{U} .

In the standard approach (**Figure 2a**), a bispherical coordinate system can be employed,

$$x = a \frac{\sin \xi \cos \phi}{\cosh \eta - \cos \xi}, \quad y = a \frac{\sin \xi \sin \phi}{\cosh \eta - \cos \xi}, \quad z = a \frac{\sinh \eta}{\cosh \eta - \cos \xi}, \quad 1.$$

such that the sphere is a coordinate surface at constant $\eta = \bar{\eta}$ (with center $x = 0, y = 0, z = a \coth \bar{\eta}$), with radius $R = a / \sinh \bar{\eta}$ and distance $D = a(\cosh \bar{\eta} - 1) / \sinh \bar{\eta}$, while the flat wall is given by $\eta = 0$. In this system, the imposition of boundary conditions is simply achieved by prescribing a variable, or some of its derivatives, on mesh elements with a constant coordinate η , which is also the wall-normal direction.

On the other hand, the grid of **Figure 2b** does not conform to the sphere, and several ingredients are needed in order to enforce boundary conditions on a surface crossing mesh elements in the middle of the computational domain. The first task is to identify the grid points and classify them as internal, external, or interface; for a simple geometry like the sphere in **Figure 2** this

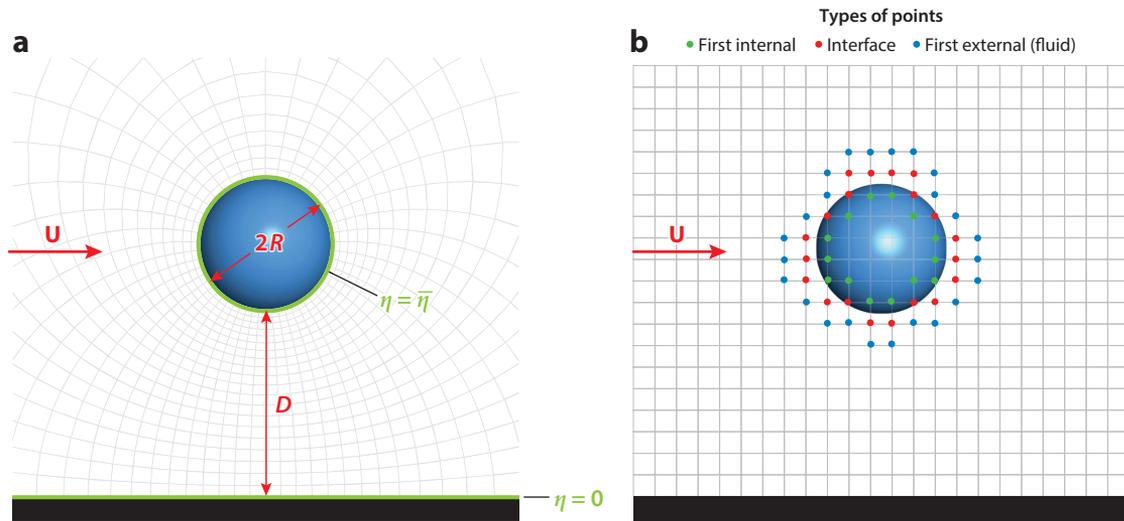


Figure 2

Possible spatial discretizations for the flow around a sphere of radius R at a distance D from a flat solid boundary. \mathbf{U} is the uniform, undisturbed horizontal velocity. (a) Body-fitted discretization using bispherical coordinates, with $\eta = 0$ giving the flat wall and $\eta = \bar{\eta}$ the sphere surface. (b) Cartesian discretization for immersed boundaries.

is straightforward, although the procedure is much more involved for a generic complex object like a road vehicle (Jindal et al. 2005), whose geometry cannot be described by simple analytical relations. In the latter case automatic procedures can be used, and reliability, more than computational cost, might be an issue, as the procedures are performed only once at the beginning of the simulation. An efficient strategy, derived from computer graphics and based on a ray-tracing algorithm, was extensively described by O'Rourke (1998).

After identifying the interface nodes, the second step consists in defining a procedure to enforce the boundary conditions at the fluid/body interface. Most of IBM methods achieve this goal by adding, explicitly or implicitly, a volumetric forcing term to the governing equations. For example, for momentum, in the Navier–Stokes equations,

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = -\frac{\nabla p}{\rho} + \nu \nabla^2 \mathbf{u} + \mathbf{f}, \quad 2.$$

the term $\mathbf{f}(\mathbf{x}, t)$ is dynamically adjusted in space and time to account for the presence of the immersed body.

There are several possible expressions for \mathbf{f} and they depend on the specific IBM technique. A very broad class of methods relies on a network of material Lagrangian markers, with positions \mathbf{X}_i , distributed over the immersed boundary and subjected to external and internal loads, the former caused by the hydrodynamic interactions and the latter depending on the boundary deformation. Each marker is thus subjected to a total forcing \mathbf{F}_i , which determines its dynamics (via Newton's second law of motion) and the reaction on the flow (via Newton's third law of motion). However, since markers are material points, each \mathbf{F}_i is localized in space as a Dirac delta function, and evaluating \mathbf{f} as $\sum_i \mathbf{F}_i \delta(\mathbf{x} - \mathbf{X}_i)$ would produce wiggles and instabilities in the numerical solution. To avoid this problem we substitute the sharp delta functions by smoother distributions $\widehat{\delta}(\mathbf{x} - \mathbf{X}_i)$ with a finite-size support, which spread \mathbf{F}_i over surrounding volumes and distribute \mathbf{f} among all the Eulerian grid points therein. There are various expressions for these regularized delta functions $\widehat{\delta}$, and they are crucial for the smoothness of the solution and conservation properties; examples are given by Peskin (1972), Saiki & Biringen (1996), and Uhlmann (2005), and a more general discussion is given by Griffith & Patankar (2020).

In another group of IBMs, often called feedback forcing methods, the local forcing \mathbf{f} is computed as a function of the difference between the actual and the desired values of the unknown. For the momentum Equation 2, with \mathbf{u}_b the velocity to be imposed at interface nodes, we can write $\mathbf{f} = \alpha \int_0^t (\mathbf{u} - \mathbf{u}_b) d\tau + \beta (\mathbf{u} - \mathbf{u}_b)$ (Goldstein et al. 1993, Saiki & Biringen 1996), where α and β are negative, user-defined parameters whose function is to dynamically drive \mathbf{u} to \mathbf{u}_b at every time step. This forcing can be considered as an application of the proportional integral derivative control to the signal error $\mathbf{u} - \mathbf{u}_b$, and this technique is widely used in many engineering fields (Aström & Murray 2010).

Penalty methods (Angot et al. 1999, Specklin & Delauré 2018) can be thought of as a particular case of the feedback approach since the body is treated as a porous volume that produces a momentum loss inversely proportional to a solidity parameter: In this context, the forcing term becomes $\mathbf{f} = (\mathbf{u} - \mathbf{u}_b)/K$, which is a feedback control with $\alpha = 0$ and $\beta = 1/K$.

It is easy to show that the combination of the time derivative term of Equation 2 with the feedback forcing can be interpreted as a damped oscillator of period $T = 2\pi/\sqrt{-\alpha - \beta^2/4}$ and time constant $-2/\beta$ (α and β are both negative). These quantities must be smaller than the integration time step if \mathbf{u} is to be forced to \mathbf{u}_b within Δt , and this imposes a minimum magnitude for α and β . In principle, the larger the values of these constants the smaller the difference $|\mathbf{u} - \mathbf{u}_b|$, although increasing α and β stiffens the equation and makes its integration more difficult. In practice, α and β are chosen as a compromise between acceptable boundary conditions and integration steps Δt ; thus, the results are unavoidably affected by user-defined parameters.

In order to avoid the use of nonobjective quantities, Mohd-Yusof (1997) considered the time-discrete version of Equation 2 of the form

$$\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} = \text{RHS}^{n+1/2} + \mathbf{f}, \quad 3.$$

with $\text{RHS}^{n+1/2}$ containing all the right-hand side of Equation 2 except for the forcing. This equation can be solved for \mathbf{f} once the velocity \mathbf{u}^{n+1} is replaced by \mathbf{u}_0 , at the interface points. This method neither requires ad hoc parameters nor alters the time stability of the integration, thus allowing complex flow simulations at the same cost as that of simple geometry counterparts. Fadlun et al. (2000) proved that this method retains second-order accuracy and also works in combination with LES turbulence models, thus largely extending its applicability.

Another key step, common to all IBMs, is the boundary reconstruction, which is needed in order to satisfy boundary conditions from forced interface points at some distance from the wall. More in detail, looking at **Figure 2b**, it is evident that the immersed surface crosses coordinate lines at positions not coinciding with the location of the unknowns. The situation is therefore that of **Figure 3** with the variable at the interface node A , which must be determined in such a way so as to satisfy the boundary value at the wall W . As an example, we show one of the simplest choices, which is to assume a linear expression for the velocity next to the immersed boundary,

$$\mathbf{u}(x, y, z) = ax + by + cz + d, \quad 4.$$

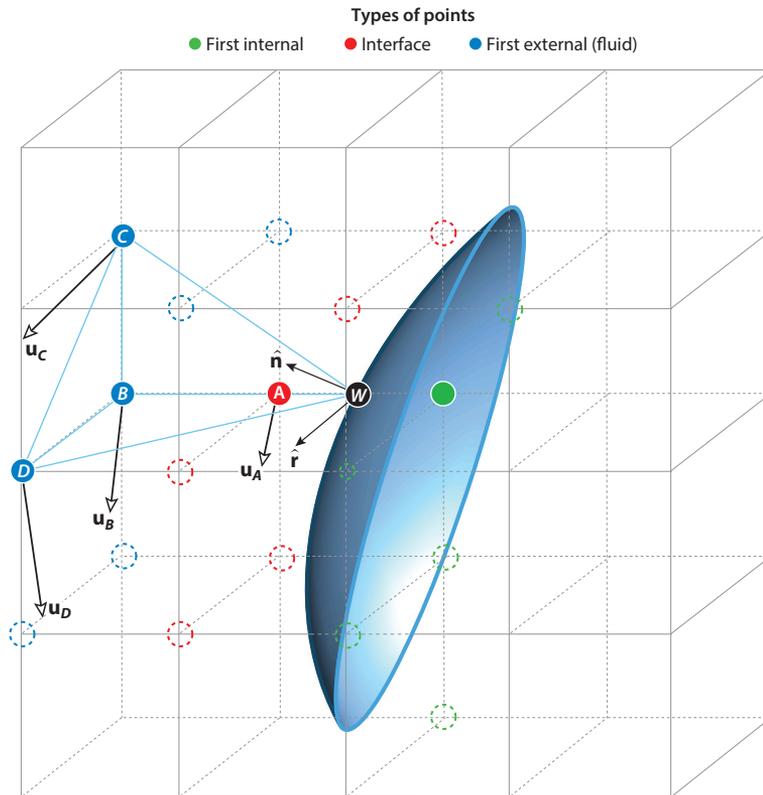


Figure 3

A possible arrangement for the boundary reconstruction using a linear interpolation, as in Equation 4. The open arrows indicate the local velocity vector.

and to determine the four constants by matching the values in B , C , D , and W ; the first three values come from the solution of Equation 2 on noninterface points, while the last is the wall boundary condition. Once the coefficients a , b , c , and d are computed, Equation 4 evaluated in A yields the value of \mathbf{u}_b to be used in one of the above forcing expressions. Note that the same approach can be adopted to impose a condition on the derivative at the wall by replacing the Dirichlet boundary condition with $\nabla \mathbf{u} \cdot \hat{\mathbf{r}}|_W = \phi_W$, with $\hat{\mathbf{r}}$ the generic unit vector at the wall and ϕ_W the prescribed value (for $\hat{\mathbf{r}} = \hat{\mathbf{n}}$, a standard surface flux boundary condition is imposed).

If the interface point is not close enough to the immersed boundary, the linear interpolation of Equation 4 might be too crude and more accurate reconstructions are necessary. Kang et al. (2009) proposed quadratic polynomials or quadratic polynomials plus a pressure gradient or momentum balance effects, while Vannella & Balaras (2009) adopted a very effective moving least-squares reconstruction to obtain a smooth distribution of surface forces. The common feature of any technique is that \mathbf{u}_b can be reconstructed as a combination of surrounding values $\mathbf{u}_b = \sum_{j=1}^N \alpha_j \mathbf{u}_j$, and that complex interpolating functions entail a large number of valid fluid nodes N , thus extending the spatial support of the stencil and reducing the local nature of the interpolation.

Despite the necessary tagging, forcing, and reconstruction steps, IBMs have computational advantages over the body-conformal approach since the resulting numerical method requires fewer operations and less communication per grid point and IBMs are more compatible with direct solvers and parallel algorithms. For moving or deforming objects, when a body-fitted grid has to be deformed or remeshed in time, IBMs become even more convenient, although some of the above points need additional care. In fact, as the body moves across the fixed grid, the computational efficiency of node tagging, which must be performed at every time step, becomes key for the feasibility of the simulation. Cells that change status in a time step (from internal to interface or external) are another issue since they do not have a valid time history to compute the explicit part of the fluid dynamics; in these cases variables must be extended from external neighboring points by extrapolation (Yang & Balaras 2006), and reduced integration time steps are usually employed. Finally, in the case of deforming boundaries, not only the position but also the shape of the interface is part of the solution: In these FSI problems, time stability is further reduced by added-mass effects (Borazjani et al. 2008) and by the auxiliary models used to compute the boundary configuration (Viola et al. 2020). These aspects have been extensively explained by Griffith & Patankar (2020) and are not discussed further in this review.

According to the previous section, IBMs have advantages but also drawbacks compared to the classical body-fitted methods, and as they are strongly problem dependent, each case should be carefully evaluated before choosing the most convenient approach. The most evident advantage of conformal meshes is that the imposition of boundary conditions is greatly simplified, as it occurs on coordinate lines without any intermediate step. The possibility of a single wall-normal direction is another important bonus since the mesh can be refined in the layers adjacent to the boundaries to properly capture the steep flow gradients at the wall without overloading the rest of the mesh. Unfortunately, grids like that of **Figure 2a** are exceptional, and they are unavailable for generic complex geometries. For configurations of practical interest, meshes are typically nonorthogonal—either unstructured or made of separate blocks that require interpolations at their overlapping regions. Furthermore, even for a simple configuration like that of **Figure 2a**, problems arise for $D \ll R$ (**Figure 4a**) owing to the coordinate lines squeezed between the sphere and wall, which make the simulation terminally expensive and the mesh eventually singular for $D \rightarrow 0$. This issue prevents the use of similar meshes in all flows involving the contact of two bodies, like the wall rebound of a ball in a viscous fluid (Zenit & Hunt 1999), the rolling of a body on a surface (Pirozzoli et al. 2012), or the closure of a heart valve (Verzicco 2022). IBMs clearly do not suffer from this limitation, and the decoupling of discretization from geometry yields

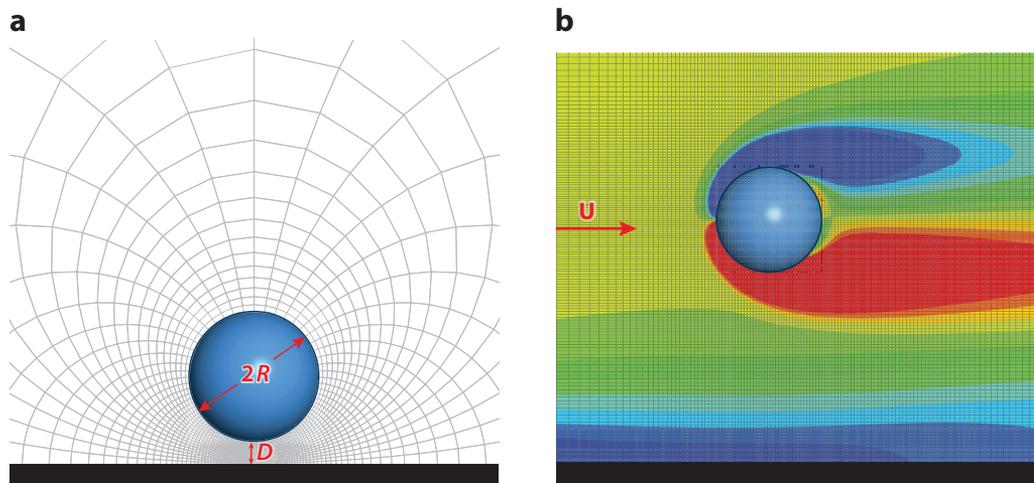


Figure 4

(a) The same discretization as in **Figure 2a** but for $D/R = 0.2$; the squeezing of coordinate lines within the gap between sphere and wall and the concurrent mesh coarsening on the opposite side of the sphere makes the simulation infeasible for $D/R \rightarrow 0$. (b) Numerical simulation of the flow using immersed boundary methods at a Reynolds number $Re = UR/\nu = 100$, with ν the viscosity. Contours are isolevels of out-of-plane vorticity and values range from -1 (blue) to 1 (red). Overlaid light gray lines are the actual Cartesian mesh. Note that inside the sphere in panel *b* a secondary flow has developed since, for this simulation, the inner points have not been forced.

simulations whose computational cost is independent of D . As D tends to 0, however, we face resolution issues because the number of grid points within the gap is insufficient for an accurate boundary reconstruction. Verzicco & Querzoli (2021) have found that when the fluid layer between two immersed boundaries is discretized by less than four points, the boundary reconstructions from the two opposing surfaces interfere and produce inaccurate or even unphysical results. In this case, however, the local Reynolds number tends to be very small and a lubrication solution can be used in lieu of an IBM reconstruction (Kempe et al. 2014, Costa et al. 2015, Picano et al. 2015).

One of the main disadvantages of IBMs is the absence of wall-normal and wall-parallel coordinates, which precludes the possibility of confining the grid refinement within near-wall layers. In a case like that of **Figure 4b**, the mesh must be refined in all directions in order to properly resolve the boundary layers at the sphere surface, and this yields a discretization that is unnecessarily fine in the bulk, where it would not be needed. The problem worsens as the Reynolds number increases owing to the thinning of the boundary layers, and eventually IBMs become unusable for high-Reynolds number turbulent flows. In fact, following Pope (2000), in a wall-bounded turbulent flow with a viscous unit $\delta_v = \nu/u_\tau$ (where $u_\tau = \sqrt{\tau_w/\rho}$ is the friction velocity and τ_w the wall shear stress), the grid spacing in the wall-normal direction can be taken to be proportional to the distance from the wall itself, which for a body-fitted mesh yields $\Delta_\perp \sim \eta$. On the other hand, in the wall-parallel directions, the grid spacing is proportional to the viscous unit $\Delta_\parallel \sim \delta_v$; thus, for a computational domain with dimensions proportional to some large-scale length L and an empirical correlation $L/\delta_v \sim Re^{0.9}$ (Reynolds 1990), the total number of grid points per cubic L increases as $N_{BF} \sim Re^{1.8} \log Re$.

If the mesh is not body fitted, any direction can be normal or tangential to the boundaries and the node distribution cannot benefit from near-wall clustering; this implies that the spacing must be proportional to δ_v in all directions and the total number of nodes per cubic L scales as $N_{IB} \sim Re^{2.7}$. Clearly the scaling is unfavorable for IBMs, and even considering the operation count per

node, which is at least one order of magnitude better for simple meshes than it is for curvilinear ones (Piomelli 2008), the ratio $N_{\text{IB}}/N_{\text{BF}} \sim Re^{0.9} \log Re$ diverges with Re , and IBMs become eventually unsuitable for high-Reynolds number flows unless additional modeling is implemented at the wall. Indeed, in several IBM applications to high-Reynolds number flows, local grid refinement (fine patches of refined mesh) is employed at the immersed surfaces, yielding a milder node count increase with Re (Durbin & Iaccarino 2002, de Tullio et al. 2007). This point is further discussed in Section 4 since it entails additional considerations on the numerical integration schemes.

Another issue in the application of IBMs to incompressible flows is the difficulty of simultaneously imposing exact velocity boundary conditions and interface cells that are locally divergence-free. In fact, although the forcing \mathbf{f} correctly enforces a given velocity via Equation 2, its value is afterward altered by the projection step onto a solenoidal field, thus generating slip and transpiration velocities at the immersed boundaries. This issue was recognized already by Fadlun et al. (2000), who proposed to iterate between Equation 2 and the divergence-free projection of the fractional step until the maximum error decreases below a given threshold. A more elegant and computationally efficient approach was proposed by Kim et al. (2001), who modified the mass conservation equation at the interface cells to make them locally divergence-free at a discrete level. Taira & Colonius (2007) introduced a new formulation in which both pressure and boundary body force were treated as distributed Lagrange multipliers; the resulting scheme was algebraically identical to a fractional step but capable of simultaneously satisfying boundary conditions and local mass conservation. Other clever amendments have been found by Uhlmann (2005), Kempe & Fröhlich (2012), and Lačis et al. (2016), although the resulting methods are mostly suited for FSIs occurring in particulate flows.

3. THE ORIGINS OF IMMERSed BOUNDARY METHODS

In the introduction we noted that IBMs have become very popular in the last two decades, although they have been used since the early 1970s when, in his seminal paper, Peskin (1972) used the 2D flow through the native mitral heart valve to show the huge potential of IBMs. The use of non-body-conformal meshes in CFD, however, is definitely older, and there have been multiple examples in which research groups have put forward the basic ideas of IBMs since the mid-1950s, when high-speed electronic computers began to be employed for the solution of differential equations. Evans & Harlow (1957) applied the particle-in-cell method of Harlow (1957) to 2D compressible flows and encountered a new difficulty in the use of reflective boundaries to mimic rigid obstacles. They observed, “with δt small enough to satisfy the accuracy criteria . . . no particle will ordinarily enter the shaded section [see figure reproduced below as **Figure 5a**]. The exception arises when the particle is in the section near the corner outlined by dots . . . [and to] overcome this [difficulty], a process may be used whereby the new coordinates of the particle . . . are replaced by . . . , which will remove the particle from the forbidden zone” (p. 26). From this statement and **Figure 5a** we recognize that constraints had been conceived to mimic the presence of boundaries in the middle of the computational domain and that special treatments were used to deal with sharp corners. Clearly the above technique could be used only for boundaries largely aligned with coordinate lines, and any inclined surface would have resulted in a stair-step distribution.

This limitation was recognized and corrected by Rich & Blackman (1962), who used a fully Eulerian finite-difference (fluid-in-cell) method to simulate 2D inviscid compressible flows. They distinguished configurations such as that in **Figure 5a**, which they called “barriers at cell boundaries,” from that of **Figure 5b**, which they called “arbitrary linear barriers within a cell.” In the former case they defined fictitious mesh cells inside the body (nowadays termed ghost cells) to enforce boundary conditions at the wall, and for the corner cell, two fictitious meshes were used depending on whether the corner was approached horizontally ($i - 1, j \rightarrow i, j$) or vertically

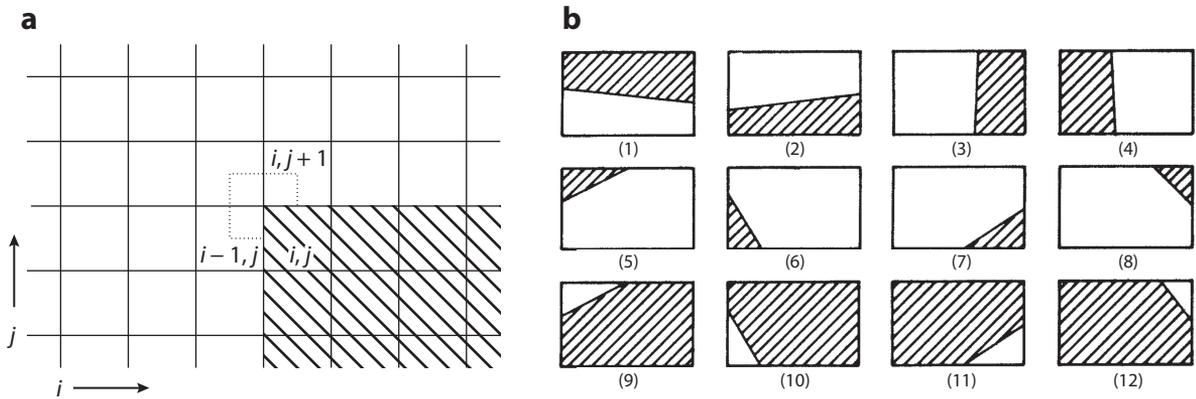


Figure 5

(a) Detail of the discretization at a sharp corner as described by Evans & Harlow (1957, p. 26). (b) “Arbitrary linear barriers within a cell” as sketched by Rich & Blackman (1962, p. 38, figure 3). Panels adapted from (a) Evans & Harlow (1957) and (b) Rich & Blackman (1962).

($i, j + 1 \rightarrow i, j$). For the general case of **Figure 5b**, 12 configurations were identified and, for each one, a set of five geometrical quantities, including fluid area and fluid/solid fraction on each side, was computed to weigh forcing terms of the equations and compute the correct fluxes across the cell faces. Although the manual cell tagging and the very limited grid resolution restricted the applications to extremely simplified examples, in retrospect this approach does not differ much from one of the boundary reconstruction methods proposed by Fadlun et al. (2000), while the metric term correction at the intersected cell faces looks very similar to the forcing procedure used by Leonardi et al. (2003) and Orlandi & Leonardi (2006) or to Cartesian cut-cell methods described by Seo & Mittal (2011).

The work of Gentry et al. (1966) is another important step in IBMs. Starting from Rich & Blackman (1962), they added flexibility to the use of reflective boundaries (**Figure 6a**) and recognized that “the use of partial cells can cause difficulty in cases where they are much smaller than

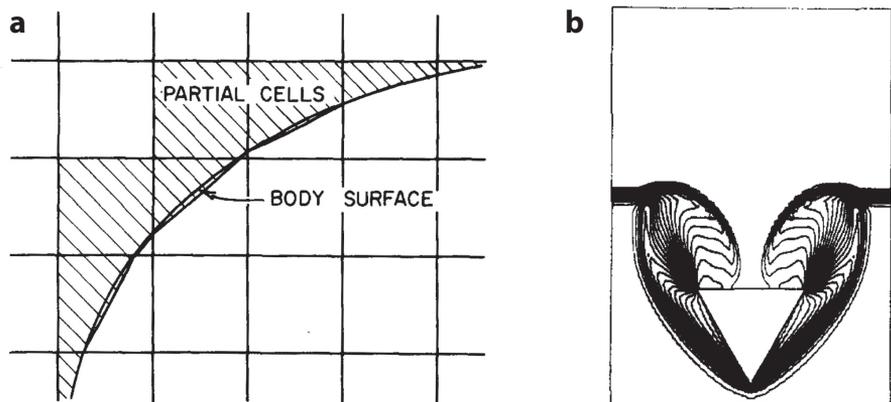


Figure 6

(a) Scheme of immersed boundary/Eulerian grid intersection are reported by Gentry et al. (1966). (b) Pressure isolines for the interaction of a shock wave with a 30° half-angle cone as obtained by Gentry et al. (1966) using an immersed boundary method. Panels adapted with permission from Gentry et al. (1966); copyright 1966 Elsevier.

full sized cells. Since the maximum value of Δt is limited for stability and accuracy reasons by the minimum cell dimension, these small partial cells could lead to prohibitively long calculation times. Such extremely small partial cells should therefore be avoided whenever possible” (p. 97). They also included numerical viscosity to stabilize the calculation, and for those cases in which small partial cells proved to be essential, they implemented a local Δt reduction, which yielded smooth and stable solutions as shown in **Figure 6b**. The problem of small fluid cells emerging from immersed boundary intersections with Cartesian grids has been evidenced more recently by Ye et al. (1999) and Chung (2006) in the framework of cut-cell methods, and they suggest the agglomeration of the smallest cells to their closest neighbors.

Viecelli (1969) presented the first method to deal with incompressible flows bounded by arbitrarily curved walls. He modified the marker-and-cell method of Welch et al. (1966) by “replacing the relaxation iteration with one in which the pressure in boundary cells is adjusted by an amount proportional to the scalar product of the particle velocity at the boundary and a unit normal defining the boundary” (p. 543). In other words, he treated the immersed boundaries as if they were free surfaces and enforced the no-penetration condition across them by introducing a forcing term proportional to the flux. Using modern terminology, this is a feedback forcing or a penalization method with a boundary pressure gradient playing the role of \mathbf{f} in Equation 2. The method was further extended to moving boundaries by Viecelli (1971) using the arbitrary boundary marker-and-cell method, which includes most elements of modern IBMs. In **Figure 7a** we show the classification of grid cells as external, fluid, or boundary, which is the cell tagging described in the previous section. Viecelli (1971) also reported that “when the boundary is stationary the (boundary) normals and Eulerian cell flags need to be determined only once at the beginning of the calculation; however when the boundary is moving it is necessary to regenerate the normals and cell flags each time the solution is advanced Δt in time” (p. 126). On the other hand, **Figure 7b** shows the procedure used to interpolate Eulerian velocities on a material point of the boundary, which is clearly a boundary reconstruction.

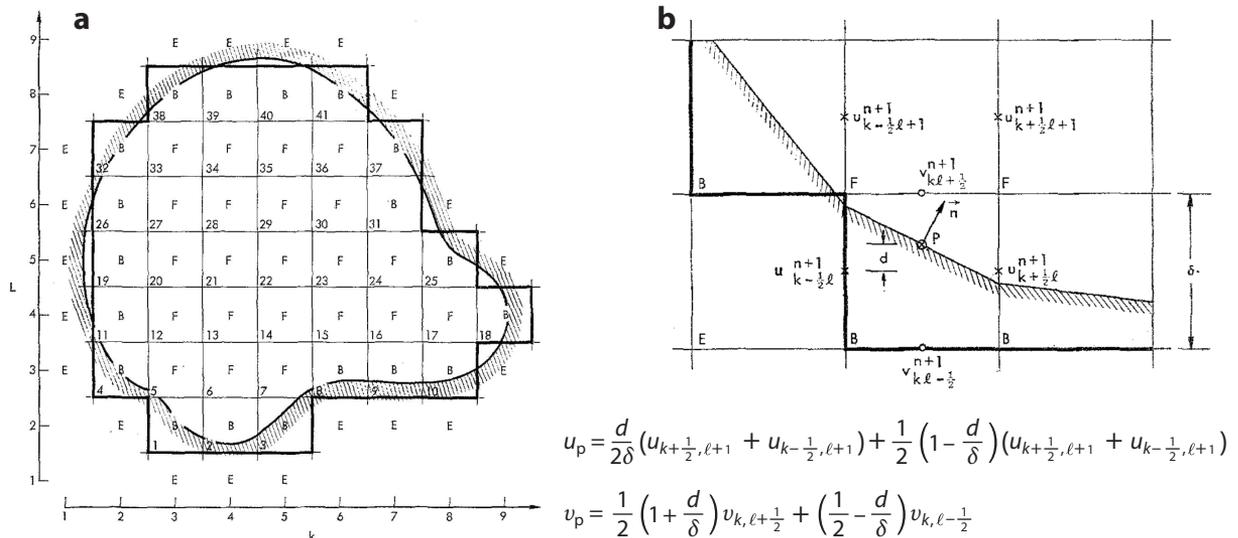


Figure 7

(a) Classification of the Eulerian cell with respect to the immersed boundary as represented by Viecelli (1971). (b) Sketch of the procedures to interpolate velocities at the boundaries taken from Viecelli (1971). The formulas below have been copied and edited to conform to house style. Panels adapted with permission from Viecelli (1971); copyright 1971 Elsevier.

Furthermore, for the dynamics of moving, flexible boundaries, Viacelli (1971) proposed to use “a Lagrangian mesh covering the wall,” and to determine their motion, “one might be solving the equations of elasticity for the wall deformations produced by the liquid pressure. Or the points might be given by some prescribed analytic formula for the boundary shape and motion” (p. 124). The paper goes even further and considers that Lagrangian and Eulerian points do not overlap; therefore, a convenient representation of the boundary is “to connect successive (Lagrangian) points with straight lines and find all of the intersections of these line segments with the underlying Eulerian mesh” (p. 125).

When comparing the content of these old papers with current IBMs, we must admit that all the basic ingredients had already been developed by the end of the 1960s and their only problem was the limited computational power of the available computers, for which 2D simulations on meshes of 30×40 nodes constituted an achievement.

Additionally, the Russian school, following its mathematical tradition, had developed an original procedure to integrate partial differential equations on non-body-conformal meshes. Saulev (1963) considered the model problem of the diffusion equation $\nabla(a\nabla u) = -f$ (with a piecewise constant diffusion coefficient a) within a domain Ω around a body B with boundary conditions $u = 0$ on the body surface $\partial\Omega$, and reformulated it on an extended domain $\widehat{\Omega} = \Omega \cup B$. The new problem read

$$\nabla(a_\epsilon \nabla u_\epsilon) = -f_\epsilon \quad \text{on } \widehat{\Omega} \quad \text{and} \quad u_\epsilon = 0 \quad \text{on } \partial\widehat{\Omega}, \quad 5.$$

with $a_\epsilon = a$ and $f_\epsilon = f$ on Ω and $a_\epsilon = 1 + 1/\epsilon$ and $f_\epsilon = 0$ on B . Proof was given for $\|u_\epsilon - \widehat{u}\| \rightarrow 0$ as ϵ approaches 0, with $\widehat{u} = u$ on Ω and $\widehat{u} = 0$ on B . Kuznetsov (2001) pointed out that Saulev (1963) introduced the fictitious domain method for the above problem only for simplicity and that “extending the formulation to more complex equations, domains, and boundary conditions would be obvious” (p. 67). He also showed that introducing the distributed Lagrange multiplier $\lambda = u/\epsilon$ on B for $\epsilon \rightarrow 0$ reduces the above formulation to the distributed Lagrange multiplier fictitious domain method proposed by Dihn et al. (1992), thus reconfirming that the main idea of the IBM had been conceived in the 1960s.

Yanenko (1967) introduced the possibility of solving fluid dynamics equations on non-body-conformal meshes, which he called “unmatched or incoherent grids” (**Figure 8b**); after having explained how to use a fractional-step method to solve flow problems on body-fitted grids (as in **Figure 8a**), he explicitly stated that “for incoherent grids additional interpolations are needed in order to impose boundary conditions at the walls” (p. 67).

Dorodnitsyn & Meller (1971) suggested a method to surrogate the no-slip condition on the boundary with a more general condition containing a small iterative parameter; although in this original study it was applied only to 2D problems, in later works the approach was extended to 3D flows (Shokin 2009). Establishing the relation between this method and current IBMs is not obvious since the scheme has been mostly discussed in terms of its mathematical features; nevertheless, it has also been implemented in computer codes and used to numerically integrate flows in non-Cartesian geometries.

4. WALL MODELING FOR IMMERSed BOUNDARY METHODS

In most engineering and practical applications, flows develop at high Reynolds numbers and their direct numerical simulation is practically impossible (Moin & Mahesh 1998). Turbulence models are therefore needed to parameterize the effects of the small flow scales on the larger ones, and for unsteady complex flows LES has proven to give accurate results within a computationally tractable framework. Indeed, IBMs have shown to be compatible with many turbulence models, and their combination with LES has already been discussed by Iaccarino & Verzicco (2003) and Cristallo

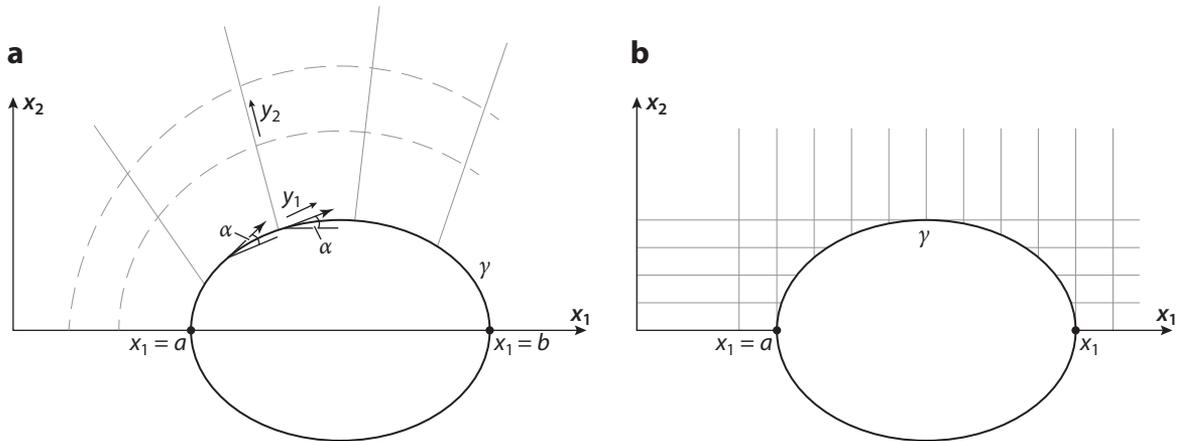


Figure 8

Representation of “matched” (a) and “unmatched” (b) grids for the flow around a curvilinear object as represented by Yanenko (1967) on p. 130 and p. 137 of his book. Panels adapted from Yanenko (1967).

& Verzicco (2006). Here, we limit the discussion to the treatment of the near-wall region, which is a relatively more recent topic and constitutes the real bottleneck for the application of IBMs to turbulent flow simulations.

The computational requirements for body-conformal meshes have been the subject of detailed studies, both in previous decades (Chapman 1979, Reynolds 1990) and more recently (Piomelli 2008, Choi & Moin 2012, Bose & Park 2018). Traditionally the flow is divided into two regions, an inner and outer layer near and far from the walls, respectively, where the size of the smallest eddies, viscous length $\delta_v \sim LRe^{-0.9}$, and Kolmogorov scale $\eta_K \sim LRe^{-0.75}$ decrease at different rates with the Reynolds number. These relations imply that for increasing Re , the computational load of the wall region becomes dominating and eventually is the limiting factor. In fact, Piomelli (2008) has shown that in a boundary layer simulation with a body-fitted mesh, for $Re \approx 10^4$ more than 50% of computational resources are used to resolve the wall region, which represents less than 10% of the flow volume; for $Re > 5 \times 10^4$ the percentage of grid points in the outer layer becomes negligible, while the thickness of the boundary layers keeps decreasing according to $\sim Re^{-0.2}$.

As anticipated at the end of Section 2, the simulation of high-Reynolds number flows using IBMs is very challenging since they have to cope with thin boundary layers without benefiting from anisotropic resolution in the wall-normal direction. In fact, while for body-fitted meshes the choice between wall-resolved and wall-modeled LES is possible, only the latter approach is feasible for IBMs, as was already recognized by Cristallo & Verzicco (2006) and has been further discussed and detailed by Cai et al. (2021).

It should be pointed out that in some early applications of LES with IBMs (Verzicco et al. 2000, 2002), no wall models were employed except for a cubic interpolation of the eddy viscosity at the wall (Pope 2000), and yet the results compared favorably with similar experiments (Figure 9c). In those cases, however, boundaries were largely aligned with coordinate lines and acceptable near-wall resolution was achieved by stretching the grid along Cartesian directions (Figure 9a). Furthermore, in those cases flow separations were determined by geometrical sharp edges (Figure 9b); thus, size and position of recirculations were largely independent of the Reynolds number.

On the other hand, in the flow of Wang & Moin (2000), the smooth suction side of the hydrofoil (Figure 10a) produced a completely viscous separation, and the underresolved IBM/LES of

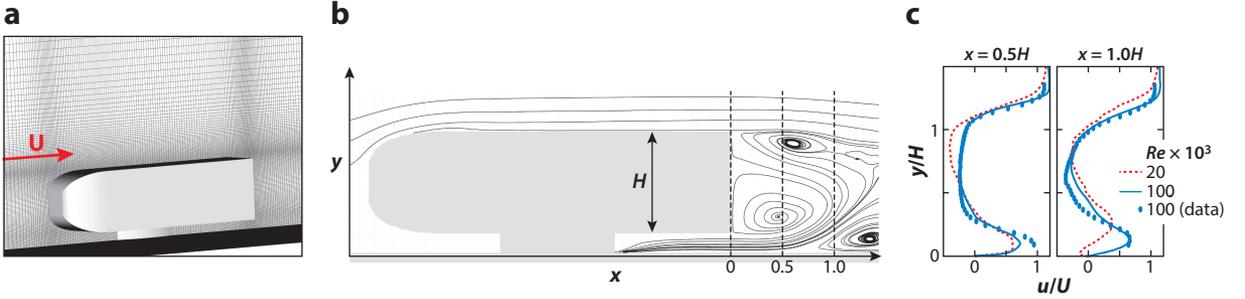


Figure 9

(a) Nonuniform Cartesian mesh with an immersed 3D model road vehicle. (b) Stream traces of the mean flow on the longitudinal symmetry plane at Reynolds number $Re = UH/\nu = 20,000$, with ν the viscosity. (c) Vertical profiles of mean streamwise velocity on the longitudinal symmetry plane at two distances from the base; symbols are experimental results from Khalighi et al. (2001). Panels adapted with permission from Verzicco et al. (2002); copyright 2022 AIAA.

Tessicini et al. (2002) completely mispredicted the wake dynamics. In fact, the best mesh they could afford yielded a wall resolution that varied between around 0 and 60 wall units, and in the latter case, a geometrical reconstruction procedure (as in **Figure 3**) was clearly inadequate, as confirmed by the results in **Figure 10b**. Another source of inaccuracy was the distance of the first external node from the wall, which varied randomly from point to point owing to the uneven crossing of the immersed boundary with the coordinate lines (**Figure 11a**).

In order to overcome these difficulties, rather than using a geometrical interpolation, Tessicini et al. (2002) determined the velocity at the first external point from a simplified boundary layer equation; following Balaras et al. (1996), the near-wall tangential velocity u_{\parallel} obeys

$$\frac{\partial}{\partial n} \left[(\nu + \nu_{\tau}) \frac{\partial u_{\parallel}}{\partial n} \right] = F, \quad \text{with} \quad F = \frac{\partial u_{\tau}}{\partial t} + \frac{\partial u_{\parallel} u_{\parallel}}{\partial \tau} + \frac{\partial u_{\parallel} u_{\perp}}{\partial n} + \frac{\partial p}{\partial \tau}, \quad 6.$$

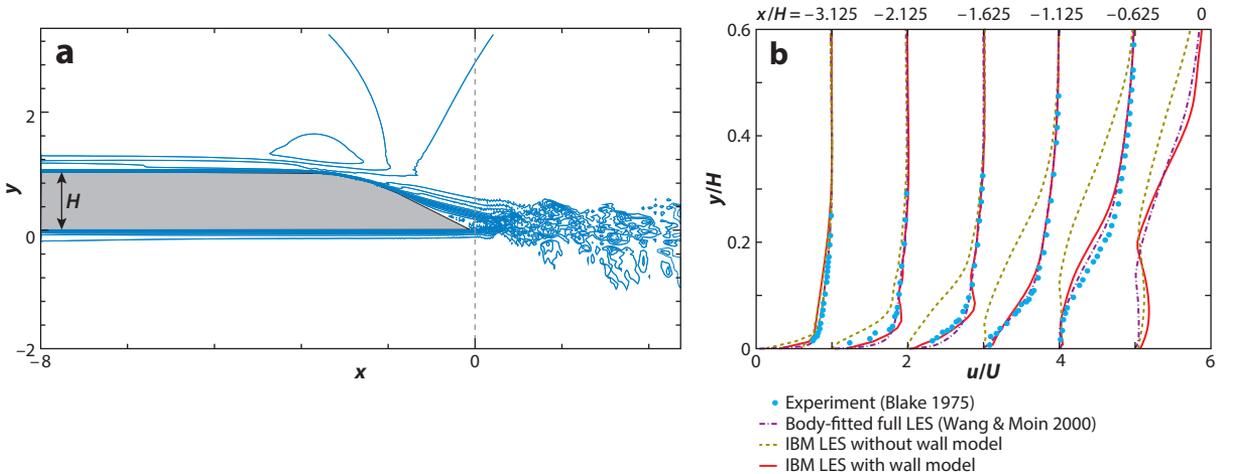


Figure 10

(a) Instantaneous snapshot of the flow past a hydrofoil trailing edge. Contours ($[-0.2, 1.2]$, with increment 0.08) show the instantaneous streamwise velocity. (b) Mean velocity magnitudes. Each profile is shifted rightward by one unit for clarity of representation. Figure adapted with permission from Tessicini et al. (2002). Abbreviations: IBM, immersed boundary method; LES, large eddy simulation.

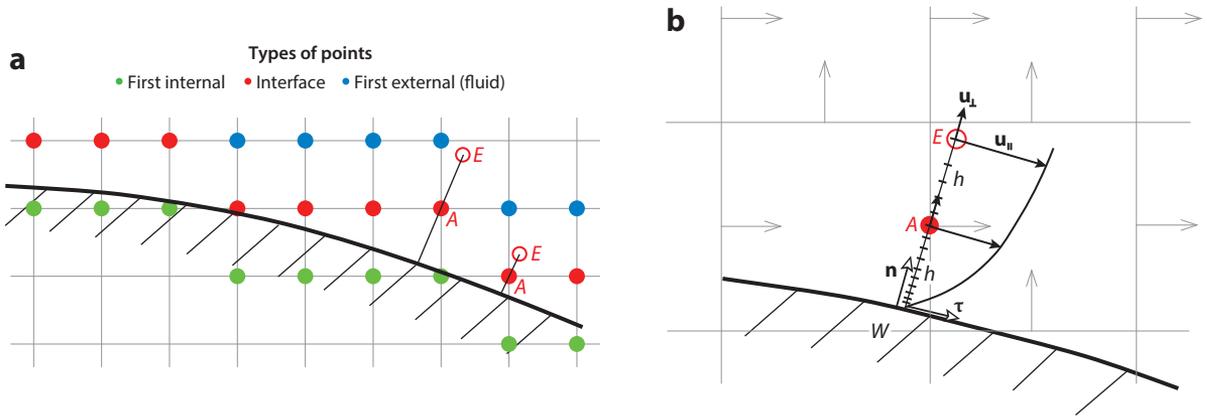


Figure 11

(a) Intersections of a generic immersed boundary with a Cartesian grid. Points E (red open circles) are used to provide the external boundary condition to the solution of Equation 6. (b) Detail of the setup for the solution of Equation 6 for one of the first external points in a background grid with a staggered discretization. Here, b is the distance from point A to the immersed boundary.

where n and τ are the local normal and tangential coordinates, respectively (for the ease of representation only the 2D equation has been written). In the context of IBMs, however, coordinates are neither normal nor tangential to the boundary; thus, from any external point A (Figure 11b), a wall-normal ray is extended on the opposite side for the same distance b . Since point E generally does not coincide with a grid node, interpolations are needed to compute its tangential and normal velocity components, which are used as boundary conditions for Equation 6 integrated over a 1D refined mesh between E and W . The eddy viscosity ν_T is obtained by a mixing length model with near-wall damping: $\nu_T = \nu \kappa n^+ (1 - e^{-n^+/A})^2$, with $\kappa = 0.4$ and $A = 19$. Here, $n^+ = n/\delta_v$ is the distance from the wall in viscous units, computed from the instantaneous local friction velocity. It must be noted that the calculation of ν_T needs n^+ , which relies on the friction velocity u_τ . The latter, in turn, is obtained from Equation 6, which contains ν_T . An iterative procedure is then required that, starting from a tentative value of u_τ (usually the value at the previous time step), solves simultaneously for Equation 6 and the definition of ν_T . The term F of Equation 6 contains derivatives along n and τ ; therefore, additional interpolations are required for its evaluation on point E . In order to avoid this step, one often imposes $F = 0$ in the model, which is termed an equilibrium stress balance model; this was adopted by Tessicini et al. (2002), with considerably improved results, as shown in Figure 10b.

It has long been thought that the model of Equation 6 would perform even better if the full term F were computed—the main reason for using the simple equilibrium stress balance model is that it reduces to a local ordinary differential equation rather than a full partial differential equation. There have been also attempts to include single terms of F , especially $\partial p/\partial \tau$, which gives the streamwise pressure gradient and the related boundary layer dynamics. However, Larsson et al. (2016) have shown that the wall model captures only the inner layer, whose dynamics is much faster than that of the outer counterpart, and it is thus mostly in equilibrium. Furthermore, the terms of F are evaluated in the outer layer where, even if they are individually large, they are in balance, thus yielding a vanishing F and justifying the good performance of the equilibrium stress balance model.

It is important to note that, in the context of IBMs, there is no control of the distance b of the first external point; thus, the model has to adjust to it. An important feature of Equation 6 is that

if the first external point is at a distance from the wall within about 10 viscous units, the model returns a linear velocity profile that is fully consistent with a boundary reconstruction, such as that of Equation 4. On the other hand, for more distant points, the solution for the instantaneous velocity is the logarithmic law of the wall.

Since the initial attempt of Tessicini et al. (2002), several improvements and generalizations have been suggested, discussed, and validated in many papers. Roman et al. (2009) used a geometrical arrangement similar to that of **Figure 11b**, although their point E was fixed along the wall normal as the closest node to the first external fluid node. For $b < 11\delta_v$, they computed the tangential velocity in A as $u_\tau(A) = (u_\tau)^2 b / \nu$, as suggested by the viscous sublayer dynamics, while for larger distances the logarithmic law of the wall, matching $u_\tau(E)$, was employed. In the latter case, the normal velocity component u_\perp was also needed in A , and this was computed by a quadratic interpolation between the wall and E . In the same paper the authors noted that reconstructing only the velocity at point A was not enough since the eddy viscosity ν_T was also needed to compute the correct viscous fluxes at the cells next to the immersed surface. Again, they separated the case $b < 11\delta_v$, where we have $\nu_T = 0$, from farther points, for which a RANS-like mixing length model was used. Using this model the authors were able to compute first- and second-order statistics for the turbulent channel flow up to $Re_\tau = 2,000$.

Chen et al. (2014) proposed a wall model that is essentially that of Equation 6 after eliminating the convective term from F ; the wall-normal velocity was instead obtained by integration of the mass conservation. The peculiarity of this model is that only the wall shear stress is provided to the external LES, while velocities are not directly affected by the model. Among several validations, particularly significant are good results for the turbulent flow in a channel with periodic smooth hills at $Re = 10,595$ since it proves that the model can work with favorable and adverse pressure gradients and in the presence of massive separations.

A slightly different method was suggested by Kang (2015), who solved the equilibrium stress model near the immersed boundary to obtain a turbulent viscosity that, in turn, was interpolated to obtain the correct total shear stress. The method performed well for turbulent channel and pipe flows, although the effect of pressure gradients and the compliance with complex geometries were not assessed.

A novel approach for wall modeling in LES was tested by Yang et al. (2015) based on the integral boundary layer equation of von Karman and Pohlhausen; in this case, rather than numerically integrate a simplified boundary layer equation, one must assume a velocity profile and determine its coefficients by matching physical constraints. The authors showed that the model combines a Reynolds-independent computational cost with the possibility of handling resolved and subgrid roughness, thus making it particularly appealing for such applications as atmospheric boundary layers of flows over complex rough surfaces.

The implementation of Dhamankar et al. (2016) essentially relies on the law of the wall for the time-averaged mean wall-tangential velocity profile given by Reichardt (1951), although it also includes interesting peculiarities. For example, with reference to **Figure 11b**, the authors skipped all first external points A to avoid the accuracy issues related to the wall distance variation along the surface. Accordingly, regardless of points A , nodes E , termed matching points, were always placed at a distance of about three grid spacings to assure that the surrounding mesh contained only valid fluid nodes and that the points themselves were well within the log-layer. Additionally, in this case an iterative procedure was needed to compute the wall shear stress from the velocity profile, which contains variables in wall units. The authors extended the model also to compressible flows by applying the van Driest transformation for the tangential velocity (Gatski & Bonnet 2013) and the Crocco–Busemann relation for the wall temperature (White 2006).

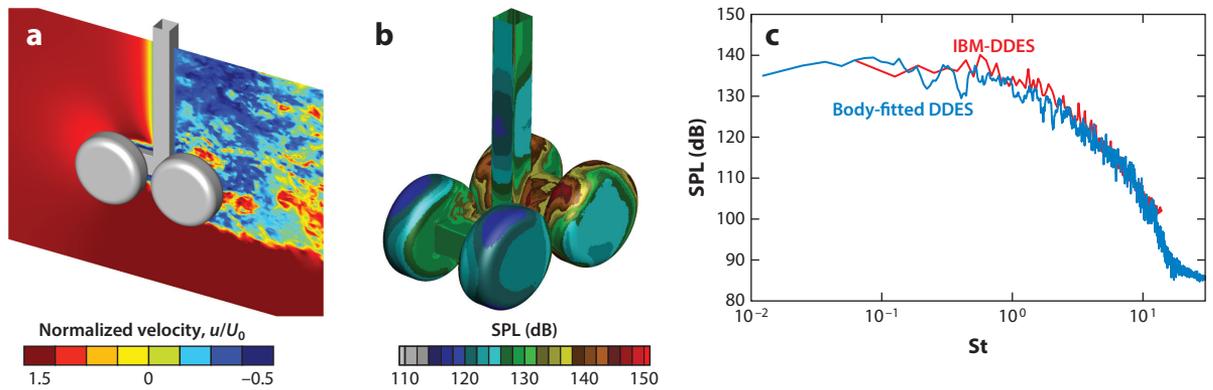


Figure 12

Visualizations of the flow around a rudimentary landing gear simulated by an immersed boundary method (IBM) with delayed detached eddy simulation (DDES) turbulence modeling. (a) Contours of streamwise velocity in the vertical symmetry plane. (b) Surface distribution of the sound pressure level (SPL) in decibels; values range from 100 dB (blue) to 150 dB (red). (c) Surface pressure frequency spectrum at the backward face of the rear wheel. The red line is the IBM-DDES, and the blue line is the body-fitted DDES. Figure adapted with permission from Bernardini et al. (2016); copyright 2016 Elsevier.

Dhamankar et al. (2016) considered the wall model applicable only in attached boundary layer regions, while flows of industrial relevance are very likely to have separations. For this reason Bernardini et al. (2016) developed a simple and effective wall model that was successfully applied to several test cases, including rudimentary landing gear, for which the results were in good agreement with a reference experiment and other numerical simulations on body-fitted meshes (Figure 12). Additionally, for this model an arrangement like that of Figure 11b was used, although the distance from the wall of nodes E (therein called reflected nodes) was always the same and fixed as the maximum among all reflected nodes. The main novelty of this model was that the equilibrium stress balance model of Equation 6 was integrated consistently with the outer turbulence model in order to alleviate the strong dependence of the numerical results on the location of the first off-wall grid node, which is particularly annoying for IBMs. Since the benchmark of the rudimentary landing gear was performed at $Re = 10^6$ and at Mach number $M = 0.2$, compressibility effects were also considered and the Crocco–Busemann relation for the wall temperature was used.

Although all the models discussed above have been implemented in LES or delayed detached eddy simulations, for which the wall resolution is the most computationally demanding part, a comparable amount of work has been devoted to RANS simulations, the first attempts of which are as old as those of LES.

Kalitzin et al. (2003) developed an IBM-compatible version of the $k-\omega$ RANS model (Wilcox 1993) by observing that $\omega \sim n^{-2}$ holds next to the wall and that a linear boundary reconstruction would be inaccurate. However, a change of variable in the form $g = 1/\sqrt{\beta^*\omega}$, with β^* a model constant, yielded $g \sim n$ at the wall, and the same reconstruction procedure used for velocity components could be applied to g . As the near-wall resolution was not yet sufficient, an additional wall model was obtained by prescribing the tangential flow velocity, at a constant distance from the immersed boundary, using a precompiled look-up table obtained from a Reichardt expression (Reichardt 1951). A similar approach was adopted by de Tullio et al. (2007), who included a semi-structured local grid refinement (Durbin & Iaccarino 2002) to capture thin boundary layers and discontinuities, like shocks, of supersonic flows.

Capizzano (2011) also used the $k-g$ turbulence model of Kalitzin et al. (2003), but for the near-wall region resolved a boundary layer equation like Equation 6 with only the pressure gradient term in F . They obtained results in very good agreement with experiments and body-fitted simulations several 2D and 3D flows around airfoil and wings. It should be pointed out, however, that the simulations of flows at $M = 0.8$ and $Re = 4 \times 10^6$ were possible thanks to a fully unstructured mesh that allowed for aggressively anisotropic local grid refinements; this approach is intermediate between body-conformal and pure IBMs since it avoids the generation of a grid fitted to the body but maintains a considerable computational overhead due to the unstructured discretization (Piomelli 2008).

Major improvements have recently been introduced by Cai et al. (2021) to avoid spurious oscillations of pressure and shear stress at the wall. To mitigate interpolation errors related to the variable distance of the first external node, they suggested interpolating the friction velocity, which varies much less than raw velocity next to the wall. They also discarded the idea of using auxiliary interpolation points in favor of the existing fluid points intersecting a wall-parallel plane. Furthermore, advanced interpolation schemes were used to compute velocity gradients and, similarly to Bernardini et al. (2016), enforce an eddy viscosity at the cells near the wall consistent with the outer-flow turbulence model. Very good results were obtained for several benchmarks, including a complex 3D wing with a slat and flap at $Re \approx 4 \times 10^6$ and $M = 0.2$. An important point of these results is the coupling of IBM and wall modeling with a lattice Boltzmann method, which allowed for local grid refinement next to the immersed boundaries, thus facilitating the resolution of steep wall gradients.

Comparing the wall models for LES with wall models for RANS, it appears that the latter are always combined with anisotropic local grid refinement, which greatly simplifies the near-wall treatment for IBMs, while the former very seldom rely on it. The reason for this difference is that locally refined grid patches introduce discontinuities in the grid spacing at their boundaries (**Figure 13a**), which produce spurious numerical stresses. This is not much of a problem for RANS simulations, which are usually run using upwind discretizations whose numerical viscosity tends to diffuse spurious stresses and prevents them from spoiling the numerical solution. In contrast, for LES, the only additional dissipation has to come from the subgrid-scale model and

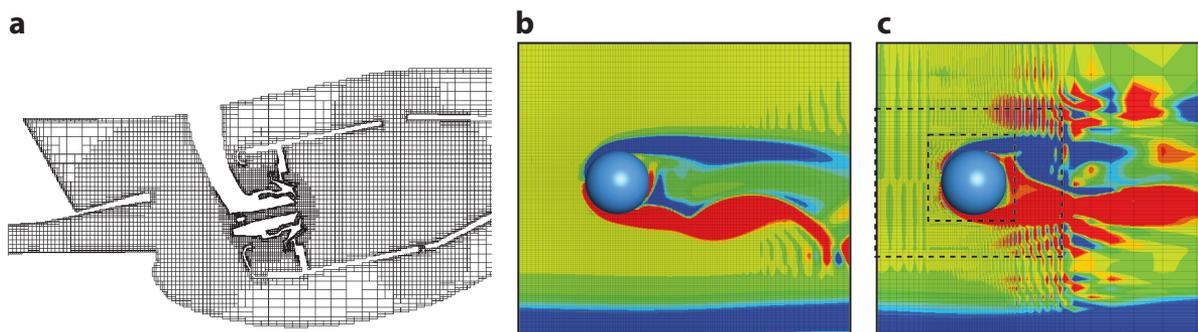


Figure 13

(a) Section in the symmetry plane of a Cartesian mesh with local grid refinements for a Pratt & Whitney combustor. Panel adapted with permission from Iaccarino & Ham (2005). (b) Large eddy simulation with dynamic Smagorinsky model for the flow around a sphere at a distance $D = 4R$, with R the sphere radius, from a solid wall at Reynolds number $Re = UR/\nu = 1,000$, with U the uniform horizontal velocity and ν the viscosity. Color contours show the vertical symmetry section of out-of-plane vorticity, with values ranging from -1 (blue) to 1 (red). Overlaid light gray lines are the actual Cartesian mesh. (c) The same as panel *b* except for the mesh, which has been abruptly coarsened at the boundaries of the two windows indicated by the thick dashed lines. Note that the wall resolution at the sphere surface and at the lower solid boundaries is the same as that for panel *b*.

central nondissipative discretizations are mandatory (Mittal & Moin 1997); in this case, discontinuities in the mesh distribution produce a footprint in the flow of wiggles and unphysical flow structures. This is shown in **Figures 13b,c**, where the flow around a sphere next to a flat wall at $Re = 1,000$ has been simulated by a staggered, second-order central finite-difference discretization and a dynamic subgrid-scale LES model. In **Figure 13b** the mesh is smoothly stretched next to the (immersed) sphere and the solid wall, while in **Figure 13c** two refinement levels are obtained by coarsening the mesh, as would be done by a local grid refinement. The comparison of the two flows shows that, despite identical grid resolution at the boundaries, in the latter case the flow is largely contaminated by numerical wiggles that, although originating at the mesh discontinuities, spread to the surrounding regions even with smooth and fine meshes.

It might be pointed out that, once a local refinement procedure has been implemented, the same strategy could be used to describe the most relevant flow structures far from the boundaries, thus avoiding the issues seen in **Figure 13c**. Indeed, this has been proposed in many studies; as an example we mention Antepará et al. (2013), who dynamically refined and coarsened the mesh, depending on the local flow gradients, for the flow around a square cylinder at $Re = 22,000$. However, given the strongly unsteady wake dynamics, the spatial location of the intense vorticity layers presumably varied in time; therefore, such an adaptive mesh refinement would require frequent remeshing and interpolations, which considerably add to the computational load.

5. IMMERSED BOUNDARY METHODS FOR MULTIPHYSICS FLOWS

The same procedures described in Section 2 for momentum can be applied to other convection–diffusion–reaction equations to model different dynamics or multiphysics phenomena, and in recent years, IBMs have indeed spread from fluid dynamics to many related topics.

Among the first applications is conjugate heat transfer in which, in addition to momentum, the temperature equation is forced using IBMs to satisfy temperature or heat flux boundary conditions (Verzicco 2002, 2004; Iaccarino & Moreau 2006). An illustrative example is given in **Figure 14a**, which shows an actual laboratory setup to study turbulent thermal convection. The aim is to mimic Rayleigh–Bénard convection; therefore, the working fluid is vertically bounded by high thermal conductivity metal plates (usually copper) at different temperatures and laterally confined by a low-conductivity wall (in this case, Plexiglas). External thermal shields are actively maintained at the mean flow temperature to prevent parasitic heat currents, and a layer of open-cell foam inhibits external convection to allow for large pressure variations (Ahlers et al. 2009).

Using IBMs, this setup is modeled by the Boussinesq equations (in which only the buoyancy term is affected by temperature-induced density variations $\Delta\rho$):

$$\begin{aligned} \frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} &= -\frac{\nabla p}{\rho} + \nu \nabla^2 \mathbf{u} + \mathbf{g} \frac{\Delta \rho}{\rho} + \mathbf{f} + \mathbf{f}', & \nabla \cdot \mathbf{u} &= 0, \\ \frac{\partial T}{\partial t} + \mathbf{u} \cdot \nabla T &= \frac{\nabla \cdot (\lambda \nabla T)}{\rho C} + k, \end{aligned} \quad 7.$$

with T , λ , and C the temperature, thermal conductivity, and specific heat, respectively. The IBM term \mathbf{f} enforces zero velocity inside the solid volumes, while $\mathbf{f}' = -\mathbf{u}/K$ models the porous convection within the foam through the porosity K (Navier–Stokes–Brinkman equation). On the other hand, in the temperature equation, a new IBM term k is used to enforce a constant temperature within the thermal shields, whereas the position-dependent material properties (ρ , λ , and C) provide the correct conjugate heat transfer solution (Stevens et al. 2014).

Among many others, some additional recent applications of IBMs to conjugate heat transfer problems include complex porous structures (Das et al. 2018), turbine blade cooling (Tafti et al.

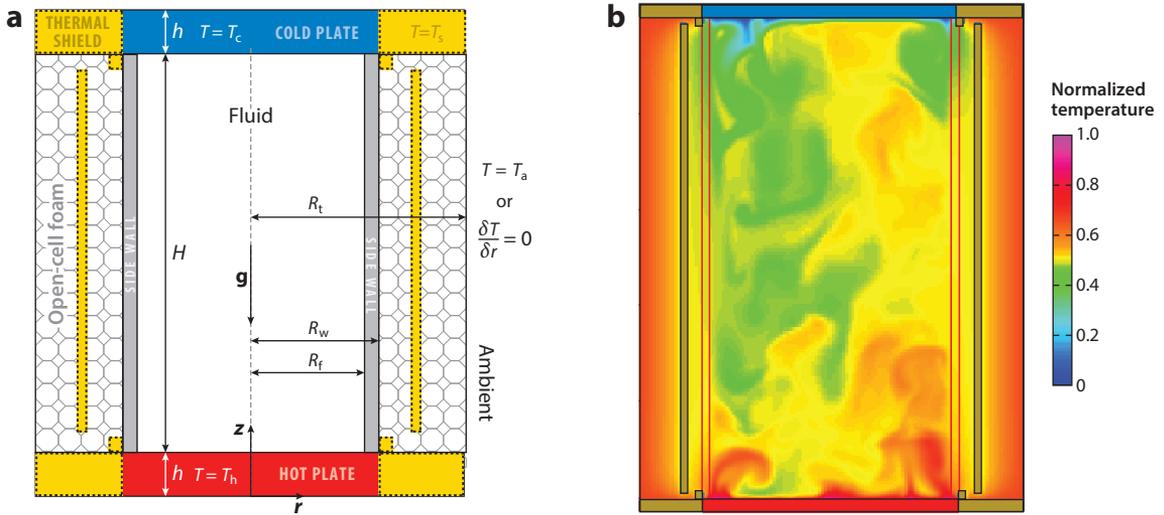


Figure 14

(a) A realistic setup for the study of thermal convection in a laboratory experiment, as inspired by the U-boot apparatus described by Ahlers et al. (2009), who used sulfur hexafluoride (SF_6) as the working fluid. A cylindrical fluid sample ($b \leq z \leq b + H$ and $0 \leq r \leq R_f$) is vertically bounded by two copper plates of thickness b , the lower with the dry surface at temperature T_h and the upper at temperature T_c (with $T_h > T_c$). The sidewall is a Plexiglas hollow cylinder of thickness $R_w - R_f$ directly in contact with the copper plates. Within the Boussinesq approximation the system attains the mean temperature of $T_m = (T_h + T_c)/2$; therefore, to prevent heat leakage some thermal shields are actively maintained at a constant temperature $T_s = T_m$. Finally, parasitic external convection (Ahlers et al. 2009) next to the side wall is inhibited by an open-cell foam, which allows one to vary the mean pressure of the system without changing volume. The temperature boundary condition external to the foam can be either zero heat flux (if additional isolation is provided) or the ambient temperature (otherwise). (b) Instantaneous temperature distribution in a vertical plane through the axis, obtained by direct numerical simulation, as described by Stevens et al. (2014) for Rayleigh number $Ra = 10^8$ and Prandtl number $Pr = 0.7$. Temperature is normalized so that it ranges in the interval $[0, 1]$ at the plates, respectively.

2014), and radiative heat transfer in complex domains with opaque and transparent media (Lapka & Furmański 2017).

Another productive IBM field is that of active surfaces, which includes chemical reactions, mass transfer with phase change, and electrolysis. The basic idea is that in any scalar convection–diffusion equation, like that for temperature in Equation 7, space- and time-dependent forcing terms can be prescribed over an immersed interface to enforce a specific physical mechanism. For example, using IBM forcing in the diffusion equation for a gas in a liquid, Zhu et al. (2018) were able to recover the analytical Epstein–Plesset solution for the growth or shrinkage of an isolated bubble in a quiescent fluid and reproduced the Ostwald ripening process of two pinned or unpinned bubbles on a solid surface. The same scheme was then used to study the diffusive interaction of multiple diffusive nanobubbles. Bao et al. (2018) extended the previous study by considering an array of surface droplets and their inhomogeneous dissolution when swept by a wall-parallel flow. Sepahi et al. (2022) analyzed the effect of buoyancy-driven convection on the growth and dissolution of bubbles on electrodes during water electrolysis.

It is important to note that, despite the complex phenomena at the interfaces and their time-dependent shape and position, the size of these droplets and bubbles is usually of the order of millimeters or less; therefore, the flow Reynolds number is very small and all of the issues related to turbulence and wall modeling of the previous section are avoided. This opens new possibilities to the application of IBMs in fields like microfluidics or miniaturized devices, in which the flow is laminar and smooth and wall resolution is not an issue.

On larger spatial scales, Lu et al. (2018) simulated the flow through a dense columnar array of hundreds of particles to mimic a bead reactor; on each particle an exothermic reaction took place and heat and mass transport was coupled with the fluid phase.

Capizzano & Iuliano (2014) and Lavoie et al. (2021) have used IBMs to simulate in-flight ice accretion over wings since in the standard body-fitted approach, the generation of highly nontrivial grids are repeated several times during the flow evolution. Indeed, in these works the focus is mostly on the Eulerian transport of water droplets, while the thermodynamics of phase change is more crude. On the other hand, in a successive study, Lavoie et al. (2022) modeled the ice accretion phenomena by a level set method.

Blais & Ilinca (2018) and Huang et al. (2021) used IBMs to study phase changes at an evolving interface. They simulated the Stefan problem and therefore included the dissolution and melting of solid phases caused by the interaction with the surrounding flow.

Design optimization in fluid mechanics (Mohammadi & Pironneau 2004) is another field in which IBMs are gaining popularity owing to the possibility of deforming a reference geometry without needing to make a new mesh or deform an existing one.

Sethian & Wiegmann (2000) applied an IBM-like method to optimize just solid structures with respect to an internal stress distribution, while Chaudhuri et al. (2011) and Gopal & Grandhi (2017) used a similar approach for shock-obstacle interactions and high-fidelity aerodynamic shape optimizations, respectively. Zheng et al. (2013) and Jenkins & Maute (2016) used IBMs for optimization with FSI, where the IBM approach is even more convenient on account of the strong coupling between flow and boundary dynamics, which results in iterative corrections.

Before concluding this section we wish to briefly mention additional applications to FSI, which perhaps constitute the fastest-growing IBM field; in fact, when the boundaries undergo large deformations or displacements, IBMs avoid the computationally expensive tasks of grid deformation or remeshing (Goza & Colonius 2017). The reader is referred to Griffith & Patankar (2020) for a dedicated review on this topic. That review, however, mainly discusses the application of IBMs/FSI to simulations of biological and biomedical flows, while these techniques can also be applied to particle-laden (Breugen 2012) and bubbly (Spandan et al. 2017, 2018) flows, soft particles in non-Newtonian fluids (Saadat et al. 2018), energy harvesting (Lahooti & Kim 2019), supersonic parachute inflation during Mars landing (Huang et al. 2020), and industrial processes (Favier et al. 2014).

A final point worth mentioning is that IBMs were initially developed for compressible flows (Gentry et al. 1966), which required numerical viscosity to stabilize the calculation; after more than 50 years it seems that the situation has not really changed and artificial viscosity is still being used to prevent the onset of wiggles resulting from IBMs. Improvements in the implementation of IBMs in wave-dominated problems are indeed necessary, especially for compressible flows with LES modeling, in which the eddy viscosity is overhung by the artificial counterpart, with a resulting possible reduction in accuracy.

6. CLOSING REMARKS

IBMs have gained popularity and credibility as efficient numerical techniques for the solution of complex geometry flows using simple Cartesian meshes. The number of scientific papers involving this subject has continuously grown over the last two decades with a constant acceleration (of 10 papers per year squared) that does not show signs of saturation or slowing down.

A few review papers on IBMs have been written already, which either focus on specific aspects of the subject or do not cover recent applications. This review is an attempt to account for the latest developments in the field of IBMs and to fill in some of the gaps left from previous reviews.

After a brief summary of the main features of the method and its benefits and pitfalls, we described the origins of IBMs in little known literature. We then discussed the main limitation of IBMs (the simulation of high-Reynolds number flows) and how this limitation can be handled by implementing suitable wall models. Finally, we attempted to present some of the fields in which IBMs are being newly applied—although, given the huge number of papers and the variety of problems in this area, the selection is necessarily incomplete and biased by personal taste.

SUMMARY POINTS

1. Most of the key ideas on immersed boundary methods had been conceived and implemented already in the 1960s, but the limited computational power of the computers of those times did not allow researchers to exploit their potential.
2. High-Reynolds number flows are the main limitation of immersed boundary methods, as their thin wall shear layers cannot be properly resolved by anisotropic grid refinement in the wall-normal direction; the problem can be alleviated by implementing suitable wall models.
3. All wall models for immersed boundary methods derive from the application of the thin boundary layer approximation at the wall. Different implementations, coupling with the numerical method and consistency with outer layer turbulence model, however, have been shown to affect the quality and reliability of the results.
4. Recent developments in microfluidics, miniaturized devices, and active interfaces have opened new opportunities for immersed boundary methods. Indeed, heat and mass transfer, phase changes, and chemical reactions can be handled using the available forcing schemes, while the small spatial dimensions and reduced velocities avoid the high-Reynolds number limitation described above.

DISCLOSURE STATEMENT

The author is not aware of any biases that might be perceived as affecting the objectivity of this review.

ACKNOWLEDGMENTS

I owe a huge debt to Paolo Orlandi, who as my PhD advisor in the early 1990s suggested that I look at the paper by Goldstein et al. (1993), which he said might contain interesting ideas; this is how I got involved in immersed boundaries. Mentoring of and collaborations with former students, and nowadays colleagues, have been key to my development of ideas and exploration of new possibilities: Gianluca Iaccarino and Marco D. de Tullio are, by far, the two individuals who have contributed most to my research on immersed boundaries, and I still have the pleasure to exchange ideas with them. Francesco Viola and Sergio Pirozzoli are gratefully acknowledged for critically reading this manuscript and providing comments and suggestions.

LITERATURE CITED

- Ahlers G, Funfshilling D, Bodenschatz E. 2009. Transitions in heat transport by turbulent convection at Rayleigh numbers up to 10^{15} . *New J. Phys.* 13:049401
- Alkhateeb O, Tsukerman I. 2013. A boundary difference method for electromagnetic scattering problems with perfect conductors and corners. *IEEE Trans. Antennas Propag.* 61:5117–26

- Andreykiv A, Rixen DJ. 2009. Numerical modelling of electromechanical coupling using fictitious domain and level set methods. *Int. J. Numer. Methods Eng.* 80:478–506
- Angot P, Bruneau CH, Fabrie P. 1999. A penalization method to take into account obstacles in incompressible viscous flows. *Numer. Math.* 81:497–520
- Antepara P, Lehmkuhl O, Chiva J, Borrell R. 2013. Parallel adaptive mesh refinement simulation of the flow around a square cylinder at $Re = 22000$. *Proc. Eng.* 61:246–50
- Aström KJ, Murray RM. 2010. *Feedback Systems: An Introduction for Scientists and Engineers*. Princeton, NJ: Princeton Univ. Press
- Balaras E, Benocci C, Piomelli U. 1996. Two-layer approximate boundary conditions for large-eddy simulations. *AIAA J.* 34:1111–19
- Bao L, Spandan V, Yang Y, Dyett B, Verzicco R, et al. 2018. Flow-induced dissolution of femtoliter surface droplet arrays. *Lab Chip* 18(7):1066–74
- Bernardini M, Modesti D, Pirozzoli S. 2016. On the suitability of the immersed boundary method for the simulation of high-Reynolds-number separated turbulent flows. *Comput. Fluids* 130:84–93
- Blais B, Ilinca F. 2018. Development and validation of a stabilized immersed boundary CFD model for freezing and melting with natural convection. *Comput. Fluids* 172:564–81
- Blake WK. 1975. *Statistical description of pressure and velocity fields of the trailing edge of a flat structure*. Tech. Rep. 4241, David Taylor Naval Ship Res. Dev. Cent., Bethesda, MD
- Borazjani I, Ge L, Sotiropoulos F. 2008. Curvilinear immersed boundary method for simulating fluid structure interaction with complex 3D rigid bodies. *J. Comput. Phys.* 227:7587–620
- Bose ST, Park GI. 2018. Wall-modeled large-eddy simulation for complex turbulent flows. *Annu. Rev. Fluid Mech.* 50:535–61
- Breugen WP. 2012. A second-order accurate immersed boundary method for fully resolved simulations of particle-laden flows. *J. Comput. Phys.* 231:4469–98
- Cai SG, Degryny J, Bousuge JF, Sagaut P. 2021. Coupling of turbulence wall models and immersed boundaries on Cartesian grids. *J. Comput. Phys.* 429:109995
- Capizzano F. 2011. Turbulent wall model for immersed boundary methods. *AIAA J.* 49(11):2367–81
- Capizzano F, Iuliano E. 2014. A Eulerian method for water droplet impingement by means of an immersed boundary technique. *J. Fluids Eng.* 136(4):040906
- Chapman DR. 1979. Computational aerodynamics development and outlook. *AIAA J.* 17(12):79–0129R
- Chaudhuri A, Hadjadj A, Chinnayya A. 2011. On the use of immersed boundary methods for shock/obstacle interactions. *J. Comput. Phys.* 230(5):1731–48
- Chen ZL, Hickel S, Devesa A, Berland J, Adams NA. 2014. Wall modeling for implicit large-eddy simulation and immersed-interface methods. *Theor. Comput. Fluid Dyn.* 28:1–21
- Choi H, Moin P. 2012. Grid-point requirements for large eddy simulation: Chapman’s estimates revisited. *Phys. Fluids* 24:011702
- Chung MH. 2006. Cartesian cut cell approach for simulating incompressible flows with rigid bodies of arbitrary shape. *Comput. Fluids* 35:607–23
- Costa P, Boersma BJ, Westerweel J, Breugem WP. 2015. Collision model for fully resolved simulations of flows laden with finite-size particles. *Phys. Rev. E* 92:053012
- Cristallo A, Verzicco R. 2006. Combined immersed boundary/large-eddy-simulations of incompressible three dimensional complex flows. *Flows Turbul. Combust.* 77:3–26
- Das S, Panda A, Deen NG, Kuipers JAM. 2018. A sharp-interface immersed boundary method to simulate convective and conjugate heat transfer through highly complex periodic porous structures. *Chem. Eng. Sci.* 191:1–18
- de Tullio MD, De Palma P, Iaccarino G, Pascasio G, Napolitano M. 2007. An immersed boundary method for compressible flows using local grid refinement. *J. Comput. Phys.* 181:2098–117
- Dhamankar NS, Blaisdell GA, Lyrantzis AS. 2016. *Implementation of a wall-modeled sharp immersed boundary method in a high-order large eddy simulation tool for jet aeroacoustics*. Paper presented at 54th AIAA Aerospace Sciences Meeting, San Diego, CA, AIAA Pap. 2016-0257
- Dihn QV, Glowinski R, He J, Kwock V, Pan TW, Periaux J. 1992. Lagrange multiplier approach to fictitious domain methods: application to fluid dynamics and electromagnetics. In *Proceedings of the Fifth*

- International Symposium on Domain Decomposition Methods for PDE*, ed. DE Keyes, TF Chan, G Meurant, JS Scroggs, RG Voigt, pp. 151–94. Philadelphia: SIAM
- Dorodnitsyn A, Meller NA. 1971. Application of the small parameters method to the solution of the Navier–Stokes equations. In *Proceedings of the Second All-Republic Conference on Hydromechanics, Heat and Mass Transfer*. Kiev: Kiev State Univ. Press
- Durbin PA, Iaccarino G. 2002. An approach to local refinement of structured grids. *J. Comput. Phys.* 181:639–53
- Evans M, Harlow FH. 1957. *The particle in cell method for hydrodynamic calculations*. Tech. Rep. 2139, Los Alamos Sci. Lab., Los Alamos, NM
- Fadlun EA, Verzicco R, Orlandi P, Mohd-Yusof J. 2000. Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations. *J. Comput. Phys.* 123:35–60
- Favier J, Revell A, Pinelli A. 2014. A lattice Boltzmann–immersed boundary method to simulate the fluid interaction with moving and slender flexible objects. *J. Comput. Phys.* 261:145–61
- Gatski T, Bonnet J. 2013. *Compressibility, Turbulence and High Speed Flow*. Oxford: Academic. 2nd ed.
- Gentry RA, Martin RE, Daly BJ. 1966. An Eulerian differencing method for unsteady compressible flow problems. *J. Comput. Phys.* 1:87–118
- Gobal K, Grandhi R. 2017. *High-fidelity aerodynamic shape optimization using immersed boundary method*. Paper presented at 35th AIAA Applied Aerodynamics Conference, Denver, AIAA Pap. 2017-4364
- Goldstein D, Handler R, Sirovich L. 1993. Modeling a no-slip flow boundary with an external force field. *J. Comput. Phys.* 105:354–66
- Goldstine H. 1980. *The Computer from Pascal to von Neumann*. Princeton, NJ: Princeton Univ. Press
- Goza D, Colonius T. 2017. A strongly-coupled immersed-boundary formulation for thin elastic structures. *J. Comput. Phys.* 336:401–11
- Griffith BE, Patankar NA. 2020. Immersed methods for fluid–structure interaction. *Annu. Rev. Fluid Mech.* 52:421–48
- Harlow FH. 1957. Hydrodynamic problems involving large fluid distortions. *J. Assoc. Comput. Mach.* 4:137–42
- Huang DZ, Avery P, Farhat C. 2020. *Modeling, simulation and validation of supersonic parachute inflation dynamics during Mars landing*. Paper presented at AIAA SciTech 2020 Forum, Orlando, FL, AIAA Pap. 2020-0313
- Huang JM, Shelley MJ, Stein DB. 2021. A stable and accurate scheme for solving the Stefan problem coupled with natural convection using the immersed boundary smooth extension method. *J. Comput. Phys.* 432:110162
- Iaccarino G, Ham F. 2005. Automatic mesh generation for LES in complex geometries. In *Annual Research Briefs 2005*, pp. 19–30. Stanford, CA: Cent. Turbul. Res.
- Iaccarino G, Moreau S. 2006. Natural and forced conjugate heat transfer in complex geometries on Cartesian adapted grids. *J. Fluids Eng.* 128(4):838–46
- Iaccarino G, Verzicco R. 2003. Immersed boundary technique for turbulent flow simulations. *Appl. Mech. Rev.* 52:421–48
- Jenkins N, Maute K. 2016. An immersed boundary approach for shape and topology optimization of stationary fluid–structure interaction problems. *Struct. Multidisc. Optim.* 54:1191–208
- Jindal S, Khalighi B, Iaccarino G. 2005. *Numerical investigation of road vehicle aerodynamics using the immersed boundary RANS approach*. Tech. Pap. 2005-01-0546, SAE Int., Warrendale, PA
- Kalitzin G, Iaccarino G, Khalighi B. 2003. *Towards an immersed boundary solver for RANS simulations*. Paper presented at 41st Aerospace Sciences Meeting and Exhibit, Reno, NV, AIAA Pap. 2003-0770
- Kang S. 2015. An improved near-wall modeling for large-eddy simulation using immersed boundary methods. *Int. J. Numer. Methods Fluids* 78:76–88
- Kang S, Iaccarino G, Moin P. 2009. Accurate immersed-boundary reconstructions for viscous flow simulations. *AIAA J.* 47(7):1750–60
- Kempe T, Fröhlich J. 2012. An improved immersed boundary method with direct forcing for the simulation of particle laden flows. *J. Comput. Phys.* 231:3663–84
- Kempe T, Vowinckel B, Fröhlich J. 2014. On the relevance of collision modeling for interface-resolving simulations of sediment transport in open channel flow. *Int. J. Multiphase Flows* 58:214–35
- Khalighi B, Zhang S, Koromilas C, Balkanyi S, Bernal L, et al. 2001. *Experimental and computational study of unsteady wake flow behind a bluff body with a drag reduction device*. Tech. Pap. 2001-01B-207, SAE Int., Warrendale, PA

- Kim J, Kim D, Choi H. 2001. An immersed-boundary finite-volume method for simulations of flow in complex geometries. *J. Comput. Phys.* 171:132–50
- Kuznetsov YA. 2001. Domain decomposition and fictitious domain methods with distributed Lagrange multipliers. In *Proceedings of the Thirteenth International Conference on Domain Decomposition Methods*, ed. N Debit, M Garbey, R Hoppe, J Periaux, D Keyes, Y Kuznetsov, pp. 67–75. Barcelona: CIMNE
- Lahooti M, Kim D. 2017. Multi-body interaction effect on the energy harvesting performance of a flapping hydrofoil. *Renew. Energy* 130:460–73
- Lapka P, Furmański P. 2017. Immersed boundary method for radiative heat transfer problems in nongray media with complex internal and external boundaries. *J. Heat Transf.* 139(2):022702
- Larsson J, Kawai S, Bodart J, Bermejo-Moreno I. 2016. Large eddy simulation with modeled wall-stress: recent progress and future directions. *Mech. Eng. Rev.* 3(1):00418
- Lačis U, Taira K, Bagheri S. 2016. A stable fluid–structure-interaction solver for low-density rigid bodies using the immersed boundary projection method. *J. Comput. Phys.* 305:300–18
- Lavoie P, Radenac E, Blanchard G, Laurendeau E, Villedieu P. 2021. Penalization method for Eulerian droplet impingement simulations toward icing applications. *ALAA J.* 60(2):641–53
- Lavoie P, Radenac E, Blanchard G, Laurendeau E, Villedieu P. 2022. Immersed boundary methodology for multistep ice accretion using a level set. *J. Aircr.* 59(4):912–26
- Leonardi S, Orlandi P, Smalley RJ, Djenidi L, Antonia R. 2003. Direct numerical simulations of turbulent channel flow with transverse square bars on one wall. *J. Fluid Mech.* 491:229–38
- Lu J, Tan MD, Peters EAJF, Kuipers JAM. 2018. Direct numerical simulations of reactive fluid–particle systems using an immersed boundary method. *Ind. Eng. Chem. Res.* 57:15565–78
- Mittal R, Iaccarino G. 2005. Immersed boundary methods. *Annu. Rev. Fluid Mech.* 37:239–61
- Mittal R, Moin P. 1997. Suitability of upwind-biased finite difference schemes for large-eddy simulation of turbulent flows. *ALAA J.* 35(8):1435–37
- Mohammadi B, Pironneau O. 2004. Shape optimization in fluid mechanics. *Annu. Rev. Fluid Mech.* 36:255–79
- Mohd-Yusof J. 1997. Combined immersed-boundary/spline methods for simulations of flow in complex geometries. In *Annual Research Briefs 1997*, pp. 317–27. Stanford, CA: Cent. Turbul. Res.
- Möes N, Dolbow JE, Belytschko T. 1999. A finite element method for crack growth without remeshing. *Int. J. Numer. Methods Eng.* 46:131–50
- Moin P, Mahesh K. 1998. Direct numerical simulation: a tool in turbulence research. *Annu. Rev. Fluid Mech.* 50:539–78
- Orlandi P, Leonardi S. 2006. DNS of turbulent channel flows with two- and three-dimensional roughness. *J. Turbul.* 7:N73
- O’Rourke J. 1998. *Computational Geometry in C*. Cambridge, UK: Cambridge Univ. Press
- Peskin CS. 1972. Flow patterns around heart valves: a numerical method. *J. Comput. Phys.* 10:252–71
- Peskin CS. 1982. The fluid dynamics of heart valves: experimental, theoretical, and computational methods. *Annu. Rev. Fluid Mech.* 14:235–59
- Peskin CS. 2002. The immersed boundary method. *Acta Numer.* 11:479–517
- Picano F, Breugen WP, Brandt L. 2015. Turbulent channel flow of dense suspensions of neutrally buoyant spheres. *J. Fluid Mech.* 764:63–487
- Piomelli U. 2008. Wall-layer models for large-eddy simulations. *Prog. Aerosp. Sci.* 44:437–46
- Pirozzoli S, Orlandi P, Bernardini M. 2012. The fluid dynamics of rolling wheels at low Reynolds number. *J. Fluid Mech.* 706:496–533
- Pope SB. 2000. *Turbulent Flows*. Cambridge, UK: Cambridge Univ. Press
- Rangarajan R, Lew AJ, Buscaglia GC. 2009. A discontinuous-Galerkin-based immersed boundary method with nonhomogeneous boundary conditions and its application to elasticity. *Comput. Methods Appl. Mech. Eng.* 198:1513–34
- Reichardt H. 1951. Vollständige Darstellung der turbulenten Geschwindigkeitsverteilung in glatten Leitungen. *Z. Angew. Math. Mech.* 31:208–19
- Reynolds WC. 1990. The potential and limitations of direct and large eddy simulations. In *Whitber Turbulence? Turbulence at the Crossroads*, ed. JL Lumley, pp. 313–43. Berlin: Springer-Verlag
- Rich M, Blackman SS. 1962. *A method for Eulerian fluid dynamics*. Tech. Rep. 2826, Los Alamos Sci. Lab., Los Alamos, NM

- Roman F, Armenio V, Fröhlich J. 2009. A simple wall layer model for large eddy simulation with immersed boundary method. *Phys. Fluids* 21:101701
- Saadat A, Guido CJ, Iaccarino G, Shaqfeh ESG. 2018. Immersed-finite-element method for deformable particle suspensions in viscous and viscoelastic media. *Phys. Rev. E* 98:063316
- Saiki EM, Biringen S. 1996. Numerical simulation of a cylinder in uniform flow: application of a virtual boundary method. *J. Comput. Phys.* 123:450–65
- Saulev VK. 1963. [On the solution of some boundary value problems on high performance computers by fictitious domain method]. *Siber. Math. J.* 4:912–25 (In Russian)
- Schillinger D, Harari I, Hsu M-C, Kamensky D, Stoter SKF, et al. 2016. The non-symmetric Nitsche method for the parameter-free imposition of weak boundary and coupling conditions in immersed finite elements. *Comput. Methods Appl. Mech. Eng.* 309:625–52
- Seo JH, Mittal R. 2011. A sharp-interface immersed boundary method with improved mass conservation and reduced spurious pressure oscillations. *J. Comput. Phys.* 230:7347–63
- Sepahi F, Pande N, Chong KL, Mul G, Verzicco R, et al. 2022. The effect of buoyancy driven convection on the growth and dissolution of bubbles on electrodes. *Electrochim. Acta* 403:139616
- Sethian JA, Wiegmann A. 2000 Structural boundary design via level set and immersed interface methods. *J. Comput. Phys.* 163(2):489–528
- Shokin YI. 2009. Computational fluid mechanics in Russia. In *100 Volumes of ‘Notes on Numerical Fluid Mechanics’*, ed. EH Hirshel, E Krause, pp. 117–31. Berlin: Springer
- Spandan V, Meschini V, Ostilla-Monico R, Lohse D, Querzoli G, et al. 2017. A parallel interaction potential approach coupled with the immersed boundary method for fully resolved simulations of deformable interfaces and membranes. *J. Comput. Phys.* 348:567–90
- Spandan V, Verzicco R, Lohse D. 2018. Physical mechanisms governing drag reduction in turbulent Taylor–Couette flow with finite-size deformable bubbles. *J. Fluid Mech.* 849:R3
- Specklin M, Delauré Y. 2018. A sharp immersed boundary method based on penalization and its application to moving boundaries and turbulent rotating flows. *Eur. J. Mech. B* 70:130–47
- Stevens RJAM, Lohse D, Verzicco R. 2014. Sidewall effects in Rayleigh–Bénard convection. *J. Fluid Mech.* 741:1–27
- Tafti DK, He L, Nagendra K. 2014. Large eddy simulation for predicting turbulent heat transfer in gas turbines. *Philos. Trans. R. Soc. A* 372:20130322
- Taira K, Colonius T. 2007. The immersed boundary method: a projection approach. *J. Comput. Phys.* 225:2118–37
- Tessicini F, Iaccarino G, Fatica M, Wang M, Verzicco R. 2002. Wall modeling for large-eddy simulation using an immersed boundary method. In *Annual Research Briefs 2002*, pp. 181–87. Stanford, CA: Cent. Turbul. Res.
- Uhlmann M. 2005. An immersed boundary method with direct forcing for the simulation of particulate flows. *J. Comput. Phys.* 2:448–76
- Vannella M, Balaras E. 2009. A moving-least-squares reconstruction for embedded-boundary formulations. *J. Comput. Phys.* 228:6617–28
- Verzicco R. 2002. Side wall finite conductivity effects in confined turbulent thermal convection. *J. Fluid Mech.* 473:201–10
- Verzicco R. 2004. Effects of non perfect thermal sources in turbulent thermal convection. *Phys. Fluids* 16(6):1965–79
- Verzicco R. 2022. Electro-fluid-mechanics of the heart. *J. Fluid Mech.* 941:P1
- Verzicco R, Iaccarino G, Fatica M, Moin P, Khalighi B. 2002. Large eddy simulation of a road vehicle with drag-reduction devices. *ALAA J.* 40(12):2447–55
- Verzicco R, Mohd-Yusof J, Orlandi P, Haworth D. 2000. Large eddy simulation in complex geometric configurations using boundary body forces. *ALAA J.* 38(3):427–33
- Verzicco R, Querzoli G. 2021. On the collision of a rigid sphere with a deformable membrane in a viscous fluid. *J. Fluid Mech.* 914:A19
- Viecelli JA. 1969. A method for including arbitrary external boundaries in the MAC incompressible fluid computing technique. *J. Comput. Phys.* 4:543–51

- Viecelli JA. 1971. A computing method for incompressible flows bounded by moving walls. *J. Comput. Phys.* 8:119–43
- Viola F, Meschini V, Verzicco R. 2020. Fluid–structure–electrophysiology interaction (FSEI) in the left-heart: a multi-way coupled computational model. *Eur. J. Mech. B* 79:212–32
- Wang M, Moin P. 2000. Computation of trailing-edge flow and noise using large-eddy simulation. *AIAA J.* 38:2001–9
- Welch JE, Harlow FH, Shannon JP, Daly BJ. 1966. *The MAC method: a computing technique for solving viscous, incompressible, transient fluid flow problems involving free surfaces*. Tech. Rep. 3425, Los Alamos Sci. Lab., Los Alamos, NM
- White F. 2006. *Viscous Flow*. Boston: McGraw-Hill. 3rd ed.
- Wilcox DC. 1993. *Turbulence Modeling for CFD*. La Cañada, CA: DCW Indust.
- Yanenko NN. 1967. [*Fractional Step Methods for Multidimensional Problems of Mathematical Physics*]. Novosibirsk, Russ.: Nauka Press (In Russian)
- Yang J, Balaras E. 2006. An embedded-boundary formulation for large-eddy simulation of turbulent flows interacting with moving boundaries. *J. Comput. Phys.* 215:12–40
- Yang XIA, Sadique J, Mittal R, Meneveau C. 2015. Integral wall model for large eddy simulations of wall-bounded turbulent flows. *Phys. Fluids* 27:025112
- Ye T, Mittal R, Udaykumar HS, Shyy W. 1999. An accurate Cartesian grid method for viscous incompressible flows with complex immersed boundaries. *J. Comput. Phys.* 156:209–40
- Zenit R, Hunt ML. 1999. Mechanics of immersed particle collisions *J. Fluids Eng.* 121:179–84
- Zheng L, Hedrick TL, Mittal R. 2013. A multi-fidelity modelling approach for evaluation and optimization of wing stroke aerodynamics in flapping flight. *J. Fluid Mech.* 721:118–54
- Zhu X, Verzicco R, Zhang X, Lohse D. 2018. Diffusive interaction of multiple surface nanobubbles: shrinkage, growth, and coarsening. *Soft Matter* 14(11):2006–14