Journal of Computational Physics 229 (2010) 701–727

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

A low numerical dissipation immersed interface method for the compressible Navier–Stokes equations

K. Karagiozis, R. Kamakoti, C. Pantano *

Department of Mechanical Science and Engineering, University of Illinois at Urbana-Champaign, 1206 W. Green St. MC-244, Urbana, IL 61801, United States

ARTICLE INFO

Article history:
Received 9 April 2009
Received in revised form 11 September 2009
Accepted 5 October 2009
Available online 14 October 2009

Keywords:
Immersed interface method
Compressible Navier–Stokes equations
Centered finite-differences
Non-dissipative methods

ABSTRACT

A numerical method to solve the compressible Navier–Stokes equations around objects of arbitrary shape using Cartesian grids is described. The approach considered here uses an embedded geometry representation of the objects and approximate the governing equations with a low numerical dissipation centered finite-difference discretization. The method is suitable for compressible flows without shocks and can be classified as an immersed interface method. The objects are sharply captured by the Cartesian mesh by appropriately adapting the discretization stencils around the irregular grid nodes, located around the boundary. In contrast with available methods, no jump conditions are used or explicitly derived from the boundary conditions, although a number of elements are adopted from previous immersed interface approaches. A new element in the present approach is the use of the summation-by-parts formalism to develop stable non-stiff first-order derivative approximations at the irregular grid points. Second-order derivative approximations, as those appearing in the transport terms, can be stiff when irregular grid points are located too close to the boundary. This is addressed using a semi-implicit time integration method. Moreover, it is shown that the resulting implicit equations can be solved explicitly in the case of constant transport properties. Convergence studies are performed for a rotating cylinder and vortex shedding behind objects of varying shapes at different Mach and Reynolds numbers.

© 2009 Elsevier Inc. All rights reserved.

1. Introduction

A number of numerical methods have been developed to solve fluid dynamical problems around complex geometries utilizing different Cartesian grid-based discretizations. These techniques present two considerable advantages: first, the fluid dynamics solver can be made very efficient by exploiting the underlying Cartesian grid topology and second, it is very inexpensive to incorporate complex objects using an embedded geometry approach in comparison with generating two- or three-dimensional meshes. These fictitious domain methods, or domain embedding methods, have been introduced by Hyman [1] and Saul'ev [2] to solve boundary value problems related to elliptic equations. The theory and application of these methods to elliptic equations is well developed and include the immersed interface method (IIM) of LeVeque and Li [3–12] and the distributed Lagrange multiplier (DLM) technique [13,14]. Extensions of these techniques to the second-order wave-equation [15] and the parabolic heat-diffusion equation [16] have been proposed. In fluid dynamics, a number of methods have been developed to maximize the advantages of the embedded geometry representation, starting from the immersed boundary method (IBM) which was further extended and popularized by Peskin [17–21]. In the IBM technique, the coupling

* Corresponding author. Tel.: +1 217 244 1412.
E-mail address: cpantano@illinois.edu (C. Pantano).

0021-9991/$ - see front matter © 2009 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2009.10.005

between a moving boundary and the fluid is modeled using a suitable distributed force over a band of grid cells surrounding the interface. The IIM and DLM techniques have also been extended to fluid dynamics problems. Most of these applications are concerned with low to moderate Reynolds number flows for incompressible fluids [22,23] and particulate flows [24]. In the case of compressible flows, the Ghost fluid method (GFM) [25] uses Eulerian level sets to capture interfaces and Godunov-type upwind discretizations to solve the inviscid Euler equations. Many methods have been developed for viscous incompressible flow [26–37] and inviscid compressible flow [38–44] but considerably fewer approaches have been developed for compressible viscous flows [45,46].

The boundary conditions in the IIM at the object boundary, which is arbitrary with respect to the Cartesian grid, alter the discretization of the governing equations next to the boundary. Specialized formulas are then utilized at these so-called irregular grid points. In IIM, the boundary conditions are expressed in terms of discontinuous changes, or jumps, in the solution and its derivatives across the boundary. Thus, the new discretization at the irregular grid points arises without additional effort if all the jumps are known at the boundary [47,48]. In the more common case, when the boundary conditions are of Dirichlet or Neumann type, i.e. not given as jumps, one needs to transform the boundary data to derive the jumps. These are then incorporated in the IIM formulation [49,50]. This problem is not as severe in the IBM method because the interface is already diffuse and there is some flexibility as to how to implement the closure. The sharp nature of the interface in the IIM is not as forgiving as in the IBM and usually the jumps derived from Dirichlet conditions result in highly stiff equations. Stiffness here refers to the singular behavior of the coefficients of the stencils obtained at the irregular points when the distance between these points to the boundary approaches zero. In formulations utilizing jumps, the stiffness may be partially obscured by the procedure that determines the jumps from the boundary conditions. In this case, the determinant of the matrix that relates the jumps to the boundary conditions generally approaches zero as the irregular grid points approach the interface. In general, this behavior cannot be avoided when employing the embedded geometry representation, since the object boundary is placed at arbitrary locations throughout the domain. Dirichlet data is important in the present context because no-slip and imposed temperature are conventional boundary conditions for the compressible Navier–Stokes equations. This difficulty of sharp interface methods is well recognized but its solution has only been partially developed and it invariably involves ill-conditioned matrices or stiff time integration problems for elliptic or parabolic equations, respectively. The latter can be converted into a problem of the former type by use of implicit or semi-implicit time integration techniques [51–54]. Tryggvason et al. [55] report that several thousand iterations are required for the implicit system of equations to converge. This particular problem can be addressed in some cases by using special iterative methods [56,57,47,58] or a splitting technique [59] to separate the problem into a hierarchy of non-stiff problems which can be solved independently using standard numerical methods. Unfortunately, this becomes cumbersome for systems of nonlinear equations or when it is impractical to assemble, or partially assemble, the full matrix of the implicit system of equations.

So far, all the methods described above have been developed from the point of view of maximizing accuracy and considerably less effort has been devoted to the understanding of their stability. The stability of these boundary treatments can be investigated by linearizing the equations and studying the spectrum of the numerical operator, which is usually referred to as the matrix method, or by using the energy method; a global stability approach [60]. While it is possible to derive conditions under which the linearized equations are stable by either method, it is unclear how to use this information to develop stable discretizations for arbitrary relative positions of the interface with respect to the mesh. Few studies have considered the stability of these methods, e.g. high-order finite-difference time-domain methods for Maxwell's equations [61] and implicit discretizations of the IBM equations [62,63,51,64]. The reason stability has received less attention is possibly that many studies using embedded methods are concerned with low Reynolds number flows or use upwind discretizations which can add substantial numerical dissipation. While this appears successful, there are undesirable effects caused by the introduction of artificial numerical dissipation. For example, the numerics can interfere with the flow physics of interest, e.g. acoustic waves or turbulence become artificially damped, motivating the interest in non-dissipative treatments. Unfortunately, if the amount of numerical dissipation is reduced, or removed completely, compressible flows tend to be far less forgiving and usually even mild numerical instabilities quickly lead to failure of the simulation (appearance of negative density or pressure). Replacing upwind by centered discretizations removes numerical dissipation but the discretization is usually unstable. Away from boundaries, this problem has been resolved by using skew-symmetric formulations to preserve stability [65–68].

In this paper, we partially address the stability problem of an immersed interface method for the compressible Navier–Stokes equations by utilizing the theory of summation-by-parts (SBP) operators [69,70] to derive a stable immersed interface approximation for the advection derivatives. Numerical experiments suggest that this approach prevents the appearance of spurious numerical instabilities, which otherwise create shock-like regions around complex boundaries. Moreover, different from IIM formulations, the new approach completely eliminates the need to deal with jumps at the object boundary. Finally, the method is combined with semi-implicit time integration to remove any stiffness present in the operators and the implicit equations are solved explicitly for the particular case of constant transport properties.

2. Approach and conventions

It is assumed that the governing equations are discretized using a Cartesian mesh, independent of the presence of objects of arbitrary shape. Although the formulation is presented for a Cartesian uniform mesh, the proposed method will be integrated with a patch-based adaptive mesh refinement solver for better resolution control in the examples in Section 5. The discretization stencils next to an object are then altered only at the so-called 'irregular' grid points close to the object boundary. In two-dimensions, let us denote a function $q(\vec{x})$ of the independent variable vector $\vec{x} = \{x, y\}$ and its approximation at grid locations $\{x_i, y_j\}$ by the grid function, $q_{i,j} = q(x_i, y_j)$, where $i,j \in \mathbb{Z}$ and $x_i = i\Delta x$ and $y_j = j\Delta y$. Explicit notation of the dependence of the function on additional independent variables or parameters will be indicated when necessary, e.g. if the function or its approximation depend on time. Moreover, constant indices will be omitted when only one index is varying and there is no ambiguity, i.e. when approximating derivatives with respect to one independent variable.

Fig. 1 shows a sketch of a two-dimensional Cartesian mesh of size $\{\Delta x, \Delta y\}$ overlapping an arbitrarily shaped boundary, Γ, which is assumed smooth enough so that its characteristic radius of curvature is well resolved by the Cartesian grid. No sharp turning or bending of the boundary within a grid cell is allowed. This is important since the type of finite-difference approximation built on the Cartesian mesh can only resolve features that vary on length scales that are equal to or larger than Δx and Δy. The physical domain, Ω, is divided into a fluid subdomain, Ω^+, and a solid (fictitious fluid) subdomain, Ω^-, such that $\Omega = \Omega^+ \cup \Omega^-$. The boundary separating the fluid from the fictitious fluid domain is denoted by $\partial\Omega^+ = \Gamma$ with coordinates given by $\vec{x}_\Gamma$. It is presumed that $\vec{x}_\Gamma$ is known and it is at least $\mathcal{C}^0$, and therefore it can be parameterized without difficulty. Each grid point on the finite-difference discretization of the domain can be accessed through the multi-index $\mathcal{I} = \{i,j \in \mathbb{Z}\}$. Therefore, a subset of $\mathcal{I}$ corresponds to cells that belong to the fluid domain, $\mathcal{I}^+ = \{i,j \in \mathbb{Z} : (x_i, y_j) \in \Omega^+\}$. Furthermore, only second-order accurate finite-difference approximations requiring up to three grid points to approximate first- and second-order derivatives will be considered. These derivatives are given by the standard formulas

$$\left.\frac{\partial q}{\partial x}\right|_{\mathrm{fd}} = \frac{q_{i+1} - q_{i-1}}{2\Delta x}, \tag{1}$$

$$\left.\frac{\partial^2 q}{\partial x^2}\right|_{\mathrm{fd}} = \frac{q_{i+1} - 2q_i + q_{i-1}}{\Delta x^2}, \tag{2}$$

$$\left.\frac{\partial^2 q}{\partial x \partial y}\right|_{\mathrm{fd}} = \frac{q_{i+1,j+1} - q_{i-1,j+1} - q_{i+1,j-1} + q_{i-1,j-1}}{4\Delta x \Delta y}. \tag{3}$$

The subscript fd is used to explicitly indicate the use of standard centered finite-difference formulas. Equivalent formulas are applied in the other directions, y for two-dimensional problems and y and z for three-dimensional problems. The cells next to the interface are irregular in the sense that the standard centered stencils, (1)–(3), cannot be employed to calculate the finite-difference approximations of the derivatives because at least one element of the stencil will be reaching into Ω^-. However, these irregular cells are a subset of $\mathcal{I}^+$, denoted by $\mathcal{I}_I = \{i,j \in \mathcal{I}^+ : \min_{y_\Gamma = y_j} d_\Gamma(x_i, y_j) \leqslant \Delta x \text{ or } \min_{x_\Gamma = x_i} d_\Gamma(x_i, y_j) \leqslant \Delta y\}$, where $d_\Gamma(x, y)$ denotes the distance between a point with coordinates (x, y) and $\vec{x}_\Gamma$. The regular points are denoted as the complement $\mathcal{I}_R = \mathcal{I}^+ \setminus \mathcal{I}_I$. Fig. 1 shows an example of a solid–fluid interface with irregular grid points, denoted by I_1–I_4. Auxiliary points arising from the intersection of the Cartesian grid lines passing through these points with the interface $\vec{x}_\Gamma$ are denoted by $\mathcal{P}_1^y, \mathcal{P}_2^x, \mathcal{P}_2^y, \mathcal{P}_3^x, \mathcal{P}_3^y$ and $\mathcal{P}_4^x$, where the superscript indicates the Cartesian line of intersection, x or y in a two-dimensional grid and the subscript indicates the corresponding irregular point associated with this auxiliary point. In the present approach, each primary grid line passing through an irregular point in $\mathcal{I}_I$ is allowed to intersect the boundary only one time. If this was not the case, the radius of curvature of the interface around this point would be less than the available resolution

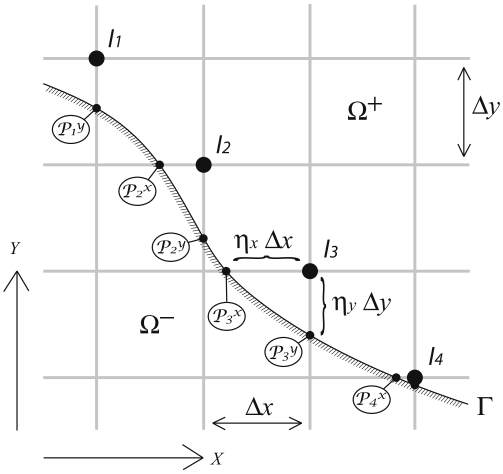


Fig. 1. Sketch denoting a two-dimensional Cartesian mesh with an embedded interface, Γ, including the notation utilized to identify different points of interest in the discretization.

or the object involved thin filament-like regions which cannot be resolved by the Cartesian mesh. The distance between $\mathcal{P}_I^x$ and its associated irregular point, I, is $|\eta_x \Delta x|$ and the sign of η_x is chosen to be negative or positive, depending on whether $\mathcal{P}_I^x$ is located to the left or right of I, respectively. Similarly, the distance of $\mathcal{P}_I^y$ to I is $|\eta_y \Delta y|$ and the sign of η_y is negative or positive, depending on whether $\mathcal{P}_I^y$ is located below or above I, respectively. A second set of irregular points appear when the interface boundary, $\vec{x}_\Gamma$, does not cross the $x = x_i$ or $y = y_j$ lines but instead it cuts through a diagonal line or corner of a cell. This set is denoted by $\mathcal{I}_M = \{i, j \in \mathcal{I}_R : \min d_\Gamma(x_i, y_j) \leqslant \sqrt{\Delta x^2 + \Delta y^2}\}$. It can be seen that the set $\mathcal{I}_M$ matters only for the corrections of cross or mixed derivative approximations since one element of (3) would reach into Ω^-.

The regular grid points involved in the discretization of the governing equations at the irregular grid locations are also classified depending on their location with respect to I. The points lying on the Cartesian lines passing through I will be referred to as the regular points and they are denoted by $\mathcal{R}_I^x$ and $\mathcal{R}_I^y$. The regular grid points that belong to the corner of the cell surrounding I are indicated by $\mathcal{C}_I^k$, where k takes values from the set of {ne, se, nw, sw}. The index k, can only take three out of the four possible values from this set, otherwise I would be a regular point. Fig. 2 shows the complete set of possible cases in two-dimensions. In this figure, the irregular points are shown by a large filled circle. All possible cases are depicted in Fig. 2 or can be transformed to these cases by a rotation or reflection. The subscript I denoting the irregular point multi-index has been dropped in these figures for simplicity. In each case, only those locations that are used by the proposed method are marked; bending of the interface within a cell such that the Cartesian grid lines are intersected more than once is not allowed. The mesh should be refined if this were to happen. Furthermore, the remaining regular points in Ω^+, where the standard centered finite-difference method (FD) is applied without corrections, are not marked. Case (a), (b) and (c) contain only

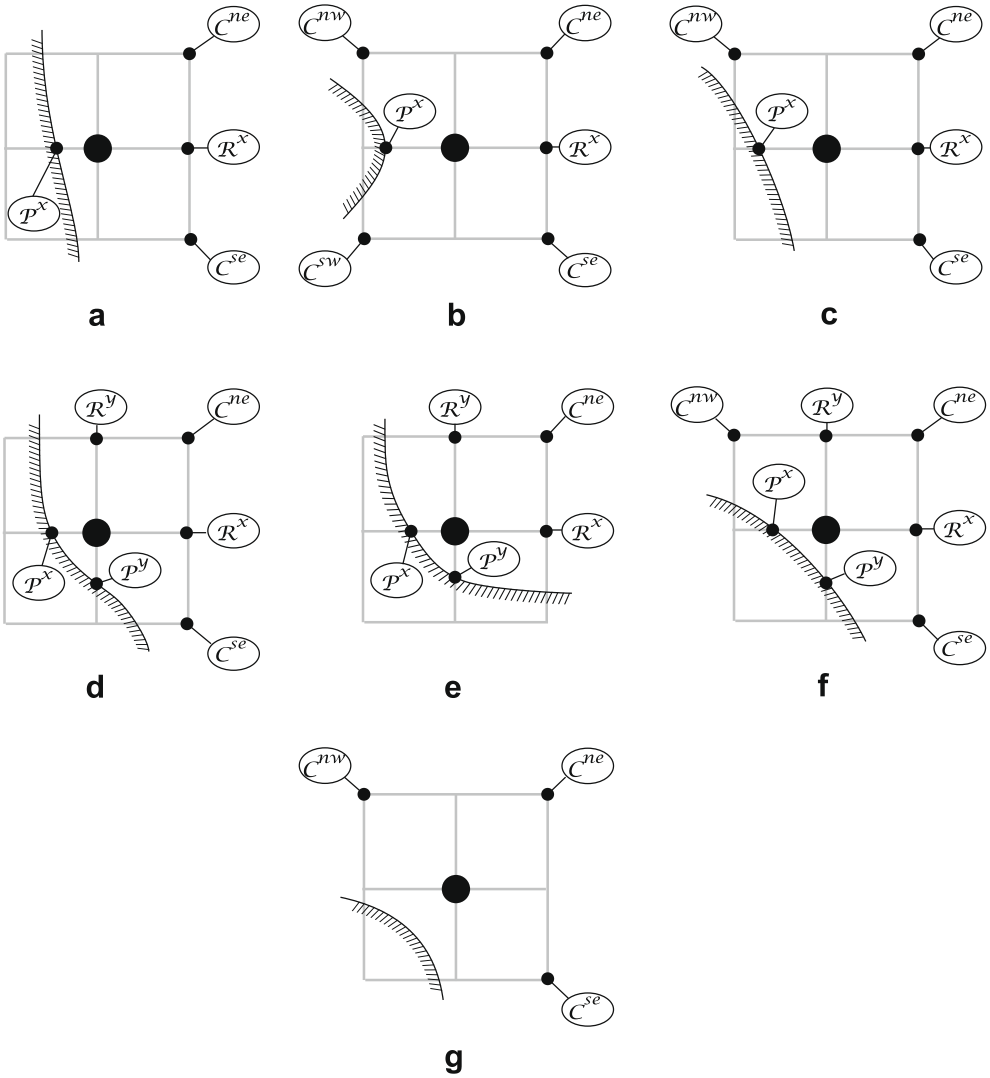


Fig. 2. Elementary cells depicting all basic boundary and cell intersections.

one boundary intersection with the primary grid lines, while case (b), (e) and (f) contain two. Special care must be employed with case (g) which only affects the approximation of mixed derivatives.

The proposed method is formulated using corrections to the standard finite-difference formulas at the irregular grid points [71,47,48], such that

$$\left.\frac{\partial q}{\partial x}\right|_{\mathrm{ir}} = \left.\frac{\partial q}{\partial x}\right|_{\mathrm{fd}} + \left[\frac{\partial q}{\partial x}\right]_{\mathrm{cr}}, \tag{4}$$

where ir denotes the approximation at the irregular grid point and $[\;]_{\mathrm{cr}}$ denotes the correction with respect to the standard finite-difference approximation. In the first stage of the fluid dynamics solver, the governing equations are discretized ignoring the presence of an embedded boundary. In the second stage, the system of equations is updated by correcting the discretization at the irregular points. Centered finite-difference formulas are employed to ensure that artificial numerical dissipation is not introduced by the discretization in the bulk of the domain and this is the reason the approach is termed a low numerical dissipation method. This applies primarily to the approximation of the first-order derivatives in the advection terms of the Navier–Stokes equations. Centered differences are also used for the second-order derivatives, appearing in the viscous and heat conduction terms, to ensure that these approximations are parabolic. Obviously these finite-difference approximations cannot be applied at the irregular points, see Fig. 1.

The presence of the boundary and type of boundary conditions applied on the boundary interface need to be appropriately treated by the numerical approximation. For example, the approach of [48] gives

$$\left.\frac{\partial q}{\partial x}\right|_{\mathrm{ir}} = \left.\frac{\partial q}{\partial x}\right|_{\mathrm{fd}} - \frac{1}{2\Delta x}\sum_{l=0}^{2}\frac{(\eta\Delta x)^l}{l!}[q^{(l)}], \tag{5}$$

$$\left.\frac{\partial^2 q}{\partial x^2}\right|_{\mathrm{ir}} = \left.\frac{\partial^2 q}{\partial x^2}\right|_{\mathrm{fd}} + \frac{1}{\Delta x^2}\sum_{l=0}^{3}\frac{(\eta\Delta x)^l}{l!}[q^{(l)}], \tag{6}$$

where the ‘jumps’ are defined, e.g. in the x-direction, by

$$[q^{(n)}] = q^{(n)+} - q^{(n)-} = \lim_{\substack{x\to x_\Gamma,\\ x\in\Omega^+}}\frac{\partial^n q}{\partial x^n} - \lim_{\substack{x\to x_\Gamma,\\ x\in\Omega^-}}\frac{\partial^n q}{\partial x^n}, \tag{7}$$

where $q^{(n)+}$ denotes the value of nth derivative of $q(x)$ at x_Γ, as one approaches it from the fluid side and $q^{(n)-}$ is the value reached from the solid side. A straight forward implementation of these corrections is achieved when all the jumps appearing in (5) and (6) are known explicitly. While this is possible for some problems, it is not generally the case for Dirichlet type boundary conditions, e.g. no-slip boundary condition, or Neumann conditions, e.g. heat flux, since only some of the jumps are known *a priori*. Despite this inconvenience, several methodologies have been proposed to recover the jumps from the boundary conditions themselves [50,10] so that (5) and (6) can be used. Note that the concept of corrections is not dependent on the use of jumps. For example, the corrections can be determined by other means, e.g. using more general unequally spaced finite-difference formulas [72,73]. This is the approach pursued in the present method. We restrict our approach to the compressible Navier–Stokes equations using Dirichlet boundary conditions, which correspond to the boundary conditions of ‘no-slip’ for the velocity and imposed temperature. The boundary condition for the density, which requires a different treatment, follows the general requirements for well-posedness of the Navier–Stokes equations [74]. Finally, to complete the specification of the problem, it is assumed that the region inside the solid objects are filled with a fluid of constant properties, i.e. fixed density, pressure and zero velocity.

3. Spatial derivatives at irregular grid points

The Navier–Stokes equations involve two types of spatial derivatives. First-order derivatives in the advection terms and second-order derivatives, both diagonal and mixed, in the viscous and thermal conduction terms. Ideally, all approximating stencils should be stable and uniformly non-stiff, i.e. independent of the location of the interface with respect to the location of the Cartesian mesh. Unfortunately, it is not always possible to achieve this objective using minimal-width stencils; stiffness is just a mathematical manifestation of the rigidity of the boundary condition, e.g. no amount of force exerted by the fluid can deform a rigid solid. Therefore, one faces the problem of constructing stable and non-stiff discretization stencils for the spatial derivatives. Stiffness is often overcome by using implicit or semi-implicit time integration methods. Unfortunately, solving large nonlinear systems of implicit equations, e.g. using iterative methods, is not always computationally practical. To avoid these difficulties, a compromise is achieved in this paper by approximating the first- and second-order derivatives with conceptually different approaches.

A low-dissipation compressible flow formulation usually requires rewriting the advection terms in a form that is discretely stable, known loosely as the skew-symmetric form (strictly speaking, this is incompressible terminology). This form of the equations, which is equivalent in the continuum case but is more suitable for centered finite-difference stencils, increases the number of first-order derivative operators in the governing equations; see for example [68] for several possible forms. In two-dimensions, see Section 4.2, the conservative equations require a minimum of eight first-order spatial derivatives while the stable skew-symmetric form requires 16 (assuming maximum reutilization of terms). Therefore, it is preferable to use non-stiff first-order derivative stencils to avoid having to deal with this problem in the time integration stage; which will otherwise result in a fully coupled implicit system of equations. In the case of the two-dimensional formulation of the viscous and heat conduction terms there are only six diagonal second-order derivatives and two mixed derivatives (after developing all the terms). These terms appear linearly in the conservation equations and therefore a stiff approximation of these terms is easily handled at a later stage by a semi-implicit time integration method. Finally, the formulation of the mixed derivatives remain an unresolved issue that the proposed approach will handle by well known least-squares methods.

The adopted methodology is in accordance with the development of an efficient two-dimensional approach that can easily be extended to the three-dimensional case without new particular developments. Therefore, focusing on strictly Cartesian closures is favored over more isotropic approaches which could potentially result in high computational overheads owing to the complex topologies that will arise with three-dimensional boundaries.

3.1. First-order spatial derivative

The objective in this section is the design of an approximation for the first-order spatial derivative that is stable, at least first-order accurate and involves stencil coefficients that are not stiff, i.e. they are uniformly bounded with respect to the distance of the interface to the location of the irregular grid point. This is the principal novelty of the proposed method for the compressible Navier–Stokes equations. In order to prevent stiff coefficients in the stencil, the approximation at the irregular points must involve at least three points and can only be of first-order accuracy. Certainly, it is possible to derive a second-order accurate stencil using just two points, one regular and one irregular, and the boundary value using unequally spaced (general) finite-differences. Unfortunately, this stencil has stiff coefficients. Alternatively, a family of non-stiff stencils can be obtained by reducing the order of accuracy of the approximation. In this case, it is found that most of these stencils are unstable even for the linear first-order hyperbolic equation.

To address this stability problem, we employ the theory of operators satisfying the SBP property [75,69]. Let us consider the grid arrangement sketched in Fig. 3 for the approximation of the first-order derivative of the one-dimensional Cauchy problem

$$\frac{\partial q}{\partial t} + a\frac{\partial q}{\partial x} = 0, \tag{8}$$

in $0 \leqslant x, t$, for $a > 0$ with boundary and initial conditions given by $q(x = 0, t) = q_{bc}(t)$ for $t \geqslant 0$ and $q(x, t = 0) = q_o(x)$, for $0 \leqslant x$, respectively. The semi-discrete approximation of (8) for the grid depicted by Fig. 3 is given by

$$\mathbf{P}\frac{\partial \vec{q}}{\partial x} = \frac{1}{\Delta x}\mathbf{Q}\vec{q}, \tag{9}$$

where Δx is the mesh spacing between the uniform-spaced nodes such that the numerical approximation of the first-order spatial derivative is $\mathbf{D}_1 = \mathbf{P}^{-1}\mathbf{Q}$. The function $q(x,t)$ is presumed known at the nodal locations $q_j(t) = q(x_j, t)$, and the vector containing all the unknowns is defined as $\vec{q} = \{q_{bc}, q_1, q_2, \ldots\}$, with

$$x_j = \begin{cases} 0 & j = 0, \\ (-\eta + j - 1)\Delta x & j > 0, \end{cases} \tag{10}$$

for $j \in \mathbb{N}_0$. In this formulation $\eta < 0$ denotes the offset from the boundary in terms of Δx where the irregular point, which requires a special stencil for the embedded geometry approach, is located. The value of η is chosen here to be negative to indicate that the irregular point is to the right of the boundary, whereas a positive value indicates the opposite. This convention is used to simplify the implementation of the immersed interface method in Section 4.2. The matrices **P** and **Q** have elements corresponding to the discrete stencil. The matrix **P** is symmetric positive definite and **Q** is skew-symmetric except for the corner elements [75]. One can show that the energy norm $E = a^{-1}\Delta x\ \vec{q}^T\mathbf{P}\vec{q}$ is discretely conserved, apart from boundary terms, and obeys the equation

$$\frac{d\dot{E}}{d\dot{t}} = -\vec{q}(\mathbf{Q}^T + \mathbf{Q})\vec{q} = q_{bc}^2(t), \tag{11}$$

where $\mathbf{Q}^T + \mathbf{Q} = \text{diag}(-1, 0, 0, \ldots)$. Therefore, the total energy of the system is bounded and grows only by that amount provided by the boundary value [75,69]; implying asymptotic stability of the semi-discrete approximation to (8). Note that while this strategy carries over to nonlinear hyperbolic problems, it does not give a general proof of stability for other

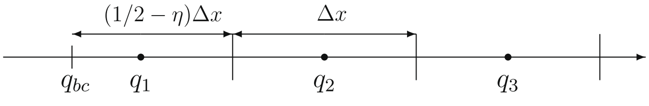


Fig. 3. Sketch of the unequally spaced boundary mesh ($\eta < 0$).

equations involving higher derivatives or multidimensional equations (unless the discrete operators can be symmetrized by the same matrix). Nevertheless, experience has shown that numerical stability is usually dominated by the discretization of the advection terms in the Navier–Stokes equations, hence the choice of (8). Although the formal derivation of the stencil assumes that a is constant, when the actual derivative operator is constructed, $\mathbf{D}_1$, one can use it with variable a or with nonlinear functions, e.g. advection terms of the Navier–Stokes equations. If the solution is sufficiently smooth (well resolved and away from shocks), the stability property of the stencil is still preserved, shown by linearization around a base solution, cf. [76,60].

The matrices $\mathbf{P}$ and $\mathbf{Q}$ for a second-order accurate approximation in the interior of the domain are given by

$$\mathbf{P} = \begin{pmatrix} p_{11} & p_{12} & 0 & \dots \\ p_{12} & p_{22} & 0 & \dots \\ 0 & 0 & 1 & \dots \\ . & . & . & \dots \end{pmatrix}, \quad \mathbf{Q} = \begin{pmatrix} -\frac{1}{2} & q_{12} & 0 & 0 & \dots \\ -q_{12} & 0 & \frac{1}{2} & 0 & \dots \\ 0 & -\frac{1}{2} & 0 & \frac{1}{2} & \dots \\ . & . & . & . & \dots \end{pmatrix}. \tag{12}$$

The unspecified coefficients p_{11}, p_{12}, p_{22} and q_{12} are chosen to satisfy the order of accuracy conditions at the boundary and the positive definite property of $\mathbf{P}$. The accuracy conditions are obtained by imposing that the approximations of the derivative of the polynomial function $f(x, x_j) = (x - x_j)^n$ for $n = 0, 1$ are exact at nodes $j = 0, 1$. These conditions are simplified by using the vectors $\vec{e}^0 = [0, -\eta, -\eta + 1, \dots]$, $\vec{e}^1 = [\eta, 0, 1, \dots]$, and $\vec{1} = [1, 1, 1, \dots]$, yielding

$$\{Q\}_0 \vec{1} = 0, \tag{13}$$

$$\{Q\}_1 \vec{1} = 0, \tag{14}$$

$$\{P\}_0 \vec{1} = \{Q\}_0 \vec{e}^0, \tag{15}$$

$$\{P\}_1 \vec{1} = \{Q\}_1 \vec{e}^1, \tag{16}$$

where the notation $\{.\}_i$ indicates the ith row of the matrix. The solution of this system of equations gives

$$p_{11} = -p_{12} - \frac{\eta}{2}, \tag{17}$$

$$p_{22} = -p_{12} + \frac{(1 - \eta)}{2}, \tag{18}$$

$$q_{12} = \frac{1}{2}, \tag{19}$$

where p_{12} is a free parameter. Evaluation of the complete derivative approximation, $\mathbf{D}_1 = \mathbf{P}^{-1}\mathbf{Q}$, shows that stiffness is present in the determinant of $\mathbf{P}$. It is imperative to avoid $\det(\mathbf{P}) \sim \eta^m$ for $\eta \to 0$, where $m \in \mathbb{N}$. In our case,

$$\det(\mathbf{P}) = p_{12}\left(\eta - \frac{1}{2}\right) - \frac{1}{4}\eta(1 - \eta), \tag{20}$$

and the particular value to avoid is $p_{12} = 0$. To complete the analysis, the eigenvalues of $\mathbf{P}$ are given by

$$\lambda^{\pm} = \frac{1 - 4p_{12} \pm \sqrt{1 + 16p_{12}^2} - 2\eta}{4}. \tag{21}$$

It can be verified that $\lambda^{\pm} > 0$ when $p_{12} \leqslant 0$ if $\eta < 0$, which is the case in the present analysis. Any value of p_{12} less than zero will suffice including, as indicated by a reviewer, the case $p_{12} = 0$ which leads to a well behaved stencil since the factor proportional to η in the denominator, $\det(\mathbf{P})$, is cancelled by the numerators which are also proportional to η. In this case, the factor of the first element of the sum defining the energy norm E in (11) is $-\eta/2$. The particular stencil that has been used satisfactorily in Section 5 uses $p_{12} = -1/2$ and gives the following approximation of the first-order derivative approximation at the irregular points,

$$\frac{\partial q}{\partial x} \approx \frac{(1 - \eta)q_2 + q_1 + (\eta - 2)q_{bc}}{\Delta x(\eta^2 - 3\eta + 1)}, \tag{22}$$

which is not stiff since the roots of the denominator are all positive, $\eta = (3 \pm \sqrt{5})/2$. Moreover, one can verify that this formula reduces to the two-point biased stencil for $\eta = 0$. Although we experimented with other values of p_{12}, obtaining indistinguishable results, a rational approach to select an optimal value of this parameter remained elusive.

The present approach differs substantially from a popular approach used with SBP operators for equally spaced grids that employs a penalty-based technique to enforce the boundary conditions [70]. In that case, q_{bc} is kept as an element of the vector of unknowns, $\vec{q}$, and a penalty term is added to the governing equations to drive the value of the node at the boundary toward the boundary condition. In principle, there is no impediment to the use of the penalty approach [70]. One drawback of this approach is that additional storage and computational cost is incurred since one has to add an additional set of elements corresponding to the floating values of the vector of state at the boundary, which are driven by the penalty method to the corresponding boundary condition, and solve for the governing equations there. In practice, the main difficulty is that the

Cartesian solver does not have a mechanism to discretize the governing equations at the boundary intersections and addressing this requires modifying the original solver beyond just using a correction stage. In the present approach, the row associated with q_{bc} is discarded once the matrix $\mathbf{D}_1$ is determined since q_{bc} is not part of the vector of unknowns and the particular formula in (22) is used instead at the irregular grid point. This approach, where the equations are segregated at the boundary, does not automatically inherit the energy-stability properties of the SBP operators. Nevertheless, it is shown next that the present approach retains the stability of the original method.

Eliminating q_{bc} from the vector of unknowns modifies the energy estimate (11). To derive the energy estimate that applies in this case consider the new vector of state $\vec{q} = \{q_1, q_2, \ldots\}$ and the elements of the approximation of the first-order derivative, given by

$$\mathbf{D}_1 = \begin{pmatrix} d_{11} & d_{12} & 0 & 0 & \ldots \\ -\frac{1}{2} & 0 & \frac{1}{2} & 0 & \ldots \\ 0 & -\frac{1}{2} & 0 & \frac{1}{2} & \ldots \\ . & . & . & . & \ldots \end{pmatrix}, \quad \vec{b} = \begin{pmatrix} d_{10} \\ 0 \\ 0 \\ . \end{pmatrix}, \tag{23}$$

where

$$d_{10} = \frac{(\eta - 2)}{\triangle}, \quad d_{11} = \frac{1}{\triangle}, \quad d_{12} = \frac{(1-\eta)}{\triangle}, \quad \triangle = (\eta^2 - 3\eta + 1), \tag{24}$$

are the coefficients in (22). The approximation to (8) is now given by

$$\frac{d\vec{q}}{dt} + \frac{1}{\Delta x}(\mathbf{D}_1\vec{q} + \vec{b}q_{bc}) = 0. \tag{25}$$

Lets define a new energy norm, $E^* = a^{-1}\Delta x\ \vec{q}^T H\vec{q}$, with

$$\mathbf{H} = \begin{pmatrix} w & 0 \\ 0 & \mathbf{I} \end{pmatrix}, \tag{26}$$

where w is a positive scalar, to be determined, and $\mathbf{I}$ is the identity matrix. Manipulating (25) in the usual manner, to arrive at an equation for E^*, gives

$$\frac{dE^*}{dt} + \vec{q}^T(\mathbf{HD}_1 + \mathbf{D}_1^T\mathbf{H})\vec{q} + 2q_{bc}\ \vec{q}^T\mathbf{H}\vec{b} = 0. \tag{27}$$

The only way to eliminate the terms involving q_2 in

$$\mathbf{HD}_1 + \mathbf{D}_1^T\mathbf{H} = \begin{pmatrix} 2wd_{11} & wd_{12} - \frac{1}{2} & 0 & 0 & \ldots \\ wd_{12} - \frac{1}{2} & 0 & 0 & 0 & \ldots \\ 0 & 0 & 0 & 0 & \ldots \\ . & . & . & . & \ldots \end{pmatrix}, \tag{28}$$

is to choose

$$w = \frac{1}{2d_{12}} = \frac{\triangle}{2(1-\eta)} > 0, \tag{29}$$

which reduces (27) to

$$\frac{dE^*}{dt} + 2w(d_{11}q_1^2 + q_{bc}q_1d_{10}) = 0. \tag{30}$$

Using the expressions for d_{11}, d_{10}, w and completing the square, gives

$$\frac{\dot{E^*}}{\dot{t}} = -\frac{(q_1 - (1-\eta/2)q_{bc})^2}{1-\eta} + \frac{(1-\eta/2)^2}{1-\eta}q_{bc}^2. \tag{31}$$

Eq. (31) establishes the stability of the approximation since the first term on the right-hand side is always negative and, therefore, cannot induce growth in E^*. The second term in the right-hand side is due to the boundary condition and it is analogous to the corresponding term in (11). This completes the proof that segregating the approximation at the boundary from the original SBP stencil retains the stability property in the new energy norm E^*.

3.2. Second-order spatial derivative

The theory behind the approximation of the first-order derivative described in the previous section could be applied to produce stable second-order derivative approximations [77]. However, it is not possible to derive non-stiff formulas for stencils using only three points at irregular points; for this case at least four points are required. This implies wider stencil approximations than those investigated here are needed and an extension of [77] to non-uniform grid spacing is necessary to remove stiff coefficients. Unfortunately, our attempts at this goal have been unsuccessful so far. Therefore, it was found that unequally spaced stencils based on the classical Taylor series method worked well when their stiff coefficients, as $\eta \to 0$, are coupled with a semi-implicit time integration method.

Using the conventions of the previous section, the unequally spaced finite-difference formula evaluated at an irregular point is given by

$$\left.\frac{\partial^2 q}{\partial x^2}\right|_{\mathrm{ir}} = 2\frac{(q_2 - q_1)}{\Delta x^2\,(1-\eta)} + 2\frac{(q_1 - q_{bc})}{\Delta x^2 \eta (1-\eta)}. \tag{32}$$

It is expected that this formula will suffer order of accuracy degradation, from second to first, when η is not of order $-1 + O(\Delta x/L)$, where L is a characteristic spatial dimension. Therefore, (32) is usually at best first-order accurate in Δx for arbitrary η [72]. This is compatible with (22) and comparable to other immersed interface methods.

From the implementation point of view, (32) is decomposed into non-stiff and stiff parts, according to

$$\left.\frac{\partial^2 q}{\partial x^2}\right|_{\mathrm{ir}} = \left.\frac{\partial^2 q}{\partial x^2}\right|_{\mathrm{ns}} + \left.\frac{\partial^2 q}{\partial x^2}\right|_{\mathrm{st}}, \tag{33}$$

where the non-stiff part is given by

$$\left.\frac{\partial^2 q}{\partial x^2}\right|_{\mathrm{ns}} = 2\frac{(q_2 - q_1)}{\Delta x^2\,(1-\eta)}, \tag{34}$$

and the stiff part is given by

$$\left.\frac{\partial^2 q}{\partial x^2}\right|_{\mathrm{st}} = \frac{S_x}{\eta}(q_1 - q_{bc}), \tag{35}$$

with

$$S_x = \frac{2}{\Delta x^2 (1-\eta)}. \tag{36}$$

This decomposition will be used later to manipulate the semi-implicit time integration method and transform it into an overall explicit method.

3.3. Mixed spatial derivatives

The approach implicitly followed throughout the development of the approximations is to consider the function $q(x,y)$ around irregular grid points as a uniformly valid Taylor expansion that is accurate to third-order, so that its first- and second-order derivatives are accurate to second- and first-order in Δx, respectively. In other terms, it was assumed that

$$q(x,y) = q_{\mathrm{ir}} + x'\left.\frac{\partial q}{\partial x}\right|_{\mathrm{ir}} + y'\left.\frac{\partial q}{\partial y}\right|_{\mathrm{ir}} + \frac{1}{2}x'^2\left.\frac{\partial^2 q}{\partial x^2}\right|_{\mathrm{ir}} + \frac{1}{2}y'^2\left.\frac{\partial^2 q}{\partial y^2}\right|_{\mathrm{ir}} + x'y'\left.\frac{\partial^2 q}{\partial x \partial y}\right|_{\mathrm{ir}} + O(\Delta x^2 \Delta y, \Delta x \Delta y^2), \tag{37}$$

where $x' = x - x_i$ and $y' = y - y_j$, is valid over each irregular grid cell, including the general boundary section cutting through it. The coefficients of this two-dimensional interpolant correspond to the different derivatives developed previously, except for the last one.

Eq. (37) could be used, in principle, in two ways. One can assume that the coefficients of the derivatives are unknown and evaluate the polynomial in x' and y' at specific locations around an irregular grid cell to generate a linear system of equations whose solution is the approximations to the derivatives. If the number of equations is equal to the number of coefficients, the unequally spaced finite-difference formulas are recovered, similar to [10]. However, these have the undesirable stiff, $\sim \eta^{-1}$, behavior inherited and must be avoided. Another approach is to choose more colocation points to evaluate (37) and solve the overly constrained problem by a least-squares method. This does not generally lead to stiff stencils but it was found that these stencils do not necessarily result in stable discretizations when applied to the first- or second-order derivatives. Two qualitatively different results were obtained using the least-squares method for all the derivatives. In one case, irregular grid stencils would result in unstable stencils and in another case the stencils were asymptotically stable but generated spurious modes which the compressible Navier–Stokes equations could not damp sufficiently fast to prevent the formation of what appeared to be shocks next to the boundaries. These spurious modes corresponded to imaginary eigenvalues in the second-order unmixed derivatives. Ideally, the eigenvalues should be all real and negative, however, the least-squares method cannot enforce this property. It is for this reason that it was first preferred to approximate the first-order derivative by constructing a stable approximation at irregular points, then construct the unequally spaced finite-difference approximation of the second-order derivative, which is stable for a diffusion problem, and leave the mixed derivatives to be approximated separately using the least-squares method.

Thus, the approach employed here uses the least-squares technique to determine the mixed spatial derivative by evaluating (3) at all diagonal regular grid nodes available as well as using the boundary condition at the intersection points $\mathcal{P}^x$ and $\mathcal{P}^y$. A similar technique was employed by Wiegmann and Bube [48] for elliptic problems and by Kirshman and Liu [78] to solve the Euler equations using embedded Cartesian meshes. In the present approach, only the mixed derivatives, which are absent in the inviscid Euler equations, are computed using a least-squares method. This solution uses all the available information around the irregular cell to estimate the mixed derivative and, given that the system is over-constrained, no stiff coefficients are recovered. The approximation of the mixed spatial derivative was obtained, for the case of two intersection points of the Cartesian grid lines with the interface, by constructing the vectors

$$\vec{\alpha} = \left[x'_{c_1}y'_{c_1}, x'_{c_2}y'_{c_2}, \ldots, x'_{p_1}y'_{p_1}, x'_{p_2}y'_{p_2}\right], \tag{38}$$

and

$$\vec{\beta} = \left[\beta\left(x'_{c_1}, y'_{c_1}\right), \beta\left(x'_{c_2}, y'_{c_2}\right), \ldots, \beta\left(x'_{p_1}, y'_{p_1}\right), \beta\left(x'_{p_2}, y'_{p_2}\right)\right], \tag{39}$$

where

$$\beta(x', y') = q - q_{\mathrm{ir}} - x'\frac{\partial q}{\partial x}\bigg|_{\mathrm{ir}} - y'\frac{\partial q}{\partial y}\bigg|_{\mathrm{ir}} - \frac{1}{2}x'^2\frac{\partial^2 q}{\partial x^2}\bigg|_{\mathrm{ir}} - \frac{1}{2}y'^2\frac{\partial^2 q}{\partial y^2}\bigg|_{\mathrm{ir}}. \tag{40}$$

The subscripts c and p refer to the corner cells $\mathcal{C}$ and boundary points $\mathcal{P}$, as seen in Fig. 2, respectively. These vectors can then be used in the normal equations of the least-squares for the mixed derivative, giving

$$\frac{\partial^2 q}{\partial x \partial y}\bigg|_{\mathrm{ir}} = \frac{\vec{\alpha}^T\vec{\beta}}{\vec{\alpha}^T\vec{\alpha}}. \tag{41}$$

The coefficients of the stencil in (41) is never stiff if the system of equations is over-constrained, $size(\vec{\alpha}) = size(\vec{\beta}) > 1$.

3.4. Time integration

Explicit time integration combined with the potentially large coefficients, i.e. (35), in the second-order derivatives present in the viscous and thermal conduction terms of the compressible Navier–Stokes equations may require excessively small time steps. One approach to deal with this stiffness is to use implicit or semi-implicit time integration methods. Of all available implicit methods, a certain balance must be achieved in terms of complexity and computational cost in most realistic applications and often only a few choices make practical sense. In our case, only few terms of the discretized governing equations involve the problematic terms $\sim \eta^{-1}$. Therefore, it appears that a semi-implicit method is the most advantageous approach in terms of the computational cost for our system. If we restrict consideration to multi-stage methods, the use of centered stencils in the bulk of the domain dictates that at least a third-order accurate Runge–Kutta scheme must be used. Consequently, a third-order accurate additive semi-implicit Runge–Kutta (ASIRK) method was used to integrate the governing equations in time, since the stiff and non-stiff terms are known beforehand. Those terms that do not contain the numerical stiff behavior are integrated explicitly while those terms that contain it are integrated implicitly.

Let us consider the system of equations

$$\frac{dq}{dt} = f(q, t) + g(q, t), \tag{42}$$

where f and g denote the nonstiff and the stiff terms in the governing equation, respectively. Time integration methods for this type of split systems are discussed in detail by Yoh and Zhong [79]. Several methods are available depending on whether g is autonomous in time or not and on the desired order of accuracy. The third-order accurate method labeled SIRK-3A by Yoh and Zhong [79] was used in this work, given by

$$q^{n+1} = q^n + w_1k_1 + w_2k_2 + w_3k_3, \tag{43}$$

and

$$k_1 = \Delta t[f(q^n, t_n + r_1\Delta t) + g(q^n + d_1k_1)], \tag{44}$$
$$k_2 = \Delta t[f(q^n + b_{21}k_1, t_n + r_2\Delta t) + g(q^n + c_{21}k_1 + d_2k_2)], \tag{45}$$
$$k_3 = \Delta t[f(q^n + b_{31}k_1 + b_{32}k_1, t_n + r_3\Delta t) + g(q^n + c_{31}k_1 + c_{32}k_2 + d_3k_3)], \tag{46}$$

where Δt is the time step size, and b_{ij} and d_i are coefficients that have been determined along with w_i to ensure third-order accuracy and stability of the numerical method. These coefficients are reproduced here from [79], giving

$$\begin{aligned}
&w_1 = \tfrac{1}{8}, \quad w_2 = \tfrac{1}{8}, \quad w_3 = \tfrac{3}{4},\\
&b_{21} = \tfrac{8}{7}, \quad b_{31} = \tfrac{71}{252}, \quad b_{32} = \tfrac{7}{36},\\
&c_{21} = \tfrac{5589}{6524}, \quad c_{31} = \tfrac{7691}{26{,}096}, \quad c_{32} = -\tfrac{26{,}335}{78{,}288},\\
&d_1 = \tfrac{3}{4}, \quad d_2 = \tfrac{75}{233}, \quad d_3 = \tfrac{65}{168},\\
&r_1 = 0, \quad r_2 = b_{21}, \quad r_3 = b_{31} + b_{32}.
\end{aligned} \tag{47}$$

The key feature of these implicit methods is that the k_i coefficients are defined implicitly only through g. If g is simple enough or involves localized interactions with neighboring grid cells, it is possible to explicitly determine the k_i, i.e. resulting in an explicit method, while retaining the stability properties of the time integration method. This feature of the proposed combined methodology is discussed below.

4. Application to the compressible Navier–Stokes equations

4.1. Governing equations

The governing equations of a two-dimensional compressible flow in conservation form for mass, momentum and energy are given by

$$\frac{\partial \vec{q}}{\partial t} + \frac{\partial \vec{f}_k(\vec{q})}{\partial x_k} = 0, \tag{48}$$

where the vector of state is defined as

$$\vec{q} = (\rho, m_1, m_2, E)^{\mathrm{T}}, \tag{49}$$

with $m_1 = \rho u_1 = \rho u, m_2 = \rho u_2 = \rho v$, where the density is denoted by ρ, the velocity components by u_k and the state relationship for the total energy E is given by

$$E = \frac{p}{\gamma - 1} + \frac{1}{2}\rho u_k u_k, \tag{50}$$

where γ is the specific heat capacity ratio, p is the pressure and repeated indices imply summation. The system of equations is closed thermodynamically by the equation of state

$$p = \rho R T, \tag{51}$$

where R is the gas constant and T is temperature.

The flux tensor can be further decomposed into inviscid and viscous components, according to

$$\vec{f}_k(\vec{q}) = \vec{f}_k^{\mathrm{inv}}(\vec{q}) + \vec{f}_k^{\mathrm{vis}}(\vec{q}), \tag{52}$$

and defined as

$$\vec{f}_k^{\mathrm{inv}}(\vec{q}) = \begin{pmatrix} \rho u_k \\ \rho u_1 u_k + \delta_{1k} p \\ \rho u_2 u_k + \delta_{2k} p \\ (E + p) u_k \end{pmatrix}, \quad \vec{f}_k^{\mathrm{vis}}(\vec{q}) = \begin{pmatrix} 0 \\ -\sigma_{1k} \\ -\sigma_{2k} \\ g_k - \sigma_{kj} u_j \end{pmatrix}. \tag{53}$$

The deviatoric Newtonian viscous stress tensor is defined as

$$\sigma_{ik} = \mu\left[\left(\frac{\partial u_k}{\partial x_i} + \frac{\partial u_i}{\partial x_k}\right) - \frac{2}{3}\frac{\partial u_j}{\partial x_j}\delta_{ik}\right], \tag{54}$$

and the heat conduction is given by

$$g_k = -\lambda \frac{\partial T}{\partial x_k}. \tag{55}$$

The shear viscosity, μ, and thermal conductivity, λ are assumed known constants.

4.2. Numerical discretization

The conservation equations (48), are discretized using a centered finite-difference approximation after rewriting the operators in skew-symmetric and discrete internal energy conservation form, for the momentum and energy equations, respectively. This approach has been shown to lead to enhanced numerical stability when the numerical method is not dissipative [68] and it has been used successfully in simulations of the Navier–Stokes equations as well as in large-eddy simulations [80]. The resulting discrete equations can be expressed as

$$\frac{d\vec{Q}}{dt} = \mathcal{N}(\vec{Q}) + \mathcal{L}(\vec{Q}), \tag{56}$$

where $\vec{Q}$ denotes the vector of the discretized conserved variables $\vec{Q}_{i,j}(t) = \vec{q}(x_i, y_j, t)$. The right-hand side of (56) has been split into the contribution from the advection terms, $\mathcal{N}(\vec{Q})$, and the contribution from the viscous and heat conduction terms, $\mathcal{L}(\vec{Q})$. The advection term is discretized according to the skew-symmetric form [68], giving

$$\mathcal{N}_\rho = -(D_x m_1 + D_y m_2), \tag{57}$$

$$\mathcal{N}_1 = -\frac{1}{2}(D_x(m_1 u_1) + D_y(m_1 u_2) + m_1 D_x u_1 + m_2 D_y u_1 - u_1 \mathcal{N}_\rho), \tag{58}$$

$$\mathcal{N}_2 = -\frac{1}{2}(D_x(m_2 u_1) + D_y(m_2 u_2) + m_1 D_x u_2 + m_2 D_y u_2 - u_2 \mathcal{N}_\rho), \tag{59}$$

$$\begin{aligned}\mathcal{N}_E = -\frac{1}{2}(&D_x(m_1 e) + D_y(m_2 e) + m_1 D_x e + m_2 D_y e - e\mathcal{N}_\rho + u_1(D_x(m_1 u_1) + D_y(m_1 u_2)) + u_2(D_x(m_2 u_1) + D_y(m_2 u_2)) \\ &+ m_1(u_1 D_x u_1 + u_2 D_y u_1) + m_2(u_1 D_x u_2 + u_2 D_y u_2)) - p(D_x u_1 + D_y u_2) - u_1 D_x p - u_2 D_y p,\end{aligned} \tag{60}$$

where $e = E/\rho - \frac{1}{2}(u^2 + v^2)$ and the viscous and heat conduction terms are discretized in the developed form,

$$\mathcal{L}_\rho = 0, \tag{61}$$

$$\mathcal{L}_1 = \mu\left(\frac{4}{3}D_{xx}u_1 + D_{yy}u_1 + \frac{1}{3}D_{xy}u_2\right), \tag{62}$$

$$\mathcal{L}_2 = \mu\left(\frac{4}{3}D_{yy}u_2 + D_{xx}u_2 + \frac{1}{3}D_{xy}u_1\right), \tag{63}$$

$$\mathcal{L}_E = \lambda(D_{xx}T + D_{yy}T) + u_1\mathcal{L}_1 + u_2\mathcal{L}_2 + \sigma_{11}D_x u_1 + \sigma_{12}(D_x u_2 + D_y u_1) + \sigma_{22}D_y u_2. \tag{64}$$

The conservative form of the Navier–Stokes equations and this particular form are equivalent in the continuous limit but the present scheme has good stability properties [68]. Unlike the incompressible equations, it has not been possible to prove discrete stability of the nonlinear equations for all time. It is possible to prove discrete stability for the constant (frozen) coefficients case [81] using SBP operators, which assuming the solution is smooth implies stability for the nonlinear case [82]. Unfortunately, this is only a necessary condition.

In the previous formulas, the derivative operators are defined according to

$$D_x = \begin{cases} \frac{\partial}{\partial x}\big|_{\mathrm{fd}} & \text{in } \mathcal{I}_R, \\ \frac{\partial}{\partial x}\big|_{\mathrm{ir}} & \text{in } \mathcal{I}_I, \end{cases} \tag{65}$$

where the regular finite-differences approximation to the first-order derivative are given by (1) and the irregular is given by (22). The same decomposition applies to all other derivatives: D_y, D_{xx}, D_{yy} and D_{xy}.

4.3. Interface corrections

In the current implementation, as is common in IIM, $\mathcal{N}$ and $\mathcal{L}$ are first computed using the regular finite-difference formulas, first case in (65), at all mesh points of the domain. No distinction is applied to different grid nodes at this stage. In a second stage, only those terms which are affected by the presence of the fluid–solid interface are updated using corrections. In the present method, all possible topologies encountered by the Cartesian mesh are shown in Fig. 2. A similar approach is commonly pursued in the incompressible case, e.g. [31].

Lets decompose the right-hand side of (56) into

$$\mathcal{N} = \mathcal{N}^{\mathrm{fd}} + \mathcal{N}^{\mathrm{cr}}, \quad \mathcal{L} = \mathcal{L}^{\mathrm{fd}} + \mathcal{L}^{\mathrm{cr}}, \tag{66}$$

where $\mathcal{N}^{\mathrm{fd}}$ and $\mathcal{L}^{\mathrm{fd}}$ correspond to (57)–(59), (60), (62)–(64), respectively, evaluated using the standard centered finite-difference formulas, first case in (65). The corrections to the advection terms at the irregular points only, $\mathcal{I}_I$, are now given by

$$\mathcal{N}_\rho^{\mathrm{cr}} = -\left(D_x^{\mathrm{cr}} m_1 + D_y^{\mathrm{cr}} m_2\right), \tag{67}$$

$$\mathcal{N}_1^{\mathrm{cr}} = -\frac{1}{2}\left(D_x^{\mathrm{cr}}(m_1 u_1) + D_y^{\mathrm{cr}}(m_1 u_2) + m_1 D_x^{\mathrm{cr}} u_1 + m_2 D_y^{\mathrm{cr}} u_1 - u_1 \mathcal{N}_\rho^{\mathrm{cr}}\right), \tag{68}$$

$$\mathcal{N}_2^{\mathrm{cr}} = -\frac{1}{2}\left(D_x^{\mathrm{cr}}(m_2 u_1) + D_y^{\mathrm{cr}}(m_2 u_2) + m_1 D_x^{\mathrm{cr}} u_2 + m_2 D_y^{\mathrm{cr}} u_2 - u_2 \mathcal{N}_\rho^{\mathrm{cr}}\right), \tag{69}$$

$$\begin{aligned}\mathcal{N}_E^{\mathrm{cr}} = -\frac{1}{2}\Big(&D_x^{\mathrm{cr}}(m_1 e) + D_y^{\mathrm{cr}}(m_2 e) + m_1 D_x^{\mathrm{cr}} e + m_2 D_y^{\mathrm{cr}} e - e\mathcal{N}_\rho^{\mathrm{cr}} + u_1\left(D_x^{\mathrm{cr}}(m_1 u_1) + D_y^{\mathrm{cr}}(m_1 u_2)\right) + u_2\left(D_x^{\mathrm{cr}}(m_2 u_1) + D_y^{\mathrm{cr}}(m_2 u_2)\right) \\ &+ m_1\left(u_1 D_x^{\mathrm{cr}} u_1 + u_2 D_y^{\mathrm{cr}} u_1\right) + m_2\left(u_1 D_x^{\mathrm{cr}} u_2 + u_2 D_y^{\mathrm{cr}} u_2\right)\Big) - p\left(D_x^{\mathrm{cr}} u_1 + D_y^{\mathrm{cr}} u_2\right) - u_1 D_x^{\mathrm{cr}} p - u_2 D_y^{\mathrm{cr}} p,\end{aligned} \tag{70}$$

where

$$D_x^{\mathrm{cr}} = \frac{\partial}{\partial x}\bigg|_{\mathrm{ir}} - \frac{\partial}{\partial x}\bigg|_{\mathrm{fd}},$$

and similarly for the other derivatives. The corrections involving e and p deserve special attention. For imposed temperature at the boundary, the case considered here,

$$e = \frac{p}{\rho(\gamma - 1)} = \frac{RT}{\gamma - 1},$$

is known and the corrections can always be evaluated. The case for the pressure is different since the density is not known at the boundary. Two obvious choices are possible since the pressure gradient term will not be integrated implicitly. In the first case,

$$D_x^{\mathrm{cr}} p = D_x^{\mathrm{cr}}(\rho RT), \tag{71}$$

while in the second case the pressure corrections are determined according to the chain rule, giving

$$D_x^{\mathrm{cr}} p = RTD_x^{\mathrm{cr}} \rho + \rho RD_x^{\mathrm{cr}} T, \tag{72}$$

and similarly for the y direction derivative. In the second case, the density correction is obtained by extrapolation of the density inside the domain, Ω^+, from the known data in the neighborhood of the interface along the corresponding Cartesian direction. This approach is fully consistent with the boundary conditions of the compressible Navier–Stokes equations [74]. The linear extrapolation of the density to $\mathcal{P}^x$ involves the value at the irregular point and $\mathcal{R}^x$, and similarly the extrapolation to $\mathcal{P}^y$ used $\mathcal{R}^y$. Both approaches were tested and produced indistinguishable results, both in terms of convergence and conservation. Therefore, the first approach (71), is used throughout the simulations presented in Section 5.

Finally, the correction to the viscous and heat conduction terms are

$$\mathcal{L}_\rho^{\mathrm{cr}} = 0, \tag{73}$$

$$\mathcal{L}_1^{\mathrm{cr}} = \mu\left(\frac{4}{3}D_{xx}^{\mathrm{cr}}u_1 + D_{yy}^{\mathrm{cr}}u_1 + \frac{1}{3}D_{xy}^{\mathrm{cr}}u_2\right), \tag{74}$$

$$\mathcal{L}_2^{\mathrm{cr}} = \mu\left(\frac{4}{3}D_{yy}^{\mathrm{cr}}u_2 + D_{xx}^{\mathrm{cr}}u_2 + \frac{1}{3}D_{xy}^{\mathrm{cr}}u_1\right), \tag{75}$$

$$\begin{aligned}\mathcal{L}_E^{\mathrm{cr}} = {} & \lambda\left(D_{xx}^{\mathrm{cr}}T + D_{yy}^{\mathrm{cr}}T\right) + u_1\mathcal{L}_1^{\mathrm{cr}} + u_2\mathcal{L}_2^{\mathrm{cr}} + \sigma_{11}^{\mathrm{cr}}D_x^{\mathrm{fd}}u_1 + \sigma_{12}^{\mathrm{cr}}\left(D_x^{\mathrm{fd}}u_2 + D_y^{\mathrm{fd}}u_1\right) + \sigma_{22}^{\mathrm{cr}}D_y^{\mathrm{fd}}u_2 + \sigma_{11}^{\mathrm{fd}}D_x^{\mathrm{cr}}u_1 \\ & + \sigma_{12}^{\mathrm{fd}}\left(D_x^{\mathrm{cr}}u_2 + D_y^{\mathrm{cr}}u_1\right) + \sigma_{22}^{\mathrm{fd}}D_y^{\mathrm{cr}}u_2 + \sigma_{11}^{\mathrm{cr}}D_x^{\mathrm{cr}}u_1 + \sigma_{12}^{\mathrm{cr}}\left(D_x^{\mathrm{cr}}u_2 + D_y^{\mathrm{cr}}u_1\right) + \sigma_{22}^{\mathrm{cr}}D_y^{\mathrm{cr}}u_2,\end{aligned} \tag{76}$$

which can easily be determined since the only part of the dissipation function in the energy equation is quadratic, and the remaining terms are linear.

4.4. Time integration

As discussed previously, the correction for second-order derivatives can be stiff depending on the relative position of the interface with respect to the Cartesian mesh, see (32). This numerical stiffness is addressed by separating the stiff component of D_{xx}^{cr} and D_{yy}^{cr} explicitly. First, let $\mathcal{L}^{\mathrm{cr}}$ be split according to

$$\mathcal{L}^{\mathrm{cr}} = \mathcal{L}^{\mathrm{ns}} + \mathcal{L}^{\mathrm{st}}, \tag{77}$$

where *ns* stands for the non-stiff terms of the irregular points (34), and *st* represents all the stiff contributions, (35). The stiff terms are given by

$$\mathcal{L}_\rho^{\mathrm{st}} = 0, \tag{78}$$

$$\mathcal{L}_1^{\mathrm{st}} = \mu\left(\frac{4}{3}\frac{S_x}{\eta_x}\left(u_1 - u_1^{\mathcal{P}^x}(t)\right) + \frac{S_y}{\eta_y}\left(u_1 - u_1^{\mathcal{P}^y}(t)\right)\right), \tag{79}$$

$$\mathcal{L}_2^{\mathrm{st}} = \mu\left(\frac{4}{3}\frac{S_y}{\eta_y}\left(u_2 - u_2^{\mathcal{P}^y}(t)\right) + \frac{S_x}{\eta_x}\left(u_2 - u_2^{\mathcal{P}^x}(t)\right)\right), \tag{80}$$

$$\mathcal{L}_E^{\mathrm{st}} = \lambda\left(\frac{S_x}{\eta_x}\left(T - T^{\mathcal{P}^x}(t)\right) + \frac{S_y}{\eta_y}\left(T - T^{\mathcal{P}^y}(t)\right)\right) + u_1\mathcal{L}_1^{\mathrm{st}} + u_2\mathcal{L}_2^{\mathrm{st}}. \tag{81}$$

Eqs. (78)–(81) constitute the implicit terms in (42), where the boundary conditions are involved. Note that the Dirichlet boundary conditions appear explicitly now and they must be evaluated at the locations $\mathcal{P}^x$ and $\mathcal{P}^y$, and need not be equal. This distinction disappears for the no-slip boundary condition of non-moving interfaces and isothermal walls.

Inspection of the form of the semi-implicit time integration method and the structure of (78)–(81) shows that one can explicitly solve what appears to be a nonlinear system of implicit equations. First, note that the resulting implicit equations are local, i.e. the values at each irregular node can be determined independently of those at any other location. Implicit time integration used in incompressible formulations also exhibit partial locality where only those equations involving the irregular points need to be solved implicitly to remove a stringent stability constraint [53]. In the present case, the implicit coupling is strictly local and does not involve neighboring points. Grouping together the stiff terms involving S_x and S_y, which behave as η_x^{-1} and η_y^{-1}, respectively, and defining the generalized inverse values

$$\tilde{\eta}_x^{-1} = \begin{cases} 0, & \text{if } \eta_x = 0 \\ \eta_x^{-1}, & \text{otherwise} \end{cases}, \quad \tilde{\eta}_y^{-1} = \begin{cases} 0, & \text{if } \eta_y = 0 \\ \eta_y^{-1}, & \text{otherwise} \end{cases}. \tag{82}$$

allows (78)–(81) to be rewritten as

$$\mathcal{L}^{\text{st}} = \tilde{\eta}_x^{-1} G_x + \tilde{\eta}_y^{-1} G_y, \tag{83}$$

where G_x and G_y are vectors containing the coefficients of the stiff operators S_x and S_y, given by

$$G_{x,1} = \frac{4}{3}\mu S_x\left(u_1 - u_1^{\mathcal{P}^x}(t)\right), \tag{84}$$

$$G_{x,2} = \mu S_x\left(u_2 - u_2^{\mathcal{P}^x}(t)\right), \tag{85}$$

$$G_{x,E} = \lambda S_x(T - T^{\mathcal{P}^x}(t)) + u_1 G_{x,1} + u_2 G_{x,2}. \tag{86}$$

and

$$G_{y,1} = \mu S_y\left(u_1 - u_1^{\mathcal{P}^y}(t)\right), \tag{87}$$

$$G_{y,2} = \frac{4}{3}\mu S_y\left(u_2 - u_2^{\mathcal{P}^y}(t)\right), \tag{88}$$

$$G_{y,E} = \lambda S_y(T - T^{\mathcal{P}^y}(t)) + u_1 G_{y,1} + u_2 G_{y,2}. \tag{89}$$

Here, it is assumed that the transport properties, μ and λ, are constant and independent of temperature. If this were not the case, one can use the present strategy to devise a Picard (fixed-point) iteration scheme for the nonlinear implicit system of equations. This simple iteration approach should converge quickly because the coupling will still be local with the temperature. Furthermore, (62)–(64) and all those derived from these equations need to be rewritten in the appropriate conservative form.

Next, the semi-implicit time integration method (43)–(46), at the irregular points can be reformulated according to

$$\vec{k}_l = \Delta t\left[\underbrace{\mathcal{N}(\vec{Q}_l) + \mathcal{L}^{\text{fd}}(\vec{Q}_l) + \mathcal{L}^{\text{ns}}(\vec{Q}_l)}_{f} + \underbrace{\mathcal{L}^{st}\left(\vec{Q}_l^* + d_l\vec{k}_l\right)}_{g}\right], \tag{90}$$

where

$$\vec{Q}_l = \begin{cases} \vec{Q}^n & l = 1, \\ \vec{Q}^n + b_{21}\vec{k}_1 & l = 2, \\ \vec{Q}^n + b_{31}\vec{k}_1 + b_{32}\vec{k}_2 & l = 3, \end{cases} \tag{91}$$

and

$$\vec{Q}_l^* = \begin{cases} \vec{Q}^n & l = 1, \\ \vec{Q}^n + c_{21}\vec{k}_1 & l = 2, \\ \vec{Q}^n + c_{31}\vec{k}_1 + c_{32}\vec{k}_2 & l = 3. \end{cases} \tag{92}$$

It is possible to rewrite (90) such that the implicit equations are solved explicitly, by introducing

$$\vec{k}_l = \vec{k}_l^f + \Delta t\left[\tilde{\eta}_x^{-1} G_x\left(\vec{Q}_l^* + d_l\vec{k}_l\right) + \tilde{\eta}_y^{-1} G_y\left(\vec{Q}_l^* + d_l\vec{k}_l\right)\right], \tag{93}$$

where

$$\vec{k}_l^f = \Delta t\left[\mathcal{N}(\vec{Q}_l) + \mathcal{L}^{\text{fd}}(\vec{Q}_l) + \mathcal{L}^{\text{ns}}(\vec{Q}_l)\right]. \tag{94}$$

Since not both $\mathcal{P}^x$ and $\mathcal{P}^y$ points are always present at all irregular points, the most restrictive, i.e. stiff, terms needs to be identified explicitly. This can be accomplished by defining $\tilde{\eta}^{-1}$ as

$$\tilde{\eta}^{-1} = \max\left(|\tilde{\eta}_x^{-1}|, |\tilde{\eta}_y^{-1}|\right), \tag{95}$$

and $\tilde{\eta} = 1/\tilde{\eta}^{-1}$. Then, the solution of the implicit equations is obtained by elimination, giving

$$k_{l,1} = \frac{\tilde{\eta} k^f_{l,1} + \Delta t \mu\left(\tfrac{4}{3} a_x S_x\left(u^*_1 - u^{\mathcal{P}^x}_1(t^*)\right) + a_y S_y\left(u^*_1 - u^{\mathcal{P}^y}_1(t^*)\right)\right)}{\tilde{\eta} - d_l \Delta t \frac{\mu}{\rho^*}\left(\tfrac{4}{3} a_x S_x + a_y S_y\right)}, \tag{96}$$

$$k_{l,2} = \frac{\tilde{\eta} k^f_{l,2} + \Delta t \mu\left(a_x S_x\left(u^*_2 - u^{\mathcal{P}^x}_2(t^*)\right) + \tfrac{4}{3} a_y S_y\left(u^*_2 - u^{\mathcal{P}^y}_2(t^*)\right)\right)}{\tilde{\eta} - d_l \Delta t \frac{\mu}{\rho^*}\left(a_x S_x + \tfrac{4}{3} a_y S_y\right)}, \tag{97}$$

$$k_{l,E} = \frac{\tilde{\eta} k^f_{l,E} + \Delta t\left[u^{**}_1 \tau^{**}_1 + u^{**}_2 \tau^{**}_2 + \lambda(a_x S_x(T^{**} - T^{\mathcal{P}^x}(t^*)) + a_y S_y(T^{**} - T^{\mathcal{P}^y}(t^*)))\right]}{\tilde{\eta} - d_l \Delta t \frac{\lambda(\gamma-1)}{\rho^*}(a_x S_x + a_y S_y)}, \tag{98}$$

where $a_x = \tilde{\eta}\tilde{\eta}_x^{-1}$ and $a_y = \tilde{\eta}\tilde{\eta}_y^{-1}$, and the intermediate values are given by

$$u^*_1 = \frac{m^*_1}{\rho^*}, \quad u^*_2 = \frac{m^*_2}{\rho^*}, \quad u^{**}_1 = \frac{m^*_1 + d_l k_{l,1}}{\rho^*}, \quad u^{**}_2 = \frac{m^*_2 + d_l k_{l,2}}{\rho^*}, \tag{99}$$

$$T^{**} = (\gamma - 1)\left(\frac{E^*}{\rho^*} - \frac{1}{2}\left(u^{**2}_1 + u^{**2}_2\right)\right), \tag{100}$$

$$\tau^{**}_1 = \mu\left(\frac{4}{3} a_x S_x(u^{**}_1 - u^{\mathcal{P}^x}_1(t^*)) + a_y S_y\left(u^{**}_1 - u^{\mathcal{P}^y}_1(t^*)\right)\right), \tag{101}$$

$$\tau^{**}_2 = \mu\left(a_x S_x\left(u^{**}_2 - u^{\mathcal{P}^x}_2(t^*)\right) + \frac{4}{3} a_y S_y\left(u^{**}_2 - u^{\mathcal{P}^y}_2(t^*)\right)\right). \tag{102}$$

In addition, by definition, $|a_x| \leqslant 1$ and $|a_y| = 1$ or $|a_x| = 1$ and $|a_y| \leqslant 1$. Therefore, all the denominators in (96)–(98) are finite and different from zero.

The procedure to obtain the k_l consists of first solving (96) and (97) using the intermediate values, k^f_l and u^*, followed by (98) using the second intermediate values, u^{**} and T^{**}. Because of the particular structure of the Navier–Stokes equations, i.e. no second-order spatial derivatives in the mass conservation equation, the implicit system of equations induced by the semi-implicit time integrator can be solved explicitly.

5. Numerical experiments

Numerical experiments using the proposed method and their results are discussed in this section. Firstly, we utilize a rotating circular cylinder problem to evaluate convergence and conservation properties of the method. This example has been used in previous investigations of incompressible flows [31]. Secondly, we apply the method to problems for compressible flow around various-shape obstacles at different Reynolds numbers and Mach numbers to investigate the effectiveness of the method. Comparisons are carried out between existing incompressible, zero Mach number, simulations around a circular cylinder and compressible simulations using the present method at the low Mach number of 0.05 (based on free-stream velocity or angular velocity at the boundary of the cylinder). Unfortunately, it is well known that the centered finite-difference approximation on a co-located mesh does not converge uniformly to the incompressible solution as the Mach number approaches zero. This is related to the momentum-pressure coupling and it is a well known difficulty of compressible solvers that are not preconditioned for low Mach number flows. This difficulty can be alleviated in several ways, one of which involve the use of filters to remove the spurious pressure and density oscillations that arise on the $2\Delta x$ wavelength, i.e. checkerboard mode, or the use of a staggered arrangement of variables. Since the purpose of the present method is not related to issues of low Mach numbers solvers, this problem was addressed, only for the low Mach number case, by adding an additional stabilizing term to the mass and energy conservation equations of the form $\epsilon\ \Delta^3\rho$ and $\epsilon/(\gamma-1)\Delta^3 p$, respectively, where ϵ is a small number that we set to $0.05(\Delta x\Delta y)^3$. This is done in the spirit of [15] and it is a well known hyperviscosity solution to prevent the appearance of high wavenumber oscillations. It is used in the present context to stabilize pressure oscillations and, at the same time, to keep the amount of numerical dissipation as low as possible. It was verified that the solution did not change appreciably when ϵ was varied around the presently chosen value. Note that no terms were added to the momentum conservation equations, which are naturally regularized by the physical viscosity, and therefore no additional dissipation is added to the kinetic energy of the flow. Furthermore, no regularizing terms were added for the cases where the Mach number was higher than 0.05. Finally, an application of the method to the scattering of a monochromatic sound wave by a cylinder is discussed.

5.1. Rotating cylinder

We investigate the flow of a compressible and viscous fluid surrounding a rotating two-dimensional cylinder of diameter D. In this numerical experiment the fluid domain, Ω, has dimensions $[0, l_x] \times [0, l_y]$ with the cylinder positioned at the center of Ω. The cylinder is allowed to rotate only around its center, without translation. The boundary of the cylinder is represented using 120 linear segments and this discretization is used to identify the irregular points. Initially, both the fluid and cylinder are at rest. At $t = 0$, a constant angular acceleration $\dot{\omega} = V_o/(Rt_c)$ is applied, where V_o is the final (constant) tangential velocity at the cylinder surface and $R = D/2$ denotes the radius of the cylinder. The acceleration lasts one acoustic time, $t_c = l_x/c_o$, where $c_o = \sqrt{\gamma p_o/\rho_o}$ denotes the speed of sound with p_o and ρ_o denoting the reference pressure and density, respectively. After the initial acceleration phase, a constant angular velocity is maintained at $\omega = V_o/R$. Comparisons are made at $t = 10l_x/c_o$. Slip and adiabatic boundary conditions are used at the boundary of Ω. No-slip and isothermal boundary conditions are employed at the cylinder surface. The Reynolds number is defined as $Re = \rho_o V_o D/\mu$ and the Mach number is defined as $M = V_o/c_o$. The Prandtl number was kept constant at 0.7. At low Mach numbers, the leading order solution for the velocity is given by the tangential momentum balance in the Navier–Stokes equations, according to

$$\rho\frac{\partial V_\theta}{\partial t} = \mu\frac{\partial}{\partial r}\left(\frac{1}{r}\frac{\partial}{\partial r}(rV_\theta)\right), \tag{103}$$

with

$$\frac{V_\theta(R,t)}{V_o} = \begin{cases} \frac{t}{t_c} & t < t_c, \\ 1 & t \geqslant t_c. \end{cases} \tag{104}$$

The time dependent solution of (103) can be obtained by Laplace transform and is given by

$$\frac{V_\theta(\eta,\tau)}{V_o} = \frac{1}{\eta} + \frac{i}{\pi\tau_c}\int_0^\infty \frac{(e^{\tau_c\zeta^2}-1)}{\zeta^3}e^{-\tau\zeta^2}\left(\frac{K_1(-i\zeta\eta)}{K_1(-i\zeta)} - \frac{K_1(i\zeta\eta)}{K_1(i\zeta)}\right)d\zeta, \tag{105}$$

where $\eta = r/R, \tau = t\mu/(\rho R^2), \tau_c = 2(l_x/R)(M/Re)$, and K_1 denotes the modified Bessel function of the second kind. The multiplication theorem of Bessel functions [83] shows that the integral in the right-hand side of (105) is purely imaginary, therefore, V_θ is real, as it should be.

Table 1 shows the error convergence from the different cases considered. The Reynolds number based on the cylinder diameter was 10 and 100, respectively. A low Mach number of 0.05 was used to compare the solution with the incompressible result of (105). The length of the computational domain was 10 times the diameter of the cylinder (which was verified to be sufficiently large for the duration of the simulations to have negligible impact on the results). The number of grid points across the diameter of the cylinder is denoted by $N_d = D/\Delta x$, with $\Delta x = l_x/N_x$ and $\Delta y = l_y/N_y$ where N_x, N_y denotes the total number of grid points across Ω in the respective directions. In all cases considered here $\Delta x = \Delta y$. The time step size was determined using the explicit CFL and viscous conditions, independently of η_x and η_y, such that

$$\Delta t = \min\left(\frac{\mathrm{CFL}}{\frac{(|u|+c)_{\max}}{\Delta x} + \frac{(|v|+c)_{\max}}{\Delta y}}, \frac{1}{2}\frac{DN}{v_{\max}(\Delta x^{-2} + \Delta y^{-2})}\right), \tag{106}$$

where $c = \sqrt{\gamma p/\rho}$ denotes the local speed of sound, $v = \mu/\rho$ is the kinematic viscosity, and the constants CFL and DN were taken to be 0.8 and 0.7, respectively. The error norms are computed according to

$$\|e_\theta\|_\infty = \max_{\mathcal{I}^+}\left|V_\theta^N(x_i,y_j) - V_\theta(x_i,y_j)\right|, \tag{107}$$

$$\|e_\theta\|_1 = \frac{1}{N_{\mathcal{I}^+}}\sum_{\mathcal{I}^+}\left|V_\theta^N(x_i,y_j) - V_\theta(x_i,y_j)\right|, \tag{108}$$

$$\|e_\theta\|_2 = \left(\frac{1}{N_{\mathcal{I}^+}}\sum_{\mathcal{I}^+}\left|V_\theta^N(x_i,y_j) - V_\theta(x_i,y_j)\right|^2\right)^{1/2}, \tag{109}$$

where V_θ^N denotes the tangential velocity obtained from the numerical solution, $\mathcal{I}^+ = \{i,j \in [1,N_x]\times[1,N_y], x_i = \Delta x(i-0.5), y_j = \Delta y(j-0.5) : (x_i,y_j) \in \Omega^+\}$ and $N_{\mathcal{I}^+} = \sum_{\mathcal{I}^+}1$. The convergence rates, p_∞, p_1 and p_2, are deduced from the respective rates of convergence of the errors at two consecutive resolutions, according to

$$p = \frac{\log(\|e_\theta\|^h/\|e_\theta\|^{h/2})}{\log 2}, \tag{110}$$

Table 1
Error norms and convergence rate of the tangential velocity for different Reynolds numbers at the end of the simulations.

Re	$N_x \times N_y$	N_d	$\|e_\theta\|_\infty$	p_∞	$\|e_\theta\|_1$	p_1	$\|e_\theta\|_2$	p_2
10	100 × 100	10	1.21×10^{-2}		1.02×10^{-4}		5.16×10^{-4}	
10	200 × 200	20	3.71×10^{-3}	1.70	2.60×10^{-5}	1.97	1.29×10^{-4}	2.00
10	400 × 400	40	1.20×10^{-3}	1.62	9.72×10^{-6}	1.42	4.28×10^{-5}	1.59
100	100 × 100	10	1.92×10^{-2}		1.42×10^{-4}		8.46×10^{-4}	
100	200 × 200	20	7.13×10^{-3}	1.43	3.29×10^{-5}	2.11	2.10×10^{-4}	2.01
100	400 × 400	40	2.01×10^{-3}	1.82	1.06×10^{-5}	1.63	7.06×10^{-5}	1.57

where $h = \Delta x = \Delta y$. Table 1 shows that the tangential velocity converges at a rate between 1.5 and 2, which is close to the expected second-order. Fig. 4 shows overlaid tangential velocity isolevels for three different resolutions as well as the analytical solution. The profiles appear to converge smoothly.

Conservation properties of the method are investigated by considering the evolution of the total mass and linear momentum, defined as

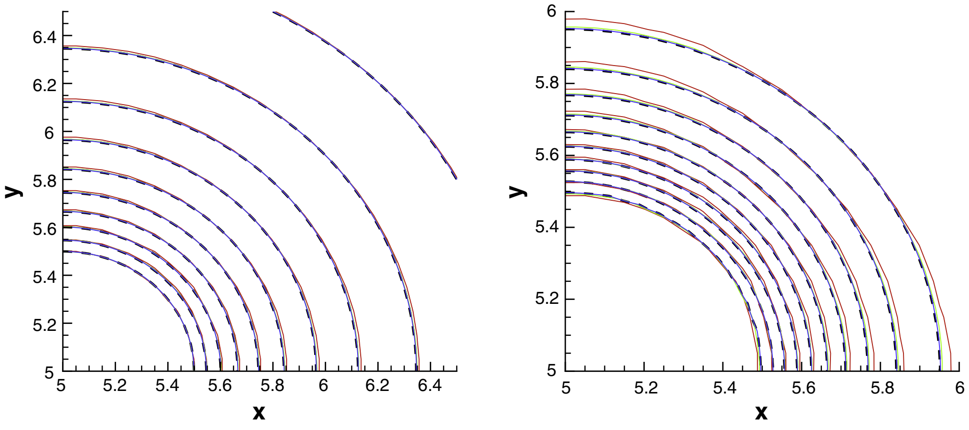
$$\langle \rho \rangle = \frac{1}{N_x N_y} \sum_{\mathcal{I}} \rho(x_i, y_j), \tag{111}$$

and

$$\langle m_1 \rangle = \frac{1}{N_x N_y} \sum_{\mathcal{I}} m_1(x_i, y_j). \tag{112}$$

Notice that these definitions include the interior of the cylinder as well (which moves as a solid body and has constant density). Table 2 shows the total percentage change of mass and linear momentum at the end of the simulations for all Reynolds numbers and resolutions investigated. It is observed that linear momentum is conserved to machine accuracy, as expected owing to the symmetry of the problem, while mass increases very slowly. These results were obtained at the end of the simulations, as previously defined, and correspond to a relatively large integration time. For a fixed Reynolds number, the total mass drift decreases with increased resolution. The smallest mass drifts are observed for the lowest Reynolds number cases. Note that the mesh sizes are the same in both sets of simulations. The total mass variation with time is affected by the irregular shape of the interface with respect to the Cartesian grid. Moreover, since there is no explicit boundary condition for the density (only for the temperature), some variations of the total mass is necessarily observed as the simulation progresses, i.e. (71) and (72) and discussion associated with these equations. An ideal method should not exhibit any variation of the total mass in the domain provided the volume over which the mass should be conserved is well defined. In practice, the small variations observed here appear to be inconsequential in all simulations considered so far.

An additional criteria to consider for the present method is the behavior of the solution as a function of the distance between the irregular points and the interface. The distribution of irregular points with respect to the cylinder is shown in Fig. 5, where η (the fraction of a grid spacing where the interface intersects one of the Cartesian lines passing through an

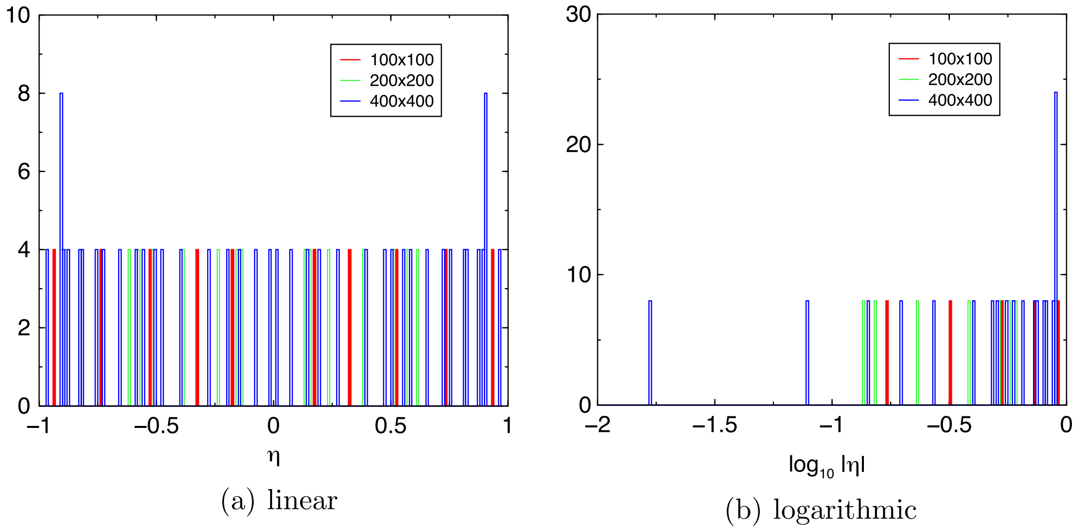


(a) $Re = 10$ (b) $Re = 100$

Fig. 4. Isolines of tangential velocity magnitude for two Reynolds numbers at the end of the simulations in one quadrant of the simulation domain. Thick broken line denotes analytical solution and thin red, green and blue lines denote results of simulations with 100 × 100, 200 × 200 and 400 × 400 grid points, respectively. In all cases, equally spaced isolines of 10% V_o are used, the center of the domain is at (5,5) and the cylinder radius is 0.5. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 2
Percentage change of total mass and total linear momentum at the end of the simulations.

Re	$N_x \times N_y$	$\%\langle\rho\rangle$	$\langle m_1 \rangle$
10	100 × 100	0.0062	-1.54×10^{-17}
10	200 × 200	0.0014	-2.31×10^{-17}
10	400 × 400	0.0005	-2.39×10^{-17}
100	100 × 100	0.0358	-2.18×10^{-17}
100	200 × 200	0.0055	-8.47×10^{-18}
100	400 × 400	0.0010	-1.91×10^{-17}



(a) linear

(b) logarithmic

Fig. 5. Histograms of $\eta = \eta_x$ or η_y (undistinguished) for all three grid arrangements. Linear (a) and logarithmic (b) representations.

Table 3
Convergence and mass conservation results of rotating cylinder at $Re = 50$ at $t = 10 l_x/c_o$ for varying $|\eta|_{\min}$. Time step criteria is given by (106), independent of η.

$\lvert\eta\rvert_{\min}$	$\lVert e_\theta \rVert_\infty$	$\lVert e_\theta \rVert_1$	$\lVert e_\theta \rVert_2$	$\%\langle\rho\rangle$
0.1357	5.47×10^{-3}	2.66×10^{-5}	1.67×10^{-4}	3.02×10^{-3}
10^{-3}	4.70×10^{-3}	2.23×10^{-5}	1.34×10^{-4}	1.86×10^{-3}
10^{-4}	4.70×10^{-3}	2.23×10^{-5}	1.34×10^{-4}	1.86×10^{-3}
10^{-5}	4.70×10^{-3}	2.23×10^{-5}	1.34×10^{-4}	1.86×10^{-3}
10^{-6}	4.70×10^{-3}	2.23×10^{-5}	1.34×10^{-4}	1.86×10^{-3}

irregular point) denotes either η_x or η_y, no distinction is made between the two here. As can be seen, the values of η are approximately uniformly distributed between −1 and 1 for all cases. Fig. 5(b) shows the histogram of the logarithm (base 10) of $|\eta|$. This helps to visualize the tails of the distribution of values of $|\eta|$, in particular the small values of $|\eta|$ that are responsible for the stiffness of the immersed interface equations. The minimum value observed, $|\eta|_{\min} = 0.0167$, appears in the simulation with the 400 × 400 grid. While this is a small value, a more controlled experiment was designed where one can explore the behavior of the method for even smaller values of η. Therefore, a new set of simulations at $Re = 50$ with a mesh of 200 × 200 grid points is used to systematically vary $|\eta|_{\min}$. This is accomplished by shifting the center of the cylinder such that the rightmost boundary sits at a predefined distance from the Cartesian mesh.

Table 3 shows the convergence rates and conservation properties as $|\eta|_{\min}$ varies from 0.1357 to 10^{-6}. The time steps used in these simulations was still controlled by (106) and it was independent of $|\eta|_{\min}$. The results indicate that no degradation of the convergence or conservation is observed.

The previous examples were carried out at a low Mach number in order to compare the analytical velocity profile (105), with the numerical results. Similar results are now described for higher Mach numbers to highlight that the simulation does not lose stability with increasing Mach number. Fig. 6 shows the variations of the tangential velocity and temperature with increasing Mach number at $Re = 50$ and all remaining parameters unchanged with respect to the previous examples.

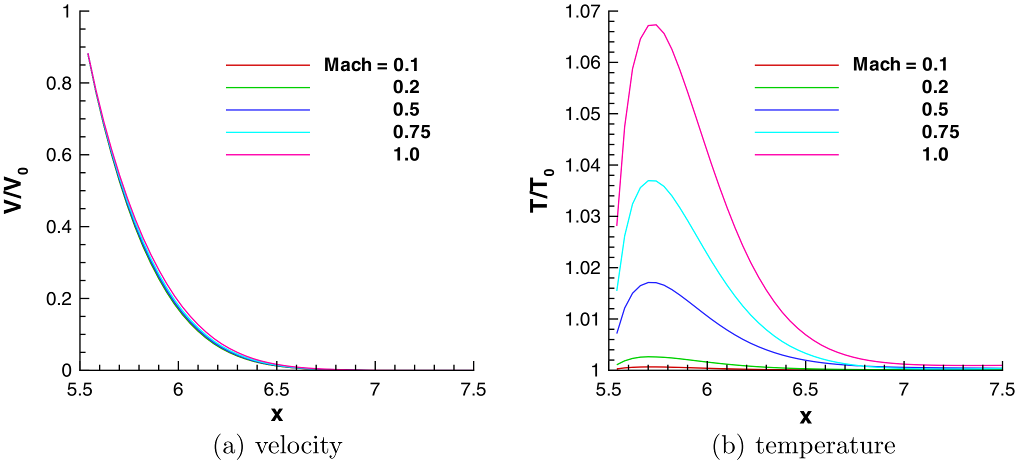
Finally, we consider a case where the temperature of the cylinder is not constant. The model problem has a cylinder temperature that varies as a function of the cylinder-attached angular coordinate φ, in the rotating frame, given by

$$T_{bc}(\varphi) = T_o(1 + 0.1\sin(\varphi)). \tag{113}$$

In this case, since an analytical solution is not available, the simulation only highlights that the case where the temperature is not uniform can be carried out without modifications. Fig. 7 show temperature isolevels at four different instants of time. As can be seen, the temperature of the flow around the cylinder is rotating and diffusing away from the surface of the cylinder as time progresses.

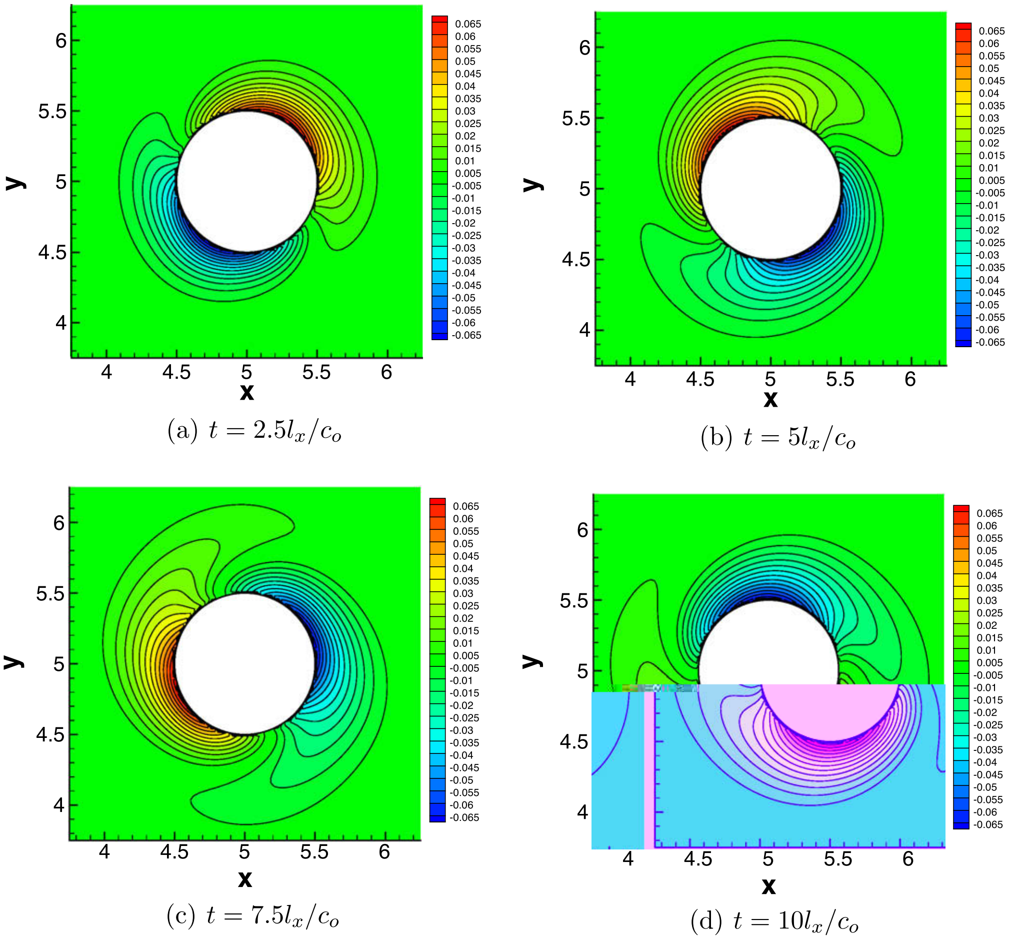
5.2. Obstacles in uniform flow

In this section we detail several simulations of rigid obstacles with simple and complex geometries immersed in an upstream uniform flow. Firstly, we consider flow over a circular cylinder and the lift and drag forces are determined at different



(a) velocity (b) temperature

Fig. 6. Effect of Mach number on normalized velocity (a) and temperature (b) profiles for $Re = 50$, using a mesh with 200 × 200 grid points at $t = 10l_x/c_o$.



(a) $t = 2.5l_x/c_o$

(b) $t = 5l_x/c_o$

(c) $t = 7.5l_x/c_o$

(d) $t = 10l_x/c_o$

Fig. 7. Isocontours of temperature fluctuation, $T - T_o$, for $Re = 50, M = 0.05$ using a mesh with 400 × 400 grid points.

Reynolds numbers. These results are compared with available experimental and computational data. Secondly, examples of systems of two and three cylinders of different shapes are discussed. In these simulations, uniform flow is imposed at the left-hand side boundary of the domain and non-reflective characteristic boundary conditions are used at the right-hand side boundary [84], implemented as described in [80]. Top and bottom boundaries utilize characteristic outgoing boundary conditions.

Fig. 8 shows vortex shedding off a circular cylinder placed in a uniform stream with $Re = \rho_\infty U_\infty D/\mu = 100$ and $M = U_\infty/c = 0.05$, where U_∞ is the upstream uniform flow velocity, c is the speed of sound, ρ_∞ is the upstream density and D the diameter of the cylinder. The domain size is $25D \times 20D$ and the cylinder is located at $x = 10D$. A convergence study was performed to investigate convergence of the Strouhal number, $St = fD/U_\infty$, the average drag coefficient, $C_D = 2\overline{F_x}/(D\rho_\infty U_\infty^2)$, and peak lift coefficient, $C_L = 2F_{y,\max}/(D\rho_\infty U_\infty^2)$, with increasing resolution. Averaging is denoted by an

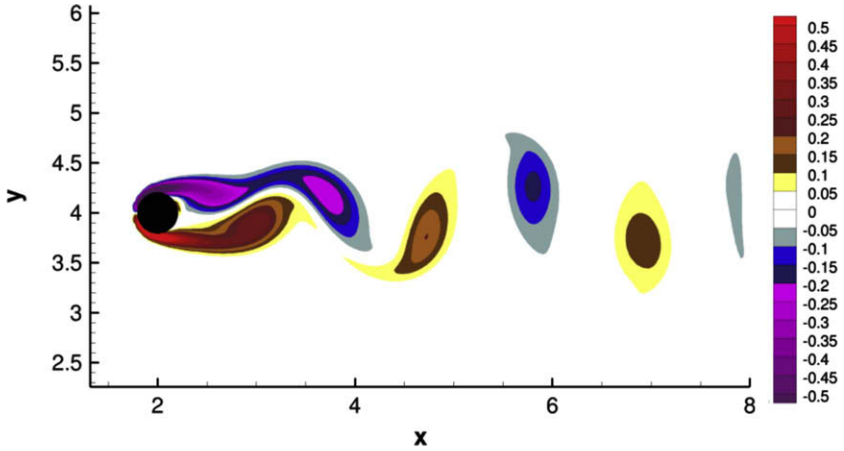
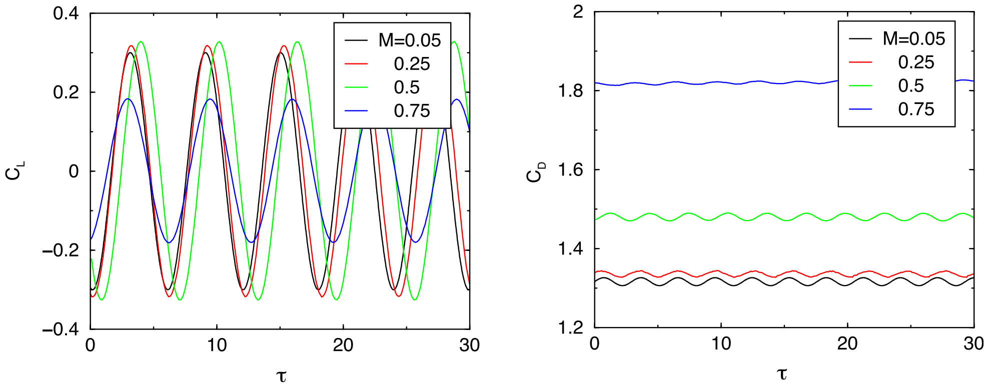


Fig. 8. Vorticity at one instant in time on a cylinder in uniform flow at $Re = 100$ and $M = 0.05$.

Table 4
Convergence of vortex shedding behind a cylinder at $Re = 100$.

	M	$\frac{D}{\Delta x}$	St	C_D	C_L
Liu et al. [86]	0		0.164	1.35	0.339
Choi et al. [32]	0	81	0.164	1.34	0.315
Kim et al. [89]	0	30	0.165	1.33	0.32
Russell and Wang [90]	0	20	0.169	1.38	0.322
Present	0.05	15	0.168	1.247	0.264
	0.05	30	0.168	1.316	0.320
	0.05	60	0.168	1.317	0.320
	0.25	15	0.165	1.275	0.272
	0.25	30	0.164	1.336	0.304
	0.25	60	0.168	1.336	0.319

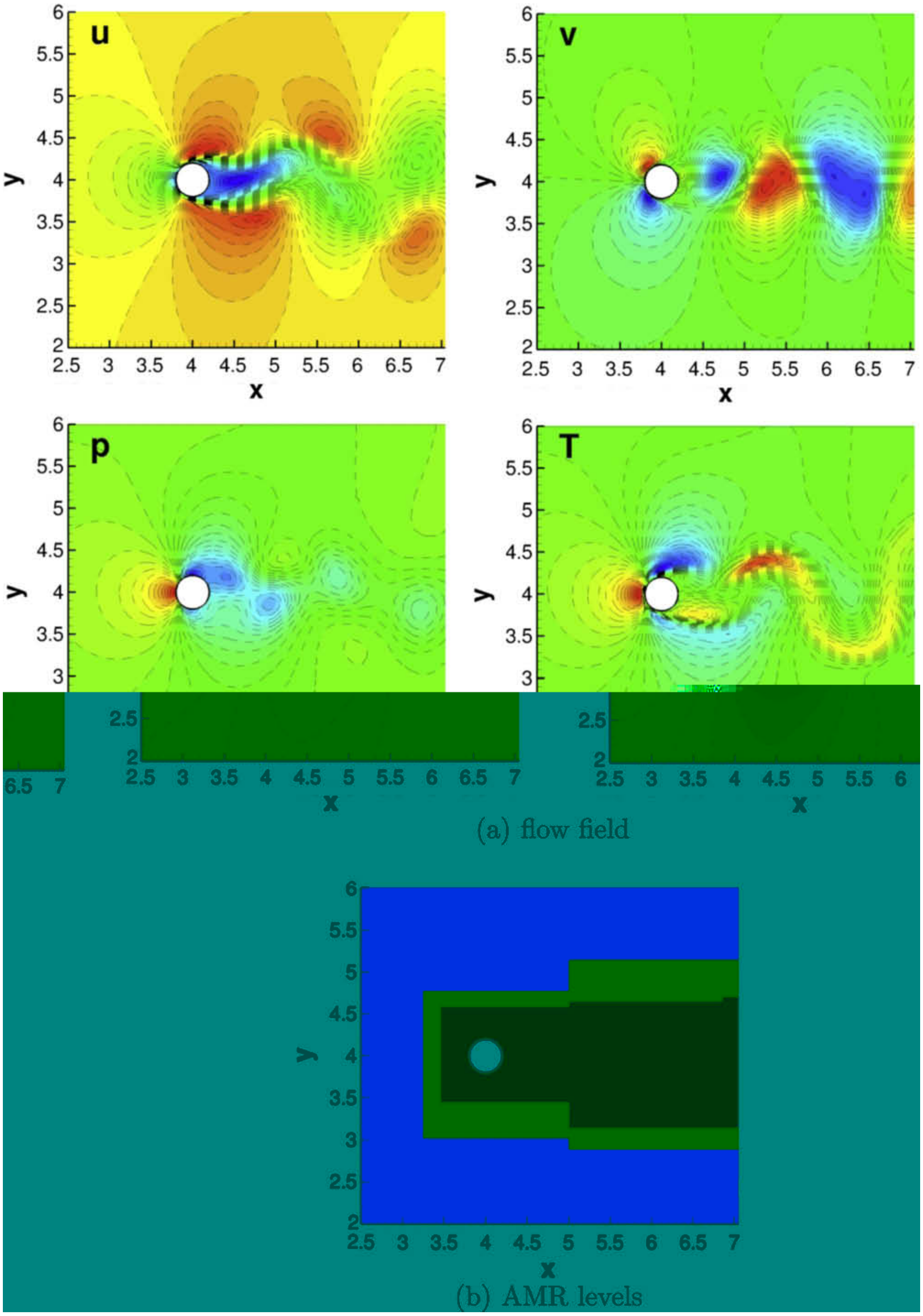


(a) lift (b) drag

Fig. 9. Normalized lift and drag coefficients as a function of normalized time, $\tau = tU_\infty/D$, for a cylinder in uniform flow at $Re = 100$ for increasing Mach numbers.

overbar in the definition of the streamwise-aligned force appearing in C_D. Table 4 shows that the Strouhal number for the low Mach number cases are in good agreement with previous experimental and numerical results of 0.165 [85,86]. The values reported by other IBM methods for this flow are also very similar to those obtained here.

Having established the accuracy of the solution we investigate the performance of the method as the free-stream Mach number is increased from 0.05 (practically incompressible) to 0.75. The latter is close to the maximum Mach number that can be reached before supersonic flow develops in the domain. Fig. 9 shows lift and drag time histories for a cylinder in a uniform stream at $Re = 100$ and Mach number between 0.05 and 0.75.



(a) flow field

(b) AMR levels

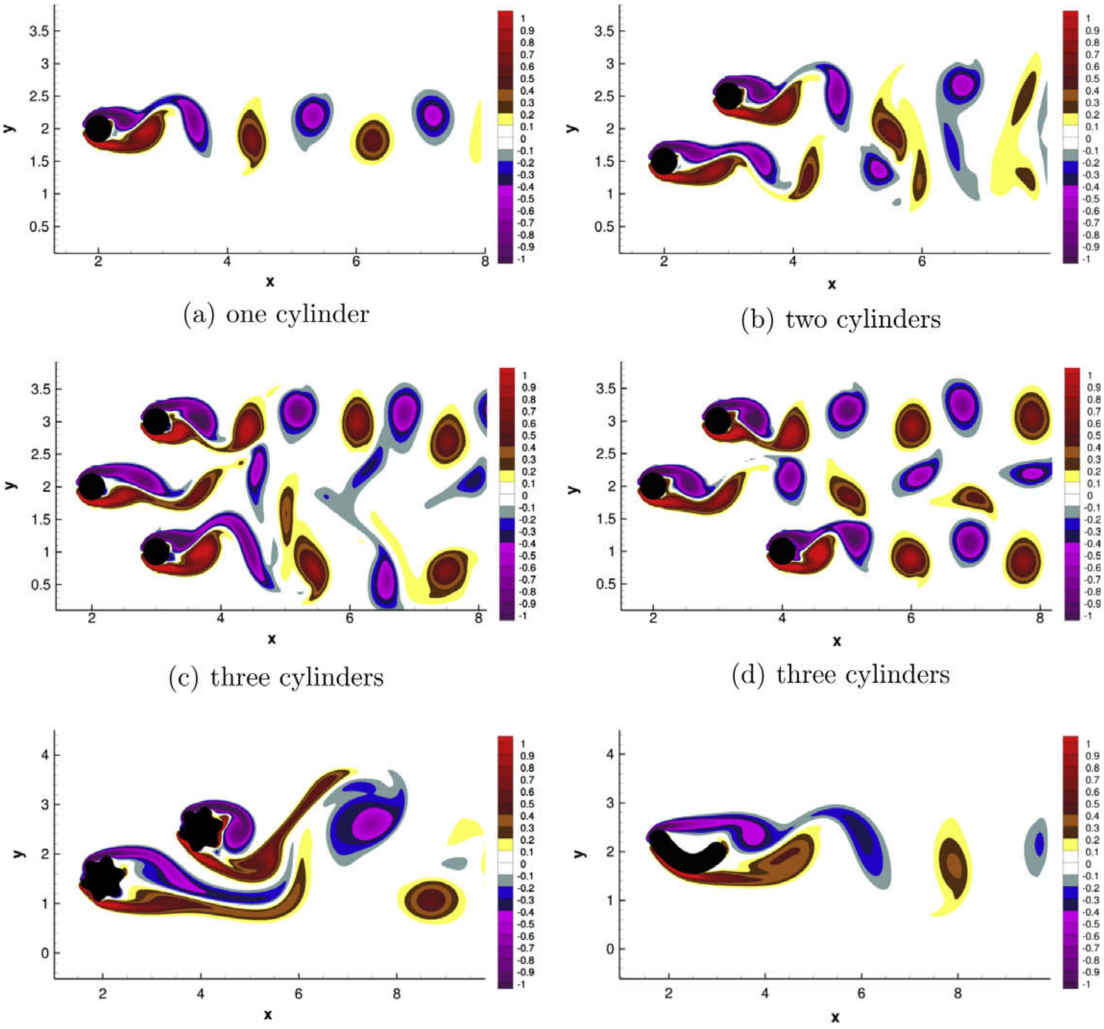
Fig. 10. Velocity, pressure, temperature and AMR levels at one instant in time for a cylinder in uniform flow at $Re = 100$ and $M = 0.5$. Blue, green and red denote coarse, intermediate and fine meshes, respectively, in subfigure (b). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

In order to conduct more interesting simulations, the method was implemented in a low numerical dissipation adaptive mesh refinement (AMR) parallel solver [80]. No particular modifications to the core numerics or the hierarchical mesh used in [80] was required, apart from the change in the time integration method. This change from the strong-stability preserving Runge–Kutta [87] used previously to the present additive semi-implicit Runge–Kutta does require two additional storage arrays to maintain k_l which is a small additional cost for the capability of incorporating arbitrarily shaped objects in the domain. Fig. 10(a) shows an instant in time of velocity, pressure and temperature for a cylinder at $Re = 100$ and $M = 0.5$. This simulations utilized a total of 3 levels of refinement with a coarse mesh using $D/\Delta x = 15$ and successive refinements of factors of 2 between levels. Fig. 10(b) shows the corresponding regions of refinement around the object and wake of the cylinder at this time. The refinement criteria is based on the requirement that the normalized velocity difference, $|\Delta u_k|/U_\infty$, between grid cells should not exceed 0.03. If this condition was not fulfilled at a particular location an additional fine level is created. Isolevels of the different quantities have been marked with broken lines.

Finally, simulations with multiple objects were carried out to test the flexibility of the method in more complex scenarios. These simulations correspond to Reynolds number of 200, based on the obstacle's mean diameter, at the low Mach number of 0.05. Fig. 11 shows an instantaneous rendering of the vorticity being shed from the objects using a total of three levels of refinement and equivalent AMR parameters as those of the simulation shown in Fig. 10.

5.3. Viscous scattering of sound

The final example corresponds to the scattering of a monochromatic sound wave in a viscous fluid by a circular cylinder. This is a classical problem in acoustics (in the inviscid limit) that is modeled presently by combining the embedded geometry



(a) one cylinder

(b) two cylinders

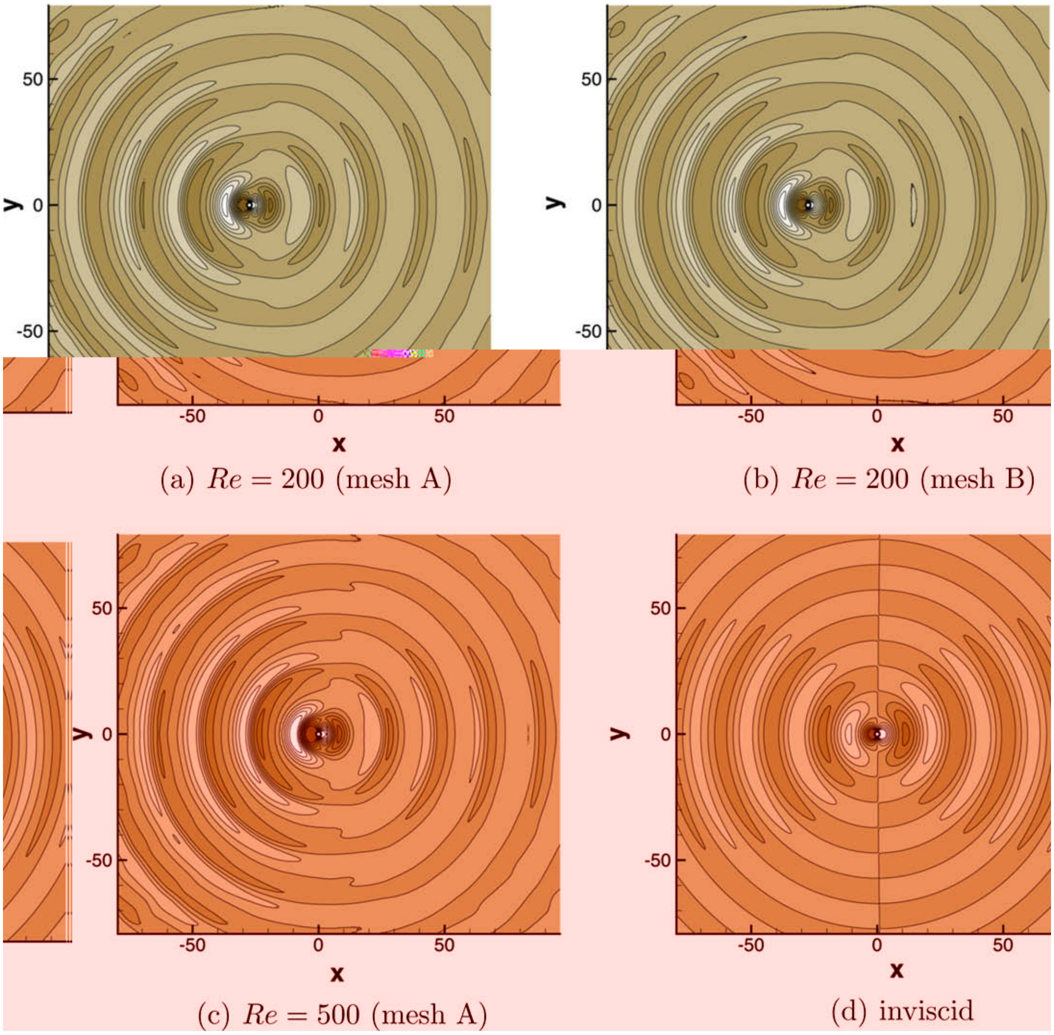
(c) three cylinders

(d) three cylinders

(e) two stars

(f) thickened parabolic section

Fig. 11. Vorticity shed from different objects in a uniform stream at $Re = 200$ and Mach 0.05.



(a) $Re = 200$ (mesh A)

(b) $Re = 200$ (mesh B)

(c) $Re = 500$ (mesh A)

(d) inviscid

Fig. 12. Scattered pressure wave at $Re = 200$ (a) and (b), 500 (c) and inviscid (d).

representation and AMR to achieve as high Reynolds number as practically possible. This is necessary since the proposed numerical method is not designed to handle inviscid (slip and adiabatic) boundary conditions, which are of mixed type as opposed to the Dirichlet type used in the present approach. We remark that low-order methods are usually unsuitable for simulation of acoustic problems due to their poor resolution and dispersion properties. In this example we have used a large number of grid points to compensate for this deficiency. Moreover, the example below poses demanding tests on the accuracy of far field boundary conditions.

A monochromatic wave of wavelength $\lambda = 20a$, where a denotes the radius of the cylinder, traveling from left to right is simulated in a domain of size $[-240\ a, 240\ a] \times [-80\ a, 80\ a]$ with the cylinder in the center. The Reynolds numbers considered are $Re = \rho_o 2ac_o/\mu = 200$, and 500, where c_o denotes the speed of sound and ρ_o the background density. Two coarse meshes were considered: mesh A with 1536×512 grid points and mesh B with 3072×1024 grid points. This corresponds to approximately 62 and 123 grid points per wavelength, respectively. The apparently large number of grid points is necessary to minimize the relatively poor-dispersion properties of the second-order accurate method. Furthermore, two and three levels of AMR with refinement factors of two were used in the simulations at Reynolds number 200 and 500, respectively, to fully resolve the flow around the cylinder. The relative amplitude of the injected sinusoidal pressure wave is $\Delta p/p_o = 0.014$, where $\Delta p = p_{\max} - p_{\min}$ and p_o denotes the reference pressure. The sound (pressure) scattering was determined by subtracting the pressure from each simulation from a baseline simulation with identical parameters that omits the cylinder. This helps in the comparisons because it removes extraneous effects (although relatively small with respect to the mean pressure) induced by the approximate one-dimensional characteristic boundary conditions used in the simulations of the sound wave in a finite domain. In this manner, the sound scattering comparisons obtained as the difference between these simulations, with and without the cylinder, cancels some of the perturbations introduced by the numerical boundary conditions

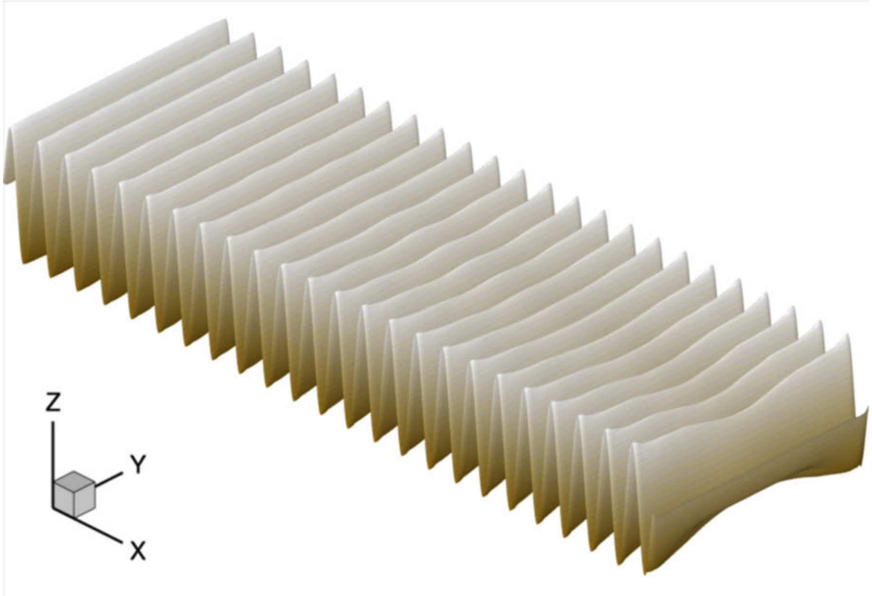


Fig. 13. Pressure wave at $Re = 200$ without cylinder showing the effect of boundary conditions for coarse resolution simulation at final time.

and the finite viscosity, which adds a small amount of physical dissipation as the sound wave travels through the large domain. Finally, the comparisons are carried out at $t = 1000a/c_o$, to allow for acoustic waves generated during the initial transient to exit the domain completely.

Fig. 12 shows comparisons of the scattered pressure wave for different resolutions and Reynolds numbers at the end of the simulation time. The isolevels are identical in all subfigures. Fig. 12(a) and (b) explore, for $Re = 200$, the effect of resolution. Minor effects of resolution are observed in the figures, which only appear exacerbated if one focuses on the location of some isolevel curves, but overall the difference between the figures is small. Fig. 12(a) and (c), at $Re = 200$ with two additional levels of refinement and 500 with three additional levels of refinement, respectively, use the same coarse resolution. These figures exhibit some minor differences in the wave pattern within this range of Reynolds numbers, suggesting that the solution should be close to inviscid. Finally, Fig. 12(d) shows the ideal inviscid solution obtained by linearization of the Euler equations, at the same time as that shown from the simulations. The general solution, valid for any wavelength, is given by an infinite series of Bessel functions and it is described in detail in [88]. It is observed that the crests and troughs of the simulated wave are qualitatively in the right location, in particular along the horizontal line intersecting the center of the cylinder. Nevertheless, the viscous solutions are not able to reproduce the observed symmetry of the ideal scattered wave. Notably, the perturbation ahead of the cylinder occupies a larger region in the simulations in comparison with the region behind the cylinder. Comparison between Fig. 12(a) and (c) does not indicate that viscous effects may play a role in the discrepancies between the simulation results and the inviscid exact solution. It appears that our approximate characteristic boundary conditions play a more important role in the results. This is illustrated in Fig. 13 showing the pressure (total) wave in the domain without the cylinder at the final instant. In this figure, it is clearly observed that the pressure wave is nonuniformly attenuated as it progresses through the domain. This is caused by accumulations of small reflections out of the corners of the computational domain that eventually pollute some of the simulation. The overall error is not large since $\Delta p/p_o$ is very small, but the scattered sound is also small and the errors are easily observed when subtracting the incident wave. Note that it is of course possible to simulate the planar wave almost perfectly by using zero-gradient boundary conditions in the vertical direction when the cylinder is not present. Unfortunately, this shortcut is not useful when the cylinder is present since the scattered sound produces small, but not negligible, inflows and outflows through the boundary and the zero-gradient condition is more reflective than our approximate characteristic boundary conditions.

In conclusion, the present method would have to be interfaced with an appropriate, if possible using high-order of accuracy, linearized Helmholtz or wave-equation solver in order to recover accurately and efficiently the scattered sound intensity. Unfortunately, this was not the objective of the present study.

6. Conclusions

A new immersed interface method for the simulation of flows governed by the compressible Navier–Stokes equations is proposed. The approach uses low numerical dissipation centered finite-differences and is not restricted by stiffness problems produced when the embedded geometry location is too close to the Cartesian grid. The proposed method combines first-order spatial derivative operators, derived using the summation-by-parts theory of stable derivative approximations, with unequally spaced finite-differences for the second-order derivatives and least-squares projection for the mixed derivatives. Stiffness is removed by use of an additive semi-implicit Runge–Kutta method whose solution is explicitly obtained, resulting in an explicit method. The new low numerical dissipation model introduces desirable properties since artificial dissipation has been removed and now complex geometrical objects can be easily treated in compressible flow simulations. Moreover, the method was integrated into an existing adaptive mesh refinement framework to allow for high resolution capability in parallel multiprocessor environments. Verification for convergence, conservation and stability of the method was investigated for a benchmark problem of a rotating cylinder in quiescent fluid and additional comparisons with experimental and computational results available in the literature are provided for the case of a circular cylinder in a uniform stream. Examples of compressible flows for Mach numbers up to 0.75 were carried out to assess the flexibility and robustness of the new method. Furthermore, a sound scattering problem was investigated to expose the limits of applicability of the method in acoustics applications.

Acknowledgments

This work was supported in part by DOE/PECASE award LLNL B574743 and by NASA NRA award NNX08AL31A (University of Minnesota is prime). The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the DOE, NASA or the US Government.

References

[1] M. Hyman, Non-iterative numerical solution of boundary value problems, Appl. Sci. Res. Sec. B2 (1952) 325–351.

[2] V. Saul'ev, A method for automatization of the solution of boundary value problems on high performance computers, Dokl. Akad. Nauk. SSSR (in Russian) 144 (1962) 497–500.

[3] R. LeVeque, Z. Li, The immersed interface method for elliptic equations with discontinuous coefficients and singular sources, SIAM J. Numer. Anal. 31 (4) (1994) 1019–1044.

[4] Z. Li, The immersed interface method: a numerical approach to partial differential equations with interfaces, Ph.D. Thesis, Department of Applied Mathematics, University of Washington, Seattle, WA, 1994.

[5] Z. Li, An overview of the immersed interface method and its applications, Taiwanese J. Math. 7 (1) (2003) 1–49.

[6] P. Berthelsen, A decomposed immersed interface method for variable coefficient elliptic equations with non-smooth and discontinuous solutions, J. Comput. Phys. 197 (1) (2004) 364–386.

[7] K. Ito, Z. Li, Y. Kyei, Higher-order, Cartesian grid based finite difference schemes for elliptic equations on irregular domains, SIAM J. Sci. Comput. 27 (1) (2005) 346–367.

[8] Y. Zhou, G. Wei, On the fictitious-domain and interpolant formulations of the matched interface and boundary (MIB) method, J. Comput. Phys. 219 (1) (2006) 228–246.

[9] S. Yu, Y. Zhou, G. Wei, Matched interface and boundary (MIB) method for elliptic problems with sharp-edged interfaces, J. Comput. Phys. 224 (2) (2007) 729–756.

[10] X. Zhong, A new high-order immersed interface method for solving elliptic equations with imbedded interface of discontinuity, J. Comput. Phys. 225 (1) (2007) 1066–1099.

[11] I.-L. Chern, Y.-C. Shu, A coupling interface method for elliptic interface problems, J. Comput. Phys. 225 (2) (2007) 2138–2174.

[12] S. Yu, G. Wei, Three-dimensional matched interface and boundary (MIB) method for treating geometric singularities, J. Comput. Phys. 227 (1) (2007) 602–632.

[13] R. Glowinski, Y. Kuznetsov, On the solution of the Dirichlet problem for linear elliptic operators by a distributed Lagrange multiplier method, C.R. Acad. Sci. Paris 327 (1998) 693–698.

[14] R. Glowinski, Y. Kuznetsov, Distributed Lagrange multipliers based on fictitious domain method for second order elliptic problems, Comput. Meth. Appl. Mech. Engrg. 196 (8) (2007) 1498–1506.

[15] H.-O. Kreiss, A. Petersson, A second order accurate embedded boundary method for the wave equation with Dirichlet data, SIAM J. Sci. Comput. 27 (4) (2006) 1141–1167.

[16] J. Kandilarov, L. Vulkov, The immersed interface method for two-dimensional heat-diffusion equations with singular own sources, Appl. Numer. Math. 57 (5) (2007) 486–497.

[17] C. Peskin, Flow patterns around heart valves: a numerical method, J. Comput. Phys. 10 (2) (1972) 252–271.

[18] C. Peskin, Numerical analysis of blood flow in the heart, J. Comput. Phys. 25 (3) (1977) 220–252.

[19] C. Peskin, D. McQueen, A three-dimensional computational method for blood flow in the heart I. Immersed elastic fibers in a viscous incompressible fluid, J. Comput. Phys. 81 (2) (1989) 372–405.

[20] M. Lai, C. Peskin, An immersed boundary method with formal second-order accuracy and reduced numerical viscosity, J. Comput. Phys. 160 (2) (2000) 705–719.

[21] C. Peskin, The immersed boundary method, Acta Numer. 1 (2002) 1–39.

[22] R. LeVeque, Z. Li, Immersed interface methods for Stokes flow with elastic boundaries or surface tension, SIAM J. Sci. Comput. 18 (3) (1997) 709–735.

[23] L. Lee, R. LeVeque, An immersed interface method for incompressible Navier–Stokes equations, SIAM J. Sci. Comput. 25 (3) (2003) 832–856.

[24] R. Glowinski, T. Pan, T. Hesla, D. Joseph, J. Périaux, A fictitious domain approach to the direct numerical simulation of incompressible viscous flow past moving rigid bodies: application to particulate flow, J. Comput. Phys. 169 (2) (2001) 363–426.

[25] R. Fedkiw, T. Aslam, B. Merriman, S. Osher, A non-oscillatory Eulerian approach to interfaces in multimaterial flows (the ghost fluid method), J. Comput. Phys. 152 (2) (1999) 457–492.

[26] Z. Li, M.-C. Lai, The immersed interface method for the Navier–Stokes equations with singular forces, J. Comput. Phys. 171 (2) (2001) 822.

[27] A. Gilmanov, F. Sotiropoulos, E. Balaras, A general reconstruction algorithm for simulating flows with complex 3D immersed boundaries on Cartesian grids, J. Comput. Phys. 191 (2) (2003) 660–669.

[28] R. Mittal, G. Iaccarino, Immersed boundary methods, Annu. Rev. Fluid Mech. 37 (2005) 239–261.

[29] M. Linnick, H. Fasel, A high-order immersed interface method for simulating unsteady incompressible flows on irregular domains, J. Comput. Phys. 204 (1) (2005) 157–192.

[30] S. Xu, Z. Wang, An immersed interface method for simulating the interaction of a fluid with moving boundaries, J. Comput. Phys. 216 (2) (2006) 454–493.

[31] D. Le, B. Khoo, J. Peraire, An immersed interface method for viscous incompressible flows involving rigid and flexible boundaries, J. Comput. Phys. 220 (1) (2006) 109–138.

[32] J.-I. Choi, R. Oberoi, J. Edwards, J. Rosati, An immersed boundary method for complex incompressible flows, J. Comput. Phys. 224 (2) (2007) 757–784.

[33] K. Taira, T. Colonius, The immersed boundary method: a projection approach, J. Comput. Phys. 225 (2) (2007) 2118–2137.

[34] T. Ikeno, T. Kajishima, Finite-difference immersed boundary method consistent with wall conditions for incompressible turbulent flow simulations, J. Comput. Phys. 226 (2) (2007) 1485–1508.

[35] L. Ge, F. Sotiropoulos, A numerical method for solving the 3D unsteady incompressible Navier–Stokes equations in curvilinear domains with complex immersed boundaries, J. Comput. Phys. 225 (2) (2007) 1782–1809.

[36] S. Xu, Z. Wang, A 3D immersed interface method for fluid–solid interaction, Comput. Meth. Appl. Mech. Engrg. 197 (25) (2008) 2068–2086.

[37] V. Rutka, Z. Li, An explicit jump immersed interface method for two-phase Navier–Stokes equations with interfaces, Comput. Meth. Appl. Mech. Engrg. 197 (25) (2008) 2317–2328.

[38] J. Piraux, B. Lombard, A new interface method for hyperbolic problems with discontinuous coefficients: one-dimensional acoustic example, J. Comput. Phys. 168 (1) (2001) 227–248.

[39] A. Dadone, B. Grossman, Ghost-cell method for inviscid two-dimensional flows on Cartesian grids, AIAA J. 42 (12) (2004) 2499–2507.

[40] E. Koh, H. Tsai, F. Liu, Euler solution using Cartesian grid with a gridless least-squares boundary treatment, AIAA J. 43 (2) (2005) 246–255.

[41] B. Lombard, R. Donat, The explicit simplified interface method for compressible multicomponent flows, SIAM J. Sci. Comput. 27 (1) (2005) 208–230.

[42] X. Lv, Y. Zhao, X. Huang, G. Xia, Z. Wang, An efficient parallel/unstructured-multigrid preconditioned implicit method for simulating 3D unsteady compressible flows with moving objects, J. Comput. Phys. 215 (2) (2006) 661–690.

[43] X. Hu, B. Khoo, N. Adams, F. Huang, A conservative interface method for compressible flows, J. Comput. Phys. 219 (2) (2006) 553–578.

[44] T. Liu, B. Khoo, The accuracy of the modified ghost fluid method for gas–gas Riemann problem, Appl. Numer. Math. 57 (5) (2007) 721–733.

[45] R. Ghias, R. Mittal, H. Dong, A sharp interface immersed boundary method for compressible viscous flows, J. Comput. Phys. 225 (1) (2007) 528–553.

[46] M. de Tullio, P.D. Palma, G. Iaccarino, G. Pascazio, M. Napolitano, An immersed boundary method for compressible flows using local grid refinement, J. Comput. Phys. 225 (2) (2007) 2098–2117.

[47] Z. Li, A fast iterative algorithm for elliptic interface problems, SIAM J. Numer. Anal. 35 (1998) 230–254.

[48] A. Wiegmann, K. Bube, The explicit-jump immersed interface method: finite difference methods for PDEs with piecewise smooth solutions, SIAM J. Numer. Anal. 37 (3) (2000) 827–862.

[49] M.-C. Lai, Z. Li, A remark on jump conditions for the three-dimensional Navier–Stokes equations involving an immersed moving membrane, Appl. Math. Lett. 14 (2) (2001) 149–154.

[50] S. Xu, Z. Wang, Systematic derivation of jump conditions for the immersed interface method in three-dimensional flow simulations, SIAM J. Sci. Comput. 27 (6) (2006) 1948–1980.

[51] E. Newren, A. Fogelson, R. Guy, R. Kirby, Unconditionally stable discretizations of the immersed boundary equations, J. Comput. Phys. 222 (2) (2007) 702–719.

[52] Y. Mori, C. Peskin, Implicit second-order immersed boundary methods with boundary mass, Comput. Meth. Appl. Mech. Engrg. 197 (2008) 25–28.

[53] D. Le, B. Khoo, K. Lim, An implicit-forcing immersed boundary method for simulating viscous flows in irregular domains, Comput. Meth. Appl. Mech. Engrg. 197 (25) (2008) 2119–2130.

[54] E. Newren, A. Fogelson, R. Guy, R. Kirby, A comparison of implicit solvers for the immersed boundary equations, Comput. Meth. Appl. Mech. Engrg. 197 (25) (2008) 2290–2304.

[55] G. Tryggvason, B. Bunner, A. Esmaeeli, D. Juric, N. Al-Rawahi, W. Tauber, J. Han, S. Nae, Y.-J. Jan, A front-tracking method for the computations of multiphase flow, J. Comput. Phys. 169 (2001) 708–759.

[56] I. Graham, M. Hagger, Unstructured additive Schwarz-conjugate gradient method for elliptic problems with highly discontinuous coefficients, SIAM J. Sci. Comput. 20 (1990) 2041–2066.

[57] T. Hou, Z. Li, S. Osher, H. Zhao, A hybrid method for moving interface problems with application to the Hele-Shaw flow, J. Comput. Phys. 134 (1997) 236–252.

[58] A. Wiegmann, K. Bube, The immersed interface method for nonlinear differential equations with discontinuous coefficients and singular sources, SIAM J. Numer. Anal. 35 (1) (1998) 177–200.

[59] I. Klapper, T. Shaw, A large jump asymptotic framework for solving elliptic and parabolic equations with interfaces and strong coefficient discontinuities, Appl. Numer. Math. 57 (5) (2007) 657–671.

[60] J. Nordström, M. Svärd, Well-posed boundary conditions for the Navier–Stokes equations, SIAM J. Numer. Anal. 43 (3) (2005) 1231–1255.

[61] S. Zhao, G. Wei, High-order FDTD methods via derivative matching for Maxwell's equations with material interfaces, J. Comput. Phys. 200 (1) (2004) 60–103.

[62] C. Tu, C. Peskin, Stability and instability in the computation of flows with moving immersed boundaries: a comparison of three methods, SIAM J. Sci. Stat. Comput. 13 (1992) 1361–1376.

[63] A. Mayo, C. Peskin, An implicit numerical method for fluid dynamics problems with immersed elastic boundaries, Contemp. Math. 141 (1993) 261–277.

[64] T. Hou, Z. Shi, Removing the stiffness of elastic force form the immersed boundary method for the 2D Stokes equations, J. Comput. Phys. 227 (21) (2008) 9138–9169.

[65] W. Feiereisen, W. Reynolds, J. Ferziger, Numerical simulation of a compressible, homogeneous, turbulent shear flow, Tech. Rep. TF-13, Dept. Mech. Engrg., Stanford University, 1981.

[66] T. Zang, On the rotation and skew-symmetric forms for incompressible flow simulation, Appl. Numer. Math. 7 (1) (1991) 27–40.

[67] G. Blaisdell, Numerical simulation of compressible homogenous turbulence, Ph.D. Thesis, Stanford University, 1991.

[68] A. Honein, P. Moin, Higher entropy conservation and numerical stability of compressible turbulence simulations, J. Comput. Phys. 201 (2) (2004) 531–545.

[69] B. Strand, Summation by parts for finite difference approximations for d/dx, J. Comput. Phys. 110 (1) (1994) 47–67.

[70] M. Carpenter, D. Gottlieb, S. Abarbanel, Time-stable boundary conditions for finite difference schemes solving hyperbolic systems: methodology and application to high-order compact schemes, J. Comput. Phys. 111 (2) (1994) 220–236.

[71] A. Mayo, The fast solution of Poisson's and the biharmonic equations on irregular regions, SIAM J. Numer. Anal. 21 (1984) 285–299.

[72] C. Fletcher, Computation techniques for fluid dynamics, Fundamental and General Techniques, vol. 1, Springer, 1996.

[73] E. Sanmiguel-Rojas, J. Ortega-Casanova, C. del Pino, R. Fernandez-Feria, A Cartesian grid finite-difference method for 2D incompressible viscous flows in irregular geometries, J. Comput. Phys. 204 (1) (2005) 302–318.

[74] J. Strikwerda, Initial boundary value problems for incompletely parabolic systems, Commun. Pure Appl. Math. 30 (6) (1977) 797–822.

[75] M. Carpenter, D. Gottlieb, S. Abarbanel, The stability of numerical boundary treatments for compact high-order finite-difference schemes, J. Comput. Phys. 108 (1993) 272–295.

[76] H.-O. Kreiss, J. Lorenz, Initial-Boundary Value Problems and the Navier–Stokes Equations, Academic Press, 1989.

[77] K. Mattsson, J. Nordström, Summation by parts operators for finite difference approximations of second derivatives, J. Comput. Phys. 199 (2) (2004) 503–540.

[78] D. Kirshman, F. Liu, A gridless boundary condition method for the solution of the Euler equations on embedded Cartesian meshes with multigrid, J. Comput. Phys. 201 (1) (2004) 119–147.

[79] J. Yoh, X. Zhong, New hybrid Runge–Kutta methods for unsteady reactive flow simulation, AIAA J. 42 (8) (2004) 1593–1600.

[80] C. Pantano, R. Deiterding, D. Hill, D. Pullin, A low numerical dissipation patch-based adaptive mesh refinement method for large-eddy simulation of compressible flows, J. Comput. Phys. 221 (1) (2007) 63–87.

[81] M. Svärd, M. Carpenter, J. Nordström, A stable high-order finite difference scheme for the compressible Navier–Stokes equations, far field boundary conditions, J. Comput. Phys. 225 (1) (2007) 1020–1038.

[82] G. Strang, Accurate partial difference methods II. Non-linear problems, Numer. Math. 6 (1964) 37–46.

[83] M. Abramowitz, I. Stegun, Handbook of Mathematical Functions, Dover, 1965.

[84] K. Thompson, Time dependent boundary conditions for hyperbolic systems, J. Comput. Phys. 68 (1) (1987) 1–24.

[85] C. Williamson, Oblique and parallel modes of vortex shedding in the wake of a circular cylinder at low Reynolds numbers, J. Fluid Mech. 206 (1989) 579–627.

[86] C. Liu, X. Zheng, C. Sung, Preconditioned multigrid methods for unsteady incompressible flows, J. Comput. Phys. 139 (1998) 35.

[87] S. Gottlieb, C.-W. Shu, E. Tadmor, Strong stability-preserving high-order time discretization methods, SIAM Rev. 43 (1) (2001) 89–112.

[88] P. Morse, K. Ingard, Theoretical Acoustics, Princeton University Press, 1986.

[89] J. Kim, D. Kim, H. Choi, An immersed-boundary finite-volume method for simulations of flow in complex geometries, J. Comput. Phys. 171 (1) (2001) 132–150.

[90] D. Russell, Z. Wang, A Cartesian grid method for modeling multiple moving objects in 2D incompressible viscous flows, J. Comput. Phys. 191 (1) (2003) 177–205.