Accepted Manuscript

A versatile embedded boundary adaptive mesh method for compressible flow in complex geometry

M. Al-Marouf, R. Samtaney

PII: S0021-9991(17)30147-X
DOI: <http://dx.doi.org/10.1016/j.jcp.2017.02.044>
Reference: YJCPH 7176

To appear in: *Journal of Computational Physics*

Received date: 8 August 2016
Revised date: 3 December 2016
Accepted date: 18 February 2017

Please cite this article in press as: M. Al-Marouf, R. Samtaney, A versatile embedded boundary adaptive mesh method for compressible flow in complex geometry, *J. Comput. Phys.* (2017), <http://dx.doi.org/10.1016/j.jcp.2017.02.044>

This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

A Versatile Embedded Boundary Adaptive Mesh Method for Compressible Flow in Complex Geometry

M. Al-Marouf, R. Samtaney*

King Abdullah University of Science and Technology, 4700 KAUST, Thuwal 23955-6900, Saudi Arabia

Abstract

We present an embedded ghost-fluid method for numerical solutions of the compressible Navier Stokes (CNS) equations in arbitrary complex domains. A PDE multidimensional extrapolation approach is used to reconstruct the solution in the ghost-fluid regions and imposing boundary conditions on the fluid-solid interface, coupled with a multi-dimensional algebraic interpolation for freshly cleared cells. The CNS equations are numerically solved by the second order multidimensional upwind method. Block-structured adaptive mesh refinement, implemented with the Chombo framework, is utilized to reduce the computational cost while keeping high resolution mesh around the embedded boundary and regions of high gradient solutions. The versatility of the method is demonstrated via several numerical examples, in both static and moving geometry, ranging from low Mach number nearly incompressible flows to supersonic flows. Our simulation results are extensively verified against other numerical results and validated against available experimental results where applicable. The significance and advantages of our implemen-

*Corresponding author, email: Ravi.Samtaney@kaust.edu.sa

tation, which revolve around balancing between the solution accuracy and implementation difficulties, are briefly discussed as well.

Keywords: Ghost cell method, PDE multidimensional extrapolation, compressible Navier Stokes equations, adaptive mesh refinement, low Mach number flow, supersonic flow

1. Introduction

Despite considerable advancements in computational science and technology, simulating fluid flow in a complex time-varying geometry continues to be one of challenging areas in computational fluid dynamics (CFD). Classically, such flows have been successfully tackled using boundary conforming methods including both structured body-fitted meshes or unstructured mesh methods. However, these methods, in general, have drawbacks including high computational cost, complex implementation and mesh generation. These drawback exacerbate further in cases of highly complex time-varying geometry. To overcome some of the drawbacks of body conforming meshes, non-boundary conforming methods, typically employing Cartesian grids, were developed as an alternative. These aim to simplify the development and implementation of the numerical solution of flows with complex boundaries [1]. Moreover, these methods replace the mesh generation steps in body conforming methods to a somewhat simpler problem of representing and tracking the Fluid-Solid Interface (FSI). In addition, the Cartesian grid methods typically require less memory and CPU time compared with boundary conforming methods [2].

A variety of Cartesian grid methods have been developed over the last

few decades. We highlight a few of these here. One method is the Immersed Boundary Method (IBM), which was originally proposed by Peskin [3] to model fluid structure interaction in the context of cardiac flows. It is categorized into two main families: the continuous (or diffuse) and discrete (or sharp) methods [4]. In the continuous method, e.g. such as those developed by Peskin [5, 6], the elastic solid boundaries are modeled through a smooth external forcing term added to the continuous momentum equation, and operate on some bounded volume surrounding the FSI. Later, this was extended to simulate flow past rigid bodies, as noted, for example, in the works of Goldstein et al. [7] and Saiki & Biringen [8], in which they modeled the forcing terms of the immersed boundaries by feedback techniques. Another continuous immersed boundary method was introduced by Lai & Peskin [9] who approximated the rigid boundaries by highly stiff elastic boundaries. Additional examples can be found in the works of Zou et al. [10], Su et al. [11] and Pan [12]. On the one hand, the main advantage of this method is that it can be easily implemented by adding source terms to the Navier Stokes equations without major adjustment. On the other hand, the boundary conditions are diffused over the operational area of the forcing terms. This reduces the accuracy of the numerical solution. Nevertheless, the discrete or sharp methods solve this problem by modifying the discretization scheme of the Navier Stokes equations or reconstructing the forcing terms in a manner that the boundary conditions are satisfied at the sharp FSI. Mohd-Yusof [13] and Fadlun et al. [14] proposed a reconstruction scheme for the solution at the fluid nodes neighboring the FSI. They proposed one dimensional linear interpolation along the appropriate grid line to implicitly

Figure 1: The ghost cell node is evaluated by linear interpolation along the normal line using both of the image point I & the FSI node S . The image point is evaluated by approximating the shaded area by a quadratic polynomial fitted using the fluid points ($F1, F2, F3$) & the FSI node S .

impose the boundary conditions exactly at the FSI, with satisfactory results for a number of different flow situations [2, 15]. However, it is not always obvious how to select the reconstruction direction, specially in the case of complex geometry. This problem can be avoided by performing the interpolation along the normal line to the FSI which was applied in the work of Balaras [16], Gilmanov et al. [17], Gilmanov & Sotiropoulos [18] and Yang & Balaras [19]. It is also possible to reconstruct the solution on ghost cell nodes (i.e. solid nodes neighboring the FSI) as can be seen in the work of Majumdar et al. [20] and Tseng & Ferziger [21]. Next we discuss two common techniques: the algebraic interpolation method and the PDE extrapolation method.

1.1. Algebraic interpolation method

A common interpolation technique that is employed in reconstructing the solution for sharp interfaces is based on approximating the solution in the neighbor fluid nodes by a multi-variable polynomial, then using it to evaluate the solution at the image or ghost fluid nodes. For example, in Figure 1, the solution variable (q) at the fluid points $[F1, F2, F3]$, in addition to the FSI node S , can be approximated by:

$$q = ax + by + cxy + d. \quad (1)$$

By evaluating this polynomial in the fluid points and the FSI node taking into account that in the case of the Neumann boundary condition the following equation is evaluated at S :

$$\frac{\partial q}{\partial n} = \frac{\partial q}{\partial x} \frac{\partial x}{\partial n} + \frac{\partial q}{\partial y} \frac{\partial y}{\partial n}. \quad (2)$$

By substituting the derivatives of equation (1) in equation (2):

$$\frac{\partial q}{\partial n} = a \frac{\partial x}{\partial n} + b \frac{\partial y}{\partial n} + c(y \frac{\partial x}{\partial n} + x \frac{\partial y}{\partial n}). \quad (3)$$

Thus the coefficients (a, b, c, d) can be obtained by solving the resulted linear system. For Dirichlet boundary condition imposed on q at S :

$$\begin{pmatrix} q_{F1} \\ q_{F2} \\ q_{F3} \\ q_S \end{pmatrix} = \begin{pmatrix} x_{F1} & y_{F1} & x_{F1}y_{F1} & 1 \\ x_{F2} & y_{F2} & x_{F2}y_{F2} & 1 \\ x_{F3} & y_{F3} & x_{F3}y_{F3} & 1 \\ x_S & y_S & x_Sy_S & 1 \end{pmatrix} \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix}, \quad (4)$$

whereas for the case of Neumann boundary condition at S , the system is:

$$\begin{pmatrix} q_{F1} \\ q_{F2} \\ q_{F3} \\ \frac{\partial q_S}{\partial n} \end{pmatrix} = \begin{pmatrix} x_{F1} & y_{F1} & x_{F1}y_{F1} & 1 \\ x_{F2} & y_{F2} & x_{F2}y_{F2} & 1 \\ x_{F3} & y_{F3} & x_{F3}y_{F3} & 1 \\ \frac{\partial x_S}{\partial n} & \frac{\partial y_S}{\partial n} & y_S \frac{\partial x_S}{\partial n} + x_S \frac{\partial y_S}{\partial n} & 0 \end{pmatrix} \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix}, \quad (5)$$

Finally, the fitted polynomial, equation (1), can be used to estimate or "reconstruct" the solution variable q at either the image node I or the ghost fluid point G (Figure 1). Since the matrices in the above linear systems only depend on the coordinates of the fluid points and FSI nodes, they can be inverted at the beginning and stored to be used during the time marching procedure. However, the cost of solving these linear systems can become significantly high for moving boundary cases because of the need to solve these linear systems at each time step, or at least after every interface movement, and this is a disadvantage for this interpolation technique. Moreover, interpolating by a higher order polynomial sometimes introduces spurious extrema to the solution at the interface [21]. The spurious extrema are also generated when the linear system becomes ill-conditioned, for instance, when the FSI node approaches one of the fluid points. In order to avoid such problems, the solution can be reconstructed by a smoother interpolation algorithm such as the Inverse Distance Weighting method (IDW).

The IDW interpolation algorithm, proposed by Franke [22], preserves local maxima while smoothly interpolating the solution by an averaging algorithm. An improved immersed boundary method based on IDW reconstruction scheme was implemented and discussed by Gao et al. [23]. They successfully utilized the IDW method in simulating incompressible flow in

two and three dimensional complex geometry domains. Additional discussion about the IDW method can be found in the work of Iaccarino and Verzicco [24], Tseng & Ferziger [21] and additional examples of this method in Lundquist et al. [25]. A similar inverse distance based interpolation scheme used in solving incompressible Navier Stokes equation in a three dimensional complex geometry with moving boundaries by Gilmanov and Sotiropoulos [18].

1.2. PDE interpolation method

In this interpolation method, the solution is reconstructed by a partial differential interpolation technique. An example of this is the Aslam algorithm [26] which extrapolates the solution to the ghost fluid region by assuming that the solution varies in some specific way along the normal curves to the FSI. This is mathematically performed by iteratively solving an advection equation in pseudo-time, that extrapolates the solution along characteristic (normal) curves until the obtained steady solution satisfies the original assumption about the nature of the solution along the normal direction to the FSI. For example, in the simplest form of this algorithm, the solution variable q is assumed to be constant along the normal lines, thus the directional derivative of q along the normal lines to the FSI is zero, i.e., $\hat{n} \cdot \nabla q = 0$ has to be satisfied by the steady solution of the following advection equation:

$$\frac{\partial q}{\partial \tau} + H(\psi)(\hat{n} \cdot \nabla q) = 0, \quad (6)$$

where $\hat{n}$ is a unit vector normal to the FSI, ψ is levelset function and $H(\psi)$ is Heaviside function that has unit value where $\psi > 0$ and zero otherwise

in order to restrict the solution of the advection equation only to the ghost fluid region.

Examples for immersed boundary method that utilizes a PDE-based reconstruction algorithm can be seen in the work of Ng et al. [27], where they interpolate the intermediate solution of the velocity field, that is obtained by the projection method, by a third order extrapolation algorithm based on the Aslam approach and the variants of Min and Gibou [28]. A two steps interpolation algorithm based on this approach was also utilized in obtaining the solution of the incompressible Navier Stokes equations for complex moving boundaries implemented and tested by Chiu et al. [29]. Recently, Uddin et al. [30] have successfully implemented an interpolation scheme to reconstruct the solution in the ghost fluid region by a PDE-based approach that is inspired by analytic continuation.

An additional examples for similar PDE approach developed for fluid structure interaction can be found in the work of Cummings et al. [31], in which they study the solid material behavior under pressure loading and shock wave interaction. Along similar lines is the work of Arienti et al. [32] in implementing an Eulerian-Lagrangian coupled solver based on levelset method for inviscid fluid structure interactions.

1.3. Overview of proposed method

In order to simulate compressible flow in complex time varying geometries, we propose a multi-stage embedded boundary method that utilizes a combination of a high order PDE extrapolation scheme and the IDW interpolation algorithm. By using PDE extrapolation scheme, a high order estimation of the solution is predicted in the ghost fluid region. A smoother corrected

solution is subsequently calculated using the IDW algorithm employing symmetric stencils of the neighboring fluid and predicted ghost fluid points. One advantage of the implemented IDW-PDE reconstruction algorithm is that it can be considered as a geometry independent embedded boundary method as long as the geometrical details of the FSI are resolved with a sufficiently fine mesh that result in low local curvature. In other words, the interpolation stencil is the same for all image nodes regardless of the geometrical structure of the FSI near these nodes. Thus, the ill-conditioning problems that occurs by a somewhat careless selection of the interpolation stencil is simply avoided. This is not the case, for instance, in the polynomial based algebraic reconstruction algorithm. Moreover, the high order estimation of the solution that is obtained by the PDE scheme is further smoothened by the IDW algorithm. Avoiding the computationally expensive (especially for the three dimensional cases) frequent matrix inversion for the moving boundary cases is another advantage. Furthermore, our proposed method is implemented using an adaptive mesh refinement (AMR) framework. Uniform meshing in Cartesian methods are somewhat wasteful of computational resources (both memory and CPU flops) and this drawback is mitigated by locally increasing the mesh resolution (local mesh refinement) using AMR. In general, one strives to maintain a uniform truncation error consistently on the global domain by error estimation tools. Alternatively, one employs heuristics to ensure a good quality solution by refining the mesh in regions of high gradients and in the vicinity of the fluid solid interface.

The novelty of the proposed multi-stage embedded boundary AMR algorithm may be summarized as follows. Our method is based on a robust high

order stencil reconstruction algorithm that can be easily implemented and utilized for a complex moving boundary without a significant overhead in computational cost compared with the fixed boundary cases. A further improvement in the accuracy and computing efficiency is achieved by utilizing the AMR technique, which supports tackling complex geometry (with possibly multiple objects), and results in an overall versatile method for solving the compressible Navier Stokes equations for a myriad of flow cases.

The overall structure of the paper is as follows. In Section 2, the governing equations and the numerical time marching scheme used to integrate them are presented. The levelset method, multi-stage embedded boundary algorithm and AMR implementation is presented and discussed as well. In Section 3, results for nearly incompressible low Mach number flows past fixed bodies (both bluff bodies and streamlined ones) are presented. This is followed by Section 4 for low Mach number flow over bodies with a prescribed motion. Convergence tests for the proposed method are presented in Section 5. High-Mach number test cases for both fixed and moving boundaries are discussed in Sections 6 & 7, respectively. In all our test cases, the emphasis is on verification with other methods (e.g. body fitted curvilinear mesh methods, or other embedded boundary methods), and validation against experiments results where applicable. We present some conclusions with an outlook of future developments in Section 8.

2. Numerical methodology

2.1. Governing equations

The viscous compressible flows are governed by the Navier Stokes equations, written below in a non-dimensional strong conservation form as:

$$\frac{\partial U}{\partial t} + \frac{\partial F_j}{\partial x_j} = \frac{\partial G_j}{\partial x_j}, \quad (7)$$

where the solution vector U and the j^{th} direction inviscid F_j and viscous G_j fluxes are given by:

$$U = \begin{pmatrix} \rho \\ \rho u_i \\ E \end{pmatrix}, \quad (8)$$

$$F_j = \begin{pmatrix} \rho u_j \\ \rho u_i u_j + p \delta_{ij} \\ (E + p) u_j \end{pmatrix}, \quad (9)$$

$$G_j = \begin{pmatrix} 0 \\ \frac{1}{Re} \tau_{ij} \\ \frac{1}{Re} \tau_{ij} u_i - \frac{1}{Re} \frac{1}{Pr} \frac{\gamma}{\gamma-1} q_j \end{pmatrix}. \quad (10)$$

The solution vector U comprises of density ρ , momentum ρu_i , and the total energy per unit volume E related to the pressure by:

$$E = \frac{p}{\gamma - 1} + \frac{1}{2} \rho u_i u_i, \quad (11)$$

where $\gamma = c_p/c_v$ is the ratio of the specific heats. Throughout this work γ is held constant and equals 1.4. Other properties (viscosity and conductivity) are also assumed constant but this is not a serious limitation. The stress

tensor (defined without the isotropic pressure contribution) and the heat flux vector are given by:

$$\tau_{ij} = \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3} \frac{\partial u_k}{\partial x_k} \delta_{ij}, \quad (12)$$

$$q_j = -\frac{\partial T}{\partial x_j}. \quad (13)$$

The system is closed by the non-dimensional perfect gas equation of state, $p = \rho T$. The non-dimensionalization is based on the inflow speed u_0 , the inflow density ρ_0 , and the characteristic length of the solid object L_0 (e.g. diameter of a circular cylinder). The Reynolds number is defined as $Re = u_0 \rho_0 L_0 / \mu$, where μ is the viscosity of the fluid. The other non-dimensional parameter is the Prandtl number $Pr = \rho \mu C_p / \kappa$, where C_p is specific heat at constant pressure and κ is the thermal conductivity of the fluid. At the solid wall, the no slip boundary condition is physically applied by evaluating Dirichlet condition for the velocity field. By assuming a thermally insulated (adiabatic) solid walls, zero gradient need to be satisfied at the boundaries. This can be numerically satisfied by homogeneous Neumann boundary condition for the density and pressure.

2.2. Scheme for fluid nodes

The governing equations are numerically solved by the unsplit second-order Godunov method of Colella [33] and modified for 3D problems by Saltzman [34]. It is basically a prediction-correction scheme. Starting by the average of solution at time level n on the control volume V_i ,

$$U_i^n = \frac{1}{\sigma_i} \int_{V_i} U(\mathbf{x}, t^n) d\mathbf{x}, \quad (14)$$

where:

$$V_i = [(i - \mathbf{u}/2)h, (i + \mathbf{u}/2)h], \quad i \in \mathbb{Z}^D, \quad \mathbf{u} = (1, 1, \dots, 1), \quad (15)$$

h is the grid spacing, and $\sigma_i = h^D$ is the volume of V_i . A summary of the solution algorithm is outlined here for completeness following [33] and [35].

The solution is integrated to the next time level by:

$$U_i^{n+1} = U_i^n - \frac{\Delta t}{h} \sum_d \left(F_{i+\frac{1}{2}e^d}^{n+\frac{1}{2},d} - F_{i-\frac{1}{2}e^d}^{n+\frac{1}{2},d} + G_{i-\frac{1}{2}e^d}^{n+\frac{1}{2},d} - G_{i+\frac{1}{2}e^d}^{n+\frac{1}{2},d} \right), \quad (16)$$

where $F_{i+\frac{1}{2}e^d}^{n+\frac{1}{2},d}$ and $G_{i+\frac{1}{2}e^d}^{n+\frac{1}{2},d}$ are the inviscid and diffusive fluxes, respectively, at the cells interfaces $(i + \frac{1}{2}e^d)$, where e^d is unit vector in the direction d . The first step in computing the half time step, interface solution is to calculate the primitive variables, W , using the given conservative variables, that is $W = W(U)$. The fluxes are computed using the evaluated solution at the half time step $n + \frac{1}{2}$ and cells interface $i + \frac{1}{2}e^d$, so that $F_{i+\frac{1}{2}e^d}^{n+\frac{1}{2},d} \equiv F^d(W_{i+\frac{1}{2}e^d}^{n+\frac{1}{2}})$. The viscous fluxes $G_{i+\frac{1}{2}e^d}^{n+\frac{1}{2},d} \equiv G^d(W_{i+\frac{1}{2}e^d}^{n+\frac{1}{2}})$ may be lagged at time level n for simplicity, i.e., $G_{i+\frac{1}{2}e^d}^{n+\frac{1}{2},d} \approx G_{i+\frac{1}{2}e^d}^{n,d} = G^d(W_{i+\frac{1}{2}e^d}^n)$, with no appreciable difference in the computed solution, between $G_{i+\frac{1}{2}e^d}^{n+\frac{1}{2},d}$ and $G_{i+\frac{1}{2}e^d}^{n,d}$.

Following the notation and method in Reference [35], we begin with a Taylor series expansion to obtain the left/right time-centered solution $W_{i+\frac{1}{2}e^d,q}^{n+\frac{1}{2},d}$ at the cells interfaces, where $q = (L, R)$ representing the left and right states. The first step is the so-called normal predictor step.

$$W_{i,\pm,d} = W_i^n + \frac{1}{2}(\pm I - \frac{\Delta t}{h} A_i^d) P_{\pm}(\Delta^d W_i), \quad (17)$$

$$A_i^d = \nabla_U W_i \cdot \nabla_U F_i^d \cdot \nabla_W U_i, \quad (18)$$

$$P_{\pm}(W) = \sum_{\pm \lambda_k > 0} (l_i^{k,d} \cdot W) r_i^{k,d}, \quad (19)$$

where $\lambda_i^{k,d}$, $k = 1 \dots M$ are the M eigenvalues that associated with the right/left eigenvectors $(r_i^{k,d}, l_i^{k,d})$ of A_i^d .

An approximation of the flux derivative can be obtained by the first order Godunov first order scheme using the fluxes at the cells interface which are computed by solving the Riemann problem for right/left states. These are later used for the transverse derivative correction step int this unsplit scheme.

$$F_{i+\frac{1}{2}e^d}^d = R(W_{i,+,d}, W_{i+e^d,-,d}, d), \quad (20)$$

In the correction step the estimation of the interface solution $W_{i+\frac{1}{2}e^d}^{n+\frac{1}{2}}$, assuming that $d \neq \hat{d}$, is calculated using transverse derivatives as follows:

$$W_{i,\pm,d}^{n+\frac{1}{2}} = W_{i,\pm,d} - \frac{\Delta t}{2h} \nabla_U W \cdot (F_{i+\frac{1}{2}e^{\hat{d}}}^{\hat{d}} - F_{i-\frac{1}{2}e^{\hat{d}}}^{\hat{d}}). \quad (21)$$

For three dimensional cases, equations (17), (20) and (21) are modified to account for the (1, 1, 1) diagonal coupling as follows:

$$W_{i,\pm,\hat{d},\bar{d}} = W_{i,\pm,\bar{d}} - \frac{\Delta t}{3h} \nabla_U W \cdot (F_{i+\frac{1}{2}e^{\bar{d}}}^{\bar{d}} - F_{i-\frac{1}{2}e^{\bar{d}}}^{\bar{d}}), \quad (22)$$

$$F_{i+\frac{1}{2}e^{\hat{d}},\bar{d}}^{\hat{d}} = R(W_{i,+, \hat{d}, \bar{d}}, W_{i+e^{\hat{d}}, -, \hat{d}, \bar{d}}, \hat{d}), \quad (23)$$

$$W_{i,\pm,d}^{n+\frac{1}{2}} = W_{i,\pm,d} - \frac{\Delta t}{2h} \nabla_U W \cdot (F_{i+\frac{1}{2}e^{\hat{d}},\bar{d}}^{\hat{d}} - F_{i-\frac{1}{2}e^{\hat{d}},\bar{d}}^{\hat{d}} + F_{i+\frac{1}{2}e^{\bar{d}},\hat{d}}^{\bar{d}} - F_{i-\frac{1}{2}e^{\bar{d}},\hat{d}}^{\bar{d}}), \quad (24)$$

where $d \neq \hat{d} \neq \bar{d}$, and $0 \leq d, \hat{d}, \bar{d} \leq D$.

The monotonized difference $\Delta^d W_i$, in equation (17), is computed by applying the van Leer limiter on the one-sided or central difference of the characteristic variables as follows. We define the expansions:

$$2(W_i - W_{i-e^d}) = \sum_{k=1}^M \alpha_{i,L}^k r_i^{k,d}, \quad (25)$$

$$\frac{1}{2}(W_{i+e^d} - W_{i-e^d}) = \sum_{k=1}^M \alpha_{i,C}^k r_i^{k,d}, \quad (26)$$

$$2(W_{i+e^d} - W_i) = \sum_{k=1}^M \alpha_{i,R}^k r_i^{k,d}. \quad (27)$$

Then we solve for $(\alpha_{i,L}^k, \alpha_{i,C}^k, \alpha_{i,R}^k)$, using the left eigenvectors:

$$\alpha_{i,L}^k = l_i^{k,d} \cdot 2(W_i - W_{i-e^d}), \quad (28)$$

$$\alpha_{i,C}^k = l_i^{k,d} \cdot \frac{1}{2}(W_{i+e^d} - W_{i-e^d}), \quad (29)$$

$$\alpha_{i,R}^k = l_i^{k,d} \cdot 2(W_{i+e^d} - W_i). \quad (30)$$

Next, the van Leer limiter is applied as follows:

$$\alpha_i^k = \begin{cases} \min(|\alpha_{i,L}^k|, |\alpha_{i,C}^k|, |\alpha_{i,R}^k|) \times \text{sgn}(\alpha_{i,C}^k) & \text{if } \alpha_{i,L}^k \cdot \alpha_{i,R}^k > 0, \\ 0 & \text{otherwise.} \end{cases} \quad (31)$$

Finally, the monotonized difference $\Delta^d W_i$ is computed as:

$$\Delta^d W_i = \sum_{k=1}^M \alpha_i^k r_i^{k,d}. \quad (32)$$

The above algorithm is slightly modified near to the physical domain boundaries. For more details regarding these modifications and the AMR implementation of this scheme, consult Reference [35].

2.3. Front representation

The solid boundary (or boundaries) in the fluid domain is represented by a union of linear line segments (although this is chosen only for simplicity in determining the signed distance levelset function). It is the accuracy in representing this boundary that determines the spacing between these nodes. In general, a high curvature region requires a larger number of closely spaced nodes to accurately represent the solid boundary. The nodes used to represent the fluid-solid boundary need not necessarily be equally spaced.

However in several cases, as part of post-processing the solution, we need to compute the hydrodynamic forces on the solid object(s) as in equation (43). For this, the distance between the FSI nodes is of the same order as that of the spacing in the surrounding fluid mesh: a choice evidently dictated by accuracy considerations. The linear elements with the nodes are used to compute a signed distance function $\psi(\mathbf{x}, t)$ which provides an implicit presentation of the FSI in the level-set approach [36]. $\psi(\mathbf{x}, t)$ is computed over the computational domain as the shortest (normal) distance from the each cell center to the FSI. In addition of representing the FSI, $\psi(\mathbf{x}, t)$ decomposes the computational domain into a physical domain (i.e. fluid sub-domain) and ghost-fluid domain (i.e. the solid object sub-domain). The ghost-fluid domain includes all the cells which their centers are located within the solid boundaries as in Figure 2. The remaining finite volume cells comprise the fluid domain in which the physical solution at the beginning of each time-level is defined on the cell center. The type of the cell (ghost or real) can be determined by extending the normal line from the cell center to the solid object and counting the number of all possible intersections with the FSI. An odd number of intersections indicates that this cell is a ghost fluid cell, otherwise it is a physical fluid cell (Figure 3). There is no restriction on the convexity of the solid object in our method.

2.4. *Embedded boundary method*

The main concept behind our developed method is to extend the physical solution into the ghost fluid region so the finite difference operator is directly (with no modification) applied near the IFS. The procedure of extending the solution into the ghost fluid region is carried out in a manner to satisfy the

Figure 2: The level-set function ψ separates the computational domain in to a fluid sub-domain and a ghost-fluid sub-domain based on its sign, and implicitly represents the FSI as the zero level contour.

boundary conditions at the FSI. For example, in viscous flow problem, the velocity field solution of the momentum equation at the FSI has to satisfy Dirichlet boundary condition:

$$V^{n+1} = V_{\Omega}^{n+1}, \quad (33)$$

where V^{n+1} is the velocity vector, and V_{Ω}^{n+1} is the prescribed velocity solution at the FSI (Ω), both at the time level $n + 1$. Since the FSI is not in general aligned with the mesh points, the prescribed boundary solution “condition” can not be directly evaluated. This problem is solved by interpolating the prescribed boundary solution to the closest neighboring grid points. In our case, these are selected from the solid side of the FSI. These points are usually called boundary or ghost fluid cells. However, the solution is not interpolated directly to the ghost fluid cells. Instead, we adopted the concept of the image point method [20], in which the solution is interpolated to some image points, and then extrapolated to the ghost fluid cells. The image point (mirror) point

Figure 3: The ray originated at the fluid point F crosses the solid object boundary twice (even) at points $[S_1, S_2]$, whereas the other ray originated in the ghost fluid point G crosses the object boundary three times (odd) at points $[S_3, S_4, S_5]$.

I , that is shown in Figure 1, is located in the fluid domain by extending a normal line from the ghost point G to the FSI node S , such that the point S equally divides the distance between I & G . Thus, to impose Dirichlet boundary condition for the solution variable q at the FSI node S :

$$q_G = 2q_S - q_I, \quad (34)$$

whereas for Neumann boundary condition:

$$q_G = h \frac{\partial q_S}{\partial n} + q_I, \quad (35)$$

where the values of q_S and $\frac{\partial q_S}{\partial n}$ are known from the boundary conditions. The solution at the image point q_I is evaluated by the algebraic IDW algorithm. The central stencil used for the IDW interpolation algorithm includes, in general, points from fluid and ghost fluid regions. Thus, the solution is evaluated at the required ghost fluid cells by the PDE extrapolation scheme.

Based on this, we consider our algorithm as prediction-correction method. The solution is predicted first at the ghost fluid cells by the PDE scheme.

Then it is corrected by the extrapolation of the image points solution that is evaluated by the algebraic IDW algorithm.

2.4.1. PDE interpolation scheme

The solution in the ghost cells domain is approximated by extrapolating the solution by Aslam's multidimensional PDE extrapolation scheme [26]. The idea of this scheme is to advect the solution along the gradient of the level-set function, that is normal to the FSI, by solving equation (6) in pseudo-time τ , until a steady solution is reached. As discussed above, ψ is the level-set function that is used to define the Heaviside function as: $H(\psi) = 1$ if $\psi > 0$, in the solid domain, and $H(\psi) = 0$ if $\psi \leq 0$, in the fluid domain; and the unit normal vector $\hat{n}$ is computed by:

$$\hat{n} = \frac{\nabla\psi}{|\nabla\psi|}. \quad (36)$$

Upon steady state, equation (6) is reduced to $\hat{n} \cdot \nabla u = 0$, implying that the variable u is constant along the normal line $\hat{n}$. In other word, the solution u is advected from the image point I in the fluid to the ghost fluid cell G as shown in Figure 4.

The concept of this PDE extrapolation (or a continuation) method is that the extrapolation can be achieved by assuming that the solution is constant, has zero directional derivative along the normal line to the FSI. This can be reasonable assumption close to the FSI. Nevertheless, the validity of this assumption is reduced as we move away from the FSI.

A more accurate "higher order" extrapolation can be accomplished by assuming, for example, that the solution changes linearly, which has constant directional derivative along the normal line; or to achieve a quadratic

Figure 4: The point I is an image of the ghost point G , defined by extending the normal from G to S to the fluid domain.

approximation, in a case where the solution has to have a constant second directional derivative along the normal line, and so on [26, 37]. Thus, to perform higher order PDE we start by extrapolating the highest constant non-zero directional derivative as explained above. Then the next lower directional derivative is extrapolated in a way that its directional derivative equals to the previously extrapolated ones. The process is repeated until the required solution is extrapolated. For example, the high order PDE extrapolation for some solution variable u is estimated to varies according to p^{th} order polynomial along the normal, starts by calculating the first non-zero directional derivatives in the fluid domain (or where its originally defined) according to:

$$q^{(m)} = \hat{n} \cdot \nabla q^{(m-1)}, \quad (37)$$

for $m = 1, 2, \dots, p$, and $q^{(0)} = q$. Next, setting $q^{(p+1)} = 0$, we solve the

following advection equation for $(m = p, \dots, 0)$:

$$\frac{\partial q^{(m)}}{\partial t} + H(\psi + r)\hat{n} \cdot \nabla q^{(m)} - q^{(m+1)} = 0. \quad (38)$$

Note that the higher order directional derivatives are not well-defined in a part of the fluid domain which requires ghost cells in the numerical estimation of the higher order derivatives $(q^1, q^2 \dots q^m)$. So we choose the domain requiring extrapolation to include part of this fluid domain by using a shift parameter denoted by r above.

2.4.2. Reconstruction by inverse distance weighting

The IDW method, which was developed by Franke [22], interpolates the value of a variable by averaging its known values at a set of neighboring scattered data points. Mathematically, the variable q can be approximated at point O using n neighboring points by:

$$q_O = \sum_{m=1}^n \frac{\omega_m}{c} q_m, \quad (39)$$

$$\omega_m = \left(\frac{R - h_m}{Rh_m} \right)^p, \quad (40)$$

$$c = \sum_{m=1}^n \left(\frac{R - h_m}{Rh_m} \right)^p, \quad (41)$$

where h_m is the distance from the point O and the m^{th} neighboring point and $R = \max(h_m)$ is the maximum distance. p is a positive number, usually equal two, and is called power parameter. The weights ω_m is a function that is decreasing smoothly as the distance between the interpolation point and the neighboring data point increases with a rate controlled by the power parameter p . The smooth reconstruction of the interpolated solution is the

main attraction to the IDW method beside its ability to maintain the local maxima.

The reconstruction part of our embedded boundary algorithm is comprised of the following steps:

- **Prediction step:** the solution is extrapolated to the ghost fluid domain by either quadratic or linear PDE extrapolation.
- **IDW interpolation step:** the image points are located and evaluated by IDW interpolation using all cells in a neighborhood (for example, a small rectangle of size $4h \times 4h$ in 2D) of the image points, which will include points from previously evaluated ghost fluid domain.
- **Correction step:** the image points with the known boundary condition on the FSI are used to reevaluate the predicted solution at the ghost cells as in Figure 4.

2.4.3. Accuracy of the reconstruction method

To assess the accuracy of the developed reconstruction algorithm, we evaluate the fluid domain close to the FSI by a third order polynomial that satisfies the Dirichlet boundary condition at the interface, and then extrapolate this solution into the ghost fluid domain. The L_2 and L_∞ norms of the errors are computed at different mesh resolutions. The convergence rate of the errors norms are illustrated in Figure 5.(a). The same procedure is repeated for estimating the reconstruction accuracy for a sinusoidal function that satisfies the Neumann boundary condition at the interface, Figure 5.(b). For both cases, a quadratic PDE extrapolation is used in the prediction step.

As can be observed from Figure 5, the Dirichlet boundary condition is imposed with a third order accuracy, whereas the extrapolated solution that imposes the Neumann boundary condition is of a second order accuracy in both of the L_2 and L_∞ norms.

Figure 5: L_2 and L_∞ norms of the error of the extrapolated solutions by the developed prediction-correction reconstruction algorithm. (a) Dirichlet boundary condition case, (b) Neumann boundary condition case.

2.4.4. Moving boundaries

An additional care has to be given to the points that change their status from solid to fluid. In other words, the points located in solid domain at the time level $t = n - 1$, by the movement of the FSI, they become fluid points at time level $t = n$, for example point G in Figure 6. Their solution stored at time level n is not valid physical solution that can be used to march to next time level. Thus, the solution at these points need to be reevaluated by an interpolation from the neighbor points, that have valid physical solution at both time levels $t = n - 1$ & n , and the boundary condition on the FSI. The

Figure 6: The dashed green curve is the FSI at time level $n - 1$, which moves to the position of the solid red curve at time level n .

prediction-correction reconstruction algorithm explained above is utilized in evaluating these points, noting that the PDE extrapolation is performed to extend the solution based on the levelset function ψ^{n-1} (Figure 6).

2.5. AMR algorithm

The lower accuracy and violation of conservation properties of the solutions near at the FSI are some of the main drawbacks of the Cartesian mesh methods [30]. To reduce the effect of these problems, we recommend combining Cartesian mesh method with adaptive mesh refinement (AMR) with the aim of utilizing the simplicity of the Cartesian mesh techniques and attaining acceptable solution accuracy at a reasonable computational cost.

The solution algorithm is implemented with a parallel block-based hierarchical adaptive mesh AMR-Godunov code using Chombo library [38]. Chombo's time stepping scheme is based on the work of Berger & Oliger [39] and Berger & Colella [40]. The Block-Based AMR divides the computational

domain to a number of blocks each of which can be treated as independent Cartesian mesh, which simplifies the implementation and the development of the solution scheme by utilizing standard finite difference/volume schemes for each block. The adaptive meshing starts by covering the computational domain by a base coarse grid. Then finer nested grids are added over the coarser one in a hierarchical manner. The mesh can be fixed either with prescribed refinement levels or dynamic that is refined or coarsened at regular time step intervals based on some error estimation criterion during the course of the simulation. Our current refinement criterion is based on the magnitude of the gradient of the density for the high Mach number flow and vorticity magnitude for the low Mach number flow. Moreover, the mesh points, at least, in some narrow band enclosing the FSI is always refined up to the maximum level of refinement. In the case of the moving boundary, a regriding step is performed whenever the FSI approaches the boundary of its current level of refinement in order to assure that the FSI is surrounded by the same (finest) mesh resolution during the entire simulation.

2.5.1. Parallel implementation and scaling

The implemented embedded-AMR solver is parallelized using the message passing interface (MPI) used within the Chombo library. It is implemented based on single program multiple data (SPMD) approach, which has an advantage of being compatible to both of shared and distributed memory platforms. Chombo distributes data in a hierarchical pattern of three ranks. The first rank is an array of the base and refinement levels. Each level, in the second rank, includes number of patches that are arranged in an array as well. The third rank, the patch, is the actual data container that holds

rectangular data array that has as the number of spatial dimensions as the computational domain has. The parallelization procedure are performed at the level rank by distributing patches to available processes. The distribution process is designed based on an objective of reducing the load balance inefficiency which leads to knapsack optimization problem. However, the knapsack problem is NP complete [41]. Thus, the heuristic Kernighan-Lin algorithm is implemented in Chombo [38] in order to optimize the load-balancing. The scalability of the developed embedded-AMR method was analyzed by executing three dimensional tests cases (similar to those discussed in section 3.2) on Shaheen II, the Cray XC40 supercomputer, ranked as the 9th fastest supercomputer by the top500.org November 2015 list. The performed scaling tests showed that the implemented method is linearly scalable up to 10^3 process. These scaling results are discussed in Reference [42] and not repeated here.

2.6. Turbulence modeling for high-Re flows

Here we digress from the main theme of the paper to briefly discuss turbulence models or lack thereof in our implementation. It is estimated that fully resolved high Reynolds number three dimensional simulations, i.e., *direct numerical simulations*, require $O(Re^{\frac{9}{4}})$ mesh points for flows with no boundaries, $O(Re^{\frac{37}{14}})$ points [43] for wall-bounded flows. From the vast body of turbulence modeling and CFD literature, it is evident that under-resolved simulations three-dimensional high-Reynolds number flows require some form of modeling for the sub-grid stresses. Our numerical Godunov-type scheme has sufficient numerical dissipation that even under-resolved high-Re simulations are stably computed. One can view the numerical dissipation in

our scheme as a turbulence model, a viewpoint commonly known as *implicit large-eddy simulation (implicit LES or ILES)* [44]. We hasten to note that most of our low-Re simulations are fully resolved. Some of our high-Re cases are computed in a two-dimensional domain. Such flows, clearly, may not be realizable in practice and may not be considered turbulent flows; nevertheless these 2D high-Re flows constitute good test cases for verification against other methods. Our method is robust enough that under-resolved 3D high-Re number simulations (not reported here), performed in the spirit of implicit LES, did not encounter any numerical instabilities or issues. One of the drawbacks of the ILES approach is that the sub-grid kinetic energy term is neglected in the filtered energy equation. Hence, we choose not to present these results here because we prefer explicit modeling of the sub-grid stresses with quantifiable Reynolds number effects. Our future plan includes implementation of sub-grid scale models and extending our simulation capability to explicit LES.

3. Results: nearly incompressible flows with fixed boundaries

In this section, we present results from simulations with the method proposed in Section 2. The results are presented with a viewpoint to demonstrate the versatility of the code: we first present results for low-speed subsonic (nearly incompressible flows) and continue with increasing Reynolds number. Moreover, we include cases with both static (this section) and moving boundaries (next section). We culminate our tests with high Mach number supersonic flows (Section 6). The emphasis is on verification with other published results, and validation with experiments wherever applicable. We also

demonstrate convergence of the basic algorithm.

3.1. Two dimensional flow tests

3.1.1. Flow past a circular cylinder

The circular cylinder problem is a classic benchmark test case for immersed boundary methods to examine flows in complex geometry. We performed the simulations at low Mach number of $M = 0.05$ as an approximation of incompressible flow, for Reynolds numbers $Re = 40, 100$, and 300 , where the Reynolds number is computed based on the free stream velocity u_∞ and density ρ_∞ and the cylinder diameter D , that is ($Re = \rho_\infty u_\infty D / \mu$). The selection of these Reynolds number is based on strong differences in flow patterns. At Reynolds number $Re = 40$ the flow field comprises of two steady symmetric vortices attached to the cylinder. This flow pattern, also known as the laminar steady recirculation region [45], characterizes the incompressible flow that occurs when the Reynolds number is less than the critical point of Bénard-von Kármán instability ($Re = 49$) [46, 47]. When the Reynolds number exceeds this critical point the Kármán vortex street develops and a two-dimensional periodic wake is observed up to $Re = 140 - 194$. As the Reynolds number increases further, three dimensional vortical structures start appearing in the wake. Here we restrict the discussion to 2D flows, and postpone the discussion on the 3D flow at $Re = 300$ until Section 3.2.

External flows, such as those past a cylinder, typically require a large computational domain to reduce the effect of the inflow and outflow boundary conditions. The use of AMR is especially conducive to the employment of a large computational domain to reduce boundary effects. In this case, we used a domain size of $[256D \times 64D]$ with the center point of the cylinder located

at $(x, y) = (32, 32)$. The domain size is larger than most those encountered in the literature for this flow. The governing equations were discretized on a cell-centered mesh with base mesh size of $[512 \times 128]$ cells which implies a cell spacing of $h = 1/2$. Four levels of refinement were employed in the vicinity of the FSI in addition to three refinement levels dynamically constructed based on the vorticity magnitude in the wake region. The refinement ratio for all levels was chosen to be 4. Thus, at the finest level, the cell spacing is $h = 1/512$. This implies that even for a domain of modest size $[20D \times 10D]$, this is equivalent to an effective resolution of $[10240 \times 5120]$. Tests were also performed with three levels of refinement with dynamic and fixed meshes.

External ghost cells are filled with values to satisfy the homogeneous Neumann boundary condition for all of the solution variables, except velocity, at all of the computational domain boundaries. The velocity vector is specified at the inflow boundary by the Dirichlet boundary condition and a no-slip boundary condition is imposed on the FSI. An adiabatic wall assumption lead to zero normal gradient for the temperature across the FSI, which is implemented indirectly by numerically implementing homogeneous Neumann boundary condition for the pressure and density. The solution was advanced in time starting with a uniform flow velocity equal to the free stream inflow condition, and uniform density and pressure. The marching process in time was carried for a sufficiently long enough time duration to correctly measure the physical characteristics of the flow. For the steady flow case $Re = 40$, the solution was marched in time until $t = 50\hat{t}$, where $\hat{t}$ is the convective time. Figure 7, illustrates the vorticity contours and the streamline plots of the steady solution. To perform a quantitative verification, we measured the

Figure 7: Vorticity contours and the streamlines for flow past a circular cylinder at $Re = 40$. Only a portion of the entire computational domain is shown.

wake length, the attached vortices location, the separation angle and drag coefficient. These are tabulated and compared with other experimental and computational results in Table 1.

The pressure and skin friction coefficients for this flow are computed. We observe that the raw values contain spurious high wavenumber oscillations. We note that the physical solution itself is smooth in the vicinity of the FSI. However any diagnostics of quantities (e.g. pressure) on the FSI have a high wave number content due to the discrete representation of the FSI on a Cartesian mesh, and the interpolation method involved. This high-wavenumber error has been previously encountered in embedded boundary simulations (see for example Reference [21]) and is unphysical. We smoothen our results by applying a first order low-pass filter modeled by the convolution operator. For instance, to smoothen the computed pressure values at the interface:

$$\tilde{p}(\theta) = \int_0^{2\pi} g(\theta - \alpha) p(\alpha) d\alpha, \quad (42)$$

Figure 8: The computed raw and filtered pressure C_p and skin friction C_f coefficients along the cylinder surface obtained by two different mesh resolutions $h = (1/128, 1/512)$: (a) C_p results for $h = 1/128$, (b) C_p results for $h = 1/512$, (c) C_f results for $h = 1/128$, (d) C_f results for $h = 1/512$.

Figure 9: (a) The smoothed pressure coefficient C_p results compared with results of Park et al. [55] and Stålberg et al. [56]. (b) The smoothed skin friction coefficient C_f results compared with results of Majumdar et al. [20] and Tseng and Ferziger [21]. Both of the pressure and skin friction coefficients are obtained at two different mesh resolutions, present-A for $h = 1/512$, and present-B for $h = 1/128$.

where $p(\theta)$ is the unfiltered pressure measured on the cylinder interface at the angle θ , and the $\tilde{p}(\theta)$ is the filtered pressure. The kernel g is an exponential function expressed as $g(\theta) = k_c e^{-k_c \theta}$, where the k_c is the cutoff frequency (wave number in our case) that is chosen to be: $k_c = 0.2$. The originally computed pressure and skin friction coefficients which were obtained at two different mesh resolutions of $h = (1/128, 1/512)$ are depicted on Figure 8. The filtered results are shown as well, and compared with other computational results (Figure 9). The good agreement that is observed verifies the correctness of the present results.

The evaluation of the drag and pressure coefficients requires computation of the hydrodynamic forces. These forces and their coefficients are evaluated by first interpolating the viscous stresses and the pressure to the solid nodes on the FSI. This process first requires reconstructing the solution in the ghost-fluid domain by the developed multi-stage algorithm. Subsequently, we compute the stress tensor on all the cells surrounding the FSI including cells from both fluid and ghost-fluid regions (in actual practice this is limited to a small band in the ghost fluid region). The stress tensor and pressure are then interpolated using bilinear interpolation to the nodes of the FSI. Finally, the net hydrodynamic force acting on the solid object is computed by numerically evaluating the following line integral by trapezoidal rule along the FSI:

$$F_i = \int (\sigma_{ij} n_j) ds, \quad (43)$$

where $\mathbf{n}$ is a unit vector normal to the FSI, and $\boldsymbol{\sigma}$ is the stress tensor with the pressure contribution defined as $\sigma_{ij} = \tau_{ij} + p\delta_{ij}$. For flow over two-dimensional objects and three dimensional objects with geometry extruded

in the span-wise direction, projecting the computed hydrodynamic force F to the stream-wise direction gives the drag force $F_D = F_x$, while the perpendicular projection produces the lift force $F_L = F_y$. Then, we compute drag and lift coefficient defined as:

$$C_D = \frac{2F_D}{\rho_\infty U_\infty^2 A}, \quad C_L = \frac{2F_L}{\rho_\infty U_\infty^2 A}, \quad (44)$$

where A is a reference area, which in the case of circular cylinder, is obtained by projecting the frontal area in the stream-wise direction. In a similar manner, the interpolated values of the pressure at the FSI can be used to compute the pressure coefficient C_p defined as:

$$C_p = \frac{2(p - p_\infty)}{\rho_\infty U_\infty^2}, \quad (45)$$

where p_∞ is the free-stream pressure.

Table 2, lists the computed hydrodynamic forces and the Strouhal number $S_r = fD/U_\infty$, where f is the shedding frequency, for the second case at Reynolds number $Re = 100$ and compared with other results reported in the literature. Our results were obtained by simulating with four and three levels of refinements on dynamic and static adaptive mesh.

Figure 10: The variation of the drag C_D (left plot) and lift C_L (right plot) coefficient with time for flow over a circular cylinder at $Re = 100$.

Figure 11: Vorticity contours for flow over a circular cylinder at $Re = 100$ and the Kármán vortex street for (a) dynamic and (b) static adaptive meshes. Only a portion of the entire computational domain is shown.

Plots for time evolution of the drag and lift coefficient are depicted in Figure 10. Moreover, the instantaneous vorticity solutions, that clearly illustrate the Kármán vortex street, and the corresponding mesh structure are shown in Figure 11. From Tables 1 & 2, we observe that the mesh resolution and dynamic adaptive meshing have no effects on the solution in the wake region and only slightly affects the solution in the vicinity of the FSI. One advantage of our method is that the wake vortices are well resolved and do not suffer the ill effects of the severe mesh stretching in the wake commonly encountered in body fitted curvilinear O and C-grids.

3.1.2. 2D flow over an airfoil

In contrast to the circular cylinder, airfoils have sharp trailing edges of infinite curvature at the tip of the trailing edge and this makes the airfoil cases more challenging for Cartesian and immersed boundary methods than body fitted meshes. Flow past an airfoil is, therefore, a somewhat hard albeit useful test case to assess the performance of our method. The computational domain setup is similar to the one in Section 3.1.1, with four levels of re-

finement constructed dynamically. This leads to an effective mesh spacing of $h = 1/512$ at the finest level.

Numerical solutions for two different cases were computed: (I) a flow at $M = 0.05$ & $Re = 2 \times 10^3$ over a NACA0008 at an angle of attack of 4° and (II) flow at $M = 0.05$ & $Re = 10^4$ over a NACA0018 at an angle of attack of 10° . In both cases, the Reynolds number is computed based on the free stream flow and the airfoil chord length. A brief discussion about the validity of assuming two dimensional flow and even more challenging topic of resolving all turbulent scales at these high Re flows was presented earlier in Section 2.6. The mean drag and lift forces are compared with other body-conforming and immersed boundary results in Table 3.

For the case I (NACA0008 airfoil) the mean hydrodynamic forces are compared with the results of Kunz & Kroo [61] and Mittal et al. [62]. For case II (NACA0018 airfoil) solutions that were obtained with different mesh resolutions, are compared with a fourth order solution obtained by body conforming structured mesh developed by Zhang & Samtaney¹ [63]. The finest mesh spacings are $h = (1/128, 1/256, 1/512, 1/1024)$, denoted by mesh A, B, C and D respectively.

Effect of the mesh resolution on the solution accuracy at the FSI (shown in Figure 12) is examined by comparing the present distributions of the mean pressure coefficient C_p with results from Zhang & Samtaney. By increasing the mesh resolution, our present results approach the Zhang & Samtaney result. Nevertheless, the difference at the trailing edge is still large. This

¹The method proposed by Zhang & Samtaney [63] was used to obtain the present two dimensional results although in their original paper 2D results were not reported.

Figure 12: Mean pressure coefficient distributions for flow past NACA0018 airfoil at $Re = 10^4$ and $AOA = 10^\circ$, obtained using effective mesh spacing of $h = (1/128, 1/256, 1/512, 1/1024)$ which are denoted by present A, B, C and D, respectively. The results are compared with a fourth order body fitted mesh based result by Zhang & Samtaney [63].

difference may be due to the fact that our method is nominally second order accurate while the results from a body-fitted fourth order method utilizes higher resolution near the airfoil surface. The capability to have large aspect ratio cells with small wall normal width is a known advantage of the boundary conforming mesh methods over the Cartesian methods. Moreover, the trailing edge with infinite curvature gets regularized when represented on a Cartesian mesh.

As suggested before, the problem of low solution accuracy of the Cartesian methods at the FSI can be overcome by aggressive local refinement using the AMR capability of the current method. This is demonstrated in Figure 12, in

Figure 13: Instantaneous plot of the vorticity field for flow over a NACA0018 airfoil at $Re = 10^4$, $AOA = 10^\circ$, at non-dimensional time $t = 280$.

Figure 14: Visualization of the wake evolution for about $50 \times C$, where C is the airfoil chord length, behind the airfoil trailing edge obtained by simulating low Mach number flow past NACA0018 airfoil at $Re = 10^4$, $AOA = 10^\circ$ for 280 non-dimensional time units.

which pressure coefficient computed using the highest mesh resolution of $h = 1/1024$ approaches the reference solution at the trailing edge. Another point of minor disagreement with the body conforming solution is the mismatch between the C_p result in the constant pressure region on the upper surface of the airfoil (approximately in the region $0.2 \leq x \leq 0.4$). This is related to the separation bubble on the top surface in this high angle-of-attack flow. As far as the wake of the airfoil is concerned, the present method can efficiently and more accurately resolve the wake solution, which is important for many practical applications. The capability of predicting the wake pattern by our method is illustrated by the instantaneous plot of the span-wise vorticity field

in Figure 13; which demonstrates the wake structure. Two alternating rows of vortex pairs is observed. The complexity of the wake configuration can be seen by visualizing larger part of the wake as in Figure 14. Body-fitted meshes have long aspect ratio cells in the wake thereby loosing accuracy and vortex structures are not nearly as well-resolved as in our present method.

3.2. Three-dimensional flow tests

3.2.1. 3D flow over circular cylinder

In this test, flow past the circular cylinder at $Re = 300$ is simulated on a $[64D \times 16D \times 6D]$ domain. Low-level random white noise perturbations, with magnitude less than or equal to $0.1U_\infty$, are introduced to the stream-wise velocity component in order to excite three dimensional perturbations. The introduction of these perturbations is not essential as numerical round-off is itself sufficient to trigger the three dimensional solution; however this requires more time steps for the flow to achieve a fully developed three dimensional state. Figure 15, shows an instantaneous iso-surface of the stream and span-wise components of the vorticity field. The boundary conditions in the transverse direction are zero normal gradient whereas periodic boundary conditions are imposed in the span-wise direction. The finest level has mesh spacing of $h = 1/32$ based on two fixed levels that is refined four times with refinement ratio of $r = 2$. The resolution of the base mesh is $[128, 32, 12]$ (stream-wise, cross-stream, span-wise), while the effective resolution (of the finest mesh) is $[2048, 512, 192]$. A brief discussion of the parallel implementation and scalability evaluation results is given in Section 2.5.1.

The present result is verified by computing the Strouhal number and the hydrodynamic forces, and comparing them with other reported data in the

literature as listed in Table 4. We believe that slight overestimation of the root mean square value of the lift coefficient in our computation is a result of low-resolution.

Figure 15: The iso-surfaces plot of x- and z-component of the vorticity field for flow over a circular cylinder with $Re = 300$ and $M = 0.05$, at non-dimensional time $t = 300$.

3.2.2. 3D flow over a sphere

In this test a three dimensional flow pattern is generated by flow past a three dimensional sphere. The flow Reynolds number based on the sphere diameter is $Re = 300$. The computational domain size is $[64D \times 32D \times 32D]$. The base mesh resolution is $[128, 64, 64]$ (stream-wise, cross-stream, span-wise), refined three times with refinement ratio of $r = (4, 4, 2)$ for an effective resolution of $[4096, 2048, 2048]$.

A verification of our result is given in Table 5. The non-zero lateral force (in the cross-stream direction) is due to the asymmetrically vortex

shedding as in Figure 16. It is established in the literature (e.g. see Johnson and Patel [69] and Mittal [70]) that the incompressible viscous flow past a sphere remains steady and axisymmetric up to $Re \approx 210$ beyond which the first transition occurs. The flow at Reynolds number between $Re \approx 210$ to $Re \approx 270$ non-axisymmetric steady. By exceeding the $Re \approx 270$ limit, a periodic non-axisymmetric vortex shedding is observed.

Figure 16: Contours of span-wise component ω_z of the vorticity field superimposed with iso-surfaces of Q-criterion showing the vortical structure for flow past a sphere at $Re = 300$, and $M = 0.05$, at non-dimensional time $t = 180$ (Part of the simulation domain is visualized).

4. Results: nearly incompressible flows with moving boundaries

4.1. Oscillating circular cylinder

4.1.1. In-line oscillating circular cylinder

To examine the robustness of our algorithm in simulating flow passing moving objects, a number of tests are performed for moving solid objects. We first focus our attention on a circular cylinder, oscillating horizontally as

a rigid body in a stationary flow field. The oscillating motion is expressed as:

$$x(t) = -A_e \sin(2\pi f_e t), \quad (46)$$

where a and f are, respectively, the oscillation amplitude and frequency. The main parameters that determine the flow dynamics are the Reynolds number $Re = U_{max} D \rho / \mu$ and the Keulegan-Carpenter number $KC = U_{max} / f_e D$, where, $U_{max} = 2\pi f_e A_e$ is the cylinder maximum velocity and D is the cylinder diameter. Following the experimental and computational results of Dütsch et al. [72], the parameters are set to $Re = 100$ and $KC = 5$. The simulation is performed in a computational domain of size $[128D \times 128D]$ with base (coarsest) mesh resolution of $[1024 \times 1024]$. Four levels with refinement ratios of $(2, 4, 4, 2)$ were added to refine the mesh in the area surrounding the cylinder, so that the resulting effective resolution at the fluid-solid interface has a mesh spacing of $h = 1/512$. Free boundary modeled by Neumann boundary condition is imposed at the computational domain boundaries, and no-slip boundary condition are imposed at the solid object boundary.

Plots of the vorticity and pressure contours at a number of phase positions $\theta = 2\pi f_e t = (0^\circ, 96^\circ, 192^\circ, 288^\circ)$ are shown in Figure 17. We observe two symmetric vortices developed on the opposite side of the motion. When the cylinder reverses its motion direction, the attached vortices are separated away and new vortex pair are generated on the other side creating a wake reversal in agreement with established results [72].

The experimental and computational results were extracted from Dütsch et al. [72] and compared with our results. First, the velocity components (u, v) for three different phase position: $\theta = (180^\circ, 210^\circ, 330^\circ)$, are plotted at

Figure 17: Instantaneous vorticity (left) and pressure (right) of the in-line oscillating cylinder in a stationary flow at $Re = 100$ and $KC = 5$ obtained at different phase position ($\theta = 2\pi f_e t$). (a) $\theta = 0^\circ$, (b) $\theta = 96^\circ$, (c) $\theta = 192^\circ$ and (d) $\theta = 288^\circ$.

four different x -locations, $x = (-0.6D, 0.0D, 0.6D, 1.2D)$, and validated with the experimental measurement in Figure 18. We note the good agreement of our simulation results with those of [72] results. The hydrodynamic force over a full oscillation cycle is verified with Dütsch et al. [72] and results are shown in Figure 19.

4.1.2. Transversely oscillating circular cylinder in a uniform flow

For an additional assessment of our proposed algorithm for moving boundaries cases, we present results from a simulation of a uniform flow past a transversely oscillating circular cylinder. The physical setup is similar to the 2D circular cylinder case in Section 3.1.1. The computational domain size is 256×64 , and the base mesh resolution is $[512 \times 128]$ with three levels of refinement each with a refinement factor of four. The simulation was performed with Mach number of $M = 0.05$ and $Re \equiv (\rho_\infty u_\infty D / \mu) = 185$. The cylinder oscillates in the transverse direction with the following prescribed motion:

$$y(t) = A_e \cos(2\pi f_e t), \quad (47)$$

where A_e is the oscillation amplitude, and fixed for all tests at $A_e = 0.2D$; and f_e is the oscillation frequency. Simulations were performed for different f_e chosen such that $f_e/f_o = (0.8, 0.9, 1.0, 1.1, 1.12, 1.2)$, where f_o is the natural shedding frequency of a stationary circular cylinder.

Plots for the time evolution of the drag and lift coefficients are shown in Figure 20, for different oscillation frequencies. The present drag/lift results compare well with the computational results based on the immersed boundary method of Yang & Balaras [19] and the body fitted results of Guilmineau & Queutey [73].

Figure 18: Comparison of the velocity field components, u (left) and v (right), for in-line oscillation cylinder in a stationary flow at $Re = 100$ and $KC = 5$. The flow is measured at three different phase position: (a) $\theta = 180^\circ$, (b) $\theta = 210^\circ$, (c) $\theta = 330^\circ$. The data are collected at four different x -location, $x = (-0.6D, 0.0D, 0.6D, 1.2D)$. The present results (shown by lines) are compared to the experimental results (denoted by symbols) of Dütsch et al. [72].

Figure 19: The in-line force F_x acting on the in-line oscillation cylinder in a stationary flow at $Re = 100$ and $KC = 5$ compared with the computational results of Dütsch et al. [72] (shown by symbols).

The distribution of pressure and skin friction coefficients along the solid interface were measured when the cylinder is at its extreme upper position. A comparison with the Guilmineau & Queutey body-fitted results [73], quantitatively verifies our results as shown in Figures 21 & 22.

In Figure 23, we examine the effect of the oscillation frequency on mean value of the drag and lift coefficient and the phase difference between the lift coefficient and cylinder vertical displacement. The results are in good agreement compared with previous results in the literature [19, 73].

4.2. Flow over a pitching airfoil

We continue on the quest of validating and verifying our algorithm by increasing the complexity of the test case. We now examine the case of flow past a pitching NACA0012 airfoil at $Re = 1.2 \times 10^4$ and $M = 0.05$. The airfoil center of rotation is located at quarter chord length from the leading

Figure 20: The time evolution of the drag coefficients (solid curve) and lift coefficients (dotted curve) transversely oscillating circular cylinder at different oscillation frequency f_e : (a) $f_e/f_o = 0.8$, (b) $f_e/f_o = 0.9$, (c) $f_e/f_o = 1.0$, (d) $f_e/f_o = 1.1$, (e) $f_e/f_o = 1.12$, (f) $f_e/f_o = 1.2$.

Figure 21: The effect of the oscillation frequency f_e on the distribution of pressure coefficient, depicted by the solid curve, along the cylinder interface obtained when the cylinder is at its extreme upper position. The angle θ is measured clockwise from the stagnation point. The results are compared with Guilmineau and Queutey [73] (shown by symbols).

Figure 22: The effect of the oscillation frequency f_e on the distribution of skin friction coefficient, depicted by the solid curve, along the cylinder interface obtained when the cylinder is at its extreme upper position. The angle θ is measured clockwise from the stagnation point. The results are compared with Guilmineau and Queutey [73] (shown by symbols).

Figure 23: (a) The mean value of the drag coefficient, and the root mean square values of the drag and lift coefficients for flow over an oscillating circular cylinder at $Re = 185$, (b) The phase difference between the lift and coefficient and cylinder displacement. The present results are compared with the Yang & Balaras immersed boundary [19] and the Guilmineau & Queutey body-fitted [73] results.

Figure 24: The variation of mean value of the thrust coefficient C_T with reduced frequency $\hat{f}$ compared with experimental measurement of Koochesfahani [74] and the computational results of Ramamurti & Sandberg [75] and Young & Lai [76].

edge and the time dependent oscillation angle is given by:

$$\theta(t) = \theta_e \sin(2\pi f_e t), \quad (48)$$

where the pitching amplitude is fixed at $\theta_e = 2^\circ$, and the pitching frequency f_e is selected such that the reduced frequency ($\hat{f} = 2\pi f_e C/2u_\infty$) is in the range: $\hat{f} \in (0, 14)$. The thrust coefficient is obtained simply by inverting the drag coefficient sign ($C_T = -C_D$). A plot of the thrust variation with reduced frequency $\hat{f}$ is shown in Figure 24, and compared with other experimental and computational results.

The agreement between numerical simulations and experiments is good at lower values of the reduced frequency. On the other hand, for higher values of $\hat{f}$ the numerical simulations underestimate the thrust coefficients. As explained by Ramamurti & Sandberg [75], a probable cause for this discrepancy is that the results of three dimensional tunnel walls are neglected in the two dimensional numerical results. Moreover, it is very likely that the

flow fields in experiments were three-dimensional and turbulent; we cannot capture turbulence effects in a two-dimensional simulation. We note that the thrust/drag coefficient was experimentally [74] estimated by measuring the momentum gain/deficit at far downstream position by:

$$C_T = \frac{2}{C} \int_{-\infty}^{\infty} \frac{u(x, y)}{U_{\infty}} \left[\frac{u(x, y)}{U_{\infty}} - 1 \right] dy, \quad (49)$$

where x is a downstream position, C is the chord length and U_{∞} is the free-stream velocity. The pressure and fluctuating quantities are ignored which may be another possible reason for the discrepancy of the thrust coefficient between the numerical and experimental results.

Visualization of the von Kármán vortex street wake produced by the oscillatory motion of the airfoil at different frequencies is shown in Figure 25; starting with the fixed airfoil case $k = 0$ (Figure 25(a)), in which a drag-wake pattern appears to consists of two rows of alternating vortices, rotating clockwise on the upper row and counterclockwise in the lower one. Complex wake patterns of vortex dipoles evolve by increasing pitching frequency to about $k = \pi$. The upstream-tilted vortex pairs configuration is a characteristic for a drag-wake. Untilted vortex dipoles of neutral-wake conflagration start appearing by at about pitching frequency $k = 1.5\pi$ at which the thrust developed by the airfoil oscillatory rotation overcomes the generated drag. Finally, downstream-tilted vortex pairs corresponding to the thrust-wake are obtained at higher pitching frequencies. In these latter cases, the two rows of alternating vortex pattern are retrieved but with counterclockwise rotation on the upper row and the opposite in the lower one as shown in Figure 25(h). Such patterns were observed by experimental and numerical simulations for plunging airfoils [77, 76].

Figure 25: Vorticity field for pitching airfoil at different frequencies: (a) $k = 0$, (b) $k = 0.5\pi$, (c) $k = 1.0\pi$, (d) $k = 1.5\pi$, (e) $k = 1.75\pi$, (f) $k = 2.0\pi$, (g) $k = 2.25\pi$, (h) $k = 3.0\pi$,

Figure 26: The relative error in the drag coefficient C_D (left) and the root mean square value of the lift coefficient C_L (right) for flow over a circular cylinder at $Re = 100$, and Mach number $M = 0.05$. The dashed lines are a linear fit for the error with a slope of 1.79 for the drag coefficient and 1.72 for the lift coefficient. Reference lines with slopes of one and two are added for comparison.

5. Convergence tests

To examine the numerical accuracy of our method, we perform two convergence tests. In the first one, the convergence rate of the drag and lift coefficients for the flow over a circular cylinder, presented earlier in Section 3.1.1, is examined. In the second test we examine the convergence of numerical solutions of an oscillating circular cylinder in ambient fluid, similar to that studied in Section 4.1.1.

5.1. Convergence test for a static cylinder

For the flow past a cylinder at $Re = 100$, the drag and lift coefficients in Table 2, are taken as a reference. The error is computed by evaluating the relative difference ($\epsilon_D^n, \epsilon_L^n$) between the reference values and ones computed

on lower resolution meshes according to:

$$\epsilon_D^n = \frac{C_D^r - C_D^n}{C_D^r}, \quad (50)$$

where C_D^r and C_D^n are, respectively, the drag coefficients based on the reference and coarser mesh. The relative error of the lift coefficient is computed in a similar manner. The finest mesh spacing in the reference solution is $\Delta x^f = 1/512$. The mesh spacing is coarsened four times by a factor of two to give $\Delta x^n = [1/256, 1/128, 1/64, 1/32]$. By transforming the pair of the measured errors with its mesh spacing to log-scale: $(\log(\Delta x^n), \log(\epsilon^n))$, the convergence rate is, then, approximated by computing the slope of a fitted linear polynomial to the transformed data. Figure 26, shows the variation of the measured error $\log(\epsilon^n)$, versus mesh spacing $\log(\Delta x^n)$. The dashed lines is a linear fit (on the log-log scale) with a slope of $\omega_D = 1.79$ for drag and $\omega_D = 1.72$ for the lift coefficient.

5.2. Convergence test for an oscillating cylinder

To test convergence for the case of the oscillating circular cylinder, we employ a periodic two dimensional domain for convenience. The oscillation amplitude is a is equal to $D/2$, and the oscillation frequency is chosen such that the cylinder maximum speed U_{max} equals unity. The flow is simulated with fixed Reynolds number $Re = 100$, and Mach number $M = 0.05$. The computations are carried out on a single level mesh (i.e. with no mesh refinement) with a domain size of $[3D \times 3D]$. Clearly, this small domain size coupled with periodic boundary conditions in both directions makes this a somewhat unphysical test case. Nonetheless, this setup is reasonable to compute the solution accuracy of our algorithm. A solution obtained with

grid size of $[1260 \times 1260]$ is chosen as our reference solution. Then, solutions were computed on coarser grids coarsening the reference grid by an odd factor to produce the grid sizes of $[420 \times 420]$, $[252 \times 252]$, $[180 \times 180]$ and $[140 \times 140]$. This set of mesh resolutions is selected in order to avoid interpolation from the finer to coarser mesh, since the cell centers in each grid coincide with next coarser and finer grid in the mesh sequence. The CFL condition is fixed as well; thus, the time increment (Δt) is refined by the same factor. The flow solutions for all of the grids were computed for one time period $T = 1/f$. To assess the effect of the order of accuracy of PDE interpolation algorithm utilized to extrapolate the solution into the ghost fluid region, we tested both linear and quadratic PDE interpolations.

The solution error ϵ^n for grid level n is computed pointwise as the difference between the corresponding grid solution and the reference grid solution U^f by:

$$\epsilon^n = U^f - U^n. \quad (51)$$

Plots of error norms (L_1 , L_2 and L_∞ norms) of the solutions components in log-log scale are shown in Figure 27. Estimation for the convergence rates were obtained by computing ω_q , such that ω_q is the slope of linear polynomial f_q fitted to the q -norm on a log-log scale: $(\log(\Delta x), \log(L_q(\epsilon)))$; where $q = (1, 2, \infty)$. The convergence rates are presented in Table 6. These results show the effect of the reconstruction accuracy on the entire simulation accuracy. The convergence rates are between 1.4 to 1.6 when linear PDE interpolation is utilized in the reconstructing step. However, by extending the solution into the ghost fluid region by quadratic PDE interpolation the computed temporal-spatial physical solution show approximately second order global

Figure 27: L_1 , L_2 and L_∞ norms of the error in the density, pressure and velocity fields computed from the solutions of oscillating circular cylinder. First order PDE interpolation: (a) density ρ , (c) velocity u, v , (e) pressure p . Second order PDE interpolation algorithm: (b) density ρ , (d) velocity u, v , (f) pressure p .

and local convergence rates.

5.2.1. Mass conservation error

Embedded boundary methods with an implicit representation of the FSI as in our case with a levelset method have a small amount of mass, and energy leakage. Here we analyze the level of lack of mass conservation in our method by examining the total mass change with time for the oscillating cylinder case. Regardless of the fact that the reconstruction algorithm is not discretely conservative, however, it is expected that the total amount of mass accumulation or reduction ought to be negligible. Furthermore, we expect the mass conservation error to decrease as the mesh resolution increases. To evaluate the error in mass conservation, we recompute the solution of the oscillating cylinder problem, discussed above in Section 4.1.1, for five oscillation periods $t = 5T$, in computational domain of $[2D \times 2D]$ with single mesh level; and different grid resolution of $h = (1/50, 1/100, 1/200, 1/400)$. Because of the periodic boundaries, the total mass in the computational domain should not change in time. Thus, the relative mass change (Δm) is computed by taking the difference between the total mass in the current time level and the total mass in the initial time level:

$$\Delta m(t) = \left(\int \rho(t) dA - \int \rho(0) dA \right) / \int \rho(0) dA. \quad (52)$$

Figures 28(a,b), show relative mass changes computed at each midpoint and end of the cylinder oscillation period (T), that is, at times $t = nT/2$, for $n = 0, 1, \dots, 10$. The results are obtained by different mesh resolutions with quadratic and constant PDE extrapolation methods. The relative mass change for different mesh resolutions at $t = 5T$, in Figure 29, shows an

Figure 28: The relative mass change for oscillating cylinder in stationary flow, computed at the midpoint and end of each oscillation period in the time interval $t \in [0, 5T]$, obtained by (a) quadratic PDE reconstruction, (b) constant PDE reconstruction.

approximate a second order convergence rate for the total mass difference obtained quadratic PDE extrapolation, while a first order convergence rate with the constant PDE extrapolation.

6. Results: compressible flows with fixed boundaries

In this section, we present results from examples where compressibility effects (e.g. as in shocks) are important. We begin with the simple shock-tube problem misaligned with the grid and increase the complexity of the test case with each example. These examples further illustrate the versatility of our method in that we have examined low Mach number (or incompressible) flow in the previous sections and now we examine high Mach number and supersonic flow cases *without* any modifications to the implemented code.

Figure 29: Convergence test for the relative mass difference obtained at $t = 5T$: (a) quadratic PDE reconstruction with an approximately second order convergence rate, (b) constant PDE reconstruction with a first order convergence rate.

6.1. Two dimensional flow tests

6.1.1. Shock-tube problem

In the first test, a Riemann problem was solved in a rotated shock tube enclosed in a Cartesian domain. Similar to [78], we initialize the problem with the following left/right states:

$$\{\rho_L, u_L, p_L\} = \{1, 0, 1\}, \quad (53)$$

$$\{\rho_R, u_R, p_R\} = \{0.125, 0, 0.1\} \quad (54)$$

As shown in Figure 30, the tube has length of 1, and height of 0.1 units and rotated 5 degree counterclockwise. The computational domain has a horizontal length of 1.05 and a quarter of that in the vertical dimension. The base mesh size is $[512 \times 128]$. Two levels of mesh refinement with refinement ratio of $[4, 2]$ provide an effective resolution of $[4096 \times 1024]$: the mesh is always refined at the shock tube boundary and in high density gradient region. In

Figure 30: The initialization and domain setup for the rotated shock tube problem.

this case, we use a slip with no normal mass flux at the solid wall boundaries. At time $t = 0.1$, the density field, and velocity magnitude contours are shown in Figure 31. The solution is extracted along the tube center line and compared with the exact solution in Figure 32. To evaluate the effect of the rotation angle, the obtained solution is compared for rotation angles 0° , 5° , 45° in Figure 33. The results demonstrate no appreciable differences in the solutions by varying the tube orientation.

Figure 31: Density solution for the Riemann problem in rotated shock tube by $\theta = 5^\circ$. The shock and rarefaction waves are depicted by black contours of the velocity magnitude.

By examining closely the solution in the vicinity of the contact discontinuity interface, a spurious feature is observed. The contact discontinuity is

Figure 32: (a) The density and (b) velocity magnitude at the tube center line, which is obtained by rotating the shock tube by 5 degree, compared with the exact one dimensional solution of the density and velocity.

subject to an impulsive acceleration and small perturbations stemming from an inexact representation on an oblique mesh may give rise to the Richtmyer-Meshkov instability, and theoretically such perturbations grow linear in time as long as the perturbation amplitude is much smaller than their wavelength. These spurious effects are, however, reduced by increasing the mesh resolution. Figure 34, compares the contact discontinuity solutions obtained with effective resolution of $[4096 \times 1024]$ and higher effective resolution of $[16384 \times 4096]$ at two different time instant $t = 0.075$ and $t = 0.3$.

6.1.2. Shock reflection at a wedge

In this case, we examine the reflection of a Mach number $M = 7.1$ shock wave reflection at a wedge inclined at 49° . Defining $R_m = L_M/L_R$, as ratio of the Mach stem length L_M to the shock distance L_R in Figure 35, Galz et al. [79] found that the experimental Mach stem length is smaller than the

Figure 33: (a) The density and (b) velocity magnitude at the tube center line obtained by rotating the shock tube by $(0^\circ, 5^\circ, 45^\circ)$.

one obtained in their inviscid computation.

In inviscid and viscous simulations utilizing a cut-cell embedded boundary approach, Graves et al. [80] showed that the shock interaction with the boundary layer leads to that significant decrease in the Mach stem length. We simulate the same case and compute both inviscid and viscous solutions. The physical parameters of this case are given in [80]. The non-dimensionalization is performed by choosing the left sound speed c_L , density ρ_L and the tube length. The domain setup is illustrated in Figure 35, and parameters presented in Table 7.

The physical tube domain is embedded in rectangular computational domain of size $[1.3 \times 0.71]$ discretized with a base mesh resolution of $[1280 \times 700]$ and with four levels each at a refinement ratio of 2. The effective resolution is, hence, $[20480 \times 11200]$; and effective mesh spacing of $h = 6.348 \times 10^{-5}$. The mesh is refined in the vicinity of the immersed boundary as well as in

Figure 34: A closeup view for the contact discontinuity at two different time instants (a) $t = 0.075$ and (b) $t = 0.3$. The effective resolution of the left plot is $[4096 \times 1024]$ while it is $[16384 \times 4096]$ for the right plot in each of the sub-figures (a) and (b)

regions of high density gradient. Instantaneous plots for the density field for the viscous and inviscid cases at time $t = 10.01\mu s$ are shown in Figure 36. The R_m ratios are computed for both inviscid and viscous cases at time instant $t = 9.61\mu s$ and tabulated in Table 8, and compared with the results of Graves et al. [80] and Glaz et al. [79]. Our inviscid result for R_m agrees well with the reported numerical results, whereas the numerical viscous results, on the one hand, slightly underestimate the R_m ratio. The source of this difference cannot be clearly identified, and due to the lack of details in the experimental setup and measurement details; such a lack of detail was also noted by Graves et al. [80].

6.1.3. Supersonic flow over circular cylinder

In this test we examine supersonic flow past a cylinder at $M = 1.7$, and Reynolds number $Re = 2 \times 10^5$. As in the previous cases, the mesh is dynam-

Figure 35: Schematic of the physical setup for simulating shock reflecting from wedge inclined by 49° . The Mach stem length L_M and the shock distance L_R are shown schematically for an inviscid shock reflection.

ically adapted based on the magnitude of the density gradient. The size of the computational domain is $[64D \times 32D]$ discretized with a base mesh size of $[256 \times 128]$ and four level of refinements with refining ratios of $(4, 4, 4, 2)$ in regions of high density gradient and at the cylinder surface. Figure 37, depicts instantaneous density and Mach number fields, obtained after relatively long time period such that the shock wave propagation reaches its steady state position. The present values of the separation angle, shock standoff distance and the drag coefficient are tabulated in Table 9, and compared with other reported results in the literature. A good agreement between our computed separation angle and the shock standoff distance can be seen in a comparison with other computational [81] and experimental results [82].

The pressure coefficient C_P can be examined in Figure 38. This result was smoothed by a second order low-pass filter implemented by applying a two consecutive first order low pass filter, in a similar manner to equation

(42):

$$\tilde{p}(\theta) = \int_0^{2\pi} g_1(\theta - \alpha) \left(\int_0^\alpha g_2(\alpha - \beta) p(\beta) d\beta \right) d\alpha, \quad (55)$$

where the cutoff frequencies for the g_1 and g_2 kernels are $k_c^1 = 0.2$ and $k_c^2 = 0.3$. A agreement is noted with other documented computational and experimental results in the literature. We cautiously note that the flow field in the wake in the experiment is most likely three dimensional whereas our results are confined to 2D.

6.2. Three-dimensional flow tests

6.2.1. Supersonic flow past a sphere

To further examine the applicability of our algorithm, a three dimensional compressible flow example is now examined. A supersonic flow at Mach number $M = 2$ and Reynolds number $Re = 6.5 \times 10^5$ past a sphere is simulated. A large computational domain size of $[128D \times 64D \times 64D]$ is discretized with base mesh resolution of $[128, 64, 64]$ in (stream-wise, cross-stream, span-wise). Five refinement levels, locally refine the mesh by refinement ratios of $r = (4, 2, 2, 4, 2)$. This is equivalent to an effective unigrid resolution of $[16384, 8192, 8192]$.

The solution is validated and verified by comparing the computed drag to the experimental result of Bailey and Hiatt [84], and the computational embedded boundary results of Uddin et al. [30], respectively, in Table 10. The shock standoff distance is also presented in the same table. A visualization of the flow vortical structure, in Figure 39, shows an iso-surface of the span-wise component of the vorticity superimposed with the color contours

of the density field in the background illustrating the shock wave in front of the sphere.

7. Results: compressible flows with moving boundaries

7.1. Inflatable shock tube

We culminate our series of tests with the case of a compressible flow interacting with a moving solid boundary. Specifically, we simulate the dynamic interaction of an inflatable tube interaction with a moving shock wave. In this test, a shock at $M = 3$ enters a two dimensional tube with flexible walls [32, 85] (see Figure 40 for an illustration of the physical setup). The tube length is $L_T = 0.85$ and its height is $H_T = 0.2$. It is attached to the left boundary of a unit square computational domain where the tube has a uniform inflow boundary. The remaining boundaries are comprised of flexible beams in the upper and lower boundaries, each with a length of $L_B = 0.6$ capped on the right by a rigid section of length 0.05 and a reflecting wall. The base mesh size is $[200 \times 200]$, that is refined twice by ratios of (4, 2). Both of inviscid flow and viscous ($Re = 5000$) flows are considered in this test case.

The beams dynamics are modeled by Euler-Bernoulli equation as follows:

$$\rho A \frac{\partial^2 y}{\partial t^2} + EI \frac{\partial^4 y}{\partial x^4} = f, \quad (56)$$

where the Young's modulus is chosen to be $E = 10^7 \text{ Pa}$, the second moment of area is $I = 10^{-8} \text{ m}^4$, the cross-sectional area is $A = 5 \times 10^{-3} \text{ m}^2$ and the density is $\rho = 250 \text{ Kg/m}^3$. The values of these parameters are selected to allow for large beam deformation based on the initialization and the pressure

solution in the tube. Moreover, our selected parameter set is similar to that chosen by Arienti et al. [32] in order to compare with their results. The beams ends are assumed to be simply supported so that zero deflection and zero bending moment conditions have to be satisfied at the beams ends. Mathematically, the Euler-Bernoulli equation are subjected to the following boundary conditions:

$$y(0, t) = y(L_B, t) = 0 \quad (57)$$

$$\frac{\partial^2}{\partial x^2} y(0, t) = \frac{\partial^2}{\partial x^2} y(L_B, t) = 0. \quad (58)$$

Initially, the beams are at equilibrium state, that is:

$$y(x, 0) = \frac{\partial}{\partial t} y(x, 0) = 0. \quad (59)$$

The force term is computed based on the difference between inner and outer pressure acting on the tube walls. The outer pressure is constant in time; and equal to the initial pressure of the unshocked gas in front of the shock, maintaining the beams in force balance initially. The Euler-Bernoulli equation are numerically solved with a central and backward difference discretization for the spatial and temporal terms, respectively. Numerically, this results in a penta-diagonal linear system. The reason for the implicit time stepping is to synchronize the time increments of both the fluid and solid solver.

A visualization of the density field and the tube walls deflection at the non-dimensional time $t = (0.1, 0.3, 0.5)$ are given in Figure 41, for both viscous and inviscid solutions. As the shock propagates down the tube, the pressure rise behind the shock exerts a force on the beam which deflects in response. After the shock reflection off the tube end, there is an additional

increase in the pressure, and a corresponding larger deflection of the tube walls. A quantitative comparison with Arienti et al. [32] is precluded due to a lack of details of this study (e.g. they did not specify the Young's modulus and other parameters). Nonetheless, the results are in qualitative agreement.

The beam deformation at different time instants is shown in Figure 42(a), with the results from both inviscid and viscous simulations overlaid in the same plot. The beam solution is almost identical for both cases during the first half of the simulation time. At late times, the difference between the viscous and inviscid cases becomes discernible with the larger deformation in the inviscid case attributed to smaller dissipation in this case compared with the viscous case. The accuracy of the beam solution is considered by examining the beam solutions at $t = 0.5$ for the inviscid case for different mesh sizes in which the finest mesh spacing is varied as $h = (1/400, 1/200, 1/100, 1/50)$. The L_2 and L_∞ norms for the differences between the highest mesh resolution and the other solutions were computed to evaluate the rate of the convergence and the plot in Figure 42(b), indicates a first order convergence attributed to the first order backward Euler temporal discretization.

8. Conclusion

In this study we discussed our implementation of an embedded AMR method to solve the compressible Navier Stokes equations in arbitrary complex geometry. The developed algorithm is, basically, a multi-stage predictor-corrector reconstruction scheme. The algorithm is started by advecting the solution from the physical fluid domain cells, that are adjacent to the fluid-

solid interface, to the ghost fluid cells. A linear or quadratic PDE multidimensional extrapolation scheme is used in this step. Next, the solution is evaluated at image points, which is located as mirror points to the ghost fluid cell-centers, by performing an IDW interpolation procedure. In this phase, a smooth solution is obtained at the image points utilizing the nearby physical and recently populated ghost fluid points. Finally the solution is re-extrapolated (corrected) to the ghost fluid point from the image point such that the boundary condition at the fluid-solid interface is satisfied. The advantages of the PDE extrapolation are its relatively simple implementation and generalization to a higher order accuracy. Moreover, the PDE algorithm is not affected directly by the geometry but just require computing the levelset signed distance function which in itself is a relatively straightforward task. The IDW interpolation method smoothens the reconstructed solution by performing the interpolation process using a large stencil from all of the spatial locations. This method is implemented using the block-based hierarchical AMR Chombo library. The fluid equations are solved with an unsplit second-order Godunov method. The advantage of using the AMR technique is that it allows for resolving all the scales of the solution. Moreover, errors at fluid-solid interface may be effectively reduced, and solution accuracy improved by the use of local refinement.

We demonstrated the versatility of the developed method for a wide variety of test cases ranging from low-Mach number nearly incompressible flows to high-Mach number supersonic flows involving shocks. In both incompressible and compressible tests we demonstrated the method with static and moving boundary cases. For each test case the emphasis was on verifi-

cation with established methods and validation using available experimental data where applicable. Our simulation test cases included both two and three dimensional flows. A clear novelty of the proposed method is in cases of moving geometry. The moving body results were successfully verified and validated by the good agreement and the capability of the adaptive meshes to resolve interesting flow features (such as the varied vortex patterns in the wake of the oscillating airfoil) was demonstrated. Convergence tests were preformed to assess the accuracy of the developed solver. We found that a local and global second order solution is obtained when high order quadratic PDE extrapolation is utilized in the reconstruction step. This convergence rate is slightly reduced to around 1.5 if linear PDE extrapolation is employed.

Our next step is to exploit our implementation is simulating more physical cases. Improving and optimizing the parallel performance of the method is an important issue to allow for tackling large scale physical problems. As we have only tested this solver in simulating laminar flow cases, studying and simulating turbulence flows is one of the our future plans. We plan to move forward in this direction by introducing a wall model and models for sub-grid stress in large-eddy simulations of turbulent flows.

Acknowledgment

This research was supported by the KAUST Office of Competitive Research Funds under Award No. URF/1/1394-01. The Cray XC40, Shaheen II, at KAUST was utilized for some of the simulations. We thank Dr. Nar-simha Rapaka for his assistance in computing the three-dimensional cases on Shaheen II.

Study	C_D	L_W	a	b	θ
Tritton [48]	1.59	–	–	–	–
Le et al. [49]	1.56	2.22	–	–	53.6
Dennis and Chang [50]	1.52	2.35	–	–	53.8
Coutanceau and Bouard [51]	–	2.13	0.76	0.59	53.5
Gautier et al. [52]	1.49	2.24	0.71	0.59	53.6
Chiu et al. [29]	1.52	2.27	0.73	0.60	53.6
Taira and Colonius [53]	1.54	2.30	0.73	0.60	53.7
Brehm et al. [54]	1.51	2.26	0.72	0.58	52.9
Present - A	1.54	2.25	0.71	0.59	53.7
Present - B	1.52	2.26	0.71	0.59	53.3
Present - C	1.55	2.26	0.71	0.59	52.9

Table 1: Physical characteristics of the flow past a circular cylinder at $Re = 40$, compared with other numerical results reported in the literature. C_D is the drag coefficient, L_W length of the attached vortices, (a, b) are the locations of the vortex centers and θ is the separation angle. The present results denoted “A” are obtained with four dynamic levels of refinements, “B” with three dynamic levels of refinement, and “C” with three static levels of refinement. The finest mesh resolution with four levels of refinement is $h = 1/512$, whereas $h = 1/128$ for three levels of refinement.

Study	C_D	C_L	S_t
Berger and Wille [57]	–	–	0.16 – 0.17
White [58]	1.46	–	–
Stålberg et al. [56]	1.32 ± 0.0090	± 0.330	0.166
Chiu et al. [29]	1.35 ± 0.0120	± 0.303	0.167
Le et al. [49]	1.37 ± 0.0090	± 0.323	0.160
Brehm et al. [54]	1.32 ± 0.0100	± 0.320	0.165
Russell and Wang [59]	1.38 ± 0.0070	± 0.300	0.172
Liu et al. [60]	1.35 ± 0.0120	± 0.339	0.165
Present - A	1.34 ± 0.0089	± 0.325	0.166
Present - B	1.33 ± 0.0085	± 0.319	0.164
Present - C	1.33 ± 0.0091	± 0.323	0.164

Table 2: Drag & lift coefficients and the Strouhal number of the flow past a circular cylinder at $Re = 100$, compared with other experimental and numerical results reported in the literature. The present results A obtained with four dynamic levels of refinements, B obtained with three dynamic levels of refinement, and C with three static levels of refinement. The finest mesh resolution for four levels of refinement is $h = 1/512$, while for three levels of refinement, it is $h = 1/128$.

Case	Study	C_D	C_L	S_t
I	Kunz & Kroo [61]	0.080	0.272	–
	Mittal et al. [62]	0.081	0.273	–
	Present	0.078	0.273	–
II	Present - A	0.146	0.248	0.726
	Present - B	0.191	0.543	0.695
	Present - C	0.217	0.817	0.703
	Present - D	0.223	0.860	0.698
	Zhang & Samtaney [63]	0.220	0.798	0.653

Table 3: Drag & lift coefficients of the flow over airfoils, Case I: NACA0008, $Re = 2 \times 10^3$, $AOA = 4^\circ$, Case II: NACA0018, $Re = 1 \times 10^4$, $AOA = 10^\circ$. The results A, B, C and D denote present results obtained using mesh resolution of $h = (1/128, 1/256, 1/512, 1/1024)$, respectively.

Study	C_D	$C_L(\text{rms})$	S_t
Ghias et al. [64]	1.23	—	0.21
Balaras [16]	1.27	0.42	0.21
Mittal and Balachandar [65]	1.26	0.38	0.21
Rajani et al. [66]	1.28	0.50	0.20
Kravchenko et al. [67]	1.28	0.40	0.20
Hartmann et al. [46]	1.26	0.39	0.20
Nam and Lien [68]	1.33	0.55	0.21
Present	1.23	0.58	0.20

Table 4: Drag coefficients, root mean square value of the lift coefficient and the Strouhal number for 3D flow over the circular cylinder at $Re = 300$, compared with other numerical results reported in the literature.

Study	C_D	C_L	S_t
Johnson and Patel [69]	0.656	0.069	0.137
Kim et al. [71]	0.657	0.067	0.134
Hartmann et al. [46]	0.657	0.069	0.135
Present	0.629	0.066	0.137

Table 5: The present results of the drag and lateral force coefficients of the flow past a sphere at $Re=300$, compared with other results reported in the literature.

Variable	ω_1		ω_2		ω_∞	
	$\mathcal{P}_1$	$\mathcal{P}_2$	$\mathcal{P}_1$	$\mathcal{P}_2$	$\mathcal{P}_1$	$\mathcal{P}_2$
Density (ρ)	1.55	1.91	1.59	1.94	1.52	1.66
Velocity (u, v)	1.60	1.92	1.60	1.98	1.42	2.06
Pressure (p)	1.56	1.96	1.59	1.96	1.47	1.90

Table 6: The convergence rates of the L_1 , L_2 and L_∞ norms of the solutions errors of circular cylinder oscillating in a stationary flow denoted by $(\omega_1, \omega_2, \omega_\infty)$, respectively. The rates are obtained by computing slopes of fitted linear polynomial to the logarithm of the data. The $\mathcal{P}_1$ and $\mathcal{P}_2$ denote the convergence rates obtained by using linear and quadratic PDE interpolation algorithm in the reconstruction step.

$p_L = 0.6$	$p_R = 1.577 \times 10^{-3}$	$Re = 8.283 \times 10^4$
$\rho_L = 1$	$\rho_R = 9.114 \times 10^{-2}$	$Pr = 35.588$
$u_L = u_R = 0$	$v_L = v_R = 0$	$\lambda = -2/3$
$y = 1/2$	$\theta = 49^\circ$	$\gamma = 5/3$
$x_1 = 3/5$	$x_2 = 2/3$	$x_3 = 1$

Table 7: The non-dimensionalized initial values and the tube specifications of the shock reflection test case.

Study	Inviscid	Viscous	Experimental
Glaz et al. [79]	0.070	—	0.038
Graves et al. [80]	0.072	0.030	—
Present	0.071	0.027	—

Table 8: Comparison of the ratio of Mach stem length to shock distance, $R_m = L_M/L_R$ measured for a shock reflected from a 49° wedge at dimensional time instant $t = 9.61\mu s$.

Figure 36: The density field for the inviscid (top) and viscous (bottom) shock reflected from 49° wedge at dimensional time instant $t = 10.01\mu s$. Only a portion of the computational domain is shown here; and the dynamically adapted mesh structure is shown on the right.

Figure 37: Top: Mach number variation (colored field) and the density field (black contours) for supersonic flow past a circular cylinder at $M = 1.7$ and $Re = 2 \times 10^5$. Bottom: a closer view of the cylinder showing the density field and the mesh structure.

Study	Separation angle	Shock standoff distance	C_D
Bashkin et al. [82]	112°	—	1.40
Tullio et al. [81]	113°	1.040	1.41
Nam & Lien [83]	113°	—	1.38
Present	113°	1.035	1.35

Table 9: The separation angle measured from the cylinder leading edge in the clockwise direction, and the shock standoff distance for supersonic flow over circular cylinder with $M = 1.7$ compared with reported experimental results [82] and computational [81, 83]. A possible reason for the underestimation of the drag coefficient in the present result is three-dimensional effects that are ignored in the present 2D simulation. Note that the other computational results listed were obtained by utilizing turbulence models.

Figure 38: Comparison for the present filtered pressure coefficient of the supersonic flow passing circular cylinder with $M = 1.7$ with the body-fitted of Bashkin et al. [82], experimental results from the former reference and result of immersed boundary method of Tullio et al. [81].

Study	Shock standoff distance	C_D
Bailey and Hiatt [84]	—	1.00
Uddin et al. [30]	1.80	1.07
present	1.83	0.96

Table 10: The present results of the drag and shock standoff distance of a supersonic flow past a sphere at $M = 2$ and $Re = 6.5 \times 10^5$, compared with other results reported in the literature.

Figure 39: Iso-surface of the span-wise component ω_z of the vorticity showing the vortical structure for supersonic flow past a sphere at $M = 2$ and $Re = 6.5 \times 10^5$. The background colored contour plot is the density field illustrating the shock wave in front of the sphere. (Part of the simulation domain is visualized.)

Figure 40: Schematic setup of an inflatable tube interaction with a shock wave.

Figure 41: The density field of the inflatable tube problem obtained at different time instant: (a) $t = 0.1$ (b) $t = 0.3$ (c) $t = 0.5$. The left figures depict viscous case solutions, whereas the right figures show the inviscid case solution. Only a portion of the simulation domain is visualized.

Figure 42: (a) The beam deflections at different times: the solid (resp. dashed) curves correspond to the inviscid (resp. viscous) solutions. (b) The convergence rate for the inviscid case solutions of the beam equation at $t = 0.5$. A convergence rate of approximately one is observed in the L_2 and L_∞ norms for the difference between the solution at the current resolution and a reference solution at the next higher resolution.

9. References

- [1] P. Berthelsen, O. Faltinsen, A local directional ghost cell approach for incompressible viscous flow problems with irregular boundaries, *Journal of Computational Physics* 227 (9) (2008) 4354 – 4397.
- [2] R. Verzicco, J. Mohd-Yusof, P. Orlandi, D. Haworth, Large-eddy simulation in complex geometric configurations using boundary body forces, *AIAA Journal* 38 (3) (2000) 427–433.
- [3] C. Peskin, Flow patterns around heart valves: A numerical method, *Journal of Computational Physics* 10 (2) (1972) 252 – 271.
- [4] R. Mittal, G. Iaccarino, Immersed boundary methods, *Annual Review of Fluid Mechanics* 37 (2005) 239–261.
- [5] C. Peskin, Numerical analysis of blood flow in the heart, *Journal of Computational Physics* 25 (3) (1977) 220–252.
- [6] C. Peskin, Fluid dynamics of heart valves: Experimental, theoretical, and computational methods., *Annual Review of Fluid Mechanics* 14 (1982) 235–259.
- [7] D. Goldstein, R. Handler, L. Sirovich, Modeling a no-slip flow boundary with an external force field, *Journal of Computational Physics* 105 (2) (1993) 354–366.
- [8] E. Saiki, S. Biringen, Numerical simulation of a cylinder in uniform flow: Application of a virtual boundary method, *Journal of Computational Physics* 123 (2) (1996) 450–465.
- [9] M.-C. Lai, C. Peskin, An immersed boundary method with formal second-order accuracy and reduced numerical viscosity, *Journal of Computational Physics* 160 (2) (2000) 705–719.
- [10] J.-F. Zou, A.-L. Ren, J. Deng, Study on flow past two spheres in tandem arrangement using a local mesh refinement virtual boundary method, *International Journal for Numerical Methods in Fluids* 49 (5) (2005) 465–488.
- [11] S.-W. Su, M.-C. Lai, C.-A. Lin, An immersed boundary technique for simulating complex flows with rigid boundary, *Computers and Fluids* 36 (2) (2007) 313–324.
- [12] D. Pan, An immersed boundary method for incompressible flows using volume of body function, *International Journal for Numerical Methods in Fluids* 50 (6) (2006) 733–750.
- [13] J. Mohd-Yusof, Combined Immersed-Boundary/B-spline methods for simulations of flow in complex geometries, *Annual research briefs, Center for Turbulence Research* (1997).

- [14] E. Fadlun, R. Verzicco, P. Orlandi, J. Mohd-Yusof, Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations, *Journal of Computational Physics* 161 (1) (2000) 35–60.
- [15] R. Verzicco, M. Fatica, G. Iaccarino, P. Orlandi, Flow in an impeller-stirred tank using an immersed-boundary method, *AIChE Journal* 50 (6) (2004) 1109–1118.
- [16] E. Balaras, Modeling complex boundaries using an external force field on fixed Cartesian grids in large-eddy simulations, *Computers & Fluids* 33 (3) (2004) 375 – 404.
- [17] A. Gilmanov, F. Sotiropoulos, E. Balaras, A general reconstruction algorithm for simulating flows with complex 3D immersed boundaries on Cartesian grids, *Journal of Computational Physics* 191 (2) (2003) 660–669.
- [18] A. Gilmanov, F. Sotiropoulos, A hybrid Cartesian/immersed boundary method for simulating flows with 3D, geometrically complex, moving bodies, *Journal of Computational Physics* 207 (2) (2005) 457–492.
- [19] J. Yang, E. Balaras, An embedded-boundary formulation for large-eddy simulation of turbulent flows interacting with moving boundaries, *Journal of Computational Physics* 215 (1) (2006) 12–40.
- [20] S. Majumdar, G. Iaccarino, P. Durbin, RANS solvers with adaptive structured boundary non-conforming grids, *Annual Research Briefs, Center for Turbulence Research, Stanford University* (2001) 353–466.
- [21] Y.-H. Tseng, J. Ferziger, A ghost-cell immersed boundary method for flow in complex geometry, *Journal of Computational Physics* 192 (2) (2003) 593–623.
- [22] R. Franke, Scattered data interpolation: Tests of some methods, *Mathematics of Computation* 38 (157) (1982) 181–200.
- [23] T. Gao, Y.-H. Tseng, X.-Y. Lu, An improved hybrid Cartesian/immersed boundary method for fluid-solid flows, *International Journal for Numerical Methods in Fluids* 55 (12) (2007) 1189–1211.
- [24] G. Iaccarino, R. Verzicco, Immersed boundary technique for turbulent flow simulations, *Applied Mechanics Reviews* 56 (3) (2003) 331–347.
- [25] K. Lundquist, F. Chow, J. Lundquist, An immersed boundary method enabling large-eddy simulations of flow over complex terrain in the WRF model, *Monthly Weather Review* 140 (12) (2012) 3936–3955.
- [26] T. Aslam, A partial differential equation approach to multidimensional extrapolation, *Journal of Computational Physics* 193 (1) (2004) 349 – 355.

- [27] Y. Ng, C. Min, F. Gibou, An efficient fluid solid coupling algorithm for single-phase flows, *Journal of Computational Physics* 228 (23) (2009) 8807 – 8829.
- [28] C. Min, F. Gibou, A second order accurate level set method on non-graded adaptive Cartesian grids, *Journal of Computational Physics* 225 (1) (2007) 300 – 321.
- [29] P. Chiu, R. Lin, T. W. Sheu, A differentially interpolated direct forcing immersed boundary method for predicting incompressible Navier Stokes equations in time-varying complex geometries, *Journal of Computational Physics* 229 (12) (2010) 4476 – 4500.
- [30] H. Uddin, R. Kramer, C. Pantano, A Cartesian-based embedded geometry technique with adaptive high-order finite differences for compressible flow around complex geometries, *Journal of Computational Physics* 262 (0) (2014) 379 – 407.
- [31] J. Cummings, M. Aivazis, R. Samtaney, R. Radovitzky, S. Mauch, D. Meiron, A virtual test facility for the simulation of dynamic response in materials, *The Journal of Supercomputing* 23 (1) (2002) 39–50.
- [32] M. Arienti, P. Hung, E. Morano, J. Shepherd, A level set approach to Eulerian-Lagrangian coupling, *Journal of Computational Physics* 185 (1) (2003) 213–251.
- [33] P. Colella, Multidimensional upwind methods for hyperbolic conservation laws, *Journal of Computational Physics* 87 (1) (1990) 171–200.
- [34] J. Saltzman, An unsplit 3D upwind method for hyperbolic conservation laws, *Journal of Computational Physics* 115 (1) (1994) 153–168.
- [35] P. Colella, D. Graves, T. Ligocki, D. Martin, B. Van Straalen, AMR godunov unsplit algorithm and implementation, Tech. rep., Applied Numerical Algorithms Group, NERSC Division, Lawrence Berkeley National Laboratory (2008).
- [36] J. Sethian, *Level-Set Method and Fast Marching Methods*, Cambridge, 1996.
- [37] F. Gibou, R. Fedkiw, A fourth order accurate discretization for the laplace and heat equations on arbitrary domains, with applications to the stefan problem, *Journal of Computational Physics* 202 (2005) 577 – 601.
- [38] M. Adams, P. Colella, D. Graves, J. Johnson, N. Keen, T. Ligocki, D. Martin, P. McCorquodale, D. Modiano, P. Schwartz, T. Sternberg, B. V. Straalen, Chombo software package for AMR applications design document.
- [39] M. Berger, J. Oliger, Adaptive mesh refinement for hyperbolic partial differential equations, *Journal of Computational Physics* 53 (3) (1984) 484 – 512.

- [40] M. Berger, P. Colella, Local adaptive mesh refinement for shock hydrodynamics, *Journal of Computational Physics* 82 (1) (1989) 64 – 84.
- [41] C. A. Rendleman, V. E. Beckner, M. Lijewski, W. Crutchfield, J. B. Bell, Parallelization of structured, hierarchical adaptive mesh refinement algorithms, *Computing and Visualization in Science* 3 (3) (2000) 147–157.
- [42] M. Al-Marouf, N. R. Rapaka, R. Samtaney, An embedded-boundary AMR method for compressible flow in moving complex geometry, 28th International Conference on Parallel Computational Fluid Dynamics, submitted (2016).
- [43] H. Choi, P. Moin, Grid-point requirements for large-eddy simulation: Chapman’s estimates revisited, *Physics of Fluids* 24 (2012) 011702.
- [44] P. Sagaut, *Large-eddy simulation for incompressible flows: an introduction*, Springer Science & Business Media, 2006.
- [45] C. Williamson, Vortex dynamics in the cylinder wake, *Annual Review of Fluid Mechanics* 28 (1) (1996) 477–539.
- [46] D. Hartmann, M. Meinke, W. Schröder, A strictly conservative Cartesian cut-cell method for compressible viscous flows on adaptive grids, *Computer Methods in Applied Mechanics and Engineering* 200 (9-12) (2011) 1038–1052.
- [47] M. Provansal, C. Mathis, L. Boyer, Bénard-von kármán instability: transient and forced regimes, *Journal of Fluid Mechanics* 182 (1987) 1–22.
- [48] D. Tritton, Experiments on the flow past a circular cylinder at low Reynolds numbers, *Journal of Fluid Mechanics* 6 (04) (1959) 547–567.
- [49] D. Le, B. Khoo, J. Peraire, An immersed interface method for viscous incompressible flows involving rigid and flexible boundaries, *Journal of Computational Physics* 220 (1) (2006) 109 – 138.
- [50] S. Dennis, G.-Z. Chang, Numerical solutions for steady flow past a circular cylinder at Reynolds numbers up to 100, *Journal of Fluid Mechanics* 42 (1970) 471–89.
- [51] M. Coutanceau, R. Bouard, Experimental determination of the main features of the viscous flow in the wake of a circular cylinder in uniform translation. I. Steady flow, *Journal of Fluid Mechanics* 79 (1977) 231–256.
- [52] R. Gautier, D. Biau, E. Lamballais, A reference solution of the flow over a circular cylinder at $Re = 40$, *Computers & Fluids* 75 (0) (2013) 103 – 111.

- [53] K. Taira, T. Colonius, The immersed boundary method: A projection approach, *Journal of Computational Physics* 225 (2) (2007) 2118–2137.
- [54] C. Brehm, C. Hader, H. Fasel, A locally stabilized immersed boundary method for the compressible Navier Stokes equations, *Journal of Computational Physics* 295 (2015) 475–504.
- [55] J. Park, K. Kwon, H. Choi, Numerical solutions of flow past a circular cylinder at Reynolds numbers up to 160, *KSME International Journal* 12 (6) (1998) 1200–1205.
- [56] E. Stålberg, A. Brüger, P. Lötstedt, A. Johansson, D. Henningson, High order accurate solution of flow past a circular cylinder, *Journal of Scientific Computing* 27 (1-3) (2006) 431–441.
- [57] E. Berger, R. Wille, Periodic flow phenomena, *Annual Review of Fluid Mechanics* 4 (1) (1972) 313–340.
- [58] F. White, *Viscous fluid flow*, 3rd Edition, McGraw-Hill New York, 2006.
- [59] D. Russell, Z. Wang, A Cartesian grid method for modeling multiple moving objects in 2D incompressible viscous flow, *Journal of Computational Physics* 191 (1) (2003) 177–205.
- [60] C. Liu, X. Zheng, C. Sung, Preconditioned multigrid methods for unsteady incompressible flows, *Journal of Computational Physics* 139 (1) (1998) 35 – 57.
- [61] I. Kroo, P. Kunz, Analysis and design of airfoils for use at ultra-low Reynolds numbers, *Fixed and Flapping Wing Aerodynamics for Micro Air Vehicle Applications* (2001) 503517.
- [62] R. Mittal, H. Dong, M. Bozkurtas, F. M. Najjar, A. Vargas, A. von Loebbecke, A versatile sharp interface immersed boundary method for incompressible flows with complex boundaries, *Journal of Computational Physics* 227 (10) (2008) 4825–4852.
- [63] W. Zhang, R. Samtaney, A direct numerical simulation investigation of the synthetic jet frequency effects on separation control of low-re flow past an airfoil, *Physics of Fluids* (1994-present) 27 (5) (2015) 055101.
- [64] R. Ghias, R. Mittal, H. Dong, A sharp interface immersed boundary method for compressible viscous flows, *J. Comput. Phys.* 225 (1) (2007) 528 – 553.
- [65] S. Mittal, R. Balachandar, On the inclusion of three dimensional effects in simulations of two-dimensional bluff body wake flows (1997).
- [66] B. Rajani, A. Kandasamy, S. Majumdar, Numerical simulation of laminar flow past a circular cylinder, *Applied Mathematical Modelling* 33 (3) (2009) 1228–1247.

- [67] A. Kravchenko, P. Moin, K. Shariff, B-spline method and zonal grids for simulations of complex turbulent flows, *Journal of Computational Physics* 151 (2) (1999) 757–789.
- [68] J. Nam, F. Lien, A ghost-cell immersed boundary method for large-eddy simulations of compressible turbulent flows, *International Journal of Computational Fluid Dynamics* 28 (1-2) (2014) 41–55.
- [69] T. Johnson, V. Patel, Flow past a sphere up to a Reynolds number of 300, *Journal of Fluid Mechanics* 378 (1999) 19–70.
- [70] R. Mittal, A Fourier-Chebyshev spectral collocation method for simulating flow past spheres and spheroids, *International Journal for Numerical Methods in Fluids* 30 (7) (1999) 921–937.
- [71] J. Kim, D. Kim, H. Choi, An immersed-boundary finite-volume method for simulations of flow in complex geometries, *Journal of Computational Physics* 171 (1) (2001) 132–150.
- [72] H. Dütsch, F. Durst, S. Becker, H. Lienhart, Low-Reynolds-number flow around an oscillating circular cylinder at low Keulegan—Carpenter numbers, *Journal of Fluid Mechanics* 360 (1998) 249–271.
- [73] E. Guilmineau, P. Queutey, A numerical simulation of vortex shedding from an oscillating circular cylinder, *Journal of Fluids and Structures* 16 (6) (2002) 773–794.
- [74] M. Koochesfahani, Vortical patterns in the wake of an oscillating airfoil, *AIAA journal* 27 (9) (1989) 1200–1205.
- [75] R. Ramamurti, W. Sandberg, Simulation of flow about flapping airfoils using finite element incompressible flow solver, *AIAA Journal* 39 (2) (2001) 253–260.
- [76] J. Young, J. Lai, Oscillation frequency and amplitude effects on the wake of a plunging airfoil, *AIAA Journal* 42 (10) (2004) 2042–2052.
- [77] J. Lai, M. Platzer, Jet characteristics of a plunging airfoil, *AIAA journal* 37 (12) (1999) 1529–1537.
- [78] R. LeVeque, *Finite volume methods for hyperbolic problems*, Vol. 31, Cambridge university press, 2002.
- [79] H. Glaz, P. Colella, I. Glass, R. Deschambault, A numerical study of oblique shock-wave reflections with experimental comparisons, in: *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, Vol. 398, The Royal Society, 1985, pp. 117–140.
- [80] D. Graves, P. Colella, D. Modiano, J. Johnson, B. Sjogreen, X. Gao, A Cartesian grid embedded boundary method for the compressible Navier Stokes equations, *Communications in Applied Mathematics and Computational Science* 8 (1) (2013) 99–122.

- [81] M. de Tullio, P. Palma, G. Iaccarino, G. Pascazio, M. Napolitano, An immersed boundary method for compressible flows using local grid refinement, *Journal of Computational Physics* 225 (2) (2007) 2098 – 2117.
- [82] V. Bashkin, A. Vaganov, I. Egorov, D. Ivanov, G. Ignatova, Comparison of calculated and experimental data on supersonic flow past a circular cylinder, *Fluid Dynamics* 37 (3) (2002) 473–483.
- [83] J. Nam, F. Lien, Assessment of ghost-cell based cut-cell method for large-eddy simulations of compressible flows at high Reynolds number, *International Journal of Heat and Fluid Flow* 53 (2015) 1–14.
- [84] A. Bailey, J. Hiatt, Sphere drag coefficients for a broad range of Mach and Reynolds numbers, *Aiaa Journal* 10 (11) (1972) 1436–1440.
- [85] J. Grétarsson, N. Kwatra, R. Fedkiw, Numerically stable fluid-structure interactions between compressible flow and solid structures, *Journal of Computational Physics* 230 (8) (2011) 3062–3084.