CrossMark

ORIGINAL PAPER

A Higher-Order Chimera Method for Finite Volume Schemes

Luis Ramírez[1] · **Xesús Nogueira**[1] · **Pablo Ouro**[2] · **Fermín Navarrina**[1] ·
Sofiane Khelladi[3] · **Ignasi Colominas**[1]

Abstract In this work a higher-order accurate finite volume method for the resolution of the Euler/Navier–Stokes equations using Chimera grid techniques is presented. The formulation is based on the use of Moving Least Squares approximations in order to obtain higher-order accurate reconstruction and connectivity between the overlapped grids. The accuracy and performance of the proposed methodology is demonstrated by solving different benchmark problems.

1 Introduction

The development of numerical methods for the simulation of problems involving highly complex geometries, which are frequent in many engineering problems, remains a very active research field in computational fluid dynamics (CFD). For these problems, the construction of multi-block structured meshes, when possible, is highly time consuming. In this context the use of unstructured grids becomes competitive but, unfortunately, this kind of mesh typology requires the use of more complex schemes when higher-order approximations are desired.

Moreover, the use of a single body-fitted mesh for the simulation of flows with moving domains, requires the deformation of the existing grid or the generation of a new one at each time step. This procedure is highly demanding in terms of computational cost for relatively large body motions. For simple movement patterns, such as rotating domains or sliding planes, it is possible to use sliding mesh techniques [1, 2]. In these cases, the computational domain is divided into two subdomains, namely moving and static subdomains. The information is transferred from one domain to the other through their interface, by using suitable interpolation techniques [1, 2].

A different approach in the simulation of moving bodies is the Immersed Boundary (IB) method, first introduced by Peskin [3] for the simulation of heart valves. Cartesian rectangular meshes are commonly used in the IB approach and they are decoupled from the immersed body mesh. Lately, IB has also been adapted to unstructured meshes [4]. In the classical IB approach, the boundary is introduced as a forcing term in the governing equations. These methods are attractive because of their simplicity. However, the major drawbacks are the occurrence of non-divergence-free velocities in incompressible flows [5], non-physical pressure oscillations in compressible flows, and the difficulty to get high-order accuracy near the walls [6]. A different class of IB methods is the cut-cell method introduced by Clarke [7], which does not exhibit of these problems. In the cut-cell method the immersed boundaries cut the mesh, creating a set of irregularly shaped cells upon which the equations are discretized. A drawback of this approach is the increased complexity, compared to the classical IB methods, since the original mesh needs to be cut by the moving bodies every time step, which forces to recompute the geometrical information and the integration points.

✉ Luis Ramírez
luis.ramirez@udc.es

1 Group of Numerical Methods in Engineering, Universidade da Coruña, Campus de Elviña, 15071 A Coruña, Spain

2 Hydro-Environmental Research Centre, School of Engineering, Cardiff University, The Parade, Cardiff CF24 3AA, UK

3 Laboratoire de Dynamique des Fluides, Arts et Métiers ParisTech, 151 Boulevard de l'Hôpital, 75013 Paris, France

In this context, the use of the overset grid approach, also known as Chimera method [8–12], has become a competitive alternative. In the Chimera method, firstly developed by Steger et al. in 1983 [8], the domain is subdivided into a set of overlapping grids. The partial differential equations are solved separately on each grid. The overlapped grids are connected through interpolation to exchange the necessary information between them. This approach exhibits flexible grid adaptation, the ability to handle complex geometries, and provides an easy mechanism to deal with the relative motion of dynamic bodies.

The interpolation between the different grids is a crucial stage for a higher-order Chimera method. In [12], it is shown that the use of linear interpolation schemes in conjunction with high-order methods leads to a decrease of the global accuracy of the solution. In fact, the overall order of accuracy is conditioned by the smallest one. As it was pointed out by several authors [10, 12], the order of the interpolation scheme must be higher or equal to the order of accuracy of the discretization scheme, in order to maintain the global accuracy of the numerical method. In this work we address this problem by using Moving Least Squares (MLS) approximations [13, 14]. The present approach may be considered as a generalization of the approach presented by the authors in [1] in the simulation of bodies under arbitrary motions. Moreover, we use a high-order finite volume method based on MLS (FV-MLS) [15–20] as the numerical scheme to solve the governing equations. In this method, high-order discretization of the governing equations is achieved using Moving Least Squares approximations for the computation of the successive derivatives that are required in the Taylor reconstruction. In this framework, the use of MLS for the exchange of data from one grid to another seems to be a natural choice. However, this approach could be used with any other numerical method.

The outline of the paper is as follows: First, the governing equations and the numerical discretization is presented in Sect. 2. Next, the Moving Least Squares method is described in Sect. 3. Then, the Chimera method is presented in Sect. 4. Next, some numerical results are presented in Sect. 5 in order to evaluate the accuracy and robustness of the proposed method when solving different two-dimensional benchmark problems. Finally, the corresponding conclusions are drawn.

2 Governing Equations

The Navier–Stokes equations, written in general form as a system of conservation laws, read as

$$\frac{\partial \boldsymbol{U}}{\partial t} + \boldsymbol{\nabla} \cdot \left(\boldsymbol{F}^H - \boldsymbol{F}^E \right) = \boldsymbol{S} \tag{1}$$

where $\boldsymbol{U}$ is the vector of variables, $\boldsymbol{S}$ represents a source term, and fluxes are split into a hyperbolic-like part, $\boldsymbol{F}^H$, and an elliptic-like part, $\boldsymbol{F}^E$, which is null for the Euler equations.

The finite volume discretization of the system of conservation laws (1) for a control volume I reads

$$A_I \frac{\partial \boldsymbol{U}_I}{\partial t} = \sum_{j=1}^{N_f} \sum_{ig=1}^{N_G} \left[\left(\boldsymbol{F}^E - \boldsymbol{F}^H \right) \cdot \hat{\boldsymbol{n}}_j \right]_{ig} \mathcal{W}_{ig} \tag{2}$$

where A_I is the area of the control volume I, $\boldsymbol{U}_I$ represents the average value of $\boldsymbol{U}$ over the control volume I, N_f is the number of edges (in 2D) of the control volume, N_G represents the number of quadrature points for each edge, $\mathcal{W}_{ig}$ is the corresponding quadrature weight for the quadrature point (ig) at each cell edge, and $\hat{\boldsymbol{n}}_j$ is the unitary normal $\boldsymbol{n}$ times the length of edge j.

The hyperbolic-like part, $\boldsymbol{F}^H$, is computed at the integration points by means of a "broken" reconstruction. This high-order reconstruction is achieved using a high-order Taylor expansion from the cell centroid. Mathematically, this statement can be expressed as $\boldsymbol{\mathcal{F}^H} = \mathcal{H}(\boldsymbol{U}^+, \boldsymbol{U}^-)$, where $\mathcal{H}$ is the numerical flux that solves the Riemann problem stated at the integration points. Reconstructed values $\boldsymbol{U}^+$ and $\boldsymbol{U}^-$ are the left and right states of cell I.

The elliptic-like fluxes are computed centered at the integration points. That is, in Fig. 1 the elliptic-like flux, $\boldsymbol{F}^E$, is computed directly at the integration points using a continuous approach. The reader is referred to [17, 21] for further details.
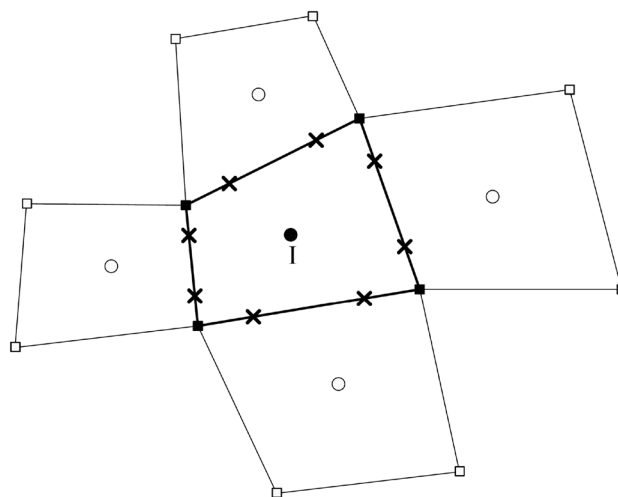

Fig. 1 Schematic illustration of the cell I and its integration points

3 Interpolation Operator: Moving Least Squares Reproducing Kernel Approximations

In this section we briefly introduce the Moving Least Squares (MLS) approximations [13], which are the key ingredient of the new high-order Chimera method presented in this paper. For a more complete description of MLS, we refer the interested reader to [14].

For the sake of brevity, we only recall here that the MLS approximation structure of a variable u at a point $\boldsymbol{x}$ can be expressed as

$$\widehat{u}(\boldsymbol{x}) = \sum_{k=1}^{n_x} N_k(\boldsymbol{x}) u_k \tag{3}$$

The approximation of u evaluated at $\boldsymbol{x}$ is written in terms of vector $\boldsymbol{N}$ of MLS shape functions N_k ($k = 1, ..., n_x$). In order to compute the MLS shape functions we need a set of n_x neighboring points, which defines the stencil where the variable u_k is known. The number of neighbors, n_x, depends on the required order of accuracy. A typical stencil for a 2D cubic basis, which is third-order accurate, is shown in Fig. 2.

The definition of the stencil is a very important issue. The stencil should be as compact as possible, and the selection of neighbors must be suitable for general grids. It is important to remark that these stencils are typically centered around the cell, shown in Fig. 2. This kind of configurations reduces the spatial bias which is often found in patch-based piece-wise polynomial approximations.

In our case, the MLS method is applied by considering the centroids of every cell of the stencil as the neighbors required for the approximation given in expression (3). The MLS shape functions, $\boldsymbol{N}^T$, are obtained as

$$\boldsymbol{N}^T(\boldsymbol{x}) = \boldsymbol{p}^T(\boldsymbol{x}) \boldsymbol{M}^{-1}(\boldsymbol{x}) \boldsymbol{P}_{\Omega_x} \boldsymbol{W}(\boldsymbol{x}) \tag{4}$$
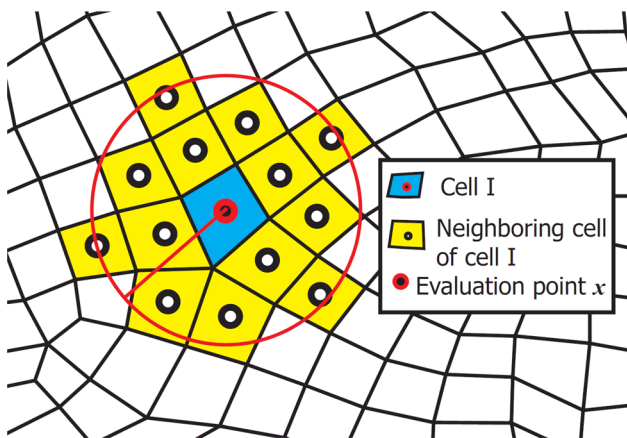

Fig. 2 Stencil for interior cells of the MLS approximation centered at cell centroid I

where $\boldsymbol{p}^T(\boldsymbol{x})$ is the vector defining the basis of functions (usually polynomials) with dimension m. For example, for a two-dimensional linear basis, $\boldsymbol{p}^T(\boldsymbol{x}) = [1, x, y]$ and $m = 3$. The order of MLS approximations is determined by the polynomial basis used in the construction of MLS shape functions. In all the examples presented in this work, a cubic polynomial basis has been used.

Moreover, $\boldsymbol{P}_{\Omega_x}$ is defined as the $m \times n_x$ matrix where the basis functions are evaluated at each point of the stencil. We also define the diagonal matrix $\boldsymbol{W}(\boldsymbol{x}) = diag\{W_i(\boldsymbol{x} - \boldsymbol{x}_i)\}$, where W is a suitable *kernel* (or smoothing function). The moment matrix $\boldsymbol{M}(\boldsymbol{x})$ (dimension $m \times m$) is obtained by minimizing an error functional (see [17] for details) and is defined as

$$\boldsymbol{M}(\boldsymbol{x}) = \boldsymbol{P}_{\Omega_x} \boldsymbol{W}(\boldsymbol{x}) \boldsymbol{P}_{\Omega_x}^T \tag{5}$$

The *kernel* function plays a very important role in the MLS method, since it assigns the corresponding weights to the different points considered in the approximation. A wide variety of *kernel* functions are found in the literature [22–25]. The choice of the *kernel* function determines the properties of the approximation [17, 19]. In this work, the so-called exponential *kernel* is used, what is defined in one dimension as

$$W_k(x_k, x, s_x) = \frac{e^{-\left(\frac{d}{c}\right)^2} - e^{-\left(\frac{d_m}{c}\right)^2}}{1 - e^{-\left(\frac{d_m}{c}\right)^2}} \tag{6}$$

for $k = 1, ..., n_x$, where $d = |x_k - x|$ and $d_m = 2 \max\left(|x_k - x|\right)$, being d_m the *smoothing length*, n_x the number of neighbors, and x the reference point where the compact support is centered.

In Eq. (6) we introduce the coefficient c, which is defined as $c = \frac{d_m}{s_x}$ where s_x is the shape parameter of the *kernel*. This parameter has a relevant influence in the *kernel* as it defines its properties and, thus, the properties of the numerical scheme [19]. In this work we have chosen the value of $s_x = 6$ for the interpolations in the Chimera scheme, as explained on Sect. 4.

The high-order approximation derivatives of field variables $u(\boldsymbol{x})$ can be expressed in terms of the derivatives of the MLS shape function. Hence the *n-th* derivative can be obtained as

$$\frac{\partial^n \widehat{u}}{\partial x^n} = \sum_{ki=1}^{n_x} \frac{\partial^n N_k(\boldsymbol{x})}{\partial x^n} u_k \tag{7}$$

The interested reader is referred to [17, 18, 21] for a complete description of the computation of MLS derivatives.

4 An Overset/Chimera Approach

One challenge in the development of a higher-order Chimera method is the preservation of the global order of the numerical scheme. The problem appears when the information is transferred between overlapping grids. For simplicity, we focus on a set of two overlapping grids, even though the methodology is valid for any arbitrary set of grids. In Fig. 3 we schematically present an example of the domain discretization addressed in this work by using two overlapping grids: a rectangular background grid with quad elements (Grid A) and a circular shape near-body grid with triangular elements (Grid B). The near-body grid could be attached to a body present on the flow (e.g. a moving body), whereas the background grid discretizes the computational domain as if the body was not present in the flow.

For each time step the following procedure is performed:

1. First, we identify the cells where the system of conservation laws is computed. We label them as non-overlapped cells. This set is composed of all the cells of the near-body grid and those cells of the background grid that are not completely covered by the near-body grid and the solid body. A hole-cutting process is used to determine the overlapped cells on the background grid. For this schematic example, non-overlapped cells are shown in Fig. 4. Note that in problems with moving bodies this procedure is performed at each time step.
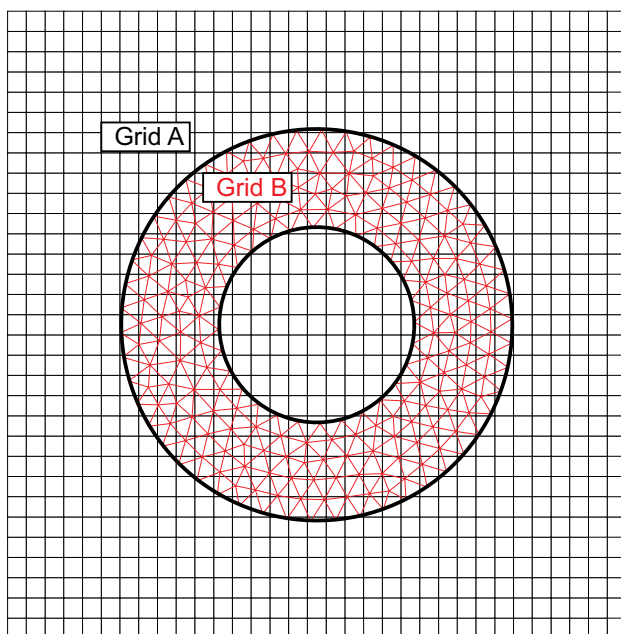
Fig. 3 Schematic representation of a circular-shaped solid using two overlapped grids. Background grid is labeled as *Grid A*, whereas the near body grid is labeled as *Grid B*
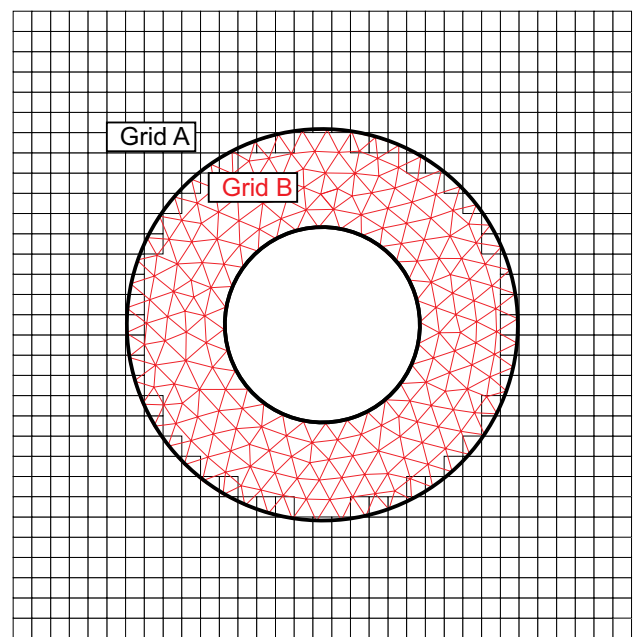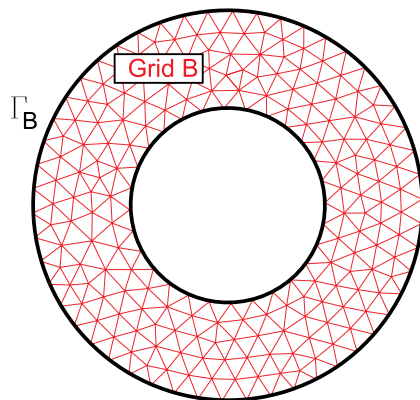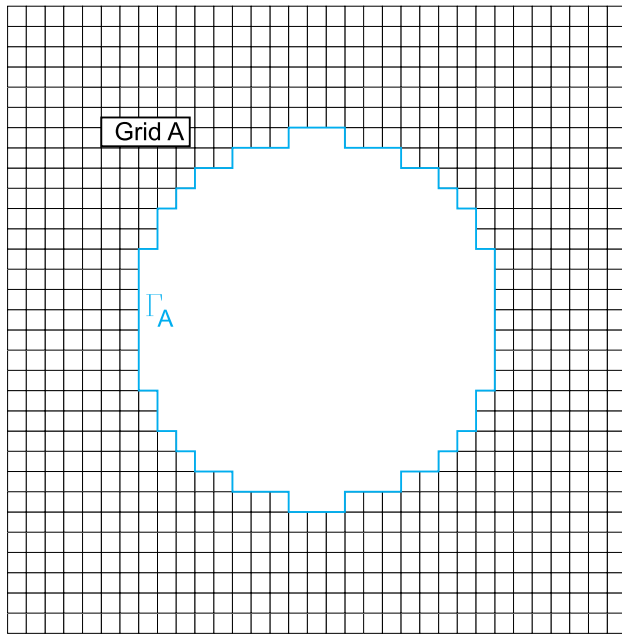
Fig. 4 Schematic representation of non-overlapped cells of Grids A and B. Note that all the cells belonging to *Grid B* are non-overlapped cells

Overlapped cells are inactive and they are not computed. Instead, the value of the variables in these cells is interpolated from the near-body grid to the background grid at the end of each time step.

2. Once the non-overlapped cells are identified, the governing equations are solved for each grid independently. In Fig. 5 we schematically plot the background and the near-body grids where equations are solved. During this procedure, two interfaces are defined inside the computational domain: Γ_A defines an instantaneous boundary for the background grid, which separates the overlapped cells from the non-overlapped cells of this grid. The other boundary, Γ_B, defines the outer boundary for the near-body grid. Note that the position and shape of Γ_A may vary in time as the near-body grid is moving.

Considering two overlapped grids, the procedure of the Chimera method presented here is defined by three sub-steps. When more than two overlapped grids are present, the procedure follows the same rationale.
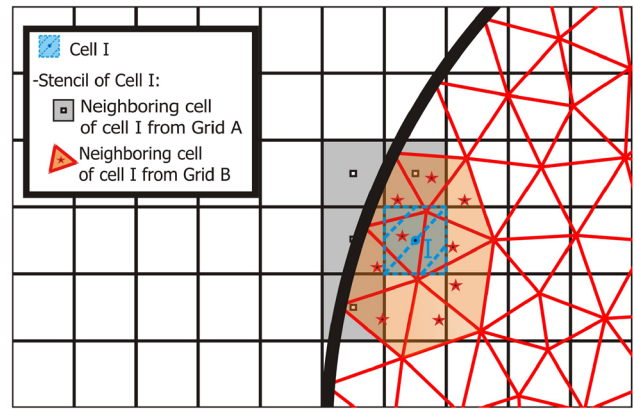
(a) Transference of data to the overlapped cells: This sub-step is performed by MLS approximations at the centroids of the overlapped cells on the Background grid. The MLS approximation for an overlapped cell *I* reads

Fig. 6 Schematic representation of the stencil of the overlapped cell. *Shaded cells* indicate the stencil used for the computation of U_I in Eq. (8)

(b) Resolution of the near-body grid: In a finite volume framework, we need to define the right and left states at both sides of the interface of a Riemann problem in order to compute the fluxes. In this work, we use the information of Grids A and B to create a fictitious state at the outer side of Γ_B, which is used to completely define the fluxes at Γ_B. Thus, for a certain cell I_B with a given edge e_I in Γ_B, we define a fictitious cell called *halo cell* (I_{halo}), which is the specular reflection of cell I_B. This is schematically shown in Fig. 7.

The mean value of the conservative variables at I_{halo} is approximated by Moving Least Squares as follows

$$U_{Halo} = \frac{1}{A_{Halo}} \int U dA = \frac{1}{A_{Halo}} \int \sum_{j=1}^{n_x} N_j(\boldsymbol{x}_{Halo}) U_j dA \quad (9)$$

Fig. 5 Schematic representation of the grids for the resolution of the equations on the background and the near-body grids. Γ_A (in *blue*) defines the instantaneous boundary of the background grid. Γ_B is the outer boundary of the near body grid

$$U_I = \frac{1}{A_I} \int U dA = \frac{1}{A_I} \int \sum_{j=1}^{S(A \cup B)} N_j(\boldsymbol{x}_I) U_j dA \quad (8)$$
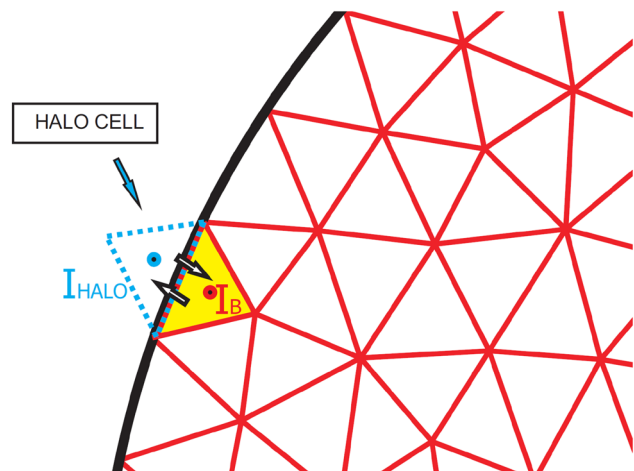
where $S(A \cup B)$ refers to the set of cells belonging to both Grids A and B which forms the stencil of centroid I. This stencil is defined as the union of the closest cells to centroid I considering the cells of Grid B and the non-overlapped cells of Grid A. A schematic representation is plotted in Fig. 6. As indicated in Sect. 3, the number of cells of the stencil is variable depending on the required order of the approximation.
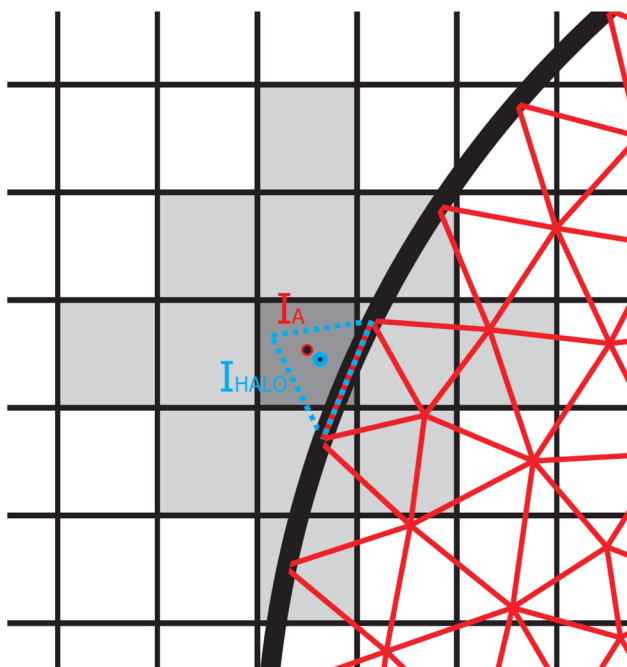

Fig. 7 Schematic representation of the flux exchange at the near-body grid. A halo cell I_{halo} is created allowing the computation of fluxes

where A_{Halo} is the area associated to the halo cell I_{halo} and $N(x_{Halo})$ is the vector of MLS shape functions centered at the centroid of I_{halo}. In order to compute the MLS shape functions $N(x_{Halo})$ is required the definition of the neighboring cells of I_{halo}. We define this set of cells as the stencil of I_A, which is the closest cell of the background grid to I_{halo}, as shown in Fig. 8.

(c) Resolution of the Background grid: The system of conservation laws is solved for all the non-overlapped cells that belong to the Background grid. Note that the values of the overlapped cells are known from the transference of data at the previous time step. Thus, the right and left states for a non-overlapped cell that shares an edge with the instantaneous boundary Γ_A are easily computed as an interior cell of a FV scheme.

5 Numerical Results

This section presents the numerical results for several benchmark problems defined with the aim of assessing the accuracy and efficiency of the proposed method for both steady and unsteady problems. In all these cases a third-order FV-MLS method is employed [15, 17, 18, 20]. In this work we have used an explicit time integration method based on a fourth-order Runge-Kutta method.


Fig. 8 Schematic representation of the stencil of the halo cell. *Shaded cells* indicate the stencil used for the computation of U_{Halo} in Eq. (9)

5.1 Ringleb Flow

The first validation test is the Ringleb flow problem. This test case is widely used as a benchmark for compressible codes [15, 26]. The flow is obtained as a solution of the hodograph equation. The transformation equations between the Cartesian variables (x, y) and the hodograph variables (V, ϑ) are described in [26].

The square domain $\Omega = [-1.15, -0.75] \times [0.15, 0.55]$ is discretized using two overlapped grids. Following the notation previously used, we call them Grid A (the background grid) and Grid B. Figure 9 shows the schematic description of the problem. The different combinations of sizes used for grids A and B are summarized in Table 1.

For this first test case we study the rate of convergence for the Chimera method with a static configuration. L_1^N, L_2^N and L_∞^N norms of the variables error and $L_2^{ent,N}$ norm of the entropy error are computed and summarized in Table 2. We observe that the optimal orders of accuracy,
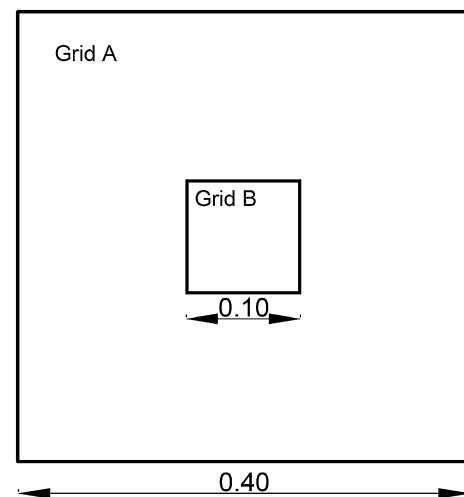
Fig. 9 Schematic setup of the Ringleb Flow test case

Table 1 Ringleb Flow test case. Different combinations of Grids A and B for the Chimera Method

Combination	Grid A $(N_x \times N_y)_A$	Grid B $(N_x \times N_y)_B$
Ringleb flow		
1	32×32	16×16
2	48×48	24×24
3	64×64	32×32
4	96×96	48×48
5	128×128	64×64
6	192×192	96×96

Table 2 Ringleb Flow. Accuracy orders for the 3rd order Chimera method in the static configuration

Combination	L_1^N error	O_1^N	L_2^N error	O_2^N	L_∞^N error	O_∞^N	$L_2^{ent,N}$ error	$O_2^{ent,N}$
Ringleb flow. Static configuration								
1	3.86×10^{-9}	–	1.14×10^{-8}	–	1.50×10^{-8}	–	2.82×10^{-9}	–
2	1.37×10^{-9}	2.58	3.72×10^{-9}	2.79	5.92×10^{-9}	2.31	9.13×10^{-10}	2.80
3	6.27×10^{-10}	2.73	1.66×10^{-9}	2.83	2.93×10^{-9}	2.46	4.03×10^{-10}	2.86
4	2.01×10^{-10}	2.82	5.18×10^{-10}	2.88	1.05×10^{-9}	2.55	1.25×10^{-10}	2.90
5	8.76×10^{-11}	2.89	2.24×10^{-10}	2.93	4.98×10^{-10}	2.59	5.38×10^{-11}	2.93
6	2.63×10^{-11}	2.98	6.69×10^{-11}	2.99	1.66×10^{-10}	2.71	1.63×10^{-11}	2.95

denoted as O_1^N, O_2^N, O_∞^N and $O_2^{ent,N}$, are recovered for all norms.

We also compute the errors and convergence rates using a standard FV-MLS method on several single-block structured meshes. The resolution of this set of grids is the same as that of the Grid A shown in Table 1. The errors and convergence rates are reported in Table 3. It is observed that errors obtained using the Chimera method are smaller than those obtained with the standard method. This is due to the finer resolution of Grid B.

5.2 Isentropic Vortex Convection

The second validation case corresponds to the unsteady vortex convection. This test case is widely used as benchmark for moving grid formulations [1, 9–11, 27], since it has analytical solution, that reads

$$\frac{u(x,y,t)}{a_\infty} = \frac{u_\infty}{a_\infty} - \frac{K}{2\pi a_\infty}\hat{y}e^{\alpha(1-r^2)/2}$$
$$\frac{v(x,y,t)}{a_\infty} = \frac{v_\infty}{a_\infty} + \frac{K}{2\pi a_\infty}\hat{x}e^{\alpha(1-r^2)/2}$$
$$\frac{T(x,y,t)}{T_\infty} = 1 - \frac{K^2(\gamma-1)}{8\alpha\pi^2 a_\infty^2}e^{\alpha(1-r^2)}$$
$$\frac{\rho(x,y,t)}{\rho_\infty} = \left(\frac{T(x,y,t)}{T_\infty}\right)^{\frac{1}{\gamma-1}}$$
$$\frac{p(x,y,t)}{p_\infty} = \left(\frac{T(x,y,t)}{T_\infty}\right)^{\frac{\gamma}{\gamma-1}}$$

where $\hat{x} = x - x_0 - u_\infty t$, $\hat{y} = y - y_0 - v_\infty t$ and $r = \sqrt{\hat{x}^2 + \hat{y}^2}$. Here, the chosen parameters are $\alpha = 1$, $\rho_\infty = 1$, $p_\infty = 1$, $(u_\infty, v_\infty) = (2, 2)$, $(x_0, y_0) = (-5, -5)$ and $K = 5$. With this set of parameters the vortex starts at the position $(x, y) = (-5, -5)$ and at $t = 5$ reaches the position $(x, y) = (5, 5)$.

In this test case we examine the accuracy when the vortex travels through the overlapped region. A schematic description of the problem is drawn in Fig. 10a.

The square domain was discretized by means of the same grids used for the Ringleb test case (see Table 1). As a reference, in Fig. 10b we show the grids for combination number 3.
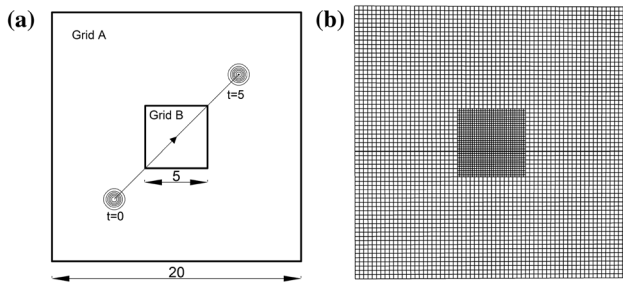
We consider several configurations for this test case: one static configuration and two different prescribed motions.

5.2.1 Static Configuration

First, we analyze the case with a static grid configuration for A and B grids. Therefore, there is not relative motion between them. The results of L_2, L_1 and L_∞ norms of the variables error and their corresponding convergence rates are presented in Table 4. As expected, the formal order of accuracy is recovered.

5.2.2 Prescribed Motion 1: Sinusoidal displacement of Grid B

Next, a relative motion between grids is added. Position of *Grid B* depends on time according to the expression

Table 3 Ringleb Flow. Accuracy orders for the 3rd order FV-MLS with a single-block structured grid

Grid A	L_1^N error	O_1^N	L_2^N error	O_2^N	L_∞^N error	O_∞^N	$L_2^{ent,N}$ error	$O_2^{ent,N}$
Ringleb flow. Single mesh configuration								
1	4.21×10^{-9}	–	1.21×10^{-8}	–	1.54×10^{-8}	–	2.89×10^{-9}	–
2	1.48×10^{-9}	2.58	3.94×10^{-9}	2.76	6.04×10^{-9}	2.31	9.43×10^{-10}	2.77
3	6.74×10^{-10}	2.73	1.76×10^{-9}	2.81	2.98×10^{-9}	2.46	4.17×10^{-10}	2.84
4	2.15×10^{-10}	2.82	5.49×10^{-10}	2.87	1.06×10^{-9}	2.55	1.29×10^{-10}	2.89
5	9.37×10^{-11}	2.89	2.37×10^{-10}	2.92	5.03×10^{-10}	2.59	5.59×10^{-11}	2.92
6	2.81×10^{-11}	2.97	7.07×10^{-11}	2.98	1.68×10^{-10}	2.70	1.69×10^{-11}	2.94

(a)

(b)

Fig. 10 a Schematic setup of the 2D Vortex Convection problem. **b** Structured mesh using combination 3 as given in Table 1

$(x, y) = (0, A \sin(2\pi f t))$. We chose an amplitude of $A = 1$ and a frequency of $f = 0.5$. This configuration is schematically shown in Fig. 10a.

The results of L_1, L_2 and L_∞ norms of the variables error and their corresponding convergence rates are found in Table 5. The formal order of accuracy is recovered and the accuracy is very similar to that of the static configuration. No numerical artifacts are observed when the vortex travels through the overlapped region.

5.2.3 Prescribed Motion 2. Grid B Following the Vortex

In this test case, we impose a movement to *Grid B* in order to follow the vortex displacement, i.e. the position of *Grid B* varies with time according to $(x, y) = (u_\infty t, v_\infty t)$. As it

can be seen in Fig. 11 the initial location of *Grid B* is different from the previous configurations of the problem.

The results are summarized in Table 6. It is observed that the order of the method is recovered. We also observe that the accuracy is increased with respect to the two previous cases. This is due to the finer mesh resolution around the vortex.

5.3 Subsonic Inviscid Flow Around a NACA 0012

The next problem corresponds to a subsonic flow around a NACA 0012 airfoil. The aim of this test case is to demonstrate the ability of the proposed methodology to deal with complex geometries and unstructured meshes. In this case, Grid B is an unstructured grid. The computational domain is a circle where the fairfield is located at 30 chords. The total number of non-overlapped cells is 13189. A close view of non-overlapped cells around the airfoil is shown in Fig. 12a. Note the disparity of element sizes between Grid A and Grid B.

The freestream Mach number is $M = 0.63$ and the angle of attack is $\alpha = 2$ degrees. Slip and adiabatic boundary conditions are applied at solid walls.

We compare the results with those obtained using a single mesh of 19716 cells, which is shown in Fig. 12b. Drag and lift coefficients are computed and summarized in Table 7. We observe that the value of these coefficients is practically identical when using single and Chimera

Table 4 Isentropic Vortex Convection. Accuracy orders for the 3rd order Chimera method in the static configuration

N	L_1^N error	O_1^N	L_2^N error	O_2^N	L_∞^N error	O_∞^N
Isentropic vortex convection. Static configuration						
1244	3.04×10^{-3}	–	9.43×10^{-2}	–	1.80×10^{-1}	–
2780	1.27×10^{-3}	2.17	4.10×10^{-2}	2.07	7.07×10^{-2}	2.32
4924	6.31×10^{-4}	2.43	2.39×10^{-2}	1.89	4.59×10^{-2}	1.51
11036	2.46×10^{-4}	2.33	9.83×10^{-3}	2.20	1.81×10^{-2}	2.30
19580	1.17×10^{-4}	2.59	4.67×10^{-3}	2.59	8.06×10^{-3}	2.83
43964	3.86×10^{-5}	2.75	1.49×10^{-4}	2.83	2.52×10^{-3}	2.87

Table 5 Isentropic Vortex Convection. Accuracy orders for the 3rd order Chimera method in the prescribed sinusoidal motion configuration

N	L_1^N error	O_1^N	L_2^N error	O_2^N	L_∞^N error	O_∞^N
Isentropic vortex convection. Sinusoidal displacement of Grid B						
1244	3.13×10^{-3}	–	9.48×10^{-2}	–	1.81×10^{-1}	–
2780	1.26×10^{-3}	2.27	4.25×10^{-2}	2.00	7.69×10^{-2}	2.12
4924	6.35×10^{-4}	2.39	2.46×10^{-2}	1.92	4.33×10^{-2}	1.76
11036	2.40×10^{-4}	2.41	9.88×10^{-3}	2.26	1.77×10^{-2}	2.40
19580	1.15×10^{-4}	2.57	4.78×10^{-3}	2.53	8.27×10^{-3}	2.65
43964	3.90×10^{-5}	2.68	1.62×10^{-4}	2.67	2.87×10^{-3}	2.62
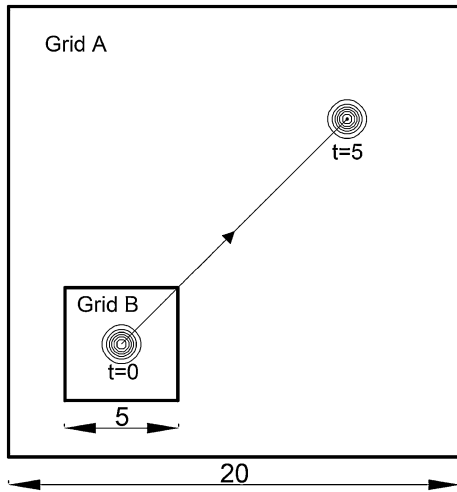
Fig. 11 Schematic setup of the 2D Vortex Convection problem for the *Grid B* following the vortex configuration

grids. Our results also agree with the expected theoretical reference values [15].

Moreover, the distribution of the pressure coefficient around the airfoil is presented in Fig. 13. Similar results are obtained with both approaches.

Mach isolines are plotted in Fig. 14, where the smooth transition between grids is appreciated. No numerical artifacts are observed.

5.4 Supersonic Flow Around a Cylinder

In this numerical example we study the supersonic flow around a cylinder of radius $R = 1$. The aim of this numerical example is to analyze the performance of the proposed methodology with supersonic flows and non-smooth solutions. The freestream Mach number is $M = 3$. Following [1], the parameters for this case are the pressure coefficient (C_P), the stand-off distance normalized by the diameter of the cylinder, and the normalized stagnation pressure (p_0).

Following [28] we have used a shock detector based on MLS and the limiter of Van Albada [29]. Slip boundary

conditions are imposed at the surface of the cylinder and freestream conditions are imposed at the outer radius $R_{outer} = 10$.

In this case we study two configurations

5.4.1 Static Configuration

First, a static configuration is analyzed. The computational domain is discretized with two non-conformal grids, as it can be seen in Fig. 15. The parameters for this configuration are compared with the ones obtained using a single mesh configuration, and results are presented in Table 8; Fig. 16.

In Fig. 17 the Mach isolines obtained with the proposed methodology are plotted and compared with the ones obtained with a single grid configuration. Note that both solution are practically coincident. Again, no numerical artifacts are observed through the interface, which location is highlighted in red.

5.4.2 Prescribed Motion

Once the performance of the Chimera method with a static configuration was analyzed in presence of a strong shock, the next step is to prescribe motion. The aim of this case is to study the behavior of the proposed methodology when *Grid B* moves through a shock. The schematic configuration of this case is represented in Fig. 18.

The overlapped grid (*Grid B*) is initially centered at $(x, y) = (-2.50, 0)$. The location of *Grid B* changes depending on the physical time according to the expression $(x, y) = (A \sin(2\pi ft), 0)$. We chose an amplitude of $A = 0.75$ and a frequency of $f = 2.0$. For this case the background is discretized with 7200 cells and the overlapped grid is discretized with a structured grid of 256 cells. The results are shown in Table 9 and Fig. 19.

In Fig. 20 the Mach isolines are plotted at different instants of time, so the location of *Grid B* is captured along its range of movement.

Table 6 Isentropic Vortex Convection. Accuracy orders for the 3rd order Chimera method for the case of Grid B following the vortex

| N	L_1^N error	O_1^N	L_2^N error	O_2^N	L_∞^N error	O_∞^N
Isentropic vortex convection. Grid B follows the vortex						
1244	1.32×10^{-3}	–	3.95×10^{-2}	–	4.91×10^{-2}	–
2780	4.49×10^{-4}	2.69	1.50×10^{-2}	2.41	2.05×10^{-2}	2.12
4924	1.98×10^{-4}	2.85	7.00×10^{-3}	2.66	9.41×10^{-3}	1.76
11036	6.20×10^{-5}	2.88	2.22×10^{-3}	2.84	2.98×10^{-3}	2.40
19580	2.73×10^{-5}	2.87	9.63×10^{-4}	2.92	1.27×10^{-3}	2.65
43964	9.31×10^{-6}	2.66	2.93×10^{-4}	2.94	3.56×10^{-4}	2.62
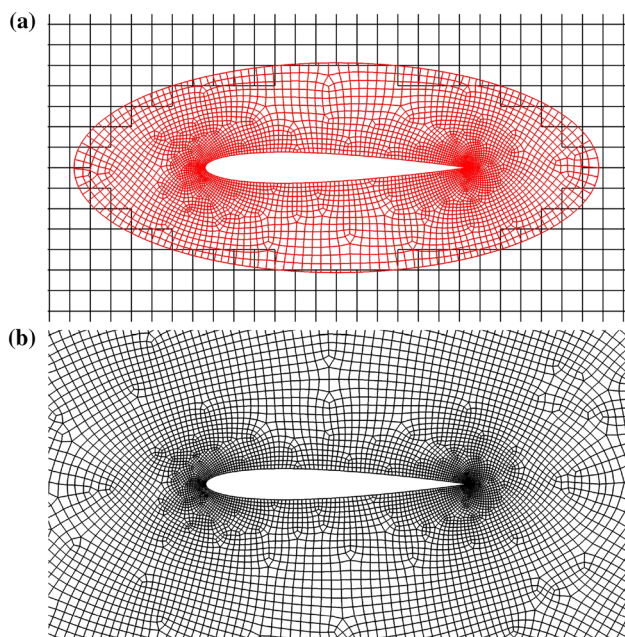
Fig. 12 Subsonic Inviscid flow around a NACA 0012. Close view **a** Non-overlapped cells and **b** Single mesh configurations

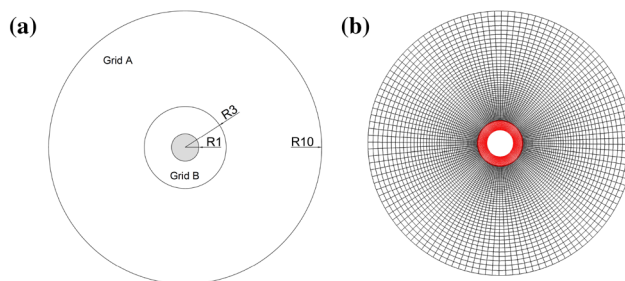**Table 7** Subsonic Inviscid flow around a NACA 0012 airfoil. Lift (C_L) and Drag (C_D) coefficients

Method	C_L	C_D
Chimera configuration	0.3336	2.81×10^{-4}
Single mesh	0.3335	2.53×10^{-4}
Theoretical results [15]	0.335	0
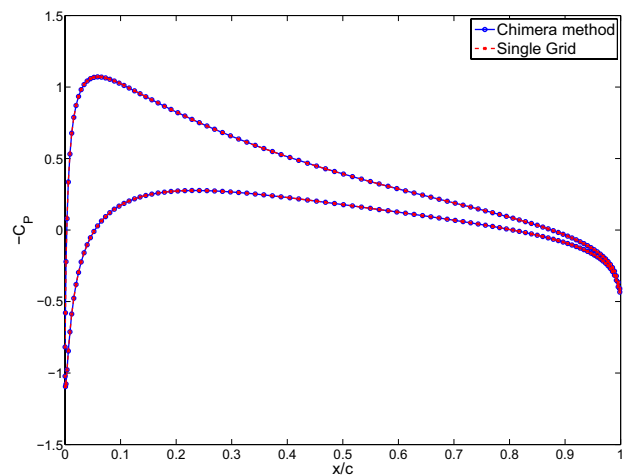
Fig. 13 Subsonic Inviscid flow around a NACA 0012. Pressure coefficient distribution around the airfoil surface
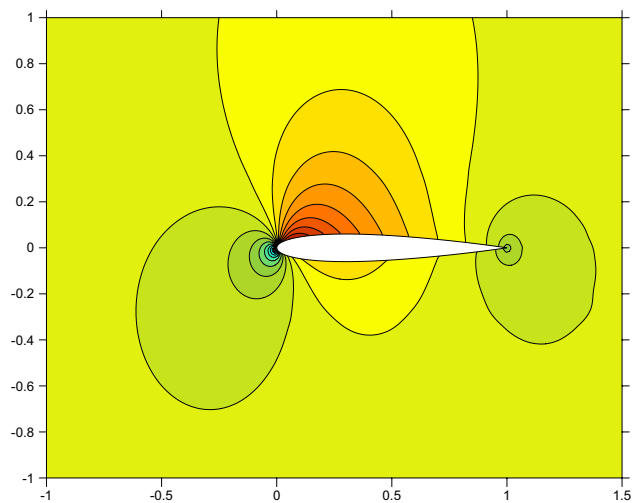
Fig. 14 Subsonic Inviscid flow around a NACA 0012. Mach field and contours

Fig. 15 Supersonic flow over a cylinder with static configuration. **a** Geometry description and **b** non-conformal grids

Table 8 Supersonic Inviscid flow around a cylinder with static configuration

Method	$p_0/(p)_\infty$	Stand-off distance/D
Single mesh	11.888	0.415
Chimera mesh	11.886	0.416
Reference solution [30]	12.061	–

5.5 Steady $Re = 40$ Flow Around a 2D Cylinder

In this test case we consider the steady flow around a circular cylinder as a validation case of the proposed Chimera method for viscous flows. The freestream Mach number is $M_\infty = 0.1$ and the Reynolds number is set as $Re = 40$. The diameter of the cylinder is $D = 1$. We impose no-slip boundary condition and adiabatic boundary condition at
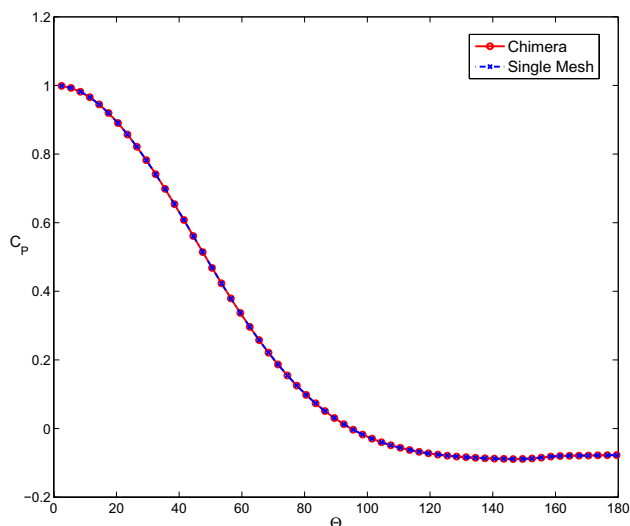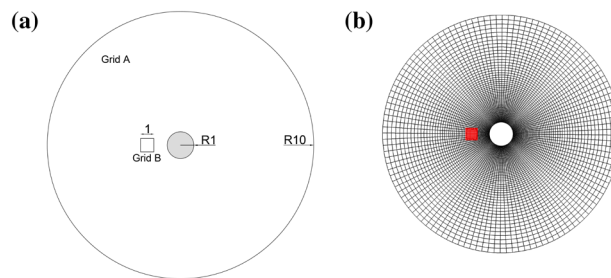
Fig. 16 Comparison of the pressure coefficient C_P distribution around the cylinder
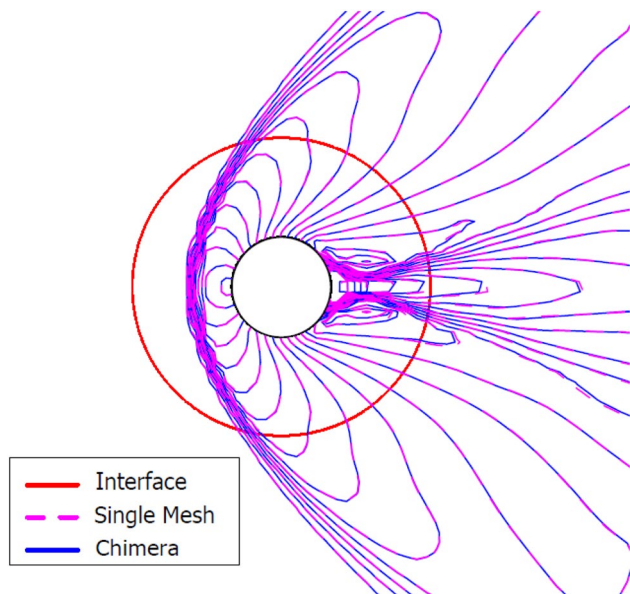
Fig. 17 Supersonic flow around a cylinder with static configuration. Comparison of Mach isolines. The location of the interface between grids is highlighted in red

Fig. 18 Supersonic flow over a cylinder with prescribed motion. **a** Geometry description and **b** non-conformal grids

Table 9 Supersonic Inviscid flow around a cylinder with prescribed motion

| Method	$p_0/(p)_\infty$	Stand-off distance/D
Single mesh	11.888	0.415
Chimera mesh	11.926	0.414
Reference solution [30]	12.061	–
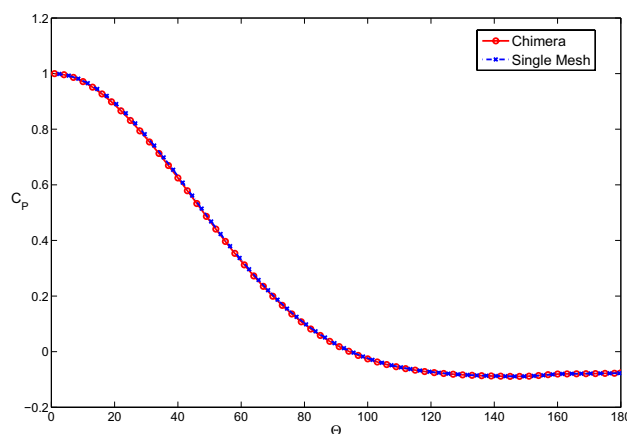
Fig. 19 Comparison of the pressure coefficient C_P distribution around the cylinder for the supersonic flow around the cylinder with prescribed motion

solid walls. In Fig. 21 we show a close view of the non-overlapped cells close to the cylinder. The computational domain is discretized with two grids, A and B, with 10114 and 3600 cells respectively, and 200 elements along the cylinder surface.

In Fig. 22 we present the velocity field and the streamlines obtained with the present Chimera method. A pair of symmetric contra-rotating vortex appear at the wake, showing that the Chimera method obtains the right wake pattern.

In Table 10 we show some of the geometrical parameters of the closed wake obtained with the Chimera method: the separation angle θ_s at the cylinder surface, the length of the circulation region (L), and the coordinates (a, b) of the vortex center. These parameters are shown schematically in Fig. 22. In addition we show the drag coefficient (C_D) and front and back pressure coefficients ($C_p(0)$ and $C_p(\pi)$). The results obtained by the new Chimera method agree well with previous computations performed by using single-grid methods [21, 31, 32] (Fig.23)
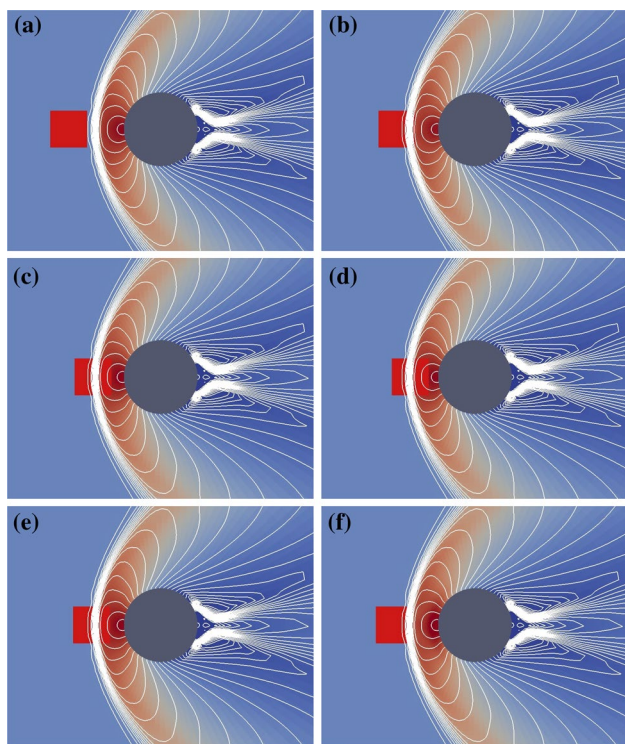
Fig. 20 Comparison of Mach isolines for the supersonic flow around a cylinder with prescribed motion. The overset grid is highlighted in red
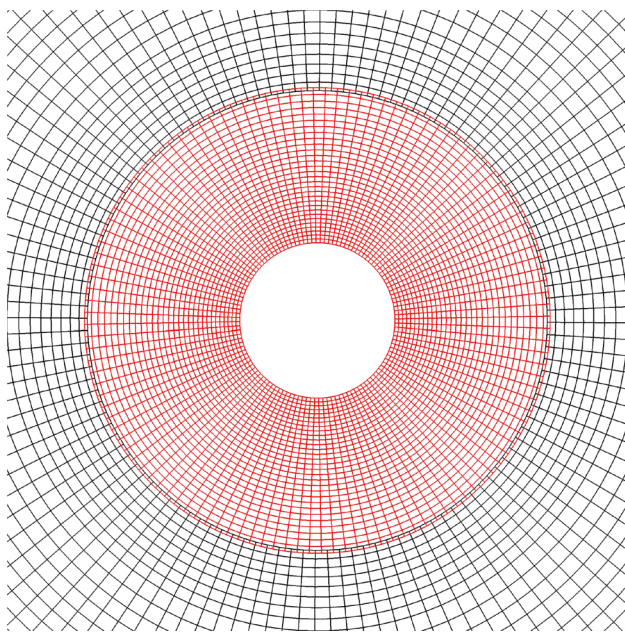
Fig. 21 Steady Laminar flow around a 2D cylinder. Non-overlapped cells
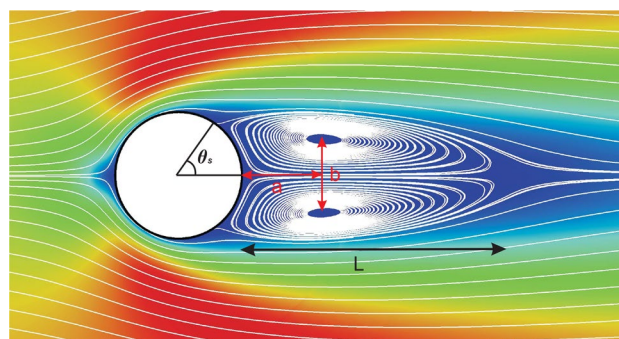
Fig. 22 Steady Laminar flow around a cylinder. Velocity contours and streamline pattern of the steady enclosed wake flow around a cylinder at Re = 40 obtained using the MLS-based Chimera method. We also show several geometrical parameters

### 5.6 Flow Past an Oscillating Circular Cylinder

In this section we address the problem of a cylinder that oscillates transversely (cross-flow) in a free stream. This case shows the ability of the proposed scheme for the simulation of moving boundary flow problems. This test case has been widely studied to validate and analyze moving boundary methods [33–37]. Following the work of Guilmineau and Queutey [33], the trajectory of the center of the circular cylinder (denoted as (x_B, y_B)) is imposed as $x_B = 0$ and $y_B = A \sin(2\pi f_e t)$, where A is the amplitude of oscillation and (f_e) is the oscillation frequency. In this work we analyze an amplitude of $A = 0.2$ and two frequencies $f_e = 0.8f_0$ and $f_e = 1.1f_0$, where the quantity f_0 denotes the natural frequency of vortex shedding for a static test case. The flow conditions are $M_\infty = 0.1$ and $Re = 185$. In a preliminary simulation the static case was computed until the vortex shedding reached an stable periodic state and we obtained a Strouhal number $S_t = \frac{f_0 D}{u_\infty} = 0.195$. The computational domain is discretized with the same mesh of the previous test case (shown in Fig. 21).

5.6.1 Prescribed Motion: $f_e = 0.8f_0$

In this configuration, the frequency of vortex shedding synchronizes with the frequency of an imposed perturbation, leading to a periodic state [33]. In Fig. 24, the vorticity contours are plotted at different times of one period of the prescribed motion (as indicated in Fig. 23). The instantaneous pressure coefficient around the cylinder is plotted in Fig. 25 and compared with the results obtained by Guilmineau and Queutey [33]. Minor discrepancies are observed, due to the use of different grids. Note that the grid used in the present computations is 4 times coarser than the one employed by Guilmineau and Queutey [33].

Table 10 Steady Laminar flow around a cylinder. Comparison of results using the new Chimera scheme with other computations using single-grid methods

| Method	C_D	L/R	$2b/D$	$2a/D$	θ_s	$C_p(0)$	$C_p(\pi)$
Present method	1.568	4.20	1.168	1.264	52.69 deg	−0.512	1.180
Chassaing et al. [21]	1.565	4.3	1.17	1.34	52.71 deg	−0.516	1.205
Niu et al. [31]	1.574	–	–	–	–	−0.555	1.147
He and Doolen [32]	1.499	4.49	–	–	52.89 deg	−0.487	1.133
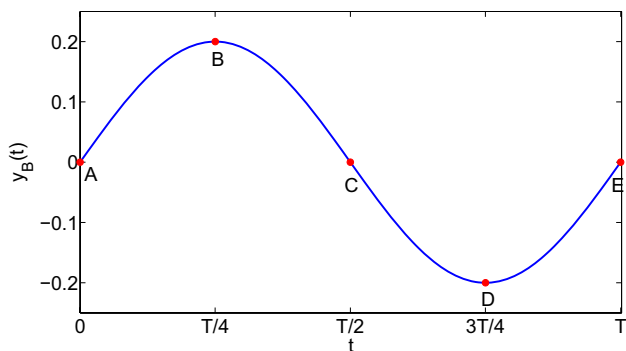
Fig. 23 Spatial location of the different snapshots shown in Fig. 24

The periodic variations of the drag and lift coefficients are plotted as a function of the vertical position of the cylinder in Fig. 26. In order to compare with other references, in Table 11 we show the time-average drag coefficient ($\overline{C}_D$), the root mean square values of the drag and lift coefficients ($C_{D,rms}$ and $C_{L,rms}$) and the time-average pressure drag coefficient ($\overline{C}_{D,p}$). The results are in good agreement with others obtained with different approaches.

5.6.2 Prescribed Motion: $f_e = 1.1f_0$.

Next, the frequency of the prescribed motion is increased to $f_e = 1.1f_0$. The drag and lift coefficients (C_D and C_L) versus time are plotted in Fig. 27. For $f_e/f_0 > 1$ the drag and lift coefficient exhibits the influence of a higher harmonic [38]. In Fig. 28 the vorticity contours are compared for the prescribed motions $f_e = 0.8f_0$ and $f_e = 1.1f_0$ when the cylinder is located at the extreme upper position. A close view around the cylinder is plotted in Fig. 29. The wake topology presents a different structure when compared with the one obtained for $f_e = 0.8f_0$.
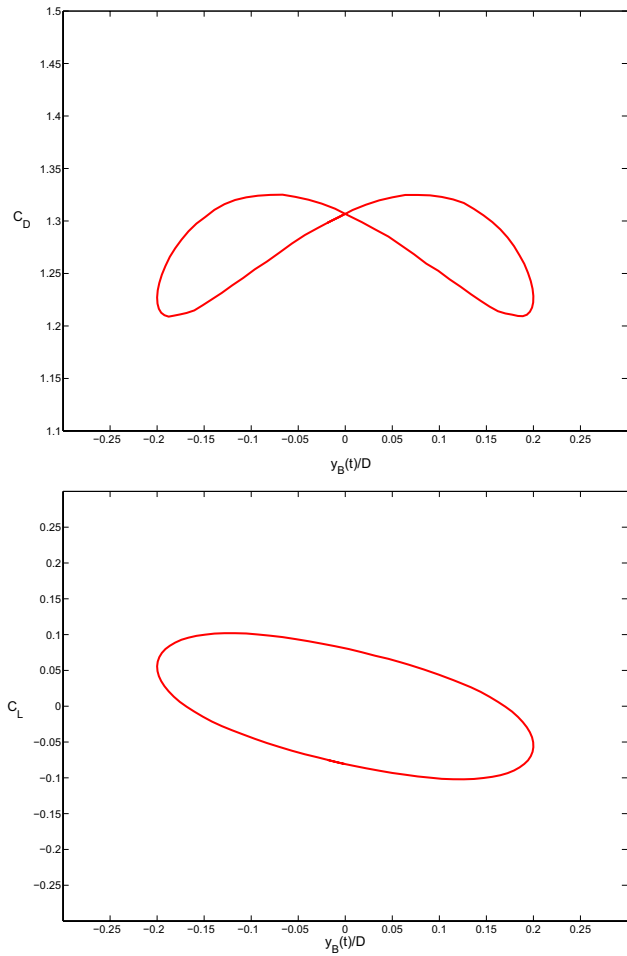
In order to compare with other references, in Table 12 we show the time-average drag coefficient ($\overline{C}_D$), the root mean square values of the drag and lift coefficients ($C_{D,rms}$ and $C_{L,rms}$). Again, the results are in good agreement with those obtained using finer grids.
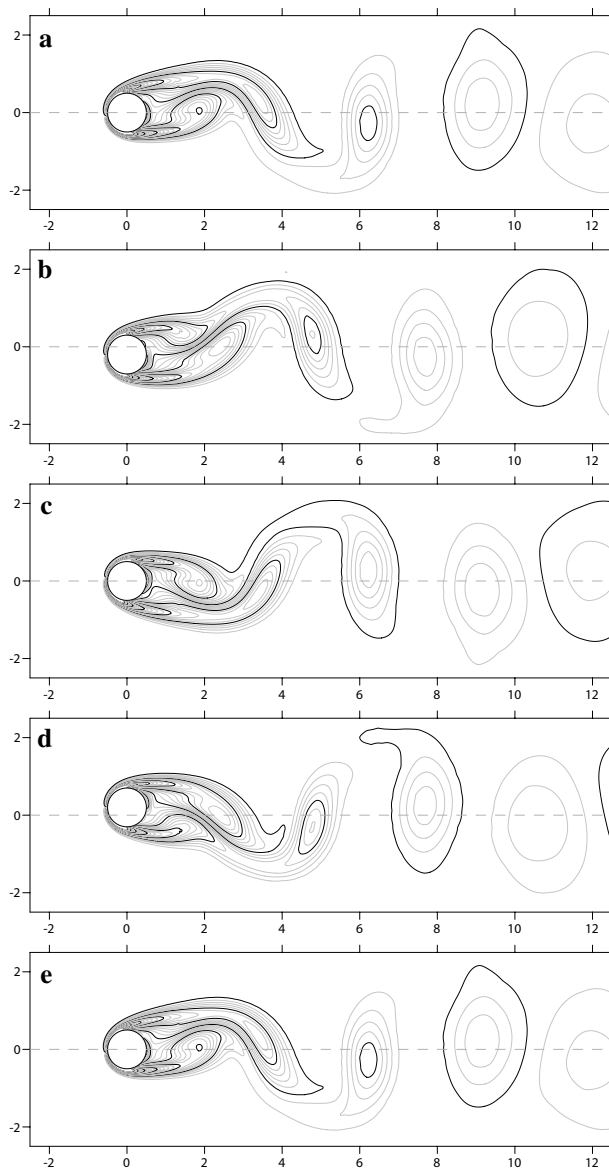
Fig. 24 Flow past an oscillating circular cylinder: Prescribed motion with $f_e = 0.8f_0$. Vorticity contours at different instants of time of the period, as indicated in Fig. 23
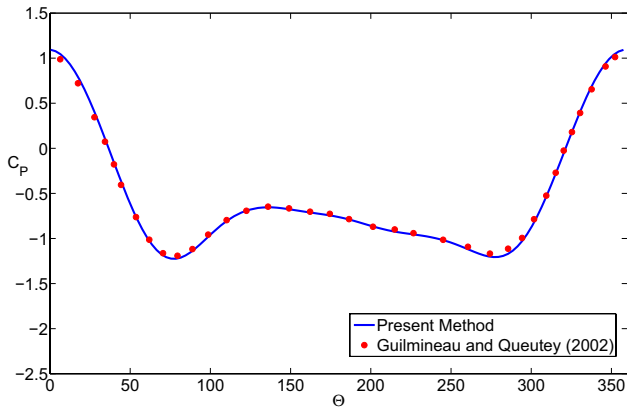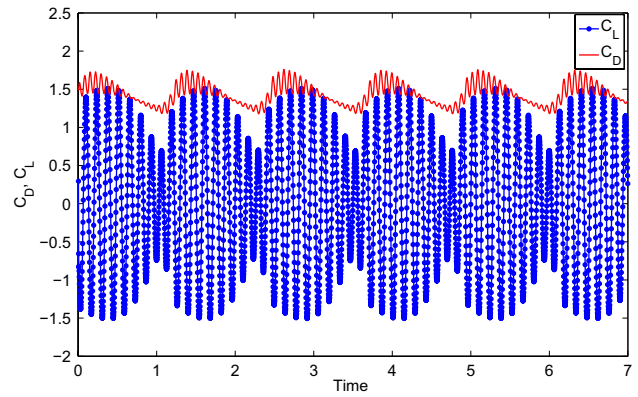
Fig. 25 Flow past an oscillating circular cylinder: Prescribed motion with $f_e = 0.8f_0$. Comparison of the instantaneous pressure coefficient C_P distribution around the cylinder with the results of Guilmineau and Queutey [33]. The location of Grid B is $(x_B, y_B) = (0, 0)$ while moving downwards
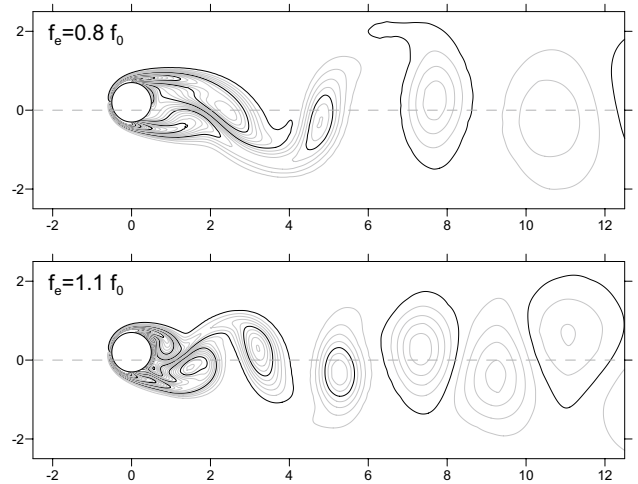
Table 11 Flow past an oscillating circular cylinder: Prescribed motion with $f_e = 0.8f_0$. The time-average drag coefficient ($\overline{C}_D$), root mean square values of the drag and lift coefficients ($C_{D,rms}$ and $C_{L,rms}$) and the the time-average pressure drag coefficient ($\overline{C}_{D,p}$)

Method	$\overline{C}_D$	$C_{D,rms}$	$C_{L,rms}$	$\overline{C}_{D,p}$
Present method	1.267	0.041	0.071	1.017
Guilmineau et al. [33]	1.195	0.036	0.08	–
Schneiders et al. [34]	1.279	0.042	0.082	1.027
Uhlmann [35]	1.354	–	0.166	–
Yang et al. [37]	1.281	0.042	0.076	–

Fig. 27 Flow past an oscillating circular cylinder: Prescribed motion with $f_e = 1.1f_0$. Drag and lift coefficients (C_D and C_L) versus time

Fig. 26 Flow past an oscillating circular cylinder: Prescribed motion with $f_e = 0.8f_0$. Time periodic drag and lift coefficients (C_D and C_L)

Fig. 28 Flow past an oscillating circular cylinder. Comparison of the vorticity contours for the prescribed motions $f_e = 0.8f_0$ and $f_e = 1.1f_0$ when the cylinder is located at its extreme upper position. Note the different structure of the wakes
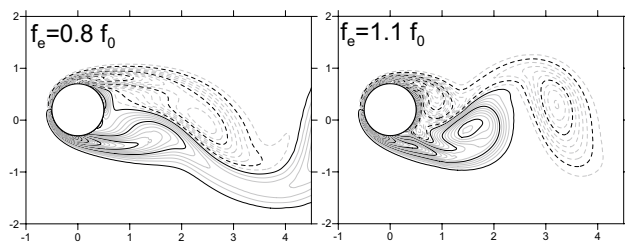
Fig. 29 Flow past an oscillating circular cylinder. Close view of the comparison of the vorticity contours for the prescribed motions $f_e = 0.8f_0$ and $f_e = 1.1f_0$ when the cylinder is located at its extreme upper position. Vorticity negative values are indicated in *dashed pattern*

Table 12 Flow past an oscillating circular cylinder: Prescribed motion with $f_e = 1.1f_0$. The time-average drag coefficient ($\overline{C}_D$) and the root mean square values of the drag and lift coefficients ($C_{D,rms}$ and $C_{L,rms}$)

Method	$\overline{C}_D$	$C_{D,rms}$	$C_{L,rms}$
Present method	1.427	0.143	0.906
Guilmineau and Queutey [33]	1.420	0.149	0.897
Wu and Shu [36]	1.454	–	0.854

6 Conclusions

In this work we have proposed a new higher-order accurate Chimera method for overlapped arbitrary grids. Moving Least Squares approximations are used to transfer the data from one grid to another. The geometrical flexibility and accuracy of the MLS approximation allow for the transfer of information between overlapped grids preserving the order (> 2) of the scheme. The new Chimera method is presented in the framework of higher-order finite volume schemes for compressible flows. The numerical examples have shown that the present methodology is applicable to a wide variety of problems, including problems where the solution presents discontinuities. In spite of that, accurate results free of numerical artifacts are obtained.

Compliance with Ethical Standards

Conflict of interest The authors declare that they have no conflict of interest.

Research Involving Human and Animal Participants The research does not involve neither human participants nor animals.

Informed Consent All the authors are informed and provided their consent.

References

1. Ramirez L, Foulquié C, Nogueira X, Khelladi S, Chassaing JC, Colominas I (2015) New high-resolution-preserving sliding mesh techniques for higher-order finite volume schemes. Comput Fluids 118:114–130
2. Ferrer E, Willden RHJ (2012) A high order discontinuous Galerkin-Fourier incompressible 3d Navier-Stokes solver with rotating sliding meshes. J Comput Phys 231:7037–7056
3. Peskin CS (1972) Flow patterns around heart valves: a numerical method. J Comput Phys 10:252–271
4. Ouro P, Cea L, Ramirez L, Nogueira X (2016) An immersed boundary method for unstructured meshes in depth averaged shallow water models. Int J Numer Methods Fluids 81:672–688
5. Cristallo A, Verzicco R (2006) Combined immersed boundary/large-eddy-simulations of incompressible three dimensional complex flows. Flow Turbul Combust 77:3–26
6. Kang C, Iaccarino G, Ham F (2009) DNS of buoyancy-dominated turbulent flows on a bluff body using the immersed boundary method. J Comput Phys 228:3189–3208
7. Clarke D, Salas M, Hassan H (1986) Euler calculations for multi-elements airfoils using Cartesian grids. AIAA J 24:353–358
8. Steger J, Dougherty F, Benek J (1982) A chimera grid scheme. ASME Mini-Symposium on Advances in Grid Generation, Houston
9. Lee KR, Park JH, Kim KH (2011) High-order interpolation method for overset grid based on finite volume method. AIAA J 49:1387–1398
10. Sherer SE, Scott JN (2005) High-order compact finite-difference methods on general overset grids. J Comput Phys 210:459–496
11. Wang G, Duchaine F, Papadogiannis D, Duran I, Moreau S, Gicquel LYM (2014) An overset grid method for large eddy simulation of turbomachinery stages. J Comput Phys 274:343–355
12. Delfs JW (2001) An overlapped grid technique for high resolution CAA schemes for complex geometries. AIAA paper 2001–2199
13. Lancaster P, Salkauskas K (1981) Surfaces generated by moving least squares methods. Math Comput 37(155):141–158
14. Liu WK, Hao W, Chen Y, Jun S, Gosz J (1997) Multiresolution reproducing kernel particle methods. Comput Mech 20:295–309
15. Nogueira X, Cueto-Felgueroso L, Colominas I, Gómez H, Navarrina F, Casteleiro M (2009) On the accuracy of finite volume and discontinuous Galerkin discretizations for compressible flow on unstructured grids. Int J Numer Methods Eng 78:1553–1584
16. Nogueira X, Khelladi S, Colominas I, Cueto-Felgueroso L, París J, Gómez H (2011) High-resolution finite volume methods on unstructured grids for turbulence and aeroacoustics. Arch Comput Methods Eng 18(3):315–340
17. Cueto-Felgueroso L, Colominas I, Nogueira X, Navarrina F, Casteleiro M (2007) Finite volume solvers and moving least-squares approximations for the compressible Navier-Stokes equations on unstructured grids. Comput Methods Appl Mech Eng 196:4712–4736
18. Khelladi S, Nogueira X, Bakir F, Colominas I (2011) Toward a higher-order unsteady finite volume solver based on reproducing Kernel particle method. Comput Methods Appl Mech Eng 200:2348–2362

19. Nogueira X, Cueto-Felgueroso L, Colominas I, Khelladi S (2010) On the simulation of wave propagation with a higher order finite volume scheme based on reproducing Kernel methods. Comput Methods Appl Mech Eng 199(155):1471–1490

20. Nogueira X, Ramirez L, Khelladi S, Chassaing JC, Colominas I (2016) A high-order density-based finite volume method for the computation of all-speed flows. Comput Methods Appl Mech Eng 298:229–251

21. Chassaing JC, Khelladi S, Nogueira X (2013) Accuracy assessment of a high-order moving least squares finite volume method for compressible flows. Comput Fluids 71:41–53

22. Liu GR, Liu MB (2003) Smoothed particle hydrodynamics. A meshfree particle method. World Scientific Publishing, Singapore

23. Liu GR, Gu YT (1999) A point interpolation method. Proc 4th Asia-Pacific Conference on Computational Mechanics, Singapore: 1009–1014

24. Liu GR, Gu YT, Dai KY (2004) Assessment and applications of point interpolation methods for computational mechanics. Int J Numer Methods Eng 59:1373–1397

25. Most T, Bucher C (2008) New concepts for moving least squares: an interpolation non-singular weighting function and weighted nodal least squares. Eng Anal Bound Elements 32:461–470

26. Chiocchia G (1985) Exact solutions to transonic and supersonic flows.Technical Report AR-211

27. Galbraith MC, Benek JA, Orkwis PD, Turner MG (2014) A discontinuous Galerkin Chimera scheme. Comput Fluids 98:27–53

28. Nogueira X, Cueto-Felgueroso L, Colominas I, Navarrina F, Casteleiro M (2010) A new shock-capturing technique based on moving least squares for higher-order numerical schemes on unstructured grids. Comput Methods Appl Mech Eng 199:2544–2558

29. Van Albada GD, Van Leer B, Roberts WW (1982) A comparative study of computational methods in cosmic gas dynamics. Astron Astrophys 108:76–84

30. Liepmann HW , Roshko A (2002) Elements of gasdynamics. Dover

31. Niu XD, Chew YT, Shu C (2003) Simulation of flows around an impulsively started circular cylinder by taylor series expansion- and least squares-based lattice boltzmann method. J Comput Phys 188:176–193

32. He X, Doolen GD (1997) Lattice boltzmann method on curvilinear coordinates system: flow around a circular cylinder. J Comput Phys 134:306–315

33. Guilmineau E, Queutey P (2002) A numerical simulation of vortex shedding from an oscillating circular cylinder. J Fluids Struct 16:773–794

34. Schneiders L, Hartmann D, Meinke M, Schröder W (2013) An accurate moving boundary formulation in cut-cell methods. J Comput Phys 235:786–809

35. Uhlmann M (2005) An immersed boundary method with direct forcing for the simulation of particle flows. J Comput Phys 209:448–476

36. Wu YL, Shu C (2008) Application of local DFD method to simulate unsteady flows around an oscillating circular cylinder. Int J Numer Methods Fluids 58:1223–1236

37. Yang X, Zhang X, Li Z, He GW (2009) A smoothing technique for discrete delta functions with application to immersed boundary method in moving boundary simulations. J Comput Phys 228:7821–7836

38. Yang J, Balaras E (2005) An embedded-bounday formulation for large-eddy simulation of turbulent flows ointeracting with moving boundaries. J Comput Phys 215:12–40