ON APPLICATION OF OVERSET/CHIMERA METHOD FOR FLOW APPROXIMATION OVER A VIBRATING BODY USING OPENFOAM

O. Winter, P. Sváček

1 Department of Technical Mathematics, Faculty of Mechanical Engineering, Czech Technical University, Karlovo náměstí 13, Praha 2, Czech Republicy

Abstract

The subject of this contribution is the numerical simulation of fluid-structure interaction problems with aid of overset method. The numerical simulation is based on the finite volume method and the arbitrary Lagrangian Eulerian method is used. Numerical results obtained by the open-source code OpenFOAM for approximation of the flow around both static and oscillating circular cylinder and oscillating NACA 0012 profile are shown.

Keywords: OpenFOAM, FVM, overset, oscillating NACA 0012

1 Introduction

Fluid-structure interaction (FSI) approximation is a challenging task, which includes moving boundaries and deformable structures. Numerous techniques and algorithms have been developed to deal with FSI problems, see e.g. [1]. Despite of existing monolithic solvers (or strongly-coupled schemes), the loosely-coupled algorithms are still widely used, see [2]. Mesh movement is realized quite often using the arbitrary Lagrangian-Eulerian (ALE) formulation, see [3]. One of the biggest drawbacks of the ALE based methods is the requirement of the mesh deformation. Eventhough it can be quite generally achieved by solving some artificial problem for the mesh displacement (see [4]), the applicability of such approach is limited. In many cases the deformation of the mesh leads to mesh distortion and a remeshing procedure is needed.

There is another possibility of using the immersed boundary methods. The immersed boundary method was introduced to study flow patterns around heart valves (cf. [5]) and has evolved into a generally useful method for problems of fluid-structure interaction, see [6]. Yet another possibility is to perform computation on overlapping (overset/chimera/composite) grids, see [7], [8]. The overset method was originally developed to address complex geometric configurations, see [9]. The overset method was being developed and nowadays it is a generic method for fluid-structure interaction problems capable of solving quite general problems such as turbine aerodynamics [10], ship engineering [11], or dragonfly aerodynamics [12]. Main advantage of the overset method is the capability to run computations on quite complex geometries with accurate representations of the boundary layers. Furthermore, the overset grids provide easy way to allow moving geometry in computations, see [11].

Particularly, the aeroelastic computations for the purpose of analyzing flutter stability requires flow approximation over a moving or vibrating structure. Aeroelastic calculations can often be very complex problem with necessity to describes complex geometry consisting of more rigid (or elastic) bodies. In this paper, the applicability of the overset method is tested.

The paper is structured as follows: the mathematical model is presented then the numerical approximation is described, and finally the numerical results are discussed. Calculation of the fluid flow around static two dimensional circular cylinder, the computation of the fluid flow around oscillating two dimensional circular cylinder and the simulation of the fluid flow past the oscillating NACA 0012 profile are shown. All cases are simulated both with mesh deformation procedure and overset method, the comparison of the results is shown and discussed. The described numerical methods are realized using the ESI OpenCFD group version of the OpenFOAM.

2 Mathematical Model

The flow over a vibrating rigid structure is considered here. Although in general, when dealing with aeroelastic problems, these vibration are induced by the fluid flow and also coupled problem
2.1 Navier-Stokes Equations

Let us denote the computational domain $\Omega_t \subset \mathbb{R}^2$ occupied by the fluid at time instant $t$, see Figure 1. It is assumed that $\Omega_t$ is a polygonal domain for any $t \in (0,T)$. The boundary of $\Omega_t$ is decomposed into distinct parts $\partial \Omega_t = \Gamma_I \cup \Gamma_D \cup \Gamma_W \cup \Gamma_O$. $\Gamma_I$ is the inlet part of the boundary $\partial \Omega_t$, $\Gamma_O$ is the outlet part of the boundary $\partial \Omega_t$, $\Gamma_W$ is the moving surface, $\Gamma_D$ is the non-penetrable wall, see Figure 1. The incompressible fluid flow is described by the system of the Navier-Stokes equations, i.e.,

$$
\frac{\partial \mathbf{v}}{\partial t} + \nabla \cdot (\mathbf{v} \otimes \mathbf{v}) = -\nabla p + \nu \Delta \mathbf{v}, \quad \nabla \cdot \mathbf{v} = 0,
$$

(1)

where $\mathbf{v}$ denotes the fluid flow velocity vector, $p$ denotes the kinematic pressure, i.e., pressure divided by the constant fluid density $\rho$ and $\nu$ denotes the kinematic viscosity of the fluid, i.e., the viscosity of the fluid $\mu$ divided by $\rho$. The system (1) is equipped with the boundary conditions

$$
a) \; \mathbf{v} = \mathbf{v}_I \text{ on } \Gamma_I, \quad b) \; \mathbf{v} = \mathbf{w} \text{ on } \Gamma_W, \quad c) \; \mathbf{v} = \mathbf{0} \text{ on } \Gamma_D,
$$

(2)

where $\mathbf{v}_I$ is the inlet velocity, $\mathbf{w}$ denotes the velocity of the boundary $\Gamma_W$. On the $\Gamma_D$ no-slip boundary condition is used. On the outlet part of the $\partial \Omega_t$ the do-nothing boundary condition can be used, but the used outlet boundary condition is realized by the numerical approximation of FVM.

2.2 ALE Formulation in Conservative Form

The time dependency of the computational domain $\Omega_t$ is treated with the aid of the ALE method, see [13], [3]. This method is based on the use of a smooth, one-to-one ALE mapping $\mathcal{A}_t : \Omega_0 \rightarrow \Omega_t$, $\mathbf{x} = \mathcal{A}_t(\mathbf{\xi})$ for any $\mathbf{\xi} \in \Omega_0$ and $t \in (0,T)$, where $\mathbf{x}$ are called the spatial coordinates and $\mathbf{\xi}$ are called the ALE coordinates. By $\mathbf{w} = \mathbf{w}(\mathbf{x},t)$ the domain velocity, i.e., the velocity of the point with a given reference $\mathbf{\xi} \in \Omega_0$, is denoted and by $D^A f / Dt$ the ALE derivative, i.e., the derivative with respect to a fixed point $\mathbf{\xi} \in \Omega_0$, is denoted, see [13]. Further by $J$ the Jacobian of the ALE mapping is denoted, which satisfies

$$
\frac{1}{J} \frac{D^A f}{Dt} = \text{div}(\mathbf{w}).
$$

(3)
Using this notations in the system of equations (1), the ALE conservative form of the Navier-Stokes equations reads
\[
\frac{1}{J} \frac{D^A(Jv)}{Dt} + \nabla \cdot (v \otimes (v - w)) = -\nabla p + \nu \Delta v, \quad \nabla \cdot v = 0. \tag{4}
\]
For the system of equations (4) the same boundary conditions as for the system of equations (1) are used.

3 Numerical Approximation

This section describes the time discretization, which is dealt with the second order backward differencing formula (BDF2). Next, the finite volume method is described together with the solution algorithm, the merged SIMPLE/PISO procedure is used, see [14]. Finally, the overset method is briefly described.

3.1 Time Discretization

Consider an equidistant partition \(0 = t_0 < t_1 < t_2 \ldots < T\) of the time interval \((0, T)\) with a constant time step \(\tau > 0\). The solution \(v(t_n), p(t_n)\), the domain velocity \(w(t_n)\) and the Jacobian of the ALE mapping \(J(t_n)\) defined in \(\Omega_n\) are approximated at time \(t_n\) by \(v^n, p^n, w^n\) and \(J^n\), respectively. For a time discretization, the BDF2 scheme is used. Using the computed approximate solution \(v^{n-1}\) in \(\Omega_{n-1}\) and \(v^n\) in \(\Omega_n\), the ALE derivative is approximated by
\[
\frac{1}{J} \frac{D^A(Jv)}{Dt} \mid_{x,t=t_{n+1}} \approx \frac{1}{\tau J^{n+1}} \left[ \frac{3}{2} v^{n+1} J^{n+1} - 2 v^n J^n + \frac{1}{2} v^{n-1} J^{n-1} \right], \quad \text{in} \ \Omega_{n+1}. \tag{5}
\]
where \(v^n = v^n(A_{t_n}(\xi)), J^n = \frac{\partial A_{t_n}(\xi)}{\partial \xi}\) and \(v^{n-1} = v^{n-1}(A_{t_{n-1}}(\xi)), J^{n-1} = \frac{\partial A_{t_{n-1}}(\xi)}{\partial \xi}\). Using a moving set \(V(t)\) equation (5) can be written in the integral form as
\[
\int_{V(t)} \frac{1}{J} \frac{D^A(Jv)}{Dt} \, dx = \frac{d}{dt} \int_{V(t)} v \, dx \approx \frac{1}{\tau} \left[ \frac{3}{2} \int_{V(t_{n+1})} v \, dx - 2 \int_{V(t_n)} v \, dx + \frac{1}{2} \int_{V(t_{n-1})} v \, dx \right]. \tag{6}
\]

3.2 Finite Volume Method

For space discretization the finite volume method (FVM) is used. In order to apply the finite volume method the computational domain \(\Omega_n\) at time \(t_n\) is approximated by a set of closed non overlapping polygons (cells) \(\mathcal{M} = \{V^p_n\}_{P \in M_n}\), where \(M_n = \{1, 2, 3, \ldots, n_h\}\) is the index set. The system \(\mathcal{M} = \{V^p_n\}_{P \in M_n}\) is called the finite volume mesh of \(\Omega_n\), if \(\Omega_n = \cup_{P \in M_n} V^p_n\). Further, a boundary of each cell consists of straight line segments called faces. The faces can either be shared with another neighbour cell or be a part of the boundary \(\partial \Omega_n\). By \(\mathcal{N}(P)\) the set of all neighbour cells of the cell \(P\) is denoted and by \(\mathcal{F}(P)\) the set of all faces of the cell \(P\) is denoted. For each face \(f \in \mathcal{F}(P)\) by \(\mathcal{S}^f\) the area vector outwards to the cell \(P\) and normal to the face \(f\) is denoted. Furthermore, we introduce following notations: \(|V^p_n|\) denotes the volume of the cell \(P\), \(x^p_n\) denotes the location of centroid of the cell \(V^p_n\).

Using introduced notation for time discretization, the solution \(v^n(x_P), p^n(x_P)\) and the domain velocity \(w^n(x_P)\) are approximated as \(v^n_P \approx \frac{1}{|V^p_P|} \int_{V^p_P} v^n(x) \, dx\), \(p^n_P \approx \frac{1}{|V^p_P|} \int_{V^p_P} p^n(x) \, dx\), and \(w^n_P \approx \frac{1}{|V^p_P|} \int_{V^p_P} w^n(x) \, dx\), respectively.

3.3 Merged SIMPLE/PISO Algorithm

For the sake of brevity the discretization of the system of the equations (4) is described here only for the case of fixed domain, i.e., \(\Omega_n = \Omega_{n-1}\) for arbitrary \(n\) and the motion of the computational domain is neglected as well as the domain velocity \(w\).
Discrete Navier-Stokes Equations. The FVM discretization uses the integral form of equation (1) over the cell around point $P$ with the volume $|V_P|$. Using Gauss’ formula, discretized momentum equation (1) can be written as

$$\frac{3}{2\tau} v^n_{P,1}^n |V_P| + \sum_{f \in \mathcal{F}(P)} v^n_{f,1}^n (v^n_f \cdot S_f) - \nu \sum_{f \in \mathcal{F}(P)} (\nabla v^n_{f+1})_f \cdot S_f = - (\nabla p^n)_P + \frac{|V_P|}{\tau} \left(2v^n_P - \frac{1}{2}v^{n-1}_P\right)$$

(7)

for any $P \in M_h$, where $(v^n_f \cdot S_f)$ is volumetric flux, face values $v^n_{f+1}$ are obtained from cell centered values by upwind differencing method with second order accurate reconstruction. For more information of discretization of convective, diffusive and gradient terms, see [15]. Terms on the left hand side of equation (7) are evaluated implicitly and terms on the right hand side explicitly. The time-indices are dropped in the following for sake of brevity. The system (7) can be formally written as

$$A^n_P v^n_P = H^n_P(v^n) - (\nabla p^n)_P, \ \forall P \in M_h,$$

(8)

where $A^n_P$ is the diagonal coefficient and $H^n_P(v)$ are the remaining off-diagonal terms and the explicit terms, see the right hand side of equation (7). By division with the diagonal term $A^n_P$ the velocity can be written as

$$v^n_P = \frac{1}{A^n_P} H^n_P(v^n) - \frac{1}{A^n_P} (\nabla p^n)_P, \ \forall P \in M_h.$$

(9)

In general, solution of the equation (7) does not fulfill the continuity equation, i.e., velocity field is not divergence-free. To solve this problem, pressure equation is formulated by applying the discrete divergence operator on the equation (9) and setting the divergence of the velocity on the left hand side to zero, which finally leads to pressure equation, i.e.,

$$\sum_{f \in \mathcal{F}(P)} S_f \left[ \frac{H^n_P(v^n)}{A^n_P} \right]_f = \sum_{f \in \mathcal{F}(P)} |S_f| \frac{1}{A^n_P f} (\nabla p^n)_f.$$

(10)

Note that pressure equation is second-order PDE in pressure, which means that additional boundary conditions should be prescribed.

Solution Algorithm. We adopted a merged SIMPLE/PISO algorithm. Given known $v^n$, $p^n$, $v^{n-1}$ and $p^{n-1}$ the algorithm can be summarized as follows

1. Solve the momentum equation (7) to obtain a new velocity guess $v^*$ of $v^{n+1}$.
3. Solve the pressure equation (10) to obtain a new pressure guess $p^*$ of $p^{n+1}$.
4. Correct the velocity with $p^*$ to obtain $v^{**}$, i.e., compute $v^{**}$ from equation (9) with $H^n_P := H^n_P(v^*)$ and $p := p^*$.
5. Update $H^n_P := H^n_P(v^{**})$ using the new velocity guess $v^{**}$.
6. Solve the pressure equation (10) to get $p^{n+1}$.
7. Correct the velocity with $p^{n+1}$ to obtain $v^{n+1}$, i.e., compute $v^{n+1}$ from equation (9) with $H^n_P(v^n) := H^n_P(v^{**})$ and $p := p^{n+1}$.

In case of stability problems additional PISO correctors are used, i.e., the steps 5-7 can be performed repeatedly. Convective term is discretized in semi-implicit way and if relatively large time steps are used additional (SIMPLE loop) iterations need to be applied. Let us also mention that in case of using SIMPLE loop the under-relaxation factors need to be introduced, for details see [14].

Mesh Deformation. In case of oscillating cylinder (or an airfoil), the ALE mapping must be constructed in every time step and also the grid velocity $w$ needs to be computed. The ALE mapping is constructed using the pseudo-elastic solver, see [16]. In this way, a deformation of the mesh is obtained for any time instant $t_n$. The discrete domain velocity is calculated with the aid of the BDF2.
3.4 Overset Method

The overset composite grids method is a way of assembling multiple grids and treating them as a single computational domain, see [17]. Basically, this method consists in generating a set of sub-grids that cover the computational domain and overlap where they meet. In this way, the method offers an effective way to reduce a geometrically complex problem into a set of simple sub-grids, allowing arbitrary movement of the component grids. Each sub-grid can have three types of cells: discretization, interpolation, and hole cells. The discretization cells are used to discretize the PDE or the boundary conditions, the interpolation cells interpolate their solution from the overlapping cells, and the hole cells are disregarded during the discretization of the PDE.

The main difficulty in the use of overset methods is in the interpolation transfer between overlapping grids and how fluxes are treated at overlapping boundaries, see [17]. The concepts of donor and receptor cells are used. The cells laying in the overlapping region between two grids are identified as either donors or fringes depending on the flow direction. The donor cells provide information used for the interpolation. Values on the fringe cells are obtained from the interpolation of the donors. The interpolation stencils are generated with aid of the inverse distance weighting scheme, see [18]. Any flow variable \( \varphi \) at a fringe cell can be expressed as (see [19])

\[
\varphi = \sum \alpha_i \varphi_i,
\]

where \( \varphi_i \) and \( \alpha_i \) are the flow variables and the corresponding interpolation weights at the \( i \)th donor cell of the fringe cell.

After the interpolation stencils are generated, they are used to couple assembled system of linear algebraic equations of calculated cells. In case of moving geometry the interpolation stencils are generated at each time instant \( t_n \).

4 Numerical Results

First, the numerical simulations according to the specifications of the benchmark test in [20], i.e., the flow around circular cylinder, are done then the case with the oscillating cylinder with same specifications is realized. Finally, the results of the approximated flow past oscillating NACA 0012 are shown.

4.1 Flow Around Static Circular Cylinder

The inflow boundary condition is \( v_1(x_2) = 4 U_m x_2 (H - x_2)/H^2 \), \( v_2 = 0 \), where \( U_m = \frac{4}{3} U \), with \( U_m = 1.5 \text{ m/s} \), yielding the Reynolds number \( \text{Re} = 100 \). The setup parameters are as follows:

\[
\rho = 1 \text{ kg/m}^3, \quad \nu = 1 \times 10^{-3} \text{ m}^2/\text{s} \text{ and } U_m = 1.5 \text{ m/s}.
\]

The drag and lift forces are computed according to

\[
F = \int_{\Gamma_W} \left( -p \mathbf{1} + \rho \mathbf{u} \cdot (\nabla \mathbf{u} + (\nabla \mathbf{u})^T) \right) \cdot \mathbf{n} \, dS, \quad F = (F_d, F_l),
\]

where \( F_l \) is lift force and \( F_d \) is drag force and \( \mathbf{n} \) is the unit outer normal to \( \partial \Omega_t \) on \( \Gamma_W \) (pointing into the cylinder). The drag \( c_d \), lift \( c_l \) coefficients and Strouhal number are

\[
c_d = \frac{2 F_d}{\rho \mathbf{U}^2 D}, \quad c_l = \frac{2 F_l}{\rho \mathbf{U}^2 D}, \quad \text{St} = \frac{D f}{\mathbf{U}},
\]

where \( f \) is frequency of separation. As a further reference value the pressure difference \( \Delta p(t) = p(x_a, y_a, t) - p(x_e, y_e, t) \) is defined, with the front and end point of the cylinder \( (x_a, y_a) = (0.15, 0.2) \) and \( (x_e, y_e) = (0.25, 0.2) \), respectively. The computations were run on three successively refined grids for both approaches, see Table 1, where U stands for unstructured grid, i.e., standard FVM and O for the overset method.

Table 1 shows obtained reference values, one can see that values obtained by both methods correspond well to each other, but there are some quantitative differences mainly in \( c_l \) and \( \Delta p \). These differences are mainly caused by the mesh structure, e.g., size and density of the overset mesh around the cylinder. Figure 2 shows lift coefficient \( c_l \) and drag coefficient \( c_d \) during the computation, results for both approaches agrees very well.
Figure 2: Lift coefficient $c_l(t)$ (a) and drag coefficient $c_d(t)$ (b) for the case of flow around static cylinder.

Table 1: Overview of obtained reference values for the case of flow past static cylinder. Lower and upper bounds are taken from [20].

<table>
<thead>
<tr>
<th>Mesh</th>
<th>$c_{d,\text{max}}$</th>
<th>$c_{l,\text{max}}$</th>
<th>$\text{St}$</th>
<th>$\Delta p(8)$</th>
<th>Cells</th>
<th>$\tau$</th>
</tr>
</thead>
<tbody>
<tr>
<td>U1</td>
<td>3.1933</td>
<td>9.4497 \cdot 10^{-1}</td>
<td>0.287</td>
<td>2.4755</td>
<td>10 844</td>
<td>1 \cdot 10^{-2}</td>
</tr>
<tr>
<td>U2</td>
<td>3.2223</td>
<td>9.7247 \cdot 10^{-1}</td>
<td>0.294</td>
<td>2.4633</td>
<td>45 060</td>
<td>5 \cdot 10^{-3}</td>
</tr>
<tr>
<td>U3</td>
<td>3.2291</td>
<td>9.8927 \cdot 10^{-1}</td>
<td>0.299</td>
<td>2.4616</td>
<td>102 420</td>
<td>2.5 \cdot 10^{-3}</td>
</tr>
<tr>
<td>O1</td>
<td>3.2800</td>
<td>9.2325 \cdot 10^{-1}</td>
<td>0.294</td>
<td>2.3459</td>
<td>12 048</td>
<td>1 \cdot 10^{-2}</td>
</tr>
<tr>
<td>O2</td>
<td>3.2351</td>
<td>9.8957 \cdot 10^{-1}</td>
<td>0.299</td>
<td>2.4645</td>
<td>49 081</td>
<td>5 \cdot 10^{-3}</td>
</tr>
<tr>
<td>O3</td>
<td>3.2280</td>
<td>9.9939 \cdot 10^{-1}</td>
<td>0.299</td>
<td>2.4739</td>
<td>111 104</td>
<td>2.5 \cdot 10^{-3}</td>
</tr>
</tbody>
</table>

| lower bound | 3.2200 | 9.9000 \cdot 10^{-1} | 0.295 | 2.4600 |
| upper bound  | 3.2400 | 10.100 \cdot 10^{-1} | 0.305 | 2.5000 |

Table 4.2 Flow Around Oscillating Circular Cylinder

Setup of this case is identical to the case of flow past static cylinder. The oscillation of the cylinder is prescribed by the harmonic motion in vertical direction, i.e., $x_2(t) = h_0 + \Delta h \sin(2\pi ft)$. The initial position $h_0 = 0$, the oscillation amplitude $\Delta h = 0.01 \text{ m}$, i.e., 10% of the diameter of the cylinder and the frequency of the oscillation $f = 3 \text{ Hz}$. Figure 3 shows good agreement for the lift coefficient $c_l$ and drag coefficient during the calculation between both approaches.

4.3 Flow Around Oscillating NACA 0012 Profile

This section presents the results of the computation of the flow past oscillating NACA 0012 profile. The following notations is used. The length of the cord is denoted by $c$ and the frequency of the oscillation by $f$. The motion of the airfoil is prescribed by the harmonic motion, i.e., $\alpha(t) = \alpha_0 + \Delta \alpha \sin(2\pi ft)$. The pressure coefficient $c_p = \frac{p_{\text{rel}} - p_{\infty}}{\frac{1}{2} \rho U_\infty^2}$, $p_{\text{rel}} = 0$, is decomposed according to $c_p = c_{p,\text{mean}} + c'_p \sin(2\pi ft) + c''_p \cos(2\pi ft)$, where $c_{p,\text{mean}}$ is the time-mean values of the $c_p$, $c'_p$ is the real part of the $c_p$ and $c''_p$ is the imaginary part of the $c_p$.

The calculation was setup as follows. The initial angle of attack $\alpha_0 = 0^\circ$, the oscillation amplitude $\Delta \alpha = 1^\circ$ and the frequency of the oscillation $f = 30 \text{ Hz}$, position of the elastic axis $x_{E1} = 0.25c$ measured from the leading edge. The free-stream velocity $U_\infty = (50, 0) \text{ m/s}$, the kinematic viscosity of the air $\nu = 1.5 \cdot 10^{-5} \text{ m}^2/\text{s}$ and $c = 0.3 \text{ m}$. One can see that numerical solutions of both non-overset and overset method corresponds well to the each other, see Figure 4.
Figure 3: Lift coefficient $c_l(t)$ (a) and drag coefficient $c_d(t)$ (b) for the case of flow around oscillating cylinder.

Figure 4: Mean value of the pressure coefficient $c_{p,\text{mean}}$ (left) and imaginary part of the pressure coefficient $c_{p}^\prime$ (right).

5 Conclusion

In this contribution the numerical method for the simulation of the interaction of viscous incompressible fluid flow and a vibrating rigid body is applied within open-source code OpenFOAM. Both the standard finite volume approach and the overset method were used. The results show good agreement between results of both method, which allows good applicability of the overset method for the fluid-structure interaction problems, e.g. aeroelastic calculations.

Acknowledgment

This work was supported by the Grant Agency of the Czech Technical University in Prague, grant No. SGS16/206/OHK3/3T/12. Authors acknowledge support from the EU Operational Programme Research, Development and Education, and from the Center of Advanced Aerospace Technology (CZ.02.1.01/0.0/0.0/16_019/0000826), Faculty of Mechanical Engineering, Czech Technical University in Prague.

References


