Advances in Applied Mathematics and Mechanics
Adv. Appl. Math. Mech., Vol. **8**, No. 1, pp. 145-165

DOI: 10.4208/aamm.2014.m590
February 2016

Reduction of Numerical Oscillations in Simulating Moving-Boundary Problems by the Local DFD Method

Yang Zhang and Chunhua Zhou*

Department of Aerodynamics, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China

Received 17 April 2014; Accepted (in revised version) 22 December 2014

Abstract. In this work, the hybrid solution reconstruction formulation proposed by Luo et al. [H. Luo, H. Dai, P. F. de Sousa and B. Yin, On the numerical oscillation of the direct-forcing immersed-boundary method for moving boundaries, Computers & Fluids, 56 (2012), pp. 61–76] for the finite-difference discretization on Cartesian meshes is implemented in the finite-element framework of the local domain-free discretization (DFD) method to reduce the numerical oscillations in the simulation of moving-boundary flows. The reconstruction formulation is applied at fluid nodes in the immediate vicinity of the immersed boundary, which combines weightly the local DFD solution with the specific values obtained via an approximation of quadratic polynomial in the normal direction to the wall. The quadratic approximation is associated with the no-slip boundary condition and the local simplified momentum equation. The weighted factor suitable for unstructured triangular and tetrahedral meshes is constructed, which is related to the local mesh intervals near the immersed boundary and the distances from exterior dependent nodes to the boundary. Therefore, the reconstructed solution can account for the smooth movement of the immersed boundary. Several numerical experiments have been conducted for two- and three-dimensional moving-boundary flows. It is shown that the hybrid reconstruction approach can work well in the finite-element context and effectively reduce the numerical oscillations with little additional computational cost, and the spatial accuracy of the original local DFD method can also be preserved.

AMS subject classifications: 76D05, 76M99

Key words: Numerical oscillation, immersed boundary method, moving boundary, domain-free discretization, boundary condition.

1 Introduction

Recently, the immersed boundary methods (IBMs) have gained a special attention for its simplicity and effectiveness in the simulation of flows with complex moving boundaries.

*Corresponding author.
Email: chzhou@nuaa.edu.cn (C. H. Zhou)

 ©2016 Global Science Press

https://doi.org/10.4208/aamm.2014.m590 Published online by Cambridge University Press

According to the classification proposed by Mittal and Iaccarino [1], IBMs can be categorized into two major groups. One is the so-called continuous forcing approach where the forcing is incorporated into the momentum equations before discretization is performed. The distinguished drawback of this approach is that the boundary condition on the immersed interface cannot be precisely satisfied and the interface is smeared. The other group corresponds to the discrete forcing approach where the boundary condition is directly introduced into the discrete equations by imposing discrete body forces, explicitly or implicitly, on the cells/nodes close to the surface of immersed body and thereby enables a sharp representation of the immersed interface.

Temporal oscillations can be found in the vast majority of the numerical results obtained by the discrete-forcing IBMs [2–5]. Lee et al. [3] indicated that the spatial discontinuity of pressure and the temporal discontinuity of velocity are the two sources of these numerical oscillations. In the work of Kim et al. [4], the oscillations are found to be reduced by introducing the mass source/sink terms. In the work of Seo and Mittal for the ghost-cell IBM [2], the numerical oscillations are reduced by adopting the Cartesian cut-cell approach [6,7] with a virtual merging technique [8]. Later, Lee and You [9] proposed a fully-implicit ghost-cell IBM coupled with a mass source/sink term to remedy this problem. Nevertheless, all the aforementioned approaches are considered from the perspective of improving mass conservation near the immersed boundary. To achieve this goal, a tedious task of geometric handling, such as local reconstruction of mesh cells in the immediate vicinity of the boundary and recalculation of fluxes for these cells is inevitably required. These processes could be time-consuming and difficult to implement, especially for three-dimensional problems with complex geometries. Recently, Luo et al. [10] proposed a hybrid reconstruction formulation which is considered from the perspective of making smooth transitions of numerical descriptions at the direct-forcing points. They stated that when the boundary moves across the nodes on the fixed background mesh, the abrupt change of numerical description between the standard finite-difference formula and the flow reconstruction at these nodes could cause numerical oscillations in the pressure. To reduce the oscillations, a hybrid formulation was proposed, which combines the reconstructed solution and the solution of governing equations at the fluid nodes immediately next to the solid boundary.

Zhou and Shu recently proposed a local domain-free discretization (DFD) method to simulate the moving-boundary flows [11]. In this method, a partial differential equation is discretized at all mesh nodes inside the solution domain, but the discrete form may involve some exterior nodes. The exterior dependent nodes, which are connected to an interior node by a cell edge, serve as the role to enforce the wall boundary condition. The flow variables at an exterior dependent node are obtained via some approximate form of the solution in the vicinity of solid boundary. This method can be classified as a discrete-forcing IBM. Unsurprisingly, temporal oscillations can be observed in the previous DFD results of moving-boundary problems.

Following the work of Luo et al. [10], we formulate a hybrid solution reconstruction in the finite-element framework of the local DFD method to reduce the numerical

https://doi.org/10.4208/aamm.2014.m590 Published online by Cambridge University Press

oscillations in the results of moving-boundary problems. The formulation is applied at fluid nodes in the immediate vicinity of the immersed boundary and combines the local DFD solution with the specific values calculated by the quadratic interpolation in the normal direction to the wall. The quadratic interpolation is inspired by the linear one employed in the hybrid Cartesian immersed boundary (HCIB) method [12]. With this solution reconstruction, a smooth transition of the numerical description at these fluid nodes near the immersed boundary can be achieved, and the temporal oscillations can be reduced. Furthermore, the tedious work of geometric handling in the Cartesian cut-cell [2] and mass source/sink [4,9] approaches can be avoided. In this work, the hybrid reconstruction formulation has also been implemented for moving-boundary problems with complex three-dimensional geometries.

The reminder of this article is arranged as follows. In Section 2, the governing equations and the basic numerical schemes in the local DFD method are described briefly. In Section 3, the hybrid reconstruction approach aiming at the reduction of numerical oscillations in the simulation of moving-boundary flows is presented, and the definition of the weighted average factor is elaborated. In Section 4, we validate the present approach for both two- and three-dimensional moving-boundary flows by numerical experiments. Finally, in Section 5, we summarize this work and present the conclusions.

2 Governing equations and basic numerical schemes

The two-dimensional incompressible Navier-Stokes equations in a Cartesian coordinate system can be written as

$$\mathbf{I}^m \cdot \frac{\partial \mathbf{w}}{\partial t} + \frac{\partial \mathbf{f}_i}{\partial x_i} = \frac{\partial \mathbf{s}_i}{\partial x_i}, \quad i = 1,2, \tag{2.1}$$

where $\mathbf{w}$ is the vector of conserved variables, $\mathbf{f}_i$ and $\mathbf{s}_i$ are the convective and viscous flux vectors, respectively

$$\mathbf{w} = \begin{bmatrix} p \\ u_1 \\ u_2 \end{bmatrix}, \quad \mathbf{f}_i = \begin{bmatrix} u_i \\ u_1 u_i + p\delta_{1i} \\ u_2 u_i + p\delta_{2i} \end{bmatrix}, \quad \mathbf{s}_i = \begin{bmatrix} 0 \\ \tau_{i1} \\ \tau_{i2} \end{bmatrix}, \tag{2.2}$$

and $\mathbf{I}^m = diag(0,1,1)$ is the modified identity matrix annihilating the temporal derivative of pressure from the continuity equation. In (2.2), u_i denotes the velocity components, p the pressure, and τ_{ij} the dimensionless viscous stress tensor.

In the original local DFD method [11], the two-dimensional computational domain is discretized with a triangular mesh. Spatial discretization of the governing equations is achieved by employing the Galerkin finite element approach proposed by Mavriplis and Jameson [13]

$$\mathbf{I}^m \cdot \frac{\partial (\Omega_P \mathbf{w}_P)}{\partial t} = \sum_{e=1}^{n} \frac{\mathbf{F}^A + \mathbf{F}^B}{2} \cdot \Delta \mathbf{L}_{AB} - \sum_{e=1}^{n} \frac{3}{2} \mathbf{S}^e \cdot \Delta \mathbf{L}_{AB}. \tag{2.3}$$

https://doi.org/10.4208/aamm.2014.m590 Published online by Cambridge University Press

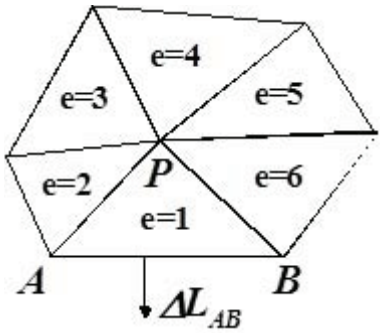


Figure 1: Influence domain of node P.

In (2.3), the summation is over all the triangles that share the node P, and Ω_P represents the area sum of these triangles. The set of these triangles defines the influence domain of P. As illustrated in Fig. 1, $\Delta\mathbf{L}_{AB}$ denotes the edge vector (outward normal) of the face of each triangle e on the boundary of the influence domain. $\mathbf{F}^A$ and $\mathbf{F}^B$ are the tensors of convective flux at the two ends of this edge, and $\mathbf{S}^e$ is the viscous flux tensor in triangle e.

Non-reflecting boundary conditions at the in- and outflow boundaries are imposed via the Steger-Warming flux splitting scheme [14]. The artificial dissipation term that consists of biharmonic operators is introduced to prevent the odd-even decoupling [15]. The semi-discrete equations (2.3) are integrated in time with a dual-time-stepping scheme [15]. The third-order temporal discretization is used in physical time marching, and the five-stage hybrid scheme proposed in [13] is adopted to march in the pseudo time.

3 Implementation of the hybrid solution reconstruction in the local DFD

3.1 Treatment of the immersed boundary in the local DFD

The treatment of the immersed boundary in the local DFD method has been described in detail in [11, 16]. Here, only a brief review is given for completeness. According to the basic numerical method presented in Section 2, the discrete form of governing equations at an interior computed node in the immediate vicinity of the immersed boundary involves at least one node in the solid domain. An example is illustrated in Fig. 2. To close the discretization, the functional values at the exterior dependent nodes are updated at each physical time step by the approximate form of solution near the immersed boundary, which is constructed through extrapolation and the simplified momentum equation along the normal direction to the boundary. In this process, the no-slip boundary condition is enforced [11].

As shown in Fig. 2, the line normal to the boundary and passing through the exterior

https://doi.org/10.4208/aamm.2014.m590 Published online by Cambridge University Press

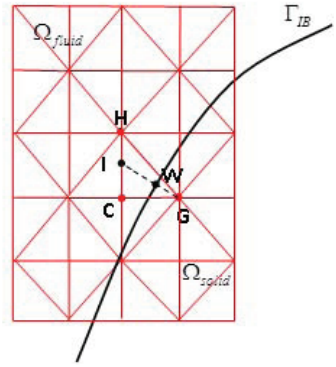


Figure 2: 2D schematic of the extrapolation stencil. H: interior computed node; G: exterior dependent node; W: boundary-intercept point; I: interpolation point; HC: interior cell edge closest to the wall and intersected by the normal line.

dependent node G intersects with the boundary at W. To obtain the flow variables at the exterior dependent node G, an interpolation point I in the fluid domain, which is called the fictitious point in [11,16], should be identified. The intersection between the normal line and the nearest interior cell edge (both ends of the edge locate in the fluid) is defined as the interpolation point I. The values of flow variables at the point I can be obtained by linear interpolation between the two end points of the edge, points H and C in Fig. 2. Therefore, the velocity components at the exterior dependent node G can be obtained by linear extrapolation between I and W

$$u_G = \frac{u_W|IG| - u_I|WG|}{|IW|}. \tag{3.1}$$

In the above equation, u_G, u_W and u_I are the Cartesian components of the velocity at G, W and I, respectively, and the absolute value means the distance between the two points. According to the no-slip boundary condition, u_W is equal to the component of the velocity of the known boundary motion.

By solving the local simplified momentum equation

$$\left(\frac{\partial p}{\partial n}\right)_W = -\left(\frac{dV_n}{dt}\right)_W, \tag{3.2}$$

where V_n is the normal component of the velocity of the boundary motion, the pressure at the exterior dependent node G can be obtained

$$p_G = p_I + |IG|\left(\frac{dV_n}{dt}\right)_W, \tag{3.3}$$

where p_G and p_I represent the pressure at the exterior dependent node G and the interpolation point I, respectively. With the known values of flow variables at all exterior dependent nodes, the discretization is closed in the whole solution domain.

https://doi.org/10.4208/aamm.2014.m590 Published online by Cambridge University Press

3.2 Hybrid reconstruction approach

Obviously, in the local DFD method, the governing equations are solved down to the fluid nodes in the immediate vicinity of the immersed boundary. However, in the HCIB approach proposed by Gilmanov and Sotiropoulos [12], the flow variables at the fluid nodes closest to the boundary (node H in Fig. 3) are obtained by linear interpolation along the normal line, and the governing equations are solved only down to the second off-wall nodes. To obtain the interpolated values of flow variables at node H, an interpolation point I in the fluid domain is constructed. It is the intersection between the normal line HW and the nearest edge BC to the boundary both ends of which are regular interior computed nodes. The values of flow variables at point I can be calculated by the interpolation between two ends of the edge. Then, the flow variables at node H can be obtained by the linear interpolation between I and W.

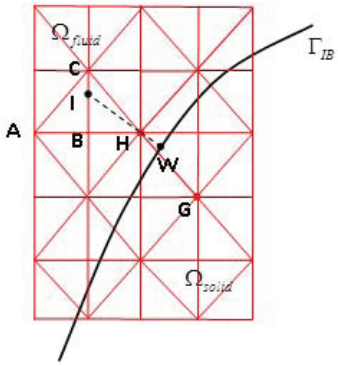


Figure 3: 2D schematic of the interpolation stencil. G: exterior dependent node; H: hybrid node; W: boundary-intercept point; I: interpolation point; B and C: regular interior computed nodes.

To enhance the interpolation accuracy, the specific values of flow variables at node H can be obtained via an approximation of quadratic polynomial in the normal direction in conjunction with the no-slip boundary condition and the simplified local momentum equation [17]. Let n be the distance measured from the boundary-intercept point W. In the vicinity of the immersed boundary, the flow variables are assumed to vary with n in a quadratic manner

$$\phi(n)=c_1n^2+c_2n+c_3, \tag{3.4}$$

where c_1, c_2, c_3 are the coefficients to be determined.

The coefficients for the distribution of velocity components can be determined by solving the following system of linear equations

$$u_W=u(0)=V_W=c_3, \tag{3.5a}$$

$$u_I=u(n_I)=c_1n_I^2+c_2n_I+c_3, \tag{3.5b}$$

$$\left(\frac{\partial u}{\partial n}\right)_I=2c_1n_I+c_2, \tag{3.5c}$$

https://doi.org/10.4208/aamm.2014.m590 Published online by Cambridge University Press

where V_W is the known velocity component of the prescribed boundary motion, n_I is the distance between I and W, and $(\frac{\partial u}{\partial n})_I$ represents the derivative at point I which can be calculated over the triangle ABC as shown in Fig. 3. Therefore, the coefficients for the distribution of velocity components can be expressed as

$$c_1=\frac{n_I\left(\frac{\partial u}{\partial n}\right)_I-u_I+V_W}{n_I^2},\quad c_2=\left(\frac{\partial u}{\partial n}\right)_I-2c_1n_I,\quad c_3=V_W. \tag{3.6}$$

Similarly, the coefficients for the distribution of pressure can be determined by solving the following system of linear equations

$$p_I=c_1n_I^2+c_2n_I+c_3, \tag{3.7a}$$

$$\left(\frac{\partial p}{\partial n}\right)_I=2c_1n_I+c_2, \tag{3.7b}$$

$$\left(\frac{\partial p}{\partial n}\right)_W=c_2. \tag{3.7c}$$

In (3.7b), $(\frac{\partial p}{\partial n})_I$ means the derivative of pressure at I, and it can be calculated in the same way as $(\frac{\partial u}{\partial n})_I$. $(\frac{\partial p}{\partial n})_W$ in (3.7c) can be obtained directly from the simplified local momentum equation (3.2). Then, the coefficients for the pressure distribution can be expressed as

$$c_1=\frac{1}{2n_I}\left(\left(\frac{\partial p}{\partial n}\right)_I+\left(\frac{dV_n}{dt}\right)_W\right),\quad c_2=-\left(\frac{dV_n}{dt}\right)_W,\quad c_3=p_I-\frac{n_I}{2}\left(\left(\frac{\partial p}{\partial n}\right)_I-\left(\frac{dV_n}{dt}\right)_W\right). \tag{3.8}$$

Therefore, the interpolated values of flow variables at the node H can be obtained by substituting $n=n_H$ into (3.4).

In the local DFD method, when the node H is close to the boundary, it may be an exterior dependent node at the next time-step. The sudden change of the role of node H due to the motion of the solid boundary will introduce numerical oscillations. Following the work of Luo et al. [10], a hybrid solution reconstruction for the local DFD method is formulated to make smooth transitions for the abrupt change of the role of the nodes like H. When the node H locates relatively far from the immersed boundary, it tends to be a regular interior computed node (see Fig. 2). In this case, the Navier-Stokes solution at H obtained by the local DFD method is more accurate. When the node H is close to the boundary, the values of flow variables at H obtained by the quadratic interpolation described by Eqs. (3.2)-(3.8) will be more accurate (see Fig. 3). Therefore, the formulation of the hybrid solution reconstruction at the nodes like H should combine weightly the Navier-Stokes solution with the specific values calculated by the quadratic interpolation. Following Luo et al. [10], we call any fluid node like H in the immediate vicinity of the immersed boundary a hybrid node. If the flow variables at the hybrid nodes obtained by solving the governing equations are denoted by ϕ_{comput} and the flow variables obtained via interpolation are denoted by ϕ_{interp}, the reconstructed solution can be expressed as

$$\phi_{\text{hybrid}}=(1-\alpha)\phi_{\text{comput}}+\alpha\phi_{\text{interp}}, \tag{3.9}$$

https://doi.org/10.4208/aamm.2014.m590 Published online by Cambridge University Press

where $0 \leq \alpha \leq 1$ is the weighted average factor and its construction will be discussed later in the next subsection.

Note that the identification procedure of boundary-intercept points and interpolation points associated with all hybrid nodes is performed only once for each physical time-step, and not involved in the pseudo-time marching. The identification procedure follows the similar way as that in the local DFD method. Furthermore, the number of hybrid nodes is relatively small. Therefore, compared to the original local DFD, the increment of computational cost due to the hybrid reconstruction is almost negligible.

3.3 Construction of the weighted average factor

As discussed in Section 3.2 and indicated by Luo et al. [10], the weighted average factor α in (3.9) should properly take into account the influence of the abrupt change of the role of the hybrid node due to the motion of the immersed boundary. Namely, the factor α should approach 1 when the hybrid node is close to the boundary, and it should approach 0 when the hybrid node is far away from the boundary (its exterior dependent nodes are close to the boundary).

Since the finite-element discretization is employed in the local DFD method, the features of the unstructured triangular meshes must be considered in the construction of the weighted average factor. The schematic for constructing the factor in the two-dimensional case is illustrated in Fig. 4. In this figure, Γ represents the immersed boundary at the current time step. The hybrid node H is assumed to have three exterior dependent nodes, i.e., G_1, G_2 and G_3. Δ_1, Δ_2 and Δ_3 are distances from G_1, G_2 and G_3 to the immersed boundary, respectively. The weighted average factor must be related to the local mesh intervals near the boundary and the distances from the exterior dependent nodes of H to the boundary. Then, the weighted average factor suitable for the unstructured triangular meshes is constructed as below

$$d_1 = \frac{\Delta_1}{|G_1H|}, \qquad d_2 = \frac{\Delta_2}{|G_2H|}, \qquad d_3 = \frac{\Delta_3}{|G_3H|}, \tag{3.10a}$$

$$\beta_1 = \frac{d_1}{d_1+d_2+d_3}, \qquad \beta_2 = \frac{d_2}{d_1+d_2+d_3}, \qquad \beta_3 = \frac{d_3}{d_1+d_2+d_3}, \tag{3.10b}$$

$$\alpha = \beta_1 d_1 + \beta_2 d_2 + \beta_3 d_3 = \frac{{d_1}^2+{d_2}^2+d_3^2}{d_1+d_2+d_3}. \tag{3.10c}$$

The general formulation of the weighted average factor α can be expressed as

$$\alpha = \sum_{n=1}^{N} {d_n}^2 \Big/ \sum_{n=1}^{N} d_n, \tag{3.11}$$

where N represents the total number of the exterior dependent nodes for a given hybrid node. Note that the value of α obtained by Eq. (3.11) may exceed 1. In this situation, α is simply set to 1.

https://doi.org/10.4208/aamm.2014.m590 Published online by Cambridge University Press

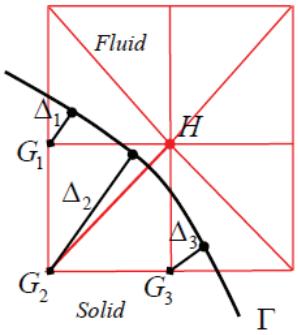


Figure 4: 2D schematic for construction of the weighted average factor. Γ: the immersed boundary; H: hybrid node; G_1, G_2, G_3: exterior dependent nodes.

The hybrid solution reconstruction approach is originally proposed by Luo et al. for the finite-difference discretization on the Cartesian meshes [10]. In this work, we implement it in the finite-element framework of the local DFD method. In the implementation, the features of the unstructured triangular meshes have been considered. The specific values of flow variables at the hybrid nodes are calculated via an approximation of quadratic polynomial in the direction normal to the wall. The weighted average factor is properly related to the distances from the exterior dependent nodes to the wall and the local intervals of the triangular mesh in the wall proximity that are represented by the distances from the hybrid node to its exterior dependent nodes.

3.4 Extension to three dimensions

For three-dimensional cases, the solution domain is discretized with a tetrahedral mesh. The body surface is discretized with an unstructured triangular mesh, which is independent of the background mesh. The extension of the local DFD method to three dimensions has been elaborated in the previous work [16]. As done in two-dimensional cases, specific values of flow variables at the hybrid nodes are also calculated via the quadratic interpolation in the direction normal to the body surface. The interpolation stencil for three-dimensional cases is illustrated in Fig. 5, which is the same as employed in the HCIB approach for inviscid compressible flows [18]. In this figure, W is the boundary-intercept point corresponding to the hybrid node H. The interpolation point I is the intersection of the normal line and the wall-nearest face of a tetrahedral cell whose three vertices are the interior computed points (triangle ABC in the figure). Then, ϕ_I, any of the flow variables at the interpolation point I, can be obtained via the linear interpolation over the triangular face ABC

$$\phi_I = \sum_{k=1}^{3} \Phi_k(x_I, y_I, z_I)\phi_k, \tag{3.12}$$

https://doi.org/10.4208/aamm.2014.m590 Published online by Cambridge University Press

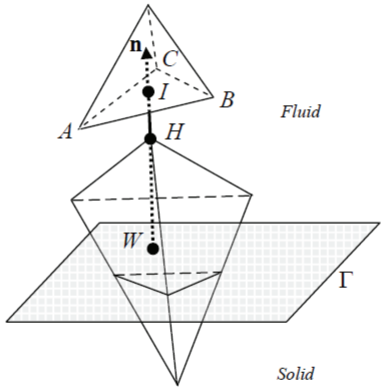


Figure 5: Three-dimensional schematic of the interpolation stencil. Γ: the immersed boundary; H: hybrid node; W: boundary-intercept point; I: interpolation point.

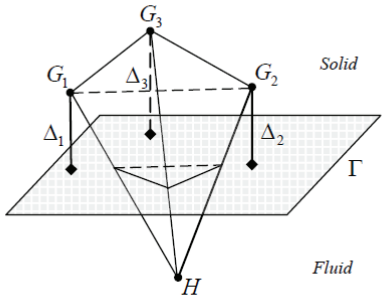


Figure 6: Three-dimensional schematic for the construction of the weighted average factor. Γ: immersed boundary; H: hybrid node; G_1, G_2, G_3: exterior dependent nodes.

where the summation over k refers to the three vertices of ABC and $\Phi_k(x_I,y_I,z_I)$ is the linear interpolation function for each vertex at the point I. Therefore, the interpolated flow variables at the hybrid node H can be obtained in the same way as described in Section 3.2 for two-dimensional cases.

The hybrid solution reconstruction and the definition of the weighted average factor can also be easily extended to three dimensions. For three-dimensional cases, the reconstructed solution at the hybrid nodes is also expressed by (3.9). The construction of the weighted average factor for three-dimensional cases is illustrated in Fig. 6. For the sake of clearness, only one tetrahedron in the influence domain of the hybrid node H is shown in this figure. The hybrid node H is assumed to have three exterior dependent nodes, i.e., G_1, G_2 and G_3. Δ_1, Δ_2 and Δ_3 are the distances from G_1, G_2 and G_3 to the immersed boundary, respectively. For each exterior dependent node, we can define a parameter $d_i=\frac{\Delta_i}{|G_iH|}$. Then, the general formulation of the weighted average factor (3.11) is still valid for three-dimensional cases.

https://doi.org/10.4208/aamm.2014.m590 Published online by Cambridge University Press

4 Numerical experiments

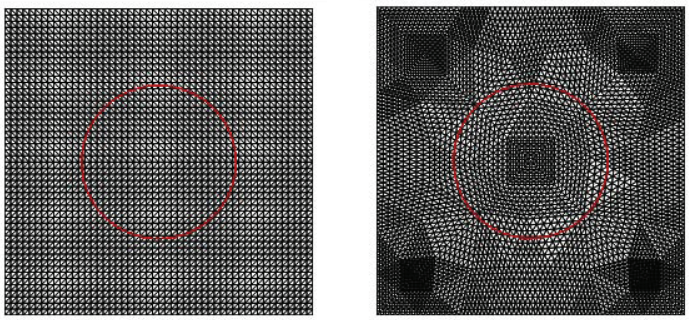
In this section, the convergence rate of the local DFD method coupled with the hybrid solution reconstruction formulation is firstly verified. Then, numerical experiments for two- and three-dimensional incompressible flows with moving-boundaries are performed to demonstrate the effectiveness of the present hybrid approach in reducing numerical oscillations. To enhance resolution locally, the meshes used in all test cases except those in Section 4.1 have been refined in the vicinity of solid body. In the subsequent discussions, the original method refers to the original local DFD method without the introduction of the hybrid solution reconstruction.

4.1 Verification of convergence rate

We consider the Navier-Stokes equations with a forcing term to verify the convergence rate of the improved local DFD method. The solution domain is bounded by $\Omega\backslash\omega$, where $\Omega=[-0.5,0.5]\times[-0.5,0.5]$ and ω is a circle with the radius of 0.25 and the center coinciding with that of Ω. The forcing term and the Dirichlet boundary condition can be determined via the analytical solution provided in [19] as below

$$\begin{cases} u_1=2\pi\sin^2(\pi x_1)\sin(\pi x_2)\cos(\pi x_2), \\ u_2=-2\pi\sin^2(\pi x_2)\sin(\pi x_1)\cos(\pi x_1), \\ p=20{x_1}^2x_2. \end{cases} \tag{4.1}$$

The L_1 and L_2 norms are chosen to measure the global error of the solution, and the L_∞ norm is chosen to measure the local error. The Navier-Stokes equations with the analytical solution (4.1) are solved by the present improved local DFD method on a series of structured-like uniform triangular meshes and a series of isotropic unstructured triangular meshes, respectively. The initial meshes are shown in Figs. 7(a) and (b). The averaged interval of the unstructured mesh is close to that of the structured-like mesh.



(a) Structured-like mesh (b) Unstructured mesh

Figure 7: Initial meshes for the verification of convergence rate.

https://doi.org/10.4208/aamm.2014.m590 Published online by Cambridge University Press

Table 1: Global and local errors of u_1 on uniform structured meshes.

Mesh interval	Global error				Local error	
	L_1 error	order	L_2 error	order	L_∞ error	order
2.0×10^{-2}	1.91×10^{-3}	–	2.84×10^{-3}	–	1.43×10^{-2}	–
1.0×10^{-2}	5.03×10^{-4}	1.93	7.61×10^{-4}	1.90	3.92×10^{-3}	1.87
5.0×10^{-3}	1.31×10^{-4}	1.94	2.04×10^{-4}	1.90	1.03×10^{-3}	1.93

Table 2: Global and local errors of p on uniform structured meshes.

Mesh interval	Global error				Local error	
	L_1 error	order	L_2 error	order	L_∞ error	order
2.0×10^{-2}	8.94×10^{-3}	–	1.27×10^{-2}	–	6.00×10^{-2}	–
1.0×10^{-2}	2.08×10^{-3}	2.10	3.07×10^{-3}	2.05	2.51×10^{-2}	1.26
5.0×10^{-3}	5.22×10^{-4}	1.99	8.18×10^{-4}	1.91	1.23×10^{-2}	1.03

Table 3: Global and local errors of u_1 on unstructured meshes.

Mesh interval	Global error				Local error	
	L_1 error	order	L_2 error	order	L_∞ error	order
1	1.40×10^{-3}	–	1.97×10^{-3}	–	6.35×10^{-3}	–
2	3.70×10^{-4}	1.92	5.28×10^{-4}	1.90	1.74×10^{-3}	1.87
3	9.50×10^{-5}	1.96	1.37×10^{-4}	1.95	4.66×10^{-4}	1.90

Table 4: Global and local errors of p on unstructured meshes.

Mesh interval	Global error				Local error	
	L_1 error	order	L_2 error	order	L_∞ error	order
1	6.97×10^{-3}	–	9.74×10^{-3}	–	5.20×10^{-2}	–
2	1.80×10^{-3}	1.95	2.58×10^{-3}	1.92	2.36×10^{-2}	1.14
3	4.68×10^{-4}	1.94	6.92×10^{-4}	1.90	1.12×10^{-2}	1.08

The error-norms and the orders of convergence rate for velocity and pressure are presented in Tables 1-4. Due to the identical status of u_1 and u_2, only the error-norms of u_1 are listed in the tables. From these tables, we can see that for both structured-like and unstructured triangular meshes the velocity components are globally and locally second-order accurate while the pressure is globally second-order accurate and locally first-order (or a little higher) accurate.

Therefore, the global second-order accuracy of the local DFD method has been well preserved.

The convergence rate of the improved local DFD method for three-dimensional cases is assessed on a series of structured-like uniform tetrahedral meshes. The solution domain is bounded by $\Omega\backslash\omega$, where $\Omega=[-0.5,0.5]\times[-0.5,0.5]\times[-0.5,0.5]$ and ω is a sphere with the radius of 0.25 and the center coinciding with that of Ω. The forcing term and the

https://doi.org/10.4208/aamm.2014.m590 Published online by Cambridge University Press

Table 5: Global and local errors of u_1 on uniform structured meshes (3D).

Mesh interval	Global error				Local error	
	L_1 error	order	L_2 error	order	L_∞ error	order
4.0×10^{-2}	4.60×10^{-4}	–	6.23×10^{-4}	–	1.06×10^{-2}	–
2.0×10^{-2}	1.05×10^{-4}	2.13	1.42×10^{-4}	2.13	2.81×10^{-3}	1.92
1.0×10^{-2}	2.78×10^{-5}	1.92	3.75×10^{-5}	1.92	7.70×10^{-4}	1.87

Dirichlet boundary condition can be determined via the following analytical solution

$$\begin{cases} u_1 = \sin(\pi x_2)\sin(\pi x_3), \\ u_2 = \sin(\pi x_1)\sin(\pi x_3), \\ u_3 = \sin(\pi x_1)\sin(\pi x_2), \\ p = x_1 x_2 x_3. \end{cases} \tag{4.2}$$

The error-norms and the orders of convergence rate for the velocity component u_1 are listed in Table 5. From this table, it can be verified that the improved local DFD method is still second-order in velocity for the three-dimensional problems.

4.2 Flows over a cross-flow oscillating circular cylinder

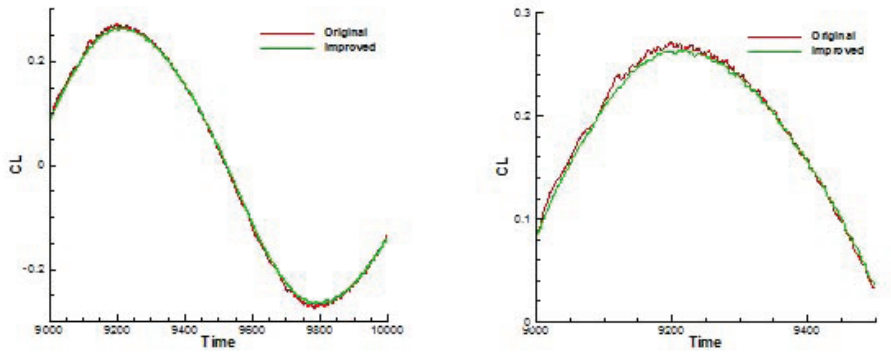
In this sub-section, a circular cylinder harmonically oscillating in the transverse direction in a free-stream [20] is considered. The Reynolds number is defined as $Re = U_\infty D/\nu$, where U_∞ is the free stream velocity and D is the diameter of the cylinder. The harmonic motion of the cylinder is described as

$$y_c(t) = A_m \sin(2\pi f_e t), \tag{4.3}$$

where y_c is the location of the cylinder center, and A_m and f_e are the amplitude and frequency of the oscillation, respectively. We have performed calculations at $Re = 185$, $A_m/D = 0.2$. f_e/f_0 varies from 0.8 to 1.2, where f_0 is the natural shedding frequency for the stationary cylinder. The size of the computational domain is $40D \times 40D$, and the number of computational nodes is around 31000. The mesh interval in the vicinity of the cylinder is about $0.0125D$. The non-dimensional time can be expressed as $t = \tau/T$, where τ is the dimensional time and T is the reference time. In this simulation, T is taken as D/U_∞. In the subsequent simulations, the non-dimensional time is defined in a similar manner. Due to the body motion, the size of physical time-step must be limited as discussed in [11]. In this test case, the size of non-dimensional physical time-step Δt is taken to be 5×10^{-3}. Comparison of time histories of the lift coefficient obtained by the original and improved DFD methods at $f_e/f_0 = 0.9$ is shown in Fig. 8. From this figure, we can see that the numerical oscillations are effectively reduced by the hybrid solution reconstruction approach.

Furthermore, to compare the numerical oscillations in a more accurate way, the magnitude of numerical oscillations is quantified by $\langle C_L \rangle_{rms}$, the root-mean-square of the

https://doi.org/10.4208/aamm.2014.m590 Published online by Cambridge University Press



(a) (b)

Figure 8: Time histories of the lift coefficient for a cross-flow oscillating cylinder at $f_e/f_0=0.9$. (b): zoomed view of (a).

difference between the fitted lift coefficient and the actual one. Comparison of $\langle C_L\rangle_{rms}$ between the original and improved DFD results for different frequencies is presented in Table 6. From this table, we can conclude that $\langle C_L\rangle_{rms}$ is reduced by roughly 40%-50% for each frequency.

Table 6: Comparison of $\langle C_L\rangle_{rms}$ between the original and improved DFD results.

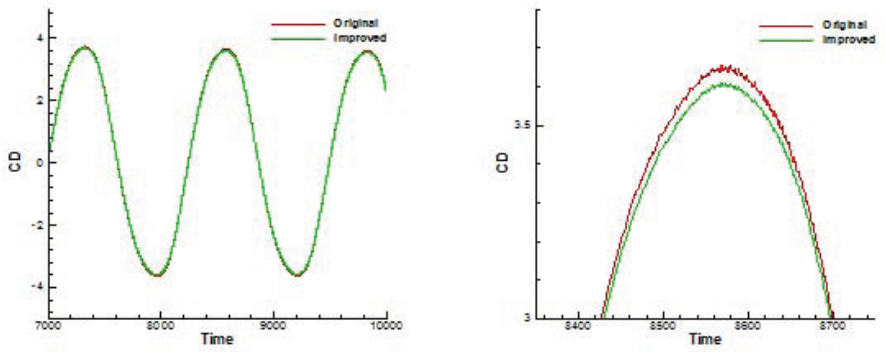
f_e/f_0	$\langle C_L\rangle_{rms}$	
	Original local DFD	Improved local DFD
0.8	1.60×10^{-3}	9.70×10^{-4}
0.9	1.74×10^{-3}	9.70×10^{-4}
1.0	1.94×10^{-3}	1.02×10^{-3}
1.1	1.81×10^{-3}	8.64×10^{-4}
1.12	1.80×10^{-3}	1.13×10^{-3}
1.2	1.87×10^{-3}	1.18×10^{-3}

4.3 Flow over an in-line oscillating circular cylinder

The flow induced by the in-line oscillation of a circular cylinder in a fluid at rest [21] is also simulated to validate the capability of the improved local DFD method for reducing numerical oscillations. The two key parameters in this case are the Reynolds number $Re=U_{\max}D/\nu$ and the Keulegan-Carpenter number $KC=U_{\max}/fD$, where $U_{\max}$ is the maximum velocity of the cylinder and f is the frequency of oscillation. The cylinder oscillates harmonically in the horizontal direction and its motion is described as

$$x_c(t)=A_m\sin(2\pi ft), \tag{4.4}$$

https://doi.org/10.4208/aamm.2014.m590 Published online by Cambridge University Press



(a) (b)

Figure 9: Variations of the drag coefficient versus time for an in-line oscillating cylinder. (b): zoomed view of (a).

where x_c is the location of the cylinder center, A_m is the amplitude of the oscillation. The parameters in the present simulation are set to be $Re=100$ and $KC=5$. The size of the computational domain is $50D\times 30D$, and the number of computational nodes is around 38000. The mesh interval in the vicinity of the cylinder is about $0.0132D$. The non-dimensional time-step size is taken to be 4×10^{-3}. Comparison of variations of the drag coefficient versus time-step is illustrated in Fig. 9. From this figure, we can see again that the numerical oscillations are effectively reduced by the hybrid reconstruction approach. In addition, $\langle C_D\rangle_{rms}$ obtained by the original DFD is 4.36×10^{-3}, while $\langle C_D\rangle_{rms}$ obtained by the improved DFD is 2.65×10^{-3}.

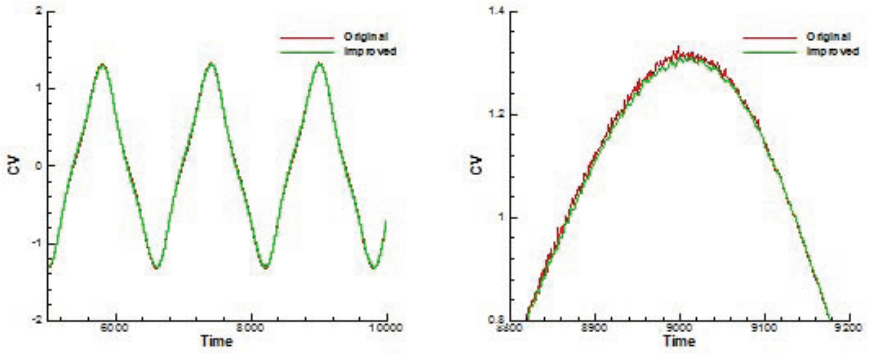
4.4 Flow over a heaving-pitching airfoil

Consider an airfoil with chord length b, moving at constant forward speed U_∞ and performing a harmonic heave motion $h(t)$ and a harmonic pitch motion $\theta(t)$. The pitch motion leads the heave motion by a phase difference ψ. Hence the heave and pitch motions can be described as

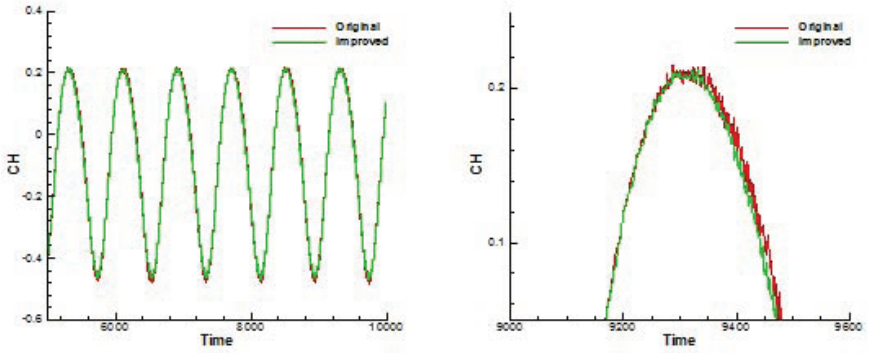
$$h(t)=h_0\sin(\omega t),\quad \theta(t)=\theta_0\sin(\omega t+\psi), \tag{4.5}$$

where h_0 denotes the amplitude of heaving, θ_0 the amplitude of pitching, ω the frequency. Following the work of Zhou et al. for the original DFD [11], the phase angle ψ varies from 50° to 115°, and the other parameters in the simulation are fixed: $h_0/b=0.50$, $\theta_0=30^\circ$, the distance of the pivot point from the leading edge $d/b=0.25$, the Strouhal number $St=h_0\omega/\pi U_\infty=0.25$, the Reynolds number $Re=1100$. The number of computational nodes is around 86400, and the mesh interval in the vicinity of the airfoil is about $0.0063b$. The size of non-dimensional time-step is taken to be 2.5×10^{-3}. Temporal variations of the vertical (C_V) and horizontal (C_H) force coefficients for $\psi=90^\circ$ obtained by the original

https://doi.org/10.4208/aamm.2014.m590 Published online by Cambridge University Press



(a) Vertical force coefficient



(b) Horizontal force coefficient

Figure 10: Variations of the force coefficient as a function of time for a heaving-pitching airfoil for $\psi = 90^\circ$. Right: zoomed view of the left figure.

and improved DFD methods are compared in Figs. 10(a) and (b), respectively. From these figures, we can observe that the numerical oscillations are greatly reduced by the hybrid reconstruction approach. To further verify our approach, comparison of $\langle C_V \rangle_{rms}$ and $\langle C_H \rangle_{rms}$ between the original and improved DFD results for different phase angles is presented in Table 7. From this table, we can see that $\langle C_V \rangle_{rms}$ and $\langle C_H \rangle_{rms}$ are reduced by approximately 29%-44% and 11%-36%, respectively.

In this test case, compared to the original DFD, the computational time of the improved one has been increased by about 1.6%. Therefore, the additional computational cost due to the hybrid reconstruction is almost negligible.

4.5 Flows of a fish-like swimming

In this sub-section, the capability of the hybrid reconstruction approach in reducing numerical oscillations for three-dimensional flows with complex moving-boundaries is

https://doi.org/10.4208/aamm.2014.m590 Published online by Cambridge University Press

Table 7: Comparison of $\langle C_V\rangle_{rms}$ and $\langle C_H\rangle_{rms}$ between the original and improved DFD results.

ψ	$\langle C_V\rangle_{rms}$		$\langle C_H\rangle_{rms}$	
	Original	Improved	Original	Improved
50	7.19×10^{-3}	5.12×10^{-3}	7.82×10^{-3}	6.05×10^{-3}
70	6.04×10^{-3}	3.59×10^{-3}	6.36×10^{-3}	4.83×10^{-3}
90	5.28×10^{-3}	2.95×10^{-3}	5.34×10^{-3}	3.41×10^{-3}
100	5.82×10^{-3}	3.80×10^{-3}	5.32×10^{-3}	4.09×10^{-3}
105	6.30×10^{-3}	4.22×10^{-3}	5.84×10^{-3}	4.43×10^{-3}
110	7.46×10^{-3}	4.62×10^{-3}	6.17×10^{-3}	5.17×10^{-3}
115	8.54×10^{-3}	5.84×10^{-3}	6.37×10^{-3}	5.68×10^{-3}

demonstrated. The three-dimensional domain is discretized by a tetrahedral mesh, and the immersed boundary (fluid/solid interface) is practically represented by a triangular surface mesh. This surface mesh is independent of the tetrahedral mesh.

We simulate the flow induced by a fish-like, flexible object of length L, moving in a straight line with constant axial velocity U and undulating in the lateral direction with a characteristic frequency f. The simulation has been performed by using the original DFD method [16]. The geometry of the three-dimensional body is provided by Sotiropoulos and Borazjani. The three-dimensional view of the body is illustrated in Fig. 11.

The swimming motion is prescribed by the following equation [12,16]

$$h(x_1,t)=a(x_1)\sin2\pi(x_1/\lambda-ft). \tag{4.6}$$

In the above equation, h is the lateral displacement of the fish backbone, x_1 is the axial (flow) direction measured from the tip of the head, $a(x_1)$ is the wave amplitude that is assumed to vary nonlinearly along the body, the non-dimensional wave length λ is taken to be 0.95, and f is the tail beat frequency which relates to the Strouhal number St through the equation $St=2fh_{\max}/U$. The expression of $a(x_1)$ can be found in [12] and the maximum lateral displacement of the fish backbone $h_{\max}=0.1$.

The computational domain is a $8L\times3L\times2L$ cuboid, discretized with a mesh containing 772172 nodes and 4550644 tetrahedral cells. The mesh interval near the body surface is about $0.008L$. The physical time-step $\Delta t=T/500$, where T is the beat period. In our simulation, the Reynolds number Re is fixed at 4000 and the Strouhal number St varies from 0.1 to 0.7.

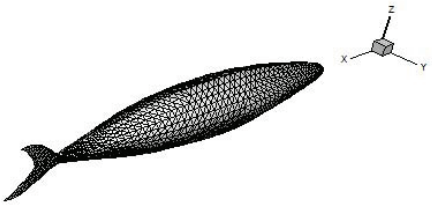


Figure 11: Three-dimensional view of the fish-like body (provided by Sotiropoulos and Borazjani).

https://doi.org/10.4208/aamm.2014.m590 Published online by Cambridge University Press

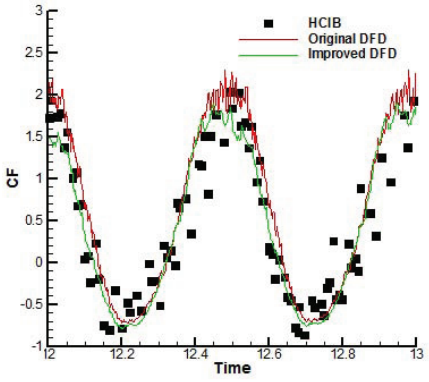


Figure 12: Time histories of the force coefficient for a fish-like swimming at $St=0.7$.

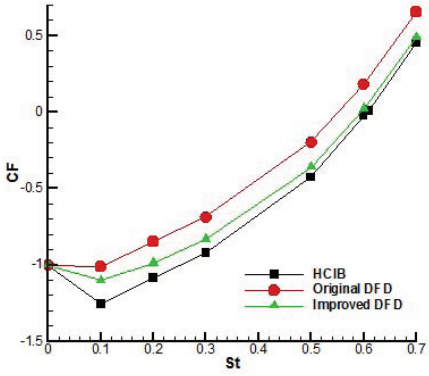


Figure 13: Variation of the mean force coefficient with Strouhal numbers ($Re=4000$).

For $St=0.7$, the time histories of the thrust coefficient obtained by HCIB, the original and improved local DFD methods are illustrated together in Fig. 12. The values of the force coefficient have been scaled with that calculated for the rigid body ($St=0$). From this figure, we can observe that the improved DFD results are much smoother than the HCIB and the original DFD results, and the numerical oscillations are effectively reduced by the hybrid reconstruction approach. Note that the difference between the original and improved DFD results is a bit large at the peak of the force coefficient. This may be attributed to the fact that the used mesh is relatively coarse. The larger the mesh intervals near the boundary, the larger the difference between the solution obtained by hybrid reconstruction ϕ_{hybrid} and the original DFD solution ϕ_{comput}. However, the difference can be surely diminished if a finer mesh is used. Variation of the mean force coefficient with Strouhal numbers is shown in Fig. 13. The improved DFD result lies between the original DFD and HCIB results. Comparison of $\langle C_F\rangle_{rms}$ between the original and improved DFD results for different Strouhal numbers is illustrated in Table 8. Obviously, $\langle C_F\rangle_{rms}$ increases with St for both the original and improved DFD method. We can see that $\langle C_F\rangle_{rms}$ is

https://doi.org/10.4208/aamm.2014.m590 Published online by Cambridge University Press

Table 8: Comparison of $\langle C_F \rangle_{rms}$ between the original and improved DFD results.

St	$\langle C_F \rangle_{rms}$	
	Original local DFD	Improved local DFD
0.1	5.43×10^{-3}	3.77×10^{-3}
0.2	6.60×10^{-3}	5.59×10^{-3}
0.3	1.44×10^{-2}	1.27×10^{-2}
0.5	3.76×10^{-2}	2.66×10^{-2}
0.6	5.58×10^{-2}	3.74×10^{-2}
0.7	7.87×10^{-2}	5.29×10^{-2}

reduced by roughly 30% for $St=0.1$, 0.5, 0.6, 0.7, 15% for $St=0.2$, and 12% for $St=0.3$.

5 Summary and conclusions

In this work, we present a hybrid solution reconstruction approach for reducing numerical oscillations in the simulation of moving-boundary flows by using the local DFD method. This approach originates from the work of Luo et al. [10] for the finite-difference discretization on Cartesian meshes, and we extend it to the finite-element discretization employed in the local DFD method. Concretely speaking, the hybrid formulation combines weightly the local DFD solution at the fluid nodes in the immediate vicinity of the immersed boundary with the specific values obtained by a quadratic interpolation along the direction normal to the wall. Considering the finite-element framework, we construct a weighted factor suitable for unstructured triangular and tetrahedral meshes, which is related to the local mesh intervals near the wall and the distances from the exterior dependent nodes to the wall. By reconstructing the solution at these specific fluid nodes, the boundary movement can be treated in a smooth way. Compared to the cut-cell or mass source/sink approaches, the hybrid reconstruction approach can be implemented easily for the three-dimensional moving-boundary problems with complex geometries.

Several numerical experiments for two- and three-dimensional moving-boundary flows have been conducted. It is shown that the present hybrid reconstruction approach can effectively reduce the numerical oscillations in simulating moving-boundary problems with little additional computational cost and the desirable accuracy of the original local DFD method can be retained.

Acknowledgments

This work has been supported by Natural Science Foundation of China under Grant No. 11472134. We are grateful to F. Sotiropoulos and I. Borazjani for providing us with the geometry of the fish-like body.

https://doi.org/10.4208/aamm.2014.m590 Published online by Cambridge University Press

References

[1] R. MITTAL AND G. IACCARINO, *Immersed boundary methods*, Ann. Rev. Fluid Mech., 37 (2005), pp. 239–261.

[2] J. H. SEO AND R. MITTAL, *A sharp-interface immersed boundary method with improved mass conservation and reduced spurious pressure oscillations,* J. Comput. Phys., 230 (2011), pp. 7347–7363.

[3] J. LEE, J. KIM, H. CHOI AND K. S. YAN, *Sources of spurious force oscillations from an immersed boundary method for moving-body problems*, J. Comput. Phys., 230 (2011), pp. 2677–2695.

[4] J. KIM, D. KIM AND H. CHOI, *An immersed-boundary finite volume method for simulations of flow in complex geometries,* J. Comput. Phys., 171 (2001), pp. 132–150.

[5] R. MITTAL, H. DONG, M. BOZKURTTAS, F. M. NAJJAR, A. VARGAS AND A. V. LOEBBECKE, *A versatile sharp interface immersed boundary method for incompressible flows with complex boundaries,* J. Comput. Phys., 227 (2008), pp. 4825–4852.

[6] T. YE, R. MITTAL, H. S. UDAYKUMAR AND W. SHYY, *An accurate Cartesian method for viscous incompressible flows with immersed complex boundaries,* J. Comput. Phys., 156 (1999), pp. 209–240.

[7] H. S. UDAYKUMAR, R. MITTAL, P. RAMPUNGGOON AND A. KHANNA, *A sharp interface Cartesian grid method for simulating flows with complex moving boundaries*, J. Comput. Phys., 174 (2001), pp. 345–380.

[8] M. MEYER, A. DEVESA, S. HICKEL, X. Y. HU AND N. A. ADAMS, *A conservative immersed interface method for large-eddy simulation of incompressible flows,* J. Comput. Phys., 229 (2010), pp. 6300–6317.

[9] J. LEE AND D. YOU, *An implicit ghost-cell immersed boundary method for simulations of moving-body problems with control of spurious force oscillations,* J. Comput. Phys., 233 (2013), pp. 295–314.

[10] H. LUO, H. DAI, P. F. DE SOUSA AND B. YIN, *On the numerical oscillation of the direct-forcing immersed-boundary method for moving boundaries,* Comput. Fluids, 56 (2012), pp. 61–76.

[11] C. H. ZHOU AND C. SHU, *A local domain-free discretization method for simulation of incompressible flows over moving bodies,* Int. J. Numer. Meth. Fluids, 66 (2011), pp. 162–182.

[12] A. GILMANOV AND F. SOTIROPOULOS, *A hybrid Cartesian/immersed boundary method for simulating flows with 3D, geometrically complex, moving bodies,* J. Comput. Phys., 207 (2005), pp. 457–492.

[13] D. J. MAVRIPLIS AND A. JAMESON, *Multigrid solution of the Navier-Stokes equations on triangular meshes,* AIAA J., 28 (1990), pp. 1415–1425.

[14] J. STEGER AND R. F. WARMING, *Flux vector splitting for the inviscid gas dynamic with applications to finite-difference methods,* J. Comput. Phys., 40 (1983), pp. 263–293.

[15] A. BELOV, L. MARTINELLI AND A. JAMESON, *A new implicit algorithm with multigrid for unsteady incompressible flow calculations,* 33rd Aerospace Sciences Meeting and Exhibit, Reno NV, (1995), 9–12.

[16] C. H. ZHOU AND C. SHU, *Extension of local domain-free discretization method to simulate 3d flows with complex moving boundaries,* Comput. Fluids, 64 (2012), pp. 98–107.

[17] Y. ZHANG, C. H. ZHOU AND J. Q. AI, *Finite volume immersed boundary method and its extension to simulate 3-D flows with complex moving boundaries (in Chinese),* Journal of Nanjing University of Aeronautics & Astronautics, 45 (2013), pp. 322–329.

[18] Y. ZHANG AND C. H. ZHOU, *An immersed boundary method for simulation of inviscid compressible flows,* Int. J. Num. Meth. Fluids, 74 (2014), pp. 775–793.

https://doi.org/10.4208/aamm.2014.m590 Published online by Cambridge University Press

[19] JAVIER DE FRUTOS AND G. A. BOSCO, *A posteriori error estimations for mixed finite-element approximations to the Navier-Stokes equations*, J. Comput. Appl. Math., 236 (2011), pp. 1103–1122.
[20] E. GUILMINEAU AND P. QUEUTEY, *A numerical simulation of vortex shedding from an oscillating cylinder*, J. Fluids Structures, 16 (2002), pp. 773–794.
[21] H. DÜTSCH, F. DURST, S. BECKER AND H. LIENHART, *Low-Reynolds-number flow around an oscillating circular cylinder at low Keulegan-Carpenter numbers*, J. Fluid Mech., 360 (1998), pp. 249–271.

https://doi.org/10.4208/aamm.2014.m590 Published online by Cambridge University Press