Journal of Computational Physics 228 (2009) 7821–7836

Contents lists available at ScienceDirect

Journal of Computational Physics

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

A smoothing technique for discrete delta functions with application to immersed boundary method in moving boundary simulations

Xiaolei Yang [a], Xing Zhang [a], Zhilin Li [b], Guo-Wei He [a,*]

[a] LNM, Institute of Mechanics, Chinese Academy of Sciences, Beijing 100190, China
[b] Department of Mathematics, North Carolina State University, Raleigh, NC 27695-8205, USA

ARTICLE INFO

Article history:
Received 18 December 2008
Received in revised form 21 June 2009
Accepted 25 July 2009
Available online 6 August 2009

Keywords:
Immersed boundary method
Moving boundary
Non-physical force oscillations
Smoothed discrete delta function

ABSTRACT

The effects of complex boundary conditions on flows are represented by a volume force in the immersed boundary methods. The problem with this representation is that the volume force exhibits non-physical oscillations in moving boundary simulations. A smoothing technique for discrete delta functions has been developed in this paper to suppress the non-physical oscillations in the volume forces. We have found that the non-physical oscillations are mainly due to the fact that the derivatives of the regular discrete delta functions do not satisfy certain moment conditions. It has been shown that the smoothed discrete delta functions constructed in this paper have one-order higher derivative than the regular ones. Moreover, not only the smoothed discrete delta functions satisfy the first two discrete moment conditions, but also their derivatives satisfy one-order higher moment condition than the regular ones. The smoothed discrete delta functions are tested by three test cases: a one-dimensional heat equation with a moving singular force, a two-dimensional flow past an oscillating cylinder, and the vortex-induced vibration of a cylinder. The numerical examples in these cases demonstrate that the smoothed discrete delta functions can effectively suppress the non-physical oscillations in the volume forces and improve the accuracy of the immersed boundary method with direct forcing in moving boundary simulations.

1. Introduction

Flows with moving boundaries and complex geometries are frequently encountered in engineering applications. Mesh generation and re-meshing process are the bottlenecks in computer simulations when conventional body-fitted methods are used to treat such problems. To reduce the complexity and the computational cost of the grid generation process, many non-body-conforming methods have been developed. Among these developments, the Immersed Boundary (IB) method receives much attention in recent years.

The IB method was first introduced by Peskin [1] in 1972 to simulate cardiac mechanics and associated blood flows. In the original IB method proposed by Peskin [1], the elastic boundary was mimicked by a set of Lagrangian points linked by springs, the force at the Lagrangian point was determined by some constitutive laws (e.g. Hooke's law), the Lagrangian variables and the Eulerian variables (such as force and the velocity) are related through a discrete delta function. A summary of the mathematical formulation and the applications of the IB method are given by Peskin [2]. The most recent review can be found in [3] by Mittal and Iaccarino.

* Corresponding author. Tel.: +86 10 82543969; fax: +86 10 62561284.
E-mail address: hgw@lnm.imech.ac.cn (G.-W. He).

0021-9991/$ - see front matter © 2009 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2009.07.023

The IB method was later extended to simulate problems with rigid boundaries by making the spring very stiff or by applying the concept of feedback control (e.g. [4,5]). Stringent restriction on the CFL number and the introduction of ad hoc parameters are the major drawbacks of these types of IB method. The direct forcing method was then proposed to overcome these drawbacks. In this type of IB method, the constitutive law was abandoned and the volume force is determined by a direct imposition of the velocity boundary condition. The direct forcing method was first implemented within the spectral framework by Mohd-Yusof [6]. Fadlun et al. [7] extended this approach to the formulation in the context of a finite difference method.

Almost all IB methods dealing with the elastic boundaries (or using a stiff spring to mimic the rigid boundaries) use a discrete delta function to spread the force to the Eulerian points. Here we only focus on the direct forcing IB methods for rigid boundaries. Among the IB methods of this category, some eliminate the use of the discrete delta function all together while others still use the discrete delta function to evaluate and distribute the force. The former variants of IB method usually resort to a local reconstruction of the velocity field, such as Fadlun et al. [7], Kim et al. [8], Tseng and Ferziger [9], Balaras [10], Yang and Balaras [11], etc., although the exact positions of forcing are slightly different among these works. In the later variants of IB method, the force is first evaluated at the Lagrangian points and then spread to its surrounding Eulerian grid points using a discrete delta function (Uhlmann [12], Su et al. [13] and Taira and Colonius [14]). There exist some slight differences among these three papers. In [12], the force at the boundary is determined explicitly while in [13,14] the force is computed implicitly by solving a linear system.

IB methods have been successfully applied to simulate stationary boundary problems, but non-physical force oscillations are produced by some variants of IB methods when dealing with moving boundaries. Uhlmann [15] tested various IB methods in simulating particulate flows. Yang and Balaras [11] used a "field-extension" approach to simulate flows with moving boundaries. Kim and Choi [16] developed an IB method in a non-inertial frame which can be used to simulate flows around an arbitrarily moving body. For the IB methods with discrete delta functions, Uhlmann [12] found that the oscillations can be controlled by a careful choice of the discrete delta function. However, the reasons behind the generation and the control of such oscillations have not been revealed.

In the present work, we develop an approach to suppress force oscillations in moving body simulations in the context of Uhlmann's IB method [12]. The central idea of the improvement is to apply a smoothing technique to the discrete delta functions. This technique can effectively reduce non-physical oscillations and lead to a significant improvement in the force prediction. The organization of the paper is as follows. The proposed smoothing technique for discrete delta functions is described in Section 2: the IB method with direct forcing for the Navier–Stokes equations is introduced in Section 2.1; the formulation of the smoothed discrete delta functions is presented in Section 2.2; the reason for the spurious force oscillations is provided in Section 2.3. Numerical results on the one-dimensional heat equation with a moving singular force, the two-dimensional flows past an oscillating cylinder and the vortex-induced vibration of a cylinder are presented in Section 3. Finally, the conclusions are drawn in Section 4.

2. The smoothing technique for discrete delta function

A discrete delta function is used as a kernel to transfer quantities between Lagrangian and Eulerian locations in the original and some variants of the IB method. The discrete delta function essentially determines the accuracy of the IB method. The commonly used discrete delta functions often introduce non-physical oscillations to the volume force in moving boundary simulations. In this section, we will develop a smoothing technique to construct new discrete delta functions from the regular ones. These smoothed discrete delta functions can significantly reduce the non-physical oscillations. In the first part of this section, we summarize the IB method used in the present work with an emphasis on how to calculate the volume forces. The smoothing technique is then presented and finally the reasons for the non-physical force oscillations in moving boundary simulation are analyzed.

2.1. The immersed boundary method with direct forcing

The incompressible Navier–Stokes equations with a volume force are

$$\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot (\mathbf{u}\mathbf{u}) = -\nabla p + \frac{1}{Re}\nabla^2 \mathbf{u} + \mathbf{f}, \tag{1}$$

$$\nabla \cdot \mathbf{u} = 0, \tag{2}$$

where $\mathbf{u}$ is the velocity vector, p the pressure and Re the Reynolds number. Here $\mathbf{f}$ denotes the volume force whose form will be given later. In the present IB method, the governing equations are solved on a fixed Cartesian grid; the geometry of the immersed surface is represented by a series of equally distributed Lagrangian markers with the arc-length comparable with the width of the surrounding Cartesian mesh (Eulerian points).

The momentum equation can be discretized in time in the following form

$$\frac{\mathbf{u}(\mathbf{x}, t^{n+1}) - \mathbf{u}(\mathbf{x}, t^n)}{\triangle t} = \mathbf{rhs}(\mathbf{x}, t^{n+1/2}) + \mathbf{f}(\mathbf{x}, t^{n+1/2}), \tag{3}$$

where $\mathbf{rhs}(\mathbf{x}, t^{n+1/2})$ represents the sum of the convection term, the viscous term and the pressure gradient at some intermediate time between t^n and t^{n+1}. The numerical procedure in the present IB method with direct forcing is similar to that in [12] and is summarized as follows:

1. Compute explicitly an estimated velocity $\tilde{\mathbf{u}}$ without volume forces;

$$\tilde{\mathbf{u}}(\mathbf{x}, t^{n+1/2}) = \mathbf{u}(\mathbf{x}, t^n) + \triangle t\ \mathbf{rhs}(\mathbf{x}, t^{n+1/2}). \tag{4}$$

2. Calculate the volume forces on the Lagrangian locations using the direct forcing method and spread them to the surrounding Eulerian locations through the discrete delta function;
3. Solve the Navier–Stokes equations on the Cartesian grid with the volume forces.

The second step is crucial to the present IB method with direct forcing. When enforcing the boundary condition, a volume force should be generated such that the desired velocities on the boundary are achieved. Therefore, the volume force can be expressed by

$$\mathbf{f}(\mathbf{x}, t^{n+1/2}) = \frac{\mathbf{v}(\mathbf{x}, t^{n+1}) - \mathbf{u}(\mathbf{x}, t^n)}{\triangle t} - \mathbf{rhs}(\mathbf{x}, t^{n+1/2}), \tag{5}$$

where $\mathbf{v}$ is the desired velocity at the boundary. For the Cartesian grid point which coincides with a Lagrangian marker, the volume force can be calculated directly using Eq. (5). However, on a moving or complex boundary, the Cartesian grid points do not necessarily coincide with any Lagrangian markers. In this case, the volume force needs to be evaluated through an interpolation at the Lagrangian markers using a discrete delta function,

$$\mathbf{f}(\mathbf{x}, t^{n+1/2}) = \sum_{l=1}^{N_l} \mathbf{F}(\mathbf{X}_l^{n+1}, t^{n+1/2}) \delta_h(\mathbf{x} - \mathbf{X}_l^{n+1}) \triangle V_l. \tag{6}$$

Here $\mathbf{F}$ is the force at the Lagrangian marker; N_l the total number of markers and $\triangle V_l$ the volume assigned to the lth Lagrangian marker (in three-dimensional case, $\triangle V_l \approx h^3$ where h is the size of the Eulerian grid [12]). Hereafter, we will use the lower-case letters for the quantities evaluated at the Eulerian locations and the upper-case letters for the ones at the Lagrangian locations. The transfer of the volume forces from the Lagrangian markers to the Cartesian grid points is usually referred to as "force spreading".

The three-dimensional discrete delta function used in IB method is usually constructed in the following form [2]:

$$\delta_h(\mathbf{x} - \mathbf{X}) = \frac{1}{h^3} \phi\left(\frac{x - X}{h}\right) \phi\left(\frac{y - Y}{h}\right) \phi\left(\frac{z - Z}{h}\right) \tag{7}$$

where $\phi(\frac{x-X}{h})/h$ is a one-dimensional discrete delta function, x, y and z are the Eulerian components of $\mathbf{x}$ and X, Y and Z the Lagrangian components of $\mathbf{X}$, respectively. It is noted that the two- or three-dimensional discrete delta functions can also be constructed by other means. For example, if the level set representation rather than the Lagrangian markers is used to track the boundary, discrete delta function can be constructed using the level set function [17–19].

Uhlmann [12] proposed to calculate the volume force at the Lagrangian marker as follows

$$\mathbf{F}(\mathbf{X}^{n+1}, t^{n+1/2}) = \frac{\mathbf{V}(\mathbf{X}^{n+1}, t^{n+1}) - \mathbf{U}(\mathbf{X}^{n+1}, t^n)}{\triangle t} - \mathbf{RHS}(\mathbf{X}^{n+1}, t^{n+1/2}). \tag{8}$$

Here $\mathbf{V}$ and $\mathbf{U}$ are the desired and estimated velocities at the Lagrangian markers, respectively. By regrouping the right-hand-side of Eq. (8), it can be expressed in the form of Eq. (9)

$$\mathbf{F}(\mathbf{X}^{n+1}, t^{n+1/2}) = \frac{\mathbf{V}(\mathbf{X}^{n+1}, t^{n+1}) - \widetilde{\mathbf{U}}(\mathbf{X}^{n+1}, t^{n+1/2})}{\triangle t}, \tag{9}$$

where

$$\widetilde{\mathbf{U}}(\mathbf{X}^{n+1}, t^{n+1/2}) = \mathbf{U}(\mathbf{X}^{n+1}, t^n) + \triangle t \mathbf{RHS}(\mathbf{X}^{n+1}, t^{n+1/2}). \tag{10}$$

By definition, $\widetilde{\mathbf{U}}$ is evaluated through an interpolation from its Eulerian counterpart $\tilde{\mathbf{u}}$,

$$\widetilde{\mathbf{U}}(\mathbf{X}^{n+1}, t^{n+1/2}) = \sum_{\mathbf{x} \in g_h} \tilde{\mathbf{u}}(\mathbf{x}, t^{n+1/2}) \delta_h(\mathbf{x} - \mathbf{X}^{n+1}) h^3. \tag{11}$$

Here g_h denotes the collection of Cartesian grid points.

2.2. A smoothed discrete delta function

In this section, we will construct a new discrete delta function using a smoothing technique. A one-dimensional function ϕ is taken as an example. The extension to two or three-dimensional discrete delta functions is straightforward since a two- or three-dimensional discrete delta function is the product of two or three of the one-dimensional ones.

For the one-dimensional function $\phi(x)$, we can define a new function ϕ^* as

$$\phi^*\left(\frac{x_j - X}{h}\right) = \frac{1}{h}\int_{x_j-0.5h}^{x_j+0.5h} \phi\left(\frac{x' - X}{h}\right)dx'. \tag{12}$$

The new function ϕ^* has one higher derivative than the original one ϕ. With this in mind, we refer to the new discrete delta function as the smoothed discrete delta function. Introducing the transformations $r = (x_j - X)h^{-1}$ and $r' = (x' - X)h^{-1}$, the new function ϕ^* can be simply reformulated as

$$\phi^*(r) = \int_{r-0.5}^{r+0.5} \phi(r')dr'. \tag{13}$$

A variety of the smoothed discrete delta functions can be constructed accordingly from the regular ones. In practice, the most commonly used discrete delta functions are the 2-point hat function [20], the 4-point cosine function [2], the 3-point discrete delta function [21], and the 4-point piecewise function [2]. The one-dimensional versions for those discrete delta functions are listed below for reference.

- A 2-point hat function ϕ_1

$$\phi_1(r) = \begin{cases} 1 - |r|, & |r| \le 1, \\ 0, & 1 \le |r|. \end{cases} \tag{14}$$

- A 4-point cosine function ϕ_2

$$\phi_2(r) = \begin{cases} \frac{1}{4}\left(1 + cos(\frac{\pi r}{2})\right), & |r| \le 2, \\ 0, & 2 \le |r|. \end{cases} \tag{15}$$

- A 3-point function ϕ_3

$$\phi_3(r) = \begin{cases} \frac{1}{3}\left(1 + \sqrt{-3r^2 + 1}\right), & |r| \le 0.5, \\ \frac{1}{6}\left(5 - 3|r| - \sqrt{-3(1 - |r|)^2 + 1}\right), & 0.5 \le |r| \le 1.5, \\ 0, & 1.5 \le |r|. \end{cases} \tag{16}$$

- A 4-point piecewise function ϕ_4

$$\phi_4(r) = \begin{cases} \frac{1}{8}\left(3 - 2|r| + \sqrt{1 + 4|r| - 4r^2}\right), & |r| \le 1, \\ \frac{1}{8}\left(5 - 2|r| - \sqrt{-7 + 12|r| - 4r^2}\right), & 1 \le |r| \le 2, \\ 0, & 2 \le |r|. \end{cases} \tag{17}$$

In terms of the definition (13), we can formulate the smoothed functions $\phi_1^*, \phi_2^*, \phi_3^*$ and ϕ_4^* corresponding to ϕ_1, ϕ_2, ϕ_3 and ϕ_4, respectively.

- A smoothed 2-point function ϕ_1^*

$$\phi_1^*(r) = \begin{cases} 3/4 - r^2, & |r| \le 0.5, \\ 9/8 - 3|r|/2 + r^2/2 & ,0.5 \le |r| \le 1.5, \\ 0, & 1.5 \le |r|. \end{cases} \tag{18}$$

- A smoothed 4-point cosine function ϕ_2^*

$$\phi_2^*(r) = \begin{cases} \frac{1}{4\pi}\left(\pi + 2\sin\left(\frac{\pi}{4}(2r+1)\right) - 2\sin\left(\frac{\pi}{4}(2r-1)\right)\right), & |r| \le 1.5, \\ -\frac{1}{8\pi}\left(-5\pi + 2\pi|r| + 4\sin\left(\frac{\pi}{4}(2|r|-1)\right)\right), & 1.5 \le |r| \le 2.5, \\ 0, & 2.5 \le |r|. \end{cases} \tag{19}$$

- A smoothed 3-point function ϕ_3^*

$$\phi_3^*(r) = \begin{cases} \frac{17}{48} + \frac{\sqrt{3}\pi}{108} + \frac{|r|}{4} - \frac{r^2}{4} + \frac{1-2|r|}{16}\sqrt{-12r^2 + 12|r| + 1} & \\ \quad - \frac{\sqrt{3}}{12}\arcsin\left(\frac{\sqrt{3}}{2}(2|r| - 1)\right), & |r| \le 1, \\ \frac{55}{48} - \frac{\sqrt{3}\pi}{108} - \frac{13|r|}{12} + \frac{r^2}{4} + \frac{2|r|-3}{48}\sqrt{-12r^2 + 36|r| - 23} & \\ \quad + \frac{\sqrt{3}}{36}\arcsin\left(\frac{\sqrt{3}}{2}(2|r| - 3)\right), & 1 \le |r| \le 2, \\ 0, & 2 \le |r|. \end{cases} \tag{20}$$

- A smoothed 4-point piecewise function ϕ_4^*

$$\phi_4^*(r)=\begin{cases}\frac{3}{8}+\frac{\pi}{32}-\frac{r^2}{4}, & |r|\le 0.5,\\ \frac{1}{4}+\frac{1-|r|}{8}\sqrt{-2+8|r|-4r^2}-\frac{1}{8}\arcsin\left(\sqrt{2}(|r|-1)\right), & 0.5\le|r|\le 1.5,\\ \frac{17}{16}-\frac{\pi}{64}-\frac{3|r|}{4}+\frac{r^2}{8}+\frac{|r|-2}{16}\sqrt{-14+16|r|-4r^2} & \\ \quad+\frac{1}{16}\arcsin\left(\sqrt{2}(|r|-2)\right), & 1.5\le|r|\le 2.5,\\ 0, & 2.5\le|r|.\end{cases} \tag{21}$$

Besides the discrete delta functions mentioned above, the 4-point hat function which is in the form of

$$\phi_5(r)=\begin{cases}0.5-0.25|r|, & |r|\le 2,\\ 0, & 2\le|r|\end{cases} \tag{22}$$

is also considered to test the smoothing effect of a discrete delta function with a wider support (this test will be discussed in Section 3.2.2).

Fig. 1 compares the smoothed discrete delta functions ϕ_i^* $(i=1,2,3,4)$ with the regular ones. It is easy to see that the new functions ϕ_i^* tend to have wider supports than the regular ones ϕ_i. However, the wider supports are not sufficient to reduce the non-physical oscillations effectively. The relevant properties are the moment conditions, which are given below for the regular and smoothed discrete delta functions. These properties will be used to analyze the force oscillations in moving boundary simulations in the next section.

1. The smoothed discrete delta functions have one higher derivative than the regular ones. In Table 1, we compare the continuity properties of the regular discrete delta functions with those of the smoothed ones. Here C^n denotes the set of function composed of the functions that have up to nth-order derivatives.
2. All the smoothed discrete delta functions and regular discrete delta functions satisfy the zeroth and first discrete moment conditions except for the regular 4-point cosine function ϕ_2 which only satisfies the zeroth discrete moment condition.

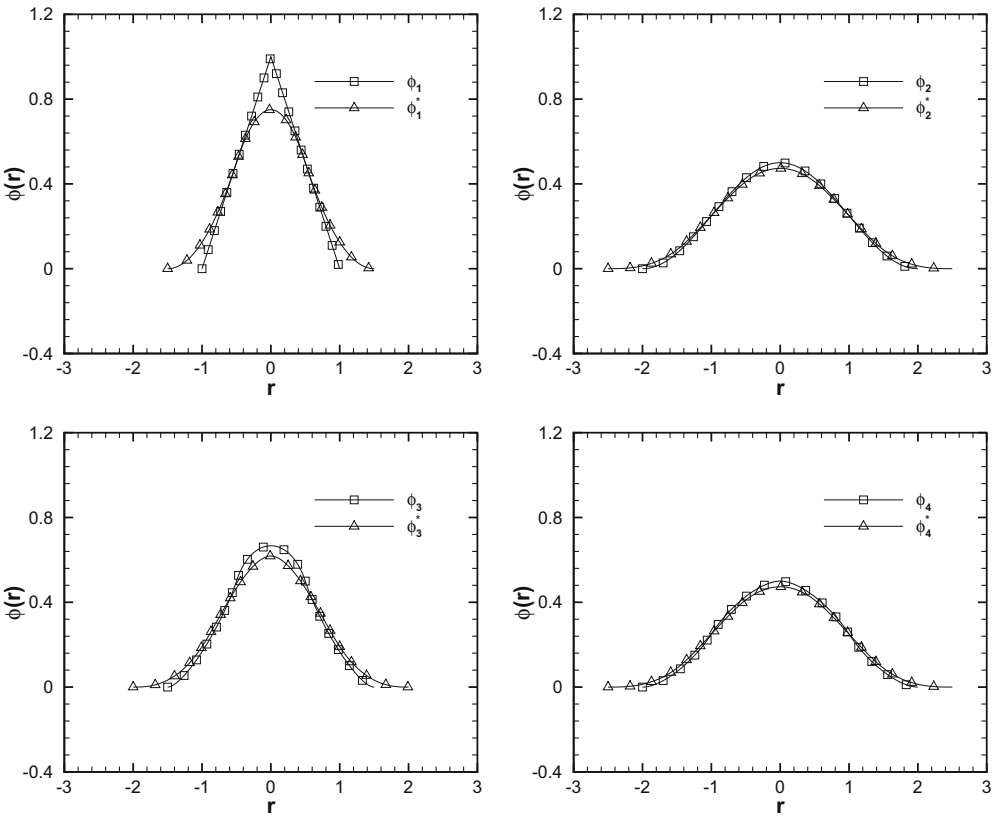

Fig. 1. Comparison of regular functions ϕ_i $(i=1,2,3,4)$ and their corresponding smoothed functions ϕ_i^* $(i=1,2,3,4)$.

Table 1
The continuity of regular discrete delta functions ϕ_i $(i = 1,2,3,4,5)$ and the smoothed ones ϕ_i^* $(i = 1,2,3,4)$.

ϕ_1	ϕ_1^*	ϕ_2	ϕ_2^*	ϕ_3	ϕ_3^*	ϕ_4	ϕ_4^*	ϕ_5
C^0	C^1	C^1	C^2	C^1	C^2	C^1	C^2	C^0

Table 2
The moment conditions for ϕ_i $(i = 1,2,3,4,5)$ and ϕ_i^* $(i = 1,2,3,4)$.

ϕ_1	ϕ_1^*	ϕ_2	ϕ_2^*	ϕ_3	ϕ_3^*	ϕ_4	ϕ_4^*	ϕ_5
1	1	0	1	1	1	1	1	1

Therefore, ϕ_2 gives a first-order approximation and the others give second-order approximations (please see [2] for detailed explanations). In Table 2, we list the discrete moment conditions for the regular discrete delta functions and the smoothed ones. Here the number n indicates that a discrete delta function satisfies up to nth order moment conditions. The proofs for the moment conditions are given below. We first check the zeroth discrete moment condition

$$\sum_j \phi_i^*(r-j) = \sum_j \int_{r-j-1/2}^{r-j+1/2} \phi_i(r')dr' = \int_{r-1/2}^{r+1/2} \sum_j \phi_i(r'-j)dr' = 1, \tag{23}$$

which holds for $i = 1,2,3,4$. It is noted that each function ϕ_i $(i = 1,2,3,4)$ also satisfies the first-order continuous moment condition

$$\int_{-\infty}^{+\infty} r'\phi_i(r')dr' = \sum_j \int_{r-j-1/2}^{r-j+1/2} r'\phi_i(r')dr' = 0. \tag{24}$$

Using the first-order continuous moment condition (24) and the zeroth discrete moment condition for the regular delta functions ϕ_i $(i = 1,2,3,4)$, we can verify that the first discrete moment condition holds for the smoothed discrete delta functions.

$$\begin{aligned}\sum_j (r-j)\phi_i^*(r-j) &= \sum_j (r-j)\int_{r-j-1/2}^{r-j+1/2} \phi_i(r')dr' = \sum_j \int_{r-j-1/2}^{r-j+1/2} (-r'+r-j)\phi_i(r')dr' \\ &= -\int_{-1/2}^{1/2} r'' \sum_j \phi_i(r''+r-j)dr'' = 0.\end{aligned} \tag{25}$$

3. It can be shown that the derivatives of the smoothed discrete delta functions ϕ_1^*, ϕ_3^* and ϕ_4^* satisfy the zeroth, the first and the second discrete moment conditions, while the derivatives of the regular ones ϕ_3 and ϕ_4 satisfy the zeroth and the first discrete moment conditions. As to the smoothed 4-point cosine function ϕ_2^*, its derivative satisfies the zeroth and the first discrete moment conditions, while the derivative of the corresponding regular one satisfies only the zeroth discrete moment condition. In Table 3, we list the discrete moment conditions satisfied by the derivatives of the discrete delta functions. Note that ϕ_1 is not differentiable, it is not included in Table 3.

The moment conditions for the derivatives of the discrete delta functions $\phi(r)$ can be formulated in the same way as the ones for the discrete delta functions themselves. These moment conditions for the derivatives are listed below

$$\sum_j \frac{d\phi(r-j)}{dr} = 0, \tag{26}$$

$$\sum_j (r-j)\frac{d\phi(r-j)}{dr} = -1, \tag{27}$$

$$\sum_j (r-j)^m \frac{d\phi(r-j)}{dr} = 0, \quad m = 2,\ldots,p. \tag{28}$$

Table 3
The moment conditions satisfied by the derivatives $d\phi_i/dr$ $(i = 2,3,4)$ and $d\phi_i^*/dr$ $(i = 1,2,3,4)$.

$d\phi_1^*/dr$	$d\phi_2/dr$	$d\phi_2^*/dr$	$d\phi_3/dr$	$d\phi_3^*/dr$	$d\phi_4/dr$	$d\phi_4^*/dr$
2	0	1	1	2	1	2

It is easy to verify that the derivatives of ϕ_i $(i = 2,3,4)$ and ϕ_i^* $(i = 1,2,3,4)$ satisfy the zeroth order moment condition

$$\sum_j \left(\frac{d\phi(r-j)}{dr}\right) = \frac{d}{dr}\left(\sum_j \phi(r-j)\right) = 0. \tag{29}$$

It can be shown that the first discrete moment condition holds only for the derivatives of the regular functions ϕ_i $(i = 3,4)$ while the same condition holds for *all* derivatives of the smoothed functions ϕ_i^* $(i = 1,2,3,4)$

$$\sum_j (r-j)\frac{d\phi(r-j)}{dr} = \sum_j \left(\frac{d}{dr}((r-j)\phi(r-j)) - \phi(r-j)\right) = \frac{d}{dr}\left(\sum_j (r-j)\phi(r-j)\right) - \sum_j \phi(r-j) = -1, \tag{30}$$

where the zeroth and the first discrete moment conditions for the function itself are used in the proof.

Finally, we show that the derivatives of the smoothed functions ϕ_i^* $(i = 1,3,4)$ satisfy the second discrete moment condition as well. The first derivative of the smoothed discrete delta functions can be written as

$$\frac{d\phi_i^*(r-j)}{dr} = \frac{d}{dr}\int_{r-j-1/2}^{r-j+1/2} \phi_i(r')dr' = \phi_i(r-j+1/2) - \phi_i(r-j-1/2). \tag{31}$$

Using Eq. (31), the zeroth and the first discrete moment conditions of the function ϕ_i $(i = 1,3,4)$, we can express the second discrete moment condition as

$$\sum_j (r-j)^2 \frac{d\phi_i^*(r-j)}{dr} = -2r + \sum_j j^2 \phi_i(r-j+1/2) - \sum_j j^2 \phi_i(r-j-1/2). \tag{32}$$

Since the second term on the right-hand-side of the above equation can be written as

$$\sum_j j^2 \phi_i(r-j-1/2) = \sum_j j^2 \phi_i(r-j+1/2) - 2r, \tag{33}$$

we thus obtain the second discrete moment condition

$$\sum_j (r-j)^2 \frac{d\phi_i^*(r-j)}{dr} = 0. \tag{34}$$

2.3. Analysis of non-physical force oscillations in moving boundary problems

The interpolation properties of the discrete delta functions are analyzed in this part to find out the reasons for the non-physical force oscillations in moving boundary problems. If $f(x,t)$ is an arbitrary function of variables x and t, $f(X^{n+1},t^{n+1})$ can be approximated by the following expression

$$f(X^{n+1},t^{n+1}) \approx \sum_j f(x_j,t^{n+1})\phi(x_j - X^{n+1}). \tag{35}$$

And the error term E^{n+1} is

$$E^{n+1} = f(X^{n+1},t^{n+1}) - \sum_j f(x_j,t^{n+1})\phi(x_j - X^{n+1}). \tag{36}$$

If $f(x,t)$ is a smooth function of x and t, then $f(X^{n+1},t^{n+1})$ can be expanded around (X^n,t^n) in a Taylor series as

$$f(X^{n+1},t^{n+1}) = f(X^n,t^n) + \frac{\partial f(X^n,t^n)}{\partial t}\Delta t + \frac{\partial f(X^n,t^n)}{\partial x}\Delta X + \sum_{m=2}^{\infty}\frac{1}{m!}\left(\Delta t\frac{\partial}{\partial t} + \Delta X\frac{\partial}{\partial x}\right)^m f(X^n,t^n) \tag{37}$$

and $f(x_j,t^{n+1})$ can also be expanded around t^n in a Taylor series as

$$f(x_j,t^{n+1}) = f(x_j,t^n) + \frac{\partial f(x_j,t^n)}{\partial t}\Delta t + \sum_{m=2}^{\infty}\frac{1}{m!}\frac{\partial^m f(x_j,t^n)}{\partial t^m}\Delta t^m. \tag{38}$$

Suppose $\phi(x_j - x)$ is a smooth function of x, then we can expand $\phi(x_j - X^{n+1})$ around $(x_j - X^n)$ in a Taylor series as

$$\phi(x_j - X^{n+1}) = \phi(x_j - X^n) + \frac{\partial\phi(x_j - X^n)}{\partial x}\Delta X + \sum_{m=2}^{\infty}\frac{1}{m!}\frac{\partial^m\phi(x_j - X^n)}{\partial x^m}\Delta X^m. \tag{39}$$

Using Eqs. (37)–(39), the error term E^{n+1} in Eq. (36) can be written as

$$E^{n+1} = E^n + \left(\frac{\partial f(X^n,t^n)}{\partial t} - \sum_j \frac{\partial f(x_j,t^n)}{\partial t}\phi(x_j - X^n)\right)\Delta t + \left(\frac{\partial f(X^n,t^n)}{\partial x} - \sum_j f(x_j,t^n)\frac{\partial \phi(x_j - X^n)}{\partial x}\right)\Delta X + \sum_{m=2}^{\infty}\frac{1}{m!}\left(\Delta t\frac{\partial}{\partial t} + \Delta X\frac{\partial}{\partial x}\right)^m\left(f(X^n,t^n) - \sum_j f(x_j,t^n)\phi(x_j - X^n)\right), \tag{40}$$

where $E^n = f(X^n,t^n) - \sum_j f(x_j,t^n)\phi(x_j - X^n)$ represents the error at time t^n. For simplicity, here we only consider the errors introduced within Δt (from time t^n to time step t^{n+1}). The second term on the right-hand-side of Eq. (40) appears in either stationary or moving body problems, while the third term associated with the body movement only appears in moving body problem. It is obvious that the second term is of order $O(h^{p+1}\Delta t)$ if the discrete delta function satisfies the first p discrete moment conditions and the pth derivative of the interpolated function $f^{(p)}(x,t)$ is Lipschitz continuous on the interval $[X^n - Mh, X^n + Mh]$. Here $2Mh$ denotes the support width of the discrete delta function. The third term can be re-formulated as

$$\left(\frac{\partial f(X^n,t^n)}{\partial x} - \sum_j f(x_j,t^n)\frac{\partial \phi(x_j - X^n)}{\partial x}\right)\Delta X = \left(\frac{\partial f(X^n,t^n)}{\partial x} - \sum_j\left(\sum_{m=0}^{\infty}\frac{1}{m!}\frac{\partial^m f(X^n,t^n)}{\partial x^m}(x_j - X^n)^m\right)\frac{\partial \phi(x_j - X^n)}{\partial x}\right)\Delta X$$
$$= \left(\frac{\partial f(X^n,t^n)}{\partial x} - \sum_{m=0}^{\infty}\frac{1}{m!}\frac{\partial^m f(X^n,t^n)}{\partial x^m}\sum_j (x_j - X^n)^m\frac{\partial \phi(x_j - X^n)}{\partial x}\right)\Delta X \tag{41}$$

Here, the first equality is obtained using the Taylor series expansion of the function $f(x_j,t^n)$ around X^n. From this expression we can see that the error of the third term on the right-hand-side of Eq. (40) is of order $h^q\Delta X$ if the derivative of the discrete delta function satisfies the first q discrete moment conditions and the qth derivative of the interpolated function $f^{(q)}(x,t)$ is Lipschitz continuous on the interval $[X^n - Mh, X^n + Mh]$.

The analysis above can be summarized below:

1. For the stationary body problems, $\triangle X = 0$ and the third term on the right-hand-side of Eq. (40) vanishes. The error can be expressed as
$$E^{n+1} = E^n + O(h^{p+1}\triangle t), \tag{42}$$
if the discrete delta function satisfies the first p moment conditions and $f^{(p)}(x,t)$ is Lipschitz continuous on the interval $[X^n - Mh, X^n + Mh]$. The approximation accuracy depends on the discrete delta function alone and no spurious force oscillation is present.
2. For the moving body problems, the third term in Eq. (40) introduces an additional error and the total error becomes
$$E^{n+1} = E^n + O(h^{p+1}\triangle t) + O(h^q\triangle X), \tag{43}$$
if the discrete delta function satisfies the first p moment conditions, its first derivative satisfies the first q moment conditions and $f^{(\max(p,q))}(x,t)$ is Lipschitz continuous on the interval $[X^n - Mh, X^n + Mh]$. It is noted that the above analysis is only valid when ϕ has at least first-order derivative. For the 2-point hat function ϕ_1 which is only of C^0 continuity, this analysis is not applicable.

3. Numerical examples

In this section, three canonical numerical examples are used to test the effect of the smoothed discrete delta functions in moving boundary simulations. The first example is the one-dimensional heat equation with a moving singular force term. Such equation is of interest because it serves as a standard model problem and the analytical solutions can be easily obtained. The second one is the simulation of flow past a circular cylinder by solving the Navier–Stokes equations. Although no analytical solutions are available for this problem, abundant numerical and experimental results can be found in the literature. The third example is the vortex-induced vibration of a cylinder. The purpose of this case is to test the validity of this approach in simulating Flow Structure Interaction (FSI) problems.

In all numerical tests presented in this paper, the same discrete delta function is used in the force spreading (Eq. (6)) and the interpolation (Eq. (11)).

3.1. The one-dimensional heat equation with a moving singular force

We consider the one-dimensional heat equation with a moving singular force,

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} + F(t)\delta(x - X(t)), \quad 0 \le x \le 1, \tag{44}$$

with the Dirichlet boundary conditions $u(0) = u(1) = 0$. Here the forcing term is non-zero only at the point $X(t)$. In this test, we compare our numerical result with the exact solution

$$u(x,t)=\begin{cases}\sin(\omega_1 x)e^{-\omega_1^2 t}, & x\le X(t),\\ \sin(\omega_2(1-x))e^{-\omega_2^2 t}, & X(t)\le x.\end{cases} \tag{45}$$

We choose $\omega_1=5\pi/4$ and $\omega_2=7\pi/4$, respectively. The position of the singular force $X(t)$ is determined by the continuity condition of the solution u at $X(t)$, i.e.

$$\sin(\omega_1 X(t))e^{-\omega_1^2 t}=\sin(\omega_2(1-X(t)))e^{-\omega_2^2 t}. \tag{46}$$

The exact forcing term is then determined by $F_e(t)=-[u_x]_{X(t)}$, which can be easily calculated from the exact solution. Here $[\,]_{X(t)}$ denotes the jump of a quantity at point $X(t)$. We can write the discretized equation in the following form

$$\frac{u(x_j,t^{n+1})-u(x_j,t^n)}{\triangle t}=\frac{L_h(u(x_j,t^{n+1}))+L_h(u(x_j,t^n))}{2}+F(t^{n+1})\delta_h(x_j-X(t^{n+1})). \tag{47}$$

Here L_h is an operator that denotes the central finite difference approximation to the second derivatives and δ_h is the discrete delta function.

Two different forcing approaches are used to determine the singular force. The first approach (explicit forcing) is analogous to the one introduced in Section 2.1 for the Navier–Stokes equations. Please note that in order to obtain the force in Eq. (44), the calculated force needs to be multiplied by a volume (which is h in the present one-dimensional example), since a volume is associated with the force in the present explicit forcing method. The second approach is the implicit forcing. Please see Beyer and LeVeque [22] for the one-dimensional heat equation and Su et al. [13] for the Navier–Stokes equations. In this approach, the force is determined from the constraint that the desired solution at the singular point X is satisfied at time step t^{n+1} when we interpolate $u(x,t^{n+1})$ to X^{n+1}. For the present one-dimensional heat equation, the implicit forcing method has two steps. In the first step, the discrete form of Eq. (47) can be written in a matrix–vector form

$$Au^{n+1}=Bu^n+\triangle tF^{n+1}\delta_h^{n+1}, \tag{48}$$

where δ_h is a column vector with elements $\delta_h(x_j-X)$. The solution of this equation is written as

$$u^{n+1}=A^{-1}Bu^n+\triangle tF^{n+1}A^{-1}\delta_h^{n+1}. \tag{49}$$

The second step is to interpolate Eq. (49) to the singular point using the same discrete delta function. Thus the force at the singular point using the implicit forcing approach is

$$F^{n+1}=(\triangle t r_{n+1}^T A^{-1}\delta_h^{n+1})^{-1}(V^{n+1}-r_{n+1}^T A^{-1}Bu^n), \tag{50}$$

where V^{n+1} is the desired value of u at the singular point at time step $n+1$. Here the row vector r^T has the components $h\delta_h(x_j-X)$. In our numerical tests, the Cartesian grids are evenly distributed with grid size h. The Crank–Nicholson scheme is used in the time advancing. The exact solution is used as the initial condition at $t=0$. The spacing h is 0.01. The results obtained by using ϕ_1 and ϕ_1^* are compared.

The time-variation of $F(t)$ by using the explicit forcing method is shown in Fig. 2. From Fig. 2, it is seen that the solution is contaminated by wiggles when the 2-point hat function ϕ_1 is used, whereas the smoothed discrete hat function ϕ_1^* effectively suppresses the oscillations and improves the quality of the solution. Fig. 3 shows that the smoothed discrete delta function also significantly reduces oscillations when the implicit forcing method is used. Grid refinement studies are performed for both methods and similar results are obtained at different mesh resolution.

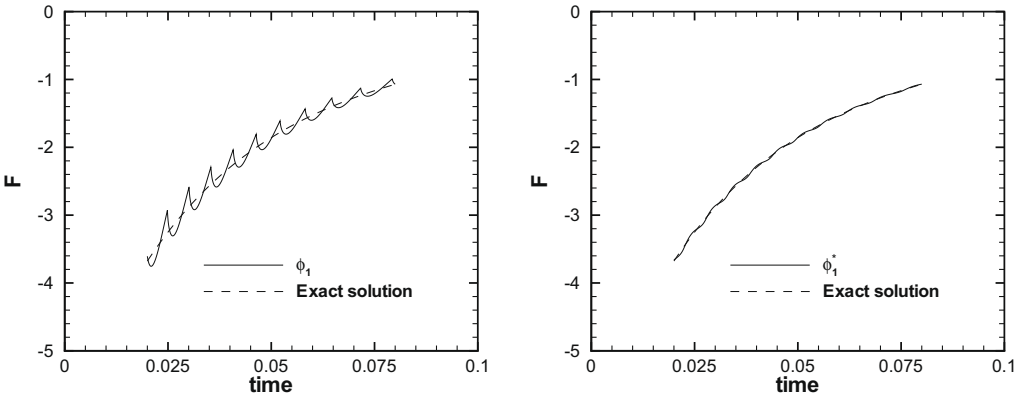

Fig. 2. Time variations of the singular force calculated by explicit forcing method for the one-dimensional heat equation with a moving singular force. Left: ϕ_1; right: ϕ_1^*.

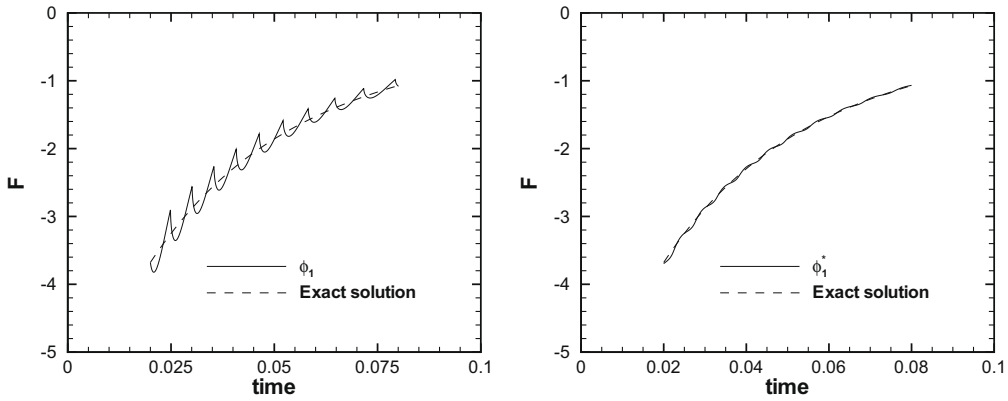

Fig. 3. Time variations of the singular force calculated by implicit forcing method for the one-dimensional heat equation with a moving singular force. Left: ϕ_1; right: ϕ_1^*.

3.2. A two-dimensional flow past a circular cylinder

We consider a two-dimensional problem of viscous flow past a stationary and oscillating cylinder. A standard fractional step method is used to solve the incompressible Navier–Stokes equations. The second-order finite volume formulation is used for the spatial discretization in the momentum equation. As to the time advancing, the third-order Runge–Kutta scheme and the Crank–Nicholson scheme are used for the explicit terms and implicit terms, respectively. The domain size is $50d \times 30d$, where d is the diameter of the cylinder. A uniform velocity boundary condition is specified at the inlet; a convective boundary condition is applied at the outlet. At the far field in the crosswise direction, shear-free boundary conditions are used.

3.2.1. Stationary cylinder

To validate the code, the flow past a stationary cylinder is computed first. The Reynolds number based on the inlet velocity and the cylinder diameter d is $Re_d = u_\infty d/\nu = 100$. The grid in the region that is occupied by cylinder is of the uniform size of $0.04d$. The number of grid points is 206 and 159 in the streamwise and crosswise directions, respectively (see Fig. 4). The time step is $\Delta t = 5 \times 10^{-3}$.

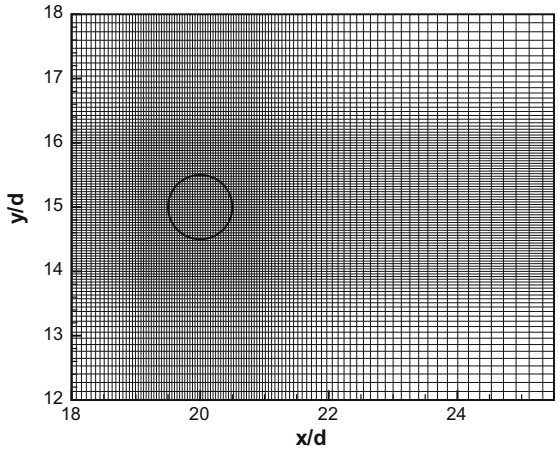

Fig. 4. A sketch of the grid employed in the vicinity of the cylinder for the present simulations.

Table 4
Drag and lift coefficients, and the Strouhal number for flow around a stationary cylinder at $Re_d = 100$.

	$\overline{C}_D$	C'_L	St
ϕ_4^*	1.393	0.335	0.165
ϕ_4	1.402	0.336	0.164
Uhlmann [12]	1.453	0.339	0.169
Liu et al. [23]	1.350	0.339	0.165

In the present study on the stationary problems, the drag and the lift on the cylinder are evaluated as the summation of the volume force as shown in Eq. (8) in the streamwise and crosswise directions, respectively. For the moving boundary problems, the inertial force of the "pseudo fluid" enclosed by the immersed boundary needs to be subtracted. For the details of the hydrodynamic force calculation, please refer to paper [15]. The results of the force prediction are summarized in Table 4. At this moderate mesh resolution, the Strouhal number ($St = fd/u_\infty$), the time-averaged drag $\overline{C}_D$ and the amplitude of lift fluctuations C'_L obtained from the smoothed function ϕ^*_4 are in good agreement with those from the regular discrete delta function ϕ_4 and the results in Ref. [12,23]. Other discrete delta functions (both the smoothed ones and the regular ones) are also tested and similar results are obtained. The time histories of the drag and the lift (not shown here) from these predictions are also compared. It is found that the smoothed discrete delta functions and the regular ones produce almost identical results and no spurious oscillations are seen in any of these force predictions.

3.2.2. Forced vibration of a cylinder

To test the effect of the smoothed discrete delta functions on the force prediction in moving body problems, simulations of the flow past an oscillating cylinder are performed. The Reynolds number based on the diameter of the cylinder is $Re_d = 185$. The cylinder moves in a sinusoidal fashion in the crosswise direction, with an amplitude of $A = 0.2d$ and a frequency of $f_o = 0.8f_s$ where f_s is the natural shedding frequency from the stationary cylinder at $Re_d = 185$ ($f_o = 0.156$ is used in the present study). Mathematically, the motion of the cylinder is prescribed as

$$y(t) = A\sin(2\pi f_o t). \tag{51}$$

To facilitate comparison, the amplitude and the frequency used in this study are the same as those in [12,24]. The grid in the region that is occupied by cylinder is of the uniform size of $0.02d$. The number of grid points is 275×239 in the streamwise and crosswise directions, respectively. The time step is $\Delta t = 2 \times 10^{-3}$.

The results are summarized in Table 5. Results from enlarged computational domains, smaller time step and refined mesh are also included. It is observed that all the simulations produce reasonably good results of the time-averaged forces, using either the smoothed discrete delta functions or the regular ones. However, if the plots of time-variations of the drag and the lift are carefully examined, noticeable effect of the smoothed discrete delta functions on the force prediction is evident.

Hereafter, in the discussion of force oscillations by using different discrete delta functions, the result from the smallest domain $50d \times 30d$ with the grid size of $0.02d$ and the time step of 2×10^{-3} will be used. Similar trends are also found in other results with different domain sizes, mesh sizes and time steps.

In Fig. 5, we show the periodic variations of the drag and the lift as a function of the cylinder's vertical position. The results obtained using ϕ_1 and ϕ^*_1 are displayed side-by-side for comparison. We can see that the force predictions obtained using the regular discrete delta function are contaminated by very strong oscillations whereas such oscillations are significantly suppressed and high frequency 'noises' are eliminated using the smoothed discrete delta function.

The comparison of the results obtained using the 4-point cosine discrete delta functions (the smoothed and the regular one) is presented in Fig. 6. Since the spurious oscillations in the drag are more evident than those in the lift, only the time-variation of the drag coefficient is plotted. Although the oscillations from the regular function ϕ_2 are not as strong as those from the hat function ϕ_1, large wiggles are clearly seen in the plot. With the use of the smoothed discrete delta function, the quality of force prediction is greatly improved.

The results from the 3-point and the 4-point piecewise discrete delta functions are shown in Figs. 7 and 8, respectively. The amplitude of oscillations is much reduced in comparison with the results from ϕ_1 and ϕ_2 (see Figs. 5 and 6). However,

Table 5
Dimensionless coefficients for flow around an oscillating cylinder at $Re_d = 185$.

Method	$\overline{C}_D$	$(C_D)_{rms}$	$(C_L)_{rms}$
$\phi^*_1(\phi_1)$	1.280[a](1.277[b])	0.042[a](0.042[b])	0.070[a](0.071[b])
$\phi^*_2(\phi_2)$	1.291[a](1.290[b])	0.043[a](0.043[b])	0.070[a](0.070[b])
$\phi^*_3(\phi_3)$	1.286[a](1.283[b])	0.042[a](0.042[b])	0.070[a](0.070[b], 0.190[c])
$\phi^*_4(\phi_4)$	1.292[a](1.290[b])	0.043[a](0.043[b])	0.070[a](0.070[b])
ϕ^*_3, enlarged domain $80d \times 50d$	1.278	0.043	0.074
ϕ^*_3, enlarged domain $80d \times 100d$	1.276	0.043	0.073
ϕ^*_3, small time step $\Delta t = 1 \times 10^{-3}$	1.273	0.041	0.083
ϕ^*_3, refined mesh[d] $h = 0.01d$	1.281	0.042	0.076
Uhlmann [12]	1.354	–	0.166[e]
Guilmineau and Queutey [24]	1.195[f]	0.036[f]	0.08[f]

[a] Results from ϕ^*_i $(i = 1,2,3,4)$.
[b] Results from ϕ_i $(i = 1,2,3,4)$.
[c] Result from ϕ_3, in which the lift force is evaluated by summing the volume force in Eq. (8).
[d] The time step Δt is 1×10^{-3}.
[e] The force is evaluated by summing the volume force in Eq. (8).
[f] The data are digitized from the Fig. 18 in Guilmineau and Queutey's paper [24].

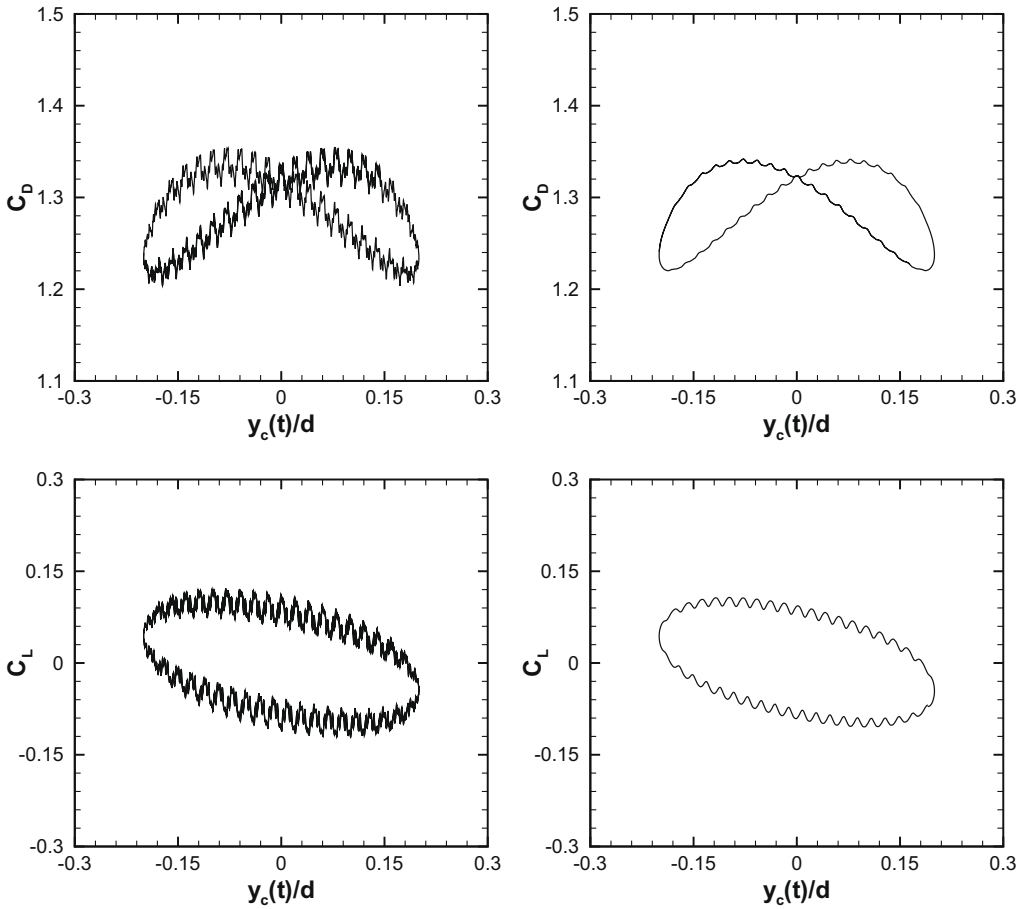

Fig. 5. Time–periodic variation of the drag and lift coefficients for flow past an oscillating cylinder at $Re_d = 185$. Left column: ϕ_1; right column: ϕ_1^*.

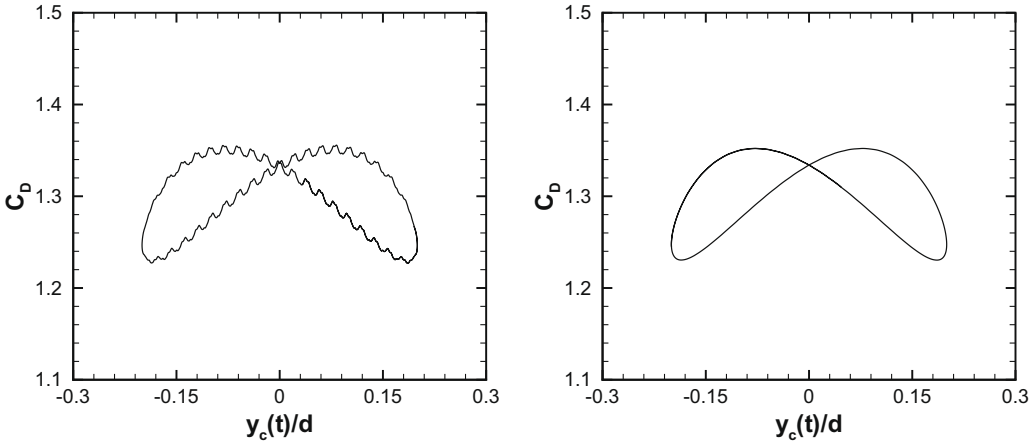

Fig. 6. Time–periodic variation of the drag coefficient for flow past an oscillating cylinder at $Re_d = 185$. Left graph: ϕ_2; right graph: ϕ_2^*.

small kinks still persist, especially near the peak value of the drag. The use of the smoothed discrete delta function provides an effective approach to eliminate these kinks.

The numerical observations can be interpreted as follows based on the analysis presented in Section 2.3. The large oscillations from the 4-point cosine discrete delta function are due to the violation of the first moment condition in its first derivative. The smoothing process rectifies this non-smooth error by making its first derivative satisfy the first moment condition. Furthermore, the smoothed discrete delta functions become continuously differentiable up to the second order, this also helps in the reduction of non-smooth errors. The smaller oscillations in the piecewise 3-point and 4-point discrete delta functions

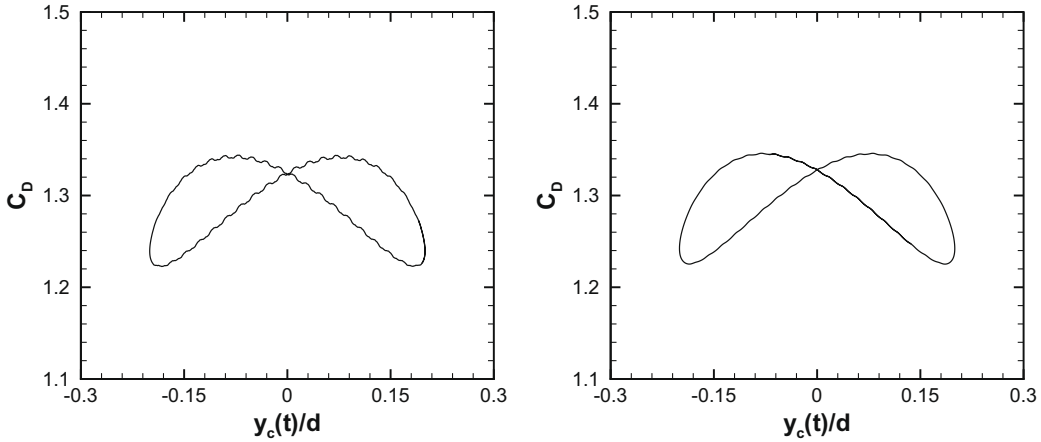

Fig. 7. Time–periodic variation of the drag coefficient for flow past an oscillating cylinder at $Re_d = 185$. Left graph: ϕ_3; right graph: ϕ_3^*.

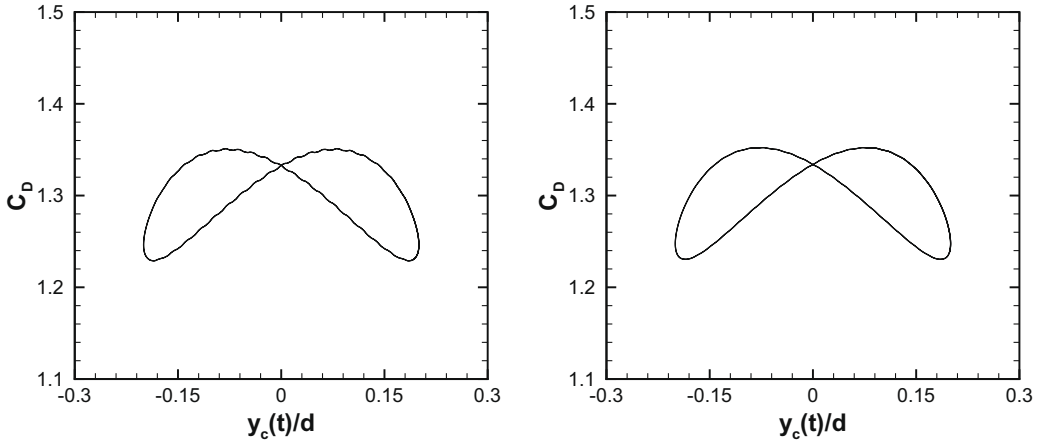

Fig. 8. Time–periodic variation of the drag coefficient for flow past an oscillating cylinder at $Re_d = 185$. Left graph: ϕ_4; right graph: ϕ_4^*.

are due to their differentiability up to the first order and the satisfaction of the moment condition up to the first order in their derivatives. Some improvements from the use of smoothed discrete delta functions in these two cases are achieved by making the discrete delta functions differentiable up to the second order and their first derivatives satisfy the second moment condition. As to the 2-point hat function, the analysis in Section 2.3 is inapplicable. Note that both ϕ_1 and ϕ_1^* satisfy the zeroth and first moment conditions. The derivative of ϕ_1 does not exist while the derivative of ϕ_1^* satisfies up to the zeroth, first and second moment conditions. These probably help to reduce the non-physical force oscillations produced by ϕ_1.

It is observed from Eqs. (18)–(21) that the smoothed discrete delta functions are distributed on a wider support than that of the regular discrete delta functions (e.g. ϕ_1 is defined on $|r| \leq 1$, while ϕ_1^* is defined on $|r| \leq 1.5$). It is then speculated that the similar smoothing effect could also be achieved by simply defining a regular discrete delta function on a wider range to include more grid points for interpolation. To test the effect of supporting width on the force prediction, we construct a 4-point hat function ϕ_5 (its form is given by Eq. (22)). The periodic variation of the drag coefficient calculated from this discrete delta function is compared with that from the 2-point hat function in Fig. 9. It is seen that in the results from ϕ_5, the non-smooth oscillations still persist although with a smaller amplitude compared with ϕ_1. By further comparing the results from ϕ_5 with the results from ϕ_1^*, it is seen that the amplitude of force oscillation produced by ϕ_1^* is much smaller than ϕ_5, although the supporting width of ϕ_1^* is smaller than ϕ_5. From these comparisons, we can see that the mechanism in the present smoothing technique is not simply an increase of the support width (or the inclusion of more grid points for interpolation).

3.2.3. Vortex-induced vibration of a cylinder

To demonstrate the validity of present approach in simulating fluid-structure interaction problem, flow past an elastically mounted cylinder is simulated. A cylinder is allowed to oscillate only in the crosswise direction and is modeled as a mass-damper-spring system. The governing equation for the motion in the crosswise direction is

$$\ddot{y} + 2\xi\left(\frac{2\pi}{U_{red}}\right)\dot{y} + \left(\frac{2\pi}{U_{red}}\right)^2 y = \frac{2}{\pi n} C_L, \tag{52}$$

where ξ is the damping ratio; $U_{red} = U_\infty / f_N d$ is the reduced velocity and n is the mass ratio which is defined as

$$n = \frac{m}{m_f} = \frac{m}{\pi \rho_f (d^2/4)}, \tag{53}$$

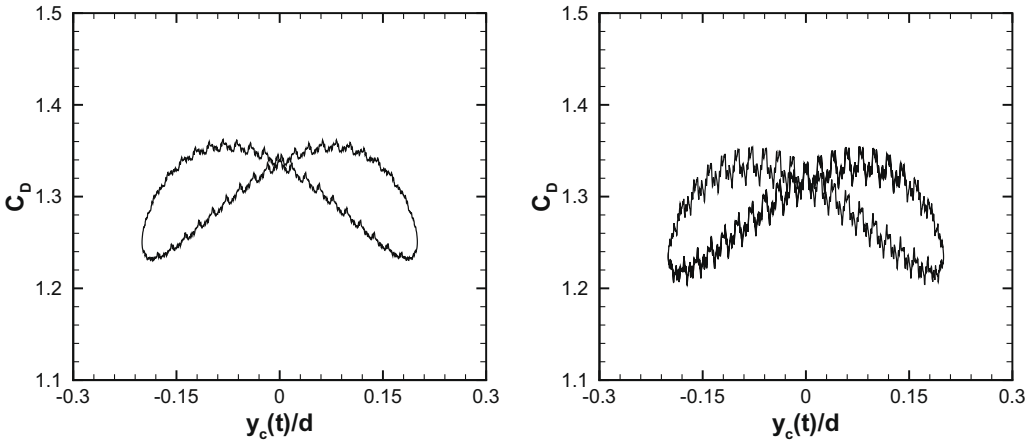

Fig. 9. Time–periodic variation of the drag coefficient for flow past an oscillating cylinder at $Re_d = 185$. Left: ϕ_5; right: ϕ_1.

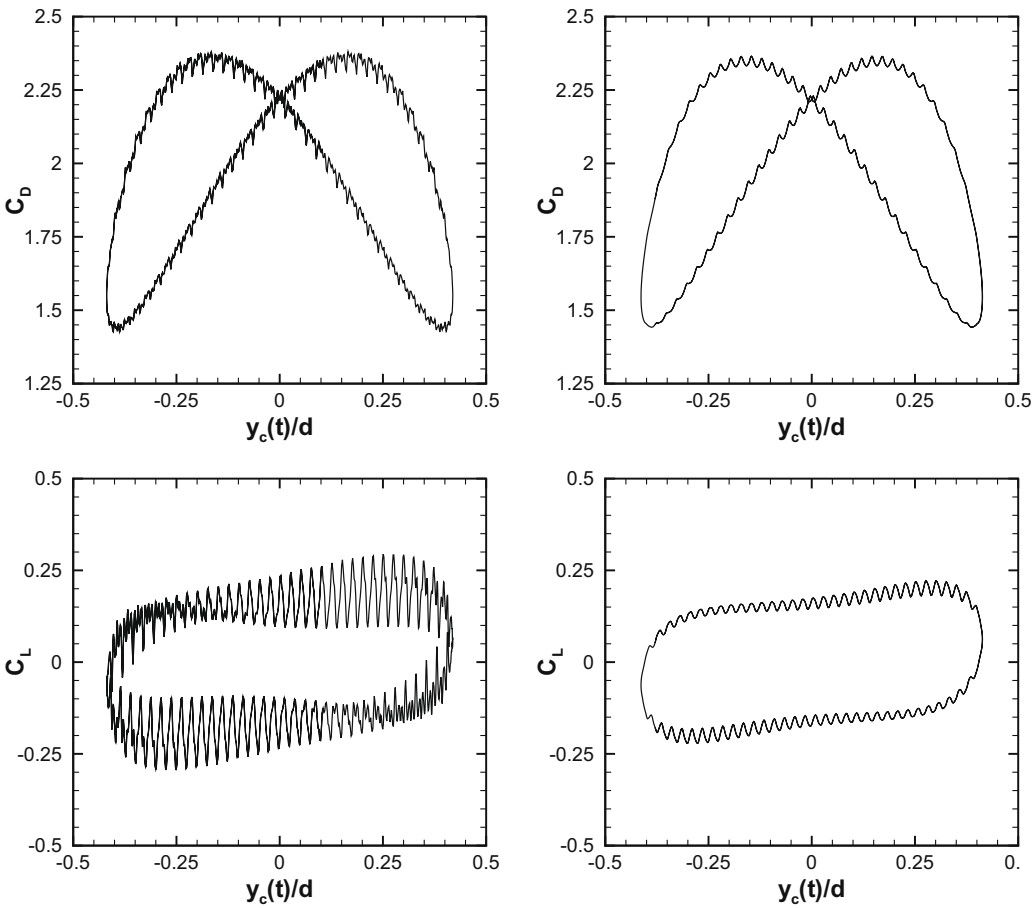

Fig. 10. Time–periodic variation of the drag and lift coefficients for flow past an elastically mounted cylinder at $Re_d = 200$ with $U_{red} = 5.1$. Left column: ϕ_1; right column: ϕ_1^*.

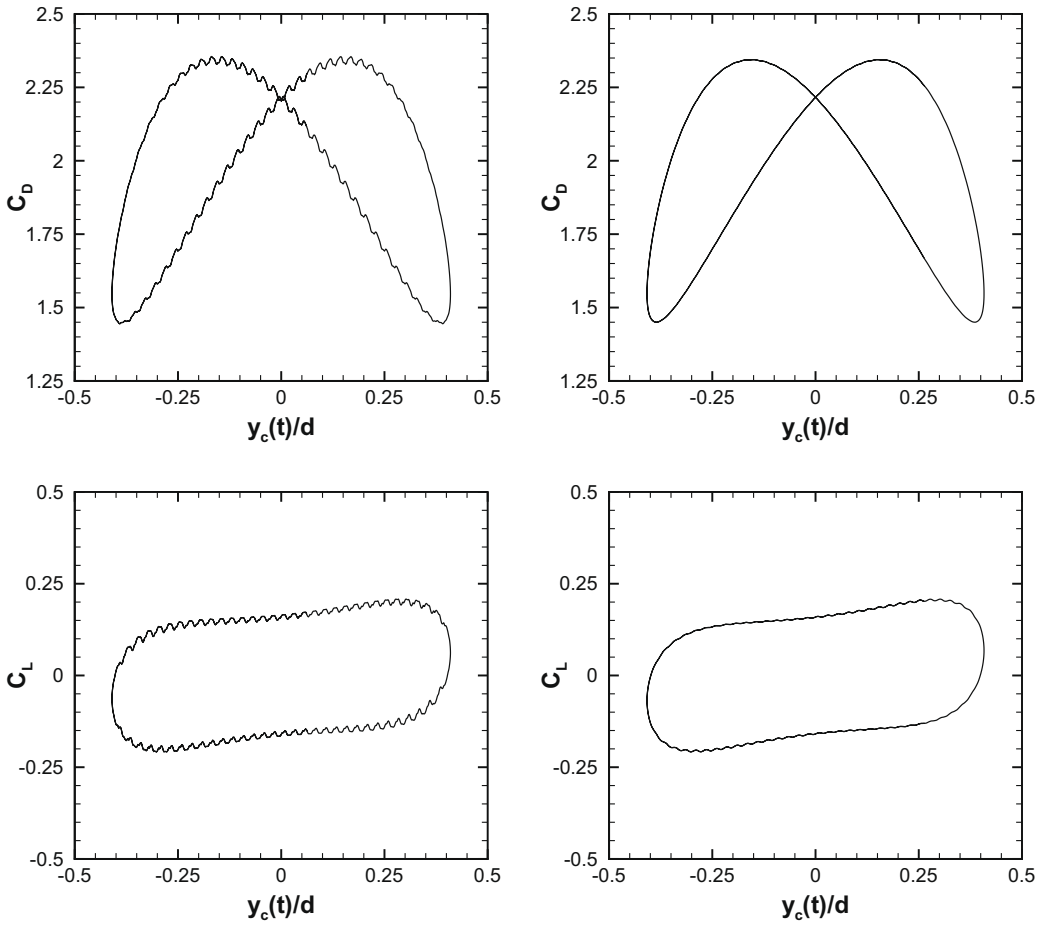

Fig. 11. Time–periodic variation of the drag and lift coefficients without inertial terms for flow past an elastically mounted cylinder at $Re_d = 200$ with $U_{red} = 5.1$. Left column: ϕ_3; right column: ϕ_3^*.

with m_f being the mass of the fluid displaced by the cylinder and ρ_f the fluid density. In the present study, Eq. (52) is discretized in time using the third-order Runge–Kutta method.

As in [25,26], the values of the mass ratio and the damping ratio are 10 and 0.01, respectively. The reduced velocity is 5.1. The Reynolds number based on inlet velocity and cylinder's diameter is 200. The computational domain is $40d \times 20d$. The number of grid points is 326×321 in the streamwise and crosswise directions, respectively. The resolution near the cylinder is $0.025d \times 0.025d$. The time step is 4×10^{-3}. The functions $\phi_1,\ \phi_1^*,\ \phi_3$ and ϕ_3^* are used.

Fig. 10 shows the drag and lift coefficients calculated from ϕ_1 and ϕ_1^*. For the results from ϕ_1, large non-physical oscillations are found in both the drag and the lift coefficients. Especially for the lift coefficient, the oscillations are of the same order of amplitude as the lift itself. Using the smoothed 2-point hat function ϕ_1^*, the amplitude of oscillations in the drag and the lift coefficient is notably reduced. Fig. 11 shows the drag and lift coefficients calculated from ϕ_3 and ϕ_3^*. The oscillations in the drag and lift coefficients obtained from ϕ_3 are of smaller amplitude comparing with those from ϕ_1. But clearly non-physical oscillations still exist. The smoothed discrete delta function ϕ_3^* effectively suppresses the non-physical force oscillations in both the drag and the lift predictions. In this numerical example, the capability of the present approach in suppressing the non-physical force oscillations is fully demonstrated for fluid-structure interactions.

4. Conclusions

The spatial interpolation using a discrete delta function may introduce non-physical oscillations when some IB methods are used to simulate moving body problems. This is mainly due to the fact that the derivatives of the regular discrete delta functions do not satisfy certain moment conditions. Compared with the regular functions, the smoothed discrete delta functions have one higher derivative. For the commonly used discrete delta functions, such as $\phi_2,\ \phi_3$ and ϕ_4, not only the smoothed discrete delta functions satisfy the zeroth and the first moment conditions which guarantee a second-order accuracy in interpolation, but also their first derivatives satisfy more discrete moment conditions. Those two properties effectively reduce non-physical oscillations and improve the force prediction.

An example of a one-dimensional heat equation with a moving singular force is used to evaluate the performance of the smoothed discrete delta functions. The spurious force oscillations are greatly reduced in both the explicit and the implicit forcing methods. A benchmark problem of a flow past an oscillating cylinder is further used to test the proposed approach. It is found that the non-physical force oscillations are effectively suppressed by using the smoothed discrete delta functions. Finally, this method is used to simulate a more complex problem – vortex-induced vibration where fluid-structure interaction is involved. The results obtained in this simulation are in good agreement with those documented in literature.

It should be noted that the proposed approach provides an effective way in minimizing the non-physical grid effect (oscillations) of the immersed boundary method when dealing with moving boundary problems. The implementation of the present approach is simple since the expressions of the smoothed discrete delta functions are given in analytical forms. A wider support of the smoothed discrete delta function does result in more operations in the interpolation and the spreading, but the increase in computational cost is rather small for the present 2-D examples. Additionally, this approach is not equivalent to a post-processing procedure of smoothing the force. An accurate force prediction is crucial to the problems in which the body dynamics and the fluid flows are coupled (e.g. vortex-induced vibration case in Section 3.2.3, self-propelled bio-locomotion, etc.). Under such circumstances, a post-processing of the force prediction is not practical and the proposed method is highly recommended.

Acknowledgments

This work was supported by Chinese Academy of Sciences under the Innovative Project "Multi-scale modeling and simulation in complex systems" (KJCX-SW-L08), "Mathematical modeling of complex system" (KJCX3-SYW-S01), National Basic Research Program of China (973 Program) under Project No. 2007CB814800, and National Natural Science Foundation of China under Project Nos. 10325211, 10628206, 10732090 and 10872201. Zhilin Li would like to acknowledge the hospitality received at LNM during his visit where he accomplished this work.

References

[1] C.S. Peskin, Flow patterns around heart valves: a numerical method, J. Comput. Phys. 10 (1972) 252–271.
[2] C.S. Peskin, The immersed boundary method, Acta Numer. 11 (2002) 479–517.
[3] R. Mittal, G. Iaccarino, Immersed boundary methods, Annu. Rev. Fluid Mech. 37 (2005) 239–261.
[4] D. Goldstein, R. Handler, L. Sirovich, Modeling a no-slip flow boundary with an external force field, J. Comput. Phys. 105 (1993) 354–366.
[5] M.-C. Lai, C.S. Peskin, An immersed boundary method with formal second-order accuracy and reduced numerical viscosity, J. Comput. Phys. 160 (2000) 705–719.
[6] J. Mohd-Yusof, Combined immersed-boundary/B-spline methods for simulations of flow in complex geometries, CTR Annual Research Briefs, NASA Ames/Stanford University, 1997, pp. 317–327.
[7] E.A. Fadlun, R. Verzicco, P. Orlandi, J. Mohd-Yusof, Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations, J. Comput. Phys. 161 (2000) 35–60.
[8] J. Kim, D. Kim, H. Choi, An immersed-boundary finite-volume method for simulations of flow in complex geometries, J. Comput. Phys. 171 (2001) 132–150.
[9] Y.-H. Tseng, J.H. Ferziger, A ghost-cell immersed boundary method for flow in complex geometry, J. Comput. Phys. 192 (2000) 593–623.
[10] E. Balaras, Modeling complex boundaries using an external force field on fixed Cartesian grids in Large-eddy simulations, Comput. Fluids 33 (2004) 375–404.
[11] J. Yang, E. Balaras, An embedded-boundary formulation for large-eddy simulation of turbulent flows interacting with moving boundaries, J. Comput. Phys. 215 (2006) 12–40.
[12] M. Uhlmann, An immersed boundary method with direct forcing for the simulation of particulate flows, J. Comput. Phys. 209 (2005) 448–476.
[13] S.-W. Su, M.-C. Lai, C.-A. Lin, An immersed boundary technique for simulating complex flows with rigid boundary, Comput. Fluids 36 (2007) 313–324.
[14] K. Taira, T. Colonius, The immersed boundary method: a projection approach, J. Comput. Phys. 225 (2007) 2118–2137.
[15] M. Uhlmann, First experiments with the simulation of particulate flows, Technical Report No. 1020, CIEMAT, Madrid, Spain, 2003, ISSN: 1135-9420.
[16] D. Kim, H. Choi, Immersed boundary method for flow around an arbitrarily moving body, J. Comput. Phys. 212 (2006) 662–680.
[17] A.-K. Tornberg, B. Engquist, Numerical approximations of singular source terms in differential equations, J. Comput. Phys. 200 (2004) 462–488.
[18] B. Engquist, A.-K. Tornberg, R. Tsai, Discretization of Dirac delta functions in level set methods, J. Comput. Phys. 207 (2005) 28–51.
[19] P. Smereka, The numerical approximation of a delta function with application to level set methods, J. Comput. Phys. 211 (2006) 77–90.
[20] R.J. LeVeque, Z. Li, The immersed interface method for elliptic equations with discontinuous coefficients and singular sources, SIAM J. Numer. Anal. 31 (1994) 1019–1044.
[21] A.M. Roma, C.S. Peskin, M.J. Berger, An adaptive version of the immersed boundary method, J. Comput. Phys. 153 (1999) 509–534.
[22] R.P. Beyer, R.J. LeVeque, Analysis of a one-dimensional model for the immersed boundary method, SIAM J. Numer. Anal. 29 (1992) 332–364.
[23] C. Liu, X. Zheng, C. Sung, Preconditioned multigrid methods for unsteady incompressible flows, J. Comput. Phys. 139 (1998) 35–57.
[24] E. Guilmineau, P. Queutey, A numerical simulation of vortex shedding from an oscillating circular cylinder, J. Fluids Struct. 16 (2002) 773–794.
[25] J.S. Leontini, M.C. Thompson, K. Hourigan, The beginning of branching behaviour of vortex-induced vibration during two-dimensional flow, J. Fluids Struct. 22 (2006) 857–864.
[26] J. Yang, S. Preidikman, E. Balaras, A strongly coupled embedded-boundary method for fluid-structure interactions of elastically mounted rigid bodies, J. Fluids Struct. 24 (2008) 167–182.