SIAM J. SCI. COMPUT.
Vol. 27, No. 6, pp. 1948–1980

SYSTEMATIC DERIVATION OF JUMP CONDITIONS FOR THE IMMERSED INTERFACE METHOD IN THREE-DIMENSIONAL FLOW SIMULATION*

SHENG XU† AND Z. JANE WANG†

Abstract. In this paper, we systematically derive jump conditions for the immersed interface method [*SIAM J. Numer. Anal.*, 31 (1994), pp. 1019–1044; *SIAM J. Sci. Comput.*, 18 (1997), pp. 709–735] to simulate three-dimensional incompressible viscous flows subject to moving surfaces. The surfaces are represented as singular forces in the Navier–Stokes equations, which give rise to discontinuities of flow quantities. The principal jump conditions across a closed surface of the velocity, the pressure, and their normal derivatives have been derived by Lai and Li [*Appl. Math. Lett.*, 14 (2001), pp. 149–154]. In this paper, we first extend their derivation to generalized surface parametrization. Starting from the principal jump conditions, we then derive the jump conditions of all first-, second-, and third-order spatial derivatives of the velocity and the pressure. We also derive the jump conditions of first- and second-order temporal derivatives of the velocity. Using these jump conditions, the immersed interface method is applicable to the simulation of three-dimensional incompressible viscous flows subject to moving surfaces, where near the surfaces the first- and second-order spatial derivatives of the velocity and the pressure can be discretized with, respectively, third- and second-order accuracy, and the first-order temporal derivatives of the velocity can be discretized with second-order accuracy.

Key words. immersed interface method, immersed boundary method, Cartesian grid methods, jump conditions, three-dimensional Navier–Stokes equations, singular force

AMS subject classifications. 76D05, 76M20, 65M06

DOI. 10.1137/040604960

1. Introduction.

Blood flow in the heart [23, 24, 25], aquatic animal locomotion [10, 15], bird and insect flight [31, 32, 3, 8, 9, 13], and flow passing a compliant wall [35, 6, 14, 4] are examples of fluid dynamics problems where it is essential to understand the coupling between moving surfaces and fluids. A main difficulty in direct numerical simulation of these problems is to accurately and efficiently resolve the moving surfaces and their effects on the fluids.

Cartesian grid methods, for example [28, 5, 30, 22, 21], avoid mesh regeneration and allow for fast flow solvers, and thus have the advantage of simplicity and efficiency for this type of problem. The immersed boundary method is a robust Cartesian grid method. It was originally proposed by Peskin [23, 24] and later further developed in [29, 26, 27, 16]. In this method, the moving surface of an immersed object is parametrized by a set of Lagrangian points comoving with a fluid. The relative positions of the Lagrangian points determine a singular force distribution on the surface based on the solid model of the object. The communication between the surface and the fluid is achieved through the spreading of the singular force and the interpolation of the surface velocity using discrete Dirac δ functions. The method has been applied to a wide variety of problems [10, 11, 25, 1, 37].

The initial implementations of the immersed boundary method were only first-order accurate in space due to the use of grid-dependent discrete Dirac δ functions.

*Received by the editors March 9, 2004; accepted for publication (in revised form) April 28, 2005; published electronically February 3, 2006.
http://www.siam.org/journals/sisc/27-6/60496.html
†Department of Theoretical and Applied Mechanics, Cornell University, Ithaca, NY 14853 (sx12@cornell.edu, jane.wang@cornell.edu).

Beyer and LeVeque [2] examined the accuracy of the method for the one-dimensional diffusion equation and found that additional terms for the discrete approximation of the Dirac δ function are sometimes necessary in order to achieve second-order accuracy, but it is unclear how to maintain the second-order accuracy by incorporating additional terms in fluid dynamics problems of higher dimensions. Although a formally second-order immersed boundary method was proposed [16], it is second-order accurate only if the Dirac δ function is replaced by a grid-independent smooth function; in practice, it is still first-order accurate. Realizing that only the divergence-free portion of the singular force contributes to the temporal evolution of the velocity, and that the projection of discrete Dirac δ functions onto a divergence-free space may be computed analytically, Cortez and Minion [7] devised the blob projection immersed boundary method, which displays the formally fourth-order convergence rate of their background flow solver. However, the analytical form of the projection depends on the velocity boundary conditions imposed on a computational domain. Thus the method may be limited to particular boundary conditions in simple geometries. It is also unclear how accurately the pressure can be recovered.

Motivated by the goal of eventually obtaining second-order accuracy in Peskin's immersed boundary method, LeVeque and Li [18, 19] have developed the immersed interface method (IIM). The IIM was originally proposed for elliptic equations [18] and Stokes equations [19]. Later, it was extended to one-dimensional nonlinear parabolic equations [33], Poisson equations with Neumann boundary conditions [34, 12], and two-dimensional incompressible Navier–Stokes equations [20]. The key idea of the IIM, which is also its main difference from the immersed boundary method, is to incorporate the known jump conditions of a solution and its derivatives into finite difference schemes in the neighborhood of the discontinuities arising from the Dirac δ function. For fluid dynamics problems with moving surfaces, the coupling between the moving surfaces and the fluids is now translated into the incorporation of the jump conditions.

If necessary jump conditions are known, the IIM can achieve second-order or even higher-order accuracy. The two-dimensional Navier–Stokes simulations by Li and Lai [20] using the IIM have indicated fully second-order accuracy for the velocity and nearly second-order accuracy for the pressure in the infinity norm. Solutions computed by the IIM are sharper across surfaces than those computed by the immersed boundary method. Furthermore, the IIM shows better conservation of the mass enclosed by a no-penetration surface.

Like any other method, the IIM has its limitations. For instance, the current IIM applies only to flows with closed smooth surfaces, as seen in its presentation later in this paper. Both the IIM and the immersed boundary method also inherit the shortcomings of fixed Cartesian grid methods. For example, thin boundary layers developed along a moving boundary and fine geometric details can be adequately resolved only if the uniform computational mesh is fine enough. It should also be noted that, for many bio-fluid dynamics problems, the computation of the singular force distribution is a modeling process, and improvement in the accuracy of the IIM or the immersed boundary method cannot eliminate modeling errors.

The applicability of the IIM depends on whether the necessary jump conditions are all known. The principal jump conditions across a closed surface of the velocity, the pressure, and their normal derivatives have been derived by Lai and Li [17] for three-dimensional incompressible viscous flows. The main contribution of our paper is to derive for the IIM the necessary jump conditions of flow variables and their derivatives to achieve a given-order discretization accuracy in three-dimensional flow simulation. Other contributions include generalized Taylor expansion, which is the basis for devising finite difference schemes for the IIM; a generalized Gauss theorem, which serves as the basic tool in the derivation of the principal jump conditions; the principal jump conditions in generalized surface parametrization, which bring the flexibility to parametrize a singular surface in practical applications; and the jump conditions of temporal derivatives, which are required to achieve first-order or higher-order temporal discretization accuracy.

The content of the paper is organized as follows. In section 2, the governing equations are described; they are the starting point for the derivation of the principal jump conditions. In section 3, the principal jump conditions are derived. In section 4, finite difference schemes with jump conditions incorporated are presented. In sections 5 and 6, the required spatial and temporal jump conditions are derived. A simple example is also provided in section 6 to address the proper discretization of temporal derivatives. In section 7, the possibility of improving the IIM to arbitrarily high-order discretization accuracy is discussed.

Since the original submission of this paper, we have implemented and tested the IIM in two-dimensional flow simulations with jump conditions obtained from our theoretical derivation below. Please refer to Xu and Wang [36] for the full numerical implementation and the test results. The test results serve in part to verify our derivation in the current paper. We have also progressed on the development of a three-dimensional code. We hope to provide the three-dimensional results soon.

2. Governing equations. Incompressible Navier–Stokes equations subject to singular force are

$$\rho\left(\frac{\partial u^i}{\partial t}+u^j\frac{\partial u^i}{\partial x^j}\right)=-\frac{\partial p}{\partial x^i}+\mu\frac{\partial^2 u^i}{\partial x^j\partial x^j}+F^i, \tag{2.1}$$

$$\frac{\partial u^i}{\partial x^i}=0, \tag{2.2}$$

where $x^i(i=1,2,3)$ is in Cartesian coordinates, t is time, ρ is fluid density, u^i is velocity, p is pressure, μ is dynamic fluid viscosity, and F^i is the singular force. Taking the divergence of momentum equation (2.1) and applying continuity condition (2.2) gives the Poisson equation for pressure p as

$$\frac{\partial^2 p}{\partial x^i\partial x^i}=\frac{\partial F^i}{\partial x^i}-\frac{\partial}{\partial x^i}\left(\rho\frac{\partial u^i}{\partial t}+\rho u^j\frac{\partial u^i}{\partial x^j}\right). \tag{2.3}$$

We consider the situation that the singular force is applied on the closed surface of an immersed object, and we call the surface singular surface $\mathcal{S}$. Referring to Figure 2.1, singular force F^i is given by

$$F^i=\int_{\mathcal{S}} f^i(\alpha^1,\alpha^2,t)\delta(\mathbf{x}-\mathbf{X}(\alpha^1,\alpha^2,t))\mathrm{d}\alpha^1\mathrm{d}\alpha^2, \tag{2.4}$$

in which $\mathbf{x}:=(x^1,x^2,x^3)$ is the Cartesian coordinates, $\mathbf{X}(\alpha^1,\alpha^2,t):=(X^1,X^2,X^3)$ is the coordinates of the singular surface, $\delta(\mathbf{x}-\mathbf{X}(\alpha^1,\alpha^2,t))$ is the three-dimensional Dirac δ function, $f^i(\alpha^1,\alpha^2,t)$ is force density, and α^1 and α^2 are two Lagrangian parameters which parametrize the singular surface at a reference time. We assume

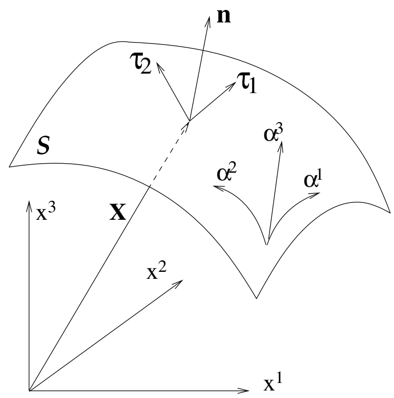

FIG. 2.1. *Cartesian coordinates $x^i (i = 1, 2, 3)$ and curvilinear coordinates $\alpha^i (i = 1, 2, 3)$. α^1 and α^2 are two Lagrangian parameters which parametrize the singular surface $\mathcal{S}$ at a reference time. $\mathbf{X}$ is the coordinates of the singular surface. $\boldsymbol{\tau_1}$, $\boldsymbol{\tau_2}$, and $\mathbf{n}$ are respectively the two tangential vectors and the normal vector at $\mathbf{X}$.*

that $f^i(\alpha^1, \alpha^2, t)$ is a smooth function of α^1, α^2, and t. Mathematically, F^i can be regarded as a distribution function with the property

$$\int_\Omega F^i \mathrm{d}\Omega = \int_{\Omega \cap \mathcal{S}} f^i(\alpha^1, \alpha^2, t) \mathrm{d}(\Omega \cap \mathcal{S}),$$

where Ω can be a volume or a surface.

3. Principal jump conditions.

We now derive the principal jump conditions of the velocity, u^i, the pressure, p, and their normal derivatives, $\frac{\partial u^i}{\partial \mathbf{n}}$ and $\frac{\partial p}{\partial \mathbf{n}}$, across the singular surface, $\mathcal{S}$.

We assume that the velocity, the pressure, and their derivatives are piecewise smooth with discontinuities only at the singular surface. The singular surface, $\mathcal{S}$: $\mathbf{X} = (X^1, X^2, X^3)$, is assumed geometrically regular and orientable. For every point on the surface, the parametrization by α^1 and α^2 generates a rank-two matrix

$$\begin{pmatrix} \dfrac{\partial X^1}{\partial \alpha^1} & \dfrac{\partial X^2}{\partial \alpha^1} & \dfrac{\partial X^3}{\partial \alpha^1} \\ \dfrac{\partial X^1}{\partial \alpha^2} & \dfrac{\partial X^2}{\partial \alpha^2} & \dfrac{\partial X^3}{\partial \alpha^2} \end{pmatrix}.$$

The row vectors in the above matrix are two independent tangential vectors at the point (see Figure 2.1)

$$\boldsymbol{\tau_1} := (\tau_1^1, \tau_1^2, \tau_1^3) = \left(\frac{\partial X^1}{\partial \alpha^1}, \frac{\partial X^2}{\partial \alpha^1}, \frac{\partial X^3}{\partial \alpha^1} \right),$$
$$\boldsymbol{\tau_2} := (\tau_2^1, \tau_2^2, \tau_2^3) = \left(\frac{\partial X^1}{\partial \alpha^2}, \frac{\partial X^2}{\partial \alpha^2}, \frac{\partial X^3}{\partial \alpha^2} \right).$$

A unit normal vector can be expressed by

$$\mathbf{n} := (n^1, n^2, n^3) = \frac{\boldsymbol{\tau_1} \times \boldsymbol{\tau_2}}{\|\boldsymbol{\tau_1} \times \boldsymbol{\tau_2}\|} = \frac{\boldsymbol{\tau_1} \times \boldsymbol{\tau_2}}{\mathcal{J}},$$

where $\|\cdot\|$ denotes the length of a vector and $\mathcal{J} := \|\boldsymbol{\tau_1} \times \boldsymbol{\tau_2}\|$; see Figure 2.1. In Cartesian coordinates, $\mathbf{n} = (n^1, n^2, n^3) = (n_1, n_2, n_3)$, where $n^i (i = 1, 2, 3)$ is a contravariant component and $n_i (i = 1, 2, 3)$ a covariant component.

PROPOSITION 3.1. *For a flow defined by* (2.1) *and* (2.2), *the velocity, u^i, is finite and continuous at the immersed singular surface, $\mathcal{S}$; i.e.,*

$$[u^i] = 0, \tag{3.1}$$

where $[\cdot]$ denotes a jump as $[\cdot] := (\cdot)_{\mathbf{X}^+} - (\cdot)_{\mathbf{X}^-}$, with $\mathbf{X}^+$ representing the point at the side of $\mathcal{S}$ in the direction of normal $\mathbf{n}$ and $\mathbf{X}^-$ the point at the other side.

Continuity equation (2.2) assures only the continuity of the normal velocity component across the singular surface. In our applications, the singular surface is a physical boundary immersed in an incompressible viscous fluid. No-slip and no-penetration conditions on this physical boundary require the velocity to be continuous across the singular surface and the singular surface to move with the local flow velocity, as expressed mathematically by Proposition 3.1.

COROLLARY 3.2.

$$\left[\frac{\partial u^i}{\partial t} + u^j \frac{\partial u^i}{\partial x^j}\right] = \left[\frac{\partial u^i}{\partial t}\right] + u^j \left[\frac{\partial u^i}{\partial x^j}\right] = 0. \tag{3.2}$$

Proof. A jump condition is a function of the time t and the surface coordinates $\mathbf{X}(\alpha^1, \alpha^2, t)$, i.e., $[\cdot] = [\cdot](\mathbf{X}, t)$. Differentiating (3.1) with respect to time t gives

$$\begin{aligned}
\frac{\mathrm{d}[u^i](\mathbf{X},t)}{\mathrm{d}t} &= \frac{\mathrm{d}u^i(\mathbf{X}^+,t)}{\mathrm{d}t} - \frac{\mathrm{d}u^i(\mathbf{X}^-,t)}{\mathrm{d}t} \\
&= \frac{\partial u^i(\mathbf{X}^+,t)}{\partial t} + \frac{\partial u^i(\mathbf{X}^+,t)}{\partial x^j}\frac{\mathrm{d}X^{j+}}{\mathrm{d}t} \\
&\quad - \left(\frac{\partial u^i(\mathbf{X}^-,t)}{\partial t} + \frac{\partial u^i(\mathbf{X}^-,t)}{\partial x^j}\frac{\mathrm{d}X^{j-}}{\mathrm{d}t}\right) \\
&= \left[\frac{\partial u^i}{\partial t} + \frac{\partial u^i}{\partial x^j}\frac{\mathrm{d}X^j}{\mathrm{d}t}\right] = 0.
\end{aligned}$$

From Proposition 3.1, we have $\frac{\mathrm{d}X^{j+}}{\mathrm{d}t} = \frac{\mathrm{d}X^{j-}}{\mathrm{d}t} = u^j$. Thus, the result follows. □

The notion of a jump condition, $[\cdot]$, commutes with a differentiation, and we can therefore write the function form of a jump condition as $[\cdot(\mathbf{X}, t)]$ to carry out the differentiation hereafter.

To derive the other principal jump conditions, we need to generalize Gauss's theorem. Gauss's theorem in the usual form reads

$$\int_{\mathcal{V}} \frac{\partial G^i}{\partial x^i} \mathrm{d}\mathcal{V} = \oint_{\mathcal{A}} G^i N_i \mathrm{d}\mathcal{A},$$

where $\mathcal{A}$ is a regular and positively oriented closed surface, $\mathcal{V}$ is the region enclosed by $\mathcal{A}$, and N_i is a normal to $\mathcal{A}$. It is required that the function G^i and its divergence $\frac{\partial G^i}{\partial x^i}$ be continuous over $\mathcal{A}$ and $\mathcal{V}$. We generalize Gauss's theorem to relax these continuity restrictions.

THEOREM 3.3 (generalized Gauss theorem). *If G^i and $\frac{\partial G^i}{\partial x^i}$ are continuous over $\mathcal{A}$ and $\mathcal{V}$ except that G^i has finite jumps or a singularity of the Dirac δ function type at the singular surface $\mathcal{S}$ enclosed in $\mathcal{V}$, then*

$$\int_{\mathcal{V}} \frac{\partial G^i}{\partial x^i} \mathrm{d}\mathcal{V} = \oint_{\mathcal{A}} G^i N_i \mathrm{d}\mathcal{A}. \tag{3.3}$$

The proof of Theorem 3.3 is given in Appendix A.

LEMMA 3.4. *For a flow defined by* (2.1) *and* (2.2), *the jump conditions across singular surface* $\mathcal{S}$ *of pressure* p *and normal velocity derivative* $\frac{\partial u^i}{\partial \mathbf{n}}$ *satisfy*

$$[p] = \frac{f^k n_k}{\mathcal{J}}, \tag{3.4}$$

$$\left[\frac{\partial u^i}{\partial \mathbf{n}}\right] = \frac{(f^k n_k) n^i - f^i}{\mu \mathcal{J}}. \tag{3.5}$$

Equation (3.4) states that the normal force on the singular surface is balanced by the difference of the pressure force across the singular surface, and (3.5) states that the tangential force on the singular surface is balanced by the difference of the shear force across the singular surface. Below, we prove Lemma 3.4 through the force balance on a control volume. The same results were obtained by Lai and Li [17] by using the test function method.

Proof. Take an infinitesimal area, $\delta\mathcal{S} = \mathcal{J}\delta\alpha^1\delta\alpha^2$, on singular surface $\mathcal{S}$, which corresponds to an infinitesimal area, $\delta\alpha^1\delta\alpha^2$, in the parameter space. Translate $\delta\mathcal{S}$ in the directions of $\mathbf{n}$ and $-\mathbf{n}$ by $\epsilon/2$, and denote the swept region $\delta\mathcal{V}$. Integrating (2.1) over $\delta\mathcal{V}$ and letting $\epsilon \to 0$ yields

$$\lim_{\epsilon\to 0}\int_{\delta\mathcal{V}} \rho\left(\frac{\partial u^i}{\partial t} + u^j\frac{\partial u^i}{\partial x^j}\right)\mathrm{d}\mathcal{V} = \lim_{\epsilon\to 0}\int_{\delta\mathcal{V}} \frac{\partial}{\partial x^j}\left(-p\delta^{ij} + \mu\frac{\partial u^i}{\partial x^j}\right)\mathrm{d}\mathcal{V} + \lim_{\epsilon\to 0}\int_{\delta\mathcal{V}} F^i \mathrm{d}\mathcal{V}, \tag{3.6}$$

where δ^{ij} is the Kronecker symbol. Applying (2.2) and the Reynolds transport theorem, the left-hand side of (3.6) becomes

$$\lim_{\epsilon\to 0}\int_{\delta\mathcal{V}} \rho\left(\frac{\partial u^i}{\partial t} + u^j\frac{\partial u^i}{\partial x^j}\right)\mathrm{d}\mathcal{V} = \lim_{\epsilon\to 0}\int_{\delta\mathcal{V}} \left(\frac{\partial \rho u^i}{\partial t} + \frac{\partial \rho u^i u^j}{\partial x^j}\right)\mathrm{d}\mathcal{V} = \lim_{\epsilon\to 0}\frac{\mathrm{d}}{\mathrm{d}t}(\rho u^i \delta\mathcal{V}) = 0.$$

Applying our generalized Gauss theorem to the first term on the right-hand side of (3.6) yields

$$\begin{aligned}\lim_{\epsilon\to 0}\int_{\delta\mathcal{V}} \frac{\partial}{\partial x^j}\left(-p\delta^{ij} + \mu\frac{\partial u^i}{\partial x^j}\right)\mathrm{d}\mathcal{V} &= \lim_{\epsilon\to 0}\int_{\delta\mathcal{S}} n_j\left[-p\delta^{ij} + \mu\frac{\partial u^i}{\partial x^j}\right]\mathrm{d}\mathcal{S}\\ &= \left(-[p]n^i + \mu\left[\frac{\partial u^i}{\partial x^k}\right]n^k\right)\delta\mathcal{S}.\end{aligned}$$

The last term in (3.6) is

$$\begin{aligned}\lim_{\epsilon\to 0}\int_{\delta\mathcal{V}} F^i \mathrm{d}\mathcal{V} &= \lim_{\epsilon\to 0}\int_{\mathcal{S}}\int_{\delta\mathcal{V}} f^i(\alpha^1,\alpha^2,t)\delta(\mathbf{x} - \mathbf{X}(\alpha^1,\alpha^2,t))\mathrm{d}\mathcal{V}\mathrm{d}\alpha^1\mathrm{d}\alpha^2\\ &= f^i(\alpha^1,\alpha^2,t)\delta\alpha^1\delta\alpha^2.\end{aligned}$$

Thus,

$$\begin{aligned}\left(-[p]n^i + \mu\left[\frac{\partial u^i}{\partial x^k}\right]n^k\right)\delta\mathcal{S} + f^i(\alpha^1,\alpha^2,t)\delta\alpha^1\delta\alpha^2 &= 0\\ \Rightarrow -[p]n^i + \mu\left[\frac{\partial u^i}{\partial x^k}\right]n^k &= -\frac{f^i(\alpha^1,\alpha^2,t)}{\mathcal{J}}.\end{aligned}$$

Multiplying n_i above and applying the facts that $n^k n_i + \tau^k \tau_i + b^k b_i = \delta_i^k$ (where δ_i^k is the Kronecker symbol, and $\mathbf{n}$, τ, and $\mathbf{b}$ are mutually orthogonal unit vectors), $\frac{\partial u^i}{\partial x^i} = 0$, and $[\frac{\partial u^i}{\partial \tau}] = [\frac{\partial u^i}{\partial \mathbf{b}}] = 0$ (from $[u^i] = 0$), we obtain

$$[p] = \frac{f^i n_i}{\mathcal{J}},$$

$$\mu \left[\frac{\partial u^i}{\partial x^k}\right] n^k = \frac{-f^i}{\mathcal{J}} + [p] n^i \Rightarrow \left[\frac{\partial u^i}{\partial \mathbf{n}}\right] = \frac{(f^k n_k) n^i - f^i}{\mu \mathcal{J}}. \qquad \square$$

Now we use the test function method to derive the jump condition of the normal pressure derivative, $\frac{\partial p}{\partial \mathbf{n}}$, across the singular surface with generalized surface parametrization. To prepare for the derivation, we first introduce a coordinate transformation (see Figure 2.1) between Cartesian coordinates $x^i (i = 1, 2, 3)$ and curvilinear coordinates $\alpha^i (i = 1, 2, 3)$ as

$$x^i = x^i(\alpha^1, \alpha^2, \alpha^3),$$
$$\alpha^i = \alpha^i(x^1, x^2, x^3),$$

where α^3 is a new coordinate with $\mathbf{x}(\alpha^1, \alpha^2, \alpha^3 = 0, t) = \mathbf{X}(\alpha^1, \alpha^2, t)$ corresponding to the singular surface, $\mathcal{S}$, and α^3 is chosen to satisfy

$$\frac{\partial \mathbf{x}(\alpha^1, \alpha^2, \alpha^3 = 0, t)}{\partial \alpha^3} = \mathbf{n}.$$

It can be shown that

$$\nabla \alpha^3 = \mathbf{n},$$

where ∇ is the gradient operator in the Cartesian coordinate system. Let $\tilde{f}^i$ be a contravariant component of the forcing density vector in the curvilinear coordinate system. It is related to f^i through

$$\tilde{f}^i = \frac{\partial \alpha^i}{\partial x^j} f^j,$$

where $\frac{\partial \alpha^i}{\partial x^j}$ satisfies

$$\frac{\partial x^k}{\partial \alpha^i} \frac{\partial \alpha^i}{\partial x^j} = \delta_j^k,$$

in which δ_j^k is the Kronecker symbol and $\frac{\partial x^k}{\partial \alpha^i}$ can be written in a matrix form on the singular surface $\mathcal{S}$ as

$$C_1^T := \begin{pmatrix} \tau_1^1 & \tau_1^2 & \tau_1^3 \\ \tau_2^1 & \tau_2^2 & \tau_2^3 \\ n^1 & n^2 & n^3 \end{pmatrix}^T, \tag{3.7}$$

where superscript T denotes the transposition of a matrix. As the determinant of C_1 is $|C_1| = \mathbf{n} \cdot (\boldsymbol{\tau_1} \times \boldsymbol{\tau_2}) \neq 0$, C_1 is nonsingular.

LEMMA 3.5. *The jump condition of normal pressure derivative* $\frac{\partial p}{\partial \mathbf{n}}$ *across the singular surface is*

$$\left[\frac{\partial p}{\partial \mathbf{n}}\right] = \frac{1}{\mathcal{J}} \left(\frac{\partial \tilde{f}^1}{\partial \alpha^1} + \frac{\partial \tilde{f}^2}{\partial \alpha^2}\right). \tag{3.8}$$

Proof. Take control volume $\mathcal{V}_s$ so that $\mathcal{V}_s$ is a layer with thickness ϵ containing singular surface $\mathcal{S}$. Denote the surface of $\mathcal{V}_s$ by $\mathcal{A}$. Multiplying a smooth test function, $\phi(\mathbf{x})$, by (2.3) and then integrating over $\mathcal{V}_s$ with $\epsilon \to 0$ yields

$$\lim_{\epsilon\to 0}\int_{\mathcal{V}_s}\phi\frac{\partial^2 p}{\partial x^i\partial x^i}\mathrm{d}\mathcal{V} = \lim_{\epsilon\to 0}\int_{\mathcal{V}_s}\phi\frac{\partial F^i}{\partial x^i}\mathrm{d}\mathcal{V} - \lim_{\epsilon\to 0}\int_{\mathcal{V}_s}\phi\frac{\partial}{\partial x^i}\left(\rho\frac{\partial u^i}{\partial t} + \rho u^j\frac{\partial u^i}{\partial x^j}\right)\mathrm{d}\mathcal{V}. \tag{3.9}$$

With Theorem 3.3, the term on the left-hand side of (3.9) can be written as

$$\begin{aligned}\lim_{\epsilon\to 0}\int_{\mathcal{V}_s}\phi\frac{\partial^2 p}{\partial x^i\partial x^i}\mathrm{d}\mathcal{V} &= \lim_{\epsilon\to 0}\int_{\mathcal{V}_s}\left(\frac{\partial}{\partial x^i}\left(\phi\frac{\partial p}{\partial x^i}\right) + p\frac{\partial^2\phi}{\partial x^i\partial x^i} - \frac{\partial}{\partial x^i}\left(p\frac{\partial\phi}{\partial x^i}\right)\right)\mathrm{d}\mathcal{V}\\ &= \int_{\mathcal{S}}\left(\phi\left[\frac{\partial p}{\partial x^i}\right]n^i - [p]\frac{\partial\phi}{\partial x^i}n^i\right)\mathrm{d}\mathcal{S}\\ &= \int_{\mathcal{S}}\left(\phi\left[\frac{\partial p}{\partial\mathbf{n}}\right] - [p]\frac{\partial\phi}{\partial\mathbf{n}}\right)\mathcal{J}\mathrm{d}\alpha^1\mathrm{d}\alpha^2.\end{aligned}$$

Due to Theorem 3.3, the first term on the right-hand side of (3.9) becomes

$$\begin{aligned}\lim_{\epsilon\to 0}\int_{\mathcal{V}_s}\phi\frac{\partial F^i}{\partial x^i}\mathrm{d}\mathcal{V} &= \lim_{\epsilon\to 0}\int_{\mathcal{V}_s}\frac{\partial(\phi F^i)}{\partial x^i}\mathrm{d}\mathcal{V} - \lim_{\epsilon\to 0}\int_{\mathcal{V}_s}F^i\frac{\partial\phi}{\partial x^i}\mathrm{d}\mathcal{V}\\ &= -\lim_{\epsilon\to 0}\int_{\mathcal{V}_s}F^i\frac{\partial\phi}{\partial x^i}\mathrm{d}\mathcal{V} = -\int_{\mathcal{S}}f^i(\alpha^1,\alpha^2,t)\frac{\partial\phi}{\partial x^i}\mathrm{d}\alpha^1\mathrm{d}\alpha^2\\ &= -\int_{\mathcal{S}}\left(\tilde{f}^1\frac{\partial\phi}{\partial\alpha^1} + \tilde{f}^2\frac{\partial\phi}{\partial\alpha^2} + \mathbf{f}\cdot\mathbf{n}\frac{\partial\phi}{\partial\mathbf{n}}\right)\mathrm{d}\alpha^1\mathrm{d}\alpha^2\\ &= \int_{\mathcal{S}}\phi\left(\frac{\partial\tilde{f}^1}{\partial\alpha^1} + \frac{\partial\tilde{f}^2}{\partial\alpha^2}\right)\mathrm{d}\alpha^1\mathrm{d}\alpha^2 - \int_{\mathcal{S}}\mathbf{f}\cdot\mathbf{n}\frac{\partial\phi}{\partial\mathbf{n}}\mathrm{d}\alpha^1\mathrm{d}\alpha^2.\end{aligned}$$

In the last step above, we used the fact that the singular surface is closed. With (3.2), the last term on the right-hand side of (3.9) becomes

$$\begin{aligned}&\lim_{\epsilon\to 0}\int_{\mathcal{V}_s}\phi\frac{\partial}{\partial x^i}\left(\rho\frac{\partial u^i}{\partial t} + \rho u^j\frac{\partial u^i}{\partial x^j}\right)\mathrm{d}\mathcal{V}\\ &= \lim_{\epsilon\to 0}\int_{\mathcal{V}_s}\frac{\partial}{\partial x^i}\left(\phi\rho\frac{\partial u^i}{\partial t} + \phi\rho u^j\frac{\partial u^i}{\partial x^j}\right)\mathrm{d}\mathcal{V} - \lim_{\epsilon\to 0}\int_{\mathcal{V}_s}\frac{\partial\phi}{\partial x^i}\left(\rho\frac{\partial u^i}{\partial t} + \rho u^j\frac{\partial u^i}{\partial x^j}\right)\mathrm{d}\mathcal{V}\\ &= \int_{\mathcal{S}}\phi\rho\left(\left[\frac{\partial u^i}{\partial t}\right] + u^j\left[\frac{\partial u^i}{\partial x^j}\right]\right)n^i\mathrm{d}\mathcal{S} - \lim_{\epsilon\to 0}\frac{\mathrm{d}}{\mathrm{d}t}\int_{\mathcal{V}_s}\left(\frac{\partial\phi}{\partial x^i}\rho u^i\right)\mathrm{d}\mathcal{V} = 0.\end{aligned}$$

Plugging the equalities for the three terms back into (3.9) and applying (3.4) gives

$$\int_{\mathcal{S}}\phi\left(\left[\frac{\partial p}{\partial\mathbf{n}}\right]\mathcal{J} - \frac{\partial\tilde{f}^1}{\partial\alpha^1} - \frac{\partial\tilde{f}^2}{\partial\alpha^2}\right)\mathrm{d}\alpha^1\mathrm{d}\alpha^2 = 0.$$

Because ϕ is arbitrary, we have

$$\left[\frac{\partial p}{\partial\mathbf{n}}\right]\mathcal{J} - \frac{\partial\tilde{f}^1}{\partial\alpha^1} - \frac{\partial\tilde{f}^2}{\partial\alpha^2} = 0,$$

which ends the proof. ∎

We have by now derived all the principal jump conditions. They are

$$[u^i] = 0, \tag{3.10}$$

$$[p] = \frac{f^k n_k}{\mathcal{J}}, \tag{3.11}$$

$$\left[\frac{\partial u^i}{\partial \mathbf{n}}\right] = \frac{(f^k n_k)n^i - f^i}{\mu \mathcal{J}}, \tag{3.12}$$

$$\left[\frac{\partial p}{\partial \mathbf{n}}\right] = \frac{1}{\mathcal{J}}\left(\frac{\partial \tilde{f}^1}{\partial \alpha^1} + \frac{\partial \tilde{f}^2}{\partial \alpha^2}\right). \tag{3.13}$$

4. Finite differencing in the IIM. The fundamental idea of the IIM in a flow simulation is to incorporate jump conditions in finite differencing at discontinuities caused by the singular force. A finite difference scheme has its usual form if its stencil does not cross the singular surface. If its stencil crosses the singular surface, it contains additional terms. The additional terms are composed of the jump conditions and are referred to the jump contribution to the finite difference scheme hereafter. To determine the form of the jump contribution in a finite difference scheme, we follow generalized Taylor expansion for a piecewise smooth function, which is given below as a lemma.

LEMMA 4.1 (generalized Taylor expansion). *Assume function $g(z)$ has discontinuity points of the first kind at $z_1, z_2, \ldots, z_m$ in $(z_0, z_{m+1}), z_0 < z_1 < z_2 < \cdots < z_m < z_{m+1}$, and $g(z) \in \mathcal{C}^\infty(z_0, z_1) \cup (z_1, z_2) \cup \cdots \cup (z_m, z_{m+1})$. $g(z)$ can be either continuous or discontinuous at z_0 and z_{m+1}. Let $[g^{(n)}(z_l)] = g^{(n)}(z_l^+) - g^{(n)}(z_l^-)(n = 1, 2, \ldots; l = 1, 2, \ldots, m)$. Then*

$$g(z_{m+1}^-) = \sum_{n=0}^{\infty} \frac{g^{(n)}(z_0^+)}{n!}(z_{m+1} - z_0)^n + \sum_{l=1}^{m}\sum_{n=0}^{\infty} \frac{[g^{(n)}(z_l)]}{n!}(z_{m+1} - z_l)^n. \tag{4.1}$$

Proof. Taylor expansion for $g^{(n)}(z_l^-)$ about z_{l-1}^+ yields

$$g^{(n)}(z_l^-) = \sum_{\beta=0}^{\infty} \frac{g^{(n+\beta)}(z_{l-1}^+)}{\beta!}(z_l - z_{l-1})^\beta.$$

With the use of the binomial theorem, we thus find that

$$\begin{aligned}\sum_{n=0}^{\infty} \frac{g^{(n)}(z_l^-)}{n!}(z_{m+1} - z_l)^n &= \sum_{n=0}^{\infty}\sum_{\beta=0}^{\infty} \frac{g^{(n+\beta)}(z_{l-1}^+)(z_l - z_{l-1})^\beta}{n!\beta!}(z_{m+1} - z_l)^n \\ &= \sum_{\gamma=0}^{\infty}\sum_{\beta=0}^{\gamma} g^{(\gamma)}(z_{l-1}^+)\frac{(z_l - z_{l-1})^\beta (z_{m+1} - z_l)^{(\gamma-\beta)}}{\beta!(\gamma-\beta)!} \\ &= \sum_{\gamma=0}^{\infty} \frac{g^{(\gamma)}(z_{l-1}^+)}{\gamma!}(z_{m+1} - z_{l-1})^\gamma.\end{aligned}$$

With $g^{(n)}(z_{l-1}^+) = g^{(n)}(z_{l-1}^-) + [g^{(n)}(z_{l-1})]$, thus

$$\begin{aligned}\sum_{n=0}^{\infty} \frac{g^{(n)}(z_l^-)}{n!}(z_{m+1} - z_l)^n = &\sum_{n=0}^{\infty} \frac{g^{(n)}(z_{l-1}^-)}{n!}(z_{m+1} - z_{l-1})^n \\ &+ \sum_{n=0}^{\infty} \frac{[g^{(n)}(z_{l-1})]}{n!}(z_{m+1} - z_{l-1})^n.\end{aligned} \tag{4.2}$$

With $g^{(n)}(z_m^+) = g^{(n)}(z_m^-) + [g^{(n)}(z_m)]$, Taylor expansion for $g(z_{m+1}^-)$ at z_m^+ yields

$$\begin{aligned} g(z_{m+1}^-) &= \sum_{n=0}^{\infty} \frac{g^{(n)}(z_m^+)}{n!}(z_{m+1} - z_m)^n \\ &= \sum_{n=0}^{\infty} \frac{g^{(n)}(z_m^-)}{n!}(z_{m+1} - z_m)^n + \sum_{n=0}^{\infty} \frac{[g^{(n)}(z_m)]}{n!}(z_{m+1} - z_m)^n. \end{aligned}$$

Using recursion (4.2) repeatedly above gives the desired result. □

COROLLARY 4.2. *The Taylor expansion for $g(z_0^+)$ about z_{m+1}^- is*

$$g(z_0^+) = \sum_{n=0}^{\infty} \frac{g^{(n)}(z_{m+1}^-)}{n!}(z_0 - z_{m+1})^n + \sum_{l=1}^{m}\sum_{n=0}^{\infty} \frac{-[g^{(n)}(z_l)]}{n!}(z_0 - z_l)^n. \tag{4.3}$$

Jump conditions at the singular surface enter a finite difference scheme in the neighborhood of the singular surface. The form of the finite difference scheme can be found by applying Lemma 4.1 and Corollary 4.2. Here, we construct central finite difference schemes for first-order and second-order derivatives in the situation where there is a discontinuity point ξ between stencil points x_{i-1} and x_i and a discontinuity point η between stencil points x_i and x_{i+1}. If there are more discontinuity points in the stencil, they can easily be included in a similar manner.

LEMMA 4.3 (generalized central finite differences). *Let $x_{i+1} - x_i = x_i - x_{i-1} = h > 0$ and $x_{i-1} < \xi < x_i \le \eta < x_{i+1}$. Suppose that $u(x)$ is infinitely smooth except at discontinuity points of the first kind ξ and η. Further, $u(x)$ can be either continuous or discontinuous at x_{i+1} and x_{i-1}. Then*

$$\begin{aligned} \frac{\mathrm{d}u(x_i^-)}{\mathrm{d}x} &= \frac{u(x_{i+1}^-) - u(x_{i-1}^+)}{2h} \\ &\quad + \frac{1}{2h}\left(\sum_{n=0}^{2} \frac{-[u^{(n)}(\xi)]}{n!}(x_{i-1}-\xi)^n - \sum_{n=0}^{2} \frac{[u^{(n)}(\eta)]}{n!}(x_{i+1} - \eta)^n\right) + O(h^2), \end{aligned} \tag{4.4}$$

$$\begin{aligned} \frac{\mathrm{d}^2u(x_i^-)}{\mathrm{d}x^2} &= \frac{u(x_{i+1}^-) - 2u(x_i) + u(x_{i-1}^+)}{h^2} \\ &\quad - \frac{1}{h^2}\left(\sum_{n=0}^{3} \frac{-[u^{(n)}(\xi)]}{n!}(x_{i-1}-\xi)^n + \sum_{n=0}^{3} \frac{[u^{(n)}(\eta)]}{n!}(x_{i+1} - \eta)^n\right) + O(h^2). \end{aligned} \tag{4.5}$$

Other finite difference schemes with different orders can also be constructed based on Lemma 4.1 and Corollary 4.2, but the number of jump conditions sets an upper limit on the order of accuracy, as stated in the following proposition.

PROPOSITION 4.4. *The highest order of a finite difference scheme for $u^{(n)}(x)$ with a stencil containing a discontinuity point ζ is $m - n + 1$, where m is the maximum number for known jump conditions $[u^{(l)}(\zeta)]$ $(l = 0, 1, 2, \ldots, m)$.*

Navier–Stokes equations (2.1), (2.2) and pressure Poisson equation (2.3) have first- and second-order spatial derivatives. According to Proposition 4.4, to discretize the first-order spatial derivatives with second-order accuracy or the second-order spatial derivatives with first-order accuracy near the singular surface, the jump conditions of the velocity, the pressure, and their first- and second-order spatial derivatives are needed. To discretize the first-order spatial derivatives with third-order accuracy or the second-order spatial derivatives with second-order accuracy near the singular surface, the jump conditions of their third-order derivatives are needed as well. All these spatial jump conditions are derived in section 5.

If the singular surface is moving, there will be jumps in the temporal derivatives of the velocity at a grid point whenever the surface crosses that grid point. Suppose that the singular surface passes the grid point at time $t_1, t_2, \ldots, t_m$ between time t_0 and t_{m+1}; then

$$u^i(t_{m+1}) = \sum_{n=0}^{\infty} \frac{\partial^n u^i(t_0)}{\partial t^n} \frac{(t_{m+1}-t_0)^n}{n!} + \sum_{l=1}^{m} \sum_{n=0}^{\infty} \left[\!\left[\frac{\partial^n u^i(t_l)}{\partial t} \right]\!\right] \frac{(t_{m+1}-t_l)^n}{n!}, \tag{4.6}$$

where $[\![\cdot]\!]$ denotes a jump at time t as $[\![\cdot]\!] := (\cdot)_{t^+} - (\cdot)_{t^-}$. Equation (4.6) follows directly from Lemma (4.1). Thus, to achieve first-order accuracy when discretizing $\frac{\partial u^i}{\partial t}$ in (2.1), we need $[\![u^i]\!]$ and $[\![\frac{\partial u^i}{\partial t}]\!]$; to achieve second-order accuracy, we also need $[\![\frac{\partial^2 u^i}{\partial t^2}]\!]$. All these temporal jump conditions are derived in section 6.1.

It should be noted that the spatial convergence rate of a simulation even in terms of the infinity norm can be of the same order as the numerical scheme away from a singular surface, even though the discretization of some derivatives in the Navier–Stokes equations and the pressure Poisson equation is of lower order accuracy near the singular surface. Examples can be found in Li and Lai [20] and Xu and Wang [36], where, with second-order central finite difference discretization of all spatial derivatives away from a singular surface, a simulation had second-order spatial convergence rates in terms of the infinity norm for both the velocity and the pressure, even though the discretization of the Laplace operator was only first-order accurate near the singular surface.

5. Jump conditions of spatial derivatives. The spatial jump conditions of velocity u^i and pressure p are given in (3.10) and (3.11). In this section we present how to derive the spatial jump conditions of all the first-, second-, and third-order velocity and pressure derivatives:

$$\left[\frac{\partial u^i}{\partial x^j}\right], \quad \left[\frac{\partial p}{\partial x^j}\right]; \quad \left[\frac{\partial^2 u^i}{\partial x^j \partial x^k}\right], \quad \left[\frac{\partial^2 p}{\partial x^j \partial x^k}\right]; \quad \left[\frac{\partial^3 \mathbf{u}}{\partial x^i \partial x^j \partial x^k}\right], \quad \left[\frac{\partial^3 p}{\partial x^i \partial x^j \partial x^k}\right];$$

where $\mathbf{u}$ is used to represent the velocity vector and a superscript (i, j, or k) takes a value $1, 2$, or 3 in three-dimensional simulation.

5.1. Jump conditions $[\frac{\partial u^i}{\partial x^j}]$ and $[\frac{\partial p}{\partial x^j}]$. Differentiate (3.10) with respect to α^m ($m = 1, 2$) to obtain

$$\frac{\partial [u^i]}{\partial \alpha^m} = \left[\frac{\partial u^i}{\partial x^k}\frac{\partial X^k}{\partial \alpha^m}\right] = \frac{\partial X^k}{\partial \alpha^m}\left[\frac{\partial u^i}{\partial x^k}\right] = 0. \tag{5.1}$$

Write (3.12) as

$$\left[\frac{\partial u^i}{\partial \mathbf{n}}\right] = n^j \left[\frac{\partial u^i}{\partial x^j}\right] = \frac{(f^k n_k)n^i - f^i}{\mu \mathcal{J}}. \tag{5.2}$$

Combining (5.1) and (5.2) gives

$$C_1 \begin{bmatrix} \dfrac{\partial u^i}{\partial x^1} \\ \dfrac{\partial u^i}{\partial x^2} \\ \dfrac{\partial u^i}{\partial x^3} \end{bmatrix} = \begin{pmatrix} 0 \\ 0 \\ \dfrac{(f^k n_k)n^i - f^i}{\mu \mathcal{J}} \end{pmatrix}, \tag{5.3}$$

where $\left[\vdots\right]$ denotes a jump for a column vector, and where the nonsingular coefficient matrix C_1 is defined in (3.7). From (5.3), jump conditions of first-order velocity derivatives can be solved.

Similarly, jump conditions of first-order pressure derivatives are found to satisfy

$$C_1 \begin{bmatrix} \dfrac{\partial p}{\partial x^1} \\ \dfrac{\partial p}{\partial x^2} \\ \dfrac{\partial p}{\partial x^3} \end{bmatrix} = \begin{pmatrix} \dfrac{\partial}{\partial \alpha^1}\left(\dfrac{f^i n_i}{\mathcal{J}}\right) \\ \dfrac{\partial}{\partial \alpha^2}\left(\dfrac{f^i n_i}{\mathcal{J}}\right) \\ \dfrac{1}{\mathcal{J}}\left(\dfrac{\partial \tilde{f}^1}{\partial \alpha^1} + \dfrac{\partial \tilde{f}^2}{\partial \alpha^2}\right) \end{pmatrix}, \tag{5.4}$$

from which, jump conditions of first-order pressure derivatives can be solved.

5.2. Jump conditions $\left[\frac{\partial^2 u^i}{\partial x^j \partial x^k}\right]$ and $\left[\frac{\partial^2 p}{\partial x^j \partial x^k}\right]$. Differentiating equation (5.1) with respect to α^n $(n = 1, 2)$, we have

$$\frac{\partial^2 [u^i]}{\partial \alpha^m \partial \alpha^n} = \left[\frac{\partial^2 u^i}{\partial x^j \partial x^k}\frac{\partial X^j}{\partial \alpha^n}\frac{\partial X^k}{\partial \alpha^m} + \frac{\partial u^i}{\partial x^k}\frac{\partial^2 X^k}{\partial \alpha^m \partial \alpha^n}\right] = 0$$

$$\Rightarrow \frac{\partial X^j}{\partial \alpha^n}\frac{\partial X^k}{\partial \alpha^m}\left[\frac{\partial^2 u^i}{\partial x^j \partial x^k}\right] = -\frac{\partial^2 X^k}{\partial \alpha^m \partial \alpha^n}\left[\frac{\partial u^i}{\partial x^k}\right]. \tag{5.5}$$

Differentiating (3.12) with respect to α^m $(m = 1, 2)$, we have

$$\frac{\partial}{\partial \alpha^m}\left[\frac{\partial u^i}{\partial \mathbf{n}}\right] = \left[\frac{\partial^2 u^i}{\partial x^j \partial x^k}\frac{\partial X^k}{\partial \alpha^m}n^j + \frac{\partial u^i}{\partial x^j}\frac{\partial n^j}{\partial \alpha^m}\right] = \frac{\partial}{\partial \alpha^m}\frac{(f^k n_k)n^i - f^i}{\mu \mathcal{J}}$$

$$\Rightarrow n^j \frac{\partial X^k}{\partial \alpha^m}\left[\frac{\partial^2 u^i}{\partial x^j \partial x^k}\right] = \frac{\partial}{\partial \alpha^m}\frac{(f^k n_k)n^i - f^i}{\mu \mathcal{J}} - \frac{\partial n^j}{\partial \alpha^m}\left[\frac{\partial u^i}{\partial x^j}\right]. \tag{5.6}$$

With (3.2), equation (2.1) yields

$$\left[\rho\left(\frac{\partial u^i}{\partial t} + u^j \frac{\partial u^i}{\partial x^j}\right)\right] = -\left[\frac{\partial p}{\partial x^i}\right] + \mu\left[\frac{\partial^2 u^i}{\partial x^j \partial x^j}\right] = 0. \tag{5.7}$$

Combining (5.5), (5.6), and (5.7) gives

$$
(5.8)\qquad C_2 \begin{bmatrix} \dfrac{\partial^2 u^i}{\partial x^1 \partial x^1} \\ \dfrac{\partial^2 u^i}{\partial x^1 \partial x^2} \\ \dfrac{\partial^2 u^i}{\partial x^1 \partial x^3} \\ \dfrac{\partial^2 u^i}{\partial x^2 \partial x^2} \\ \dfrac{\partial^2 u^i}{\partial x^2 \partial x^3} \\ \dfrac{\partial^2 u^i}{\partial x^3 \partial x^3} \end{bmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \dfrac{\partial}{\partial \alpha^1} \dfrac{(f^k n_k) n^i - f^i}{\mu \mathcal{J}} \\ \dfrac{\partial}{\partial \alpha^2} \dfrac{(f^k n_k) n^i - f^i}{\mu \mathcal{J}} \\ \dfrac{1}{\mu} \left[\dfrac{\partial p}{\partial x^i} \right] \end{pmatrix}
- \begin{pmatrix} \dfrac{\partial^2 X^1}{\partial \alpha^1 \partial \alpha^1} & \dfrac{\partial^2 X^2}{\partial \alpha^1 \partial \alpha^1} & \dfrac{\partial^2 X^3}{\partial \alpha^1 \partial \alpha^1} \\ \dfrac{\partial^2 X^1}{\partial \alpha^2 \partial \alpha^2} & \dfrac{\partial^2 X^2}{\partial \alpha^2 \partial \alpha^2} & \dfrac{\partial^2 X^3}{\partial \alpha^2 \partial \alpha^2} \\ \dfrac{\partial^2 X^1}{\partial \alpha^1 \partial \alpha^2} & \dfrac{\partial^2 X^2}{\partial \alpha^1 \partial \alpha^2} & \dfrac{\partial^2 X^3}{\partial \alpha^1 \partial \alpha^2} \\ \dfrac{\partial n^1}{\partial \alpha^1} & \dfrac{\partial n^2}{\partial \alpha^1} & \dfrac{\partial n^3}{\partial \alpha^1} \\ \dfrac{\partial n^1}{\partial \alpha^2} & \dfrac{\partial n^2}{\partial \alpha^2} & \dfrac{\partial n^3}{\partial \alpha^2} \\ 0 & 0 & 0 \end{pmatrix} \begin{bmatrix} \dfrac{\partial u^i}{\partial x^1} \\ \dfrac{\partial u^i}{\partial x^2} \\ \dfrac{\partial u^i}{\partial x^3} \end{bmatrix},
$$

where C_2 is

$$
C_2 = \begin{pmatrix} \tau_1^1 \tau_1^1 & \tau_1^1 \tau_1^2 + \tau_1^2 \tau_1^1 & \tau_1^1 \tau_1^3 + \tau_1^3 \tau_1^1 & \tau_1^2 \tau_1^2 & \tau_1^2 \tau_1^3 + \tau_1^3 \tau_1^2 & \tau_1^3 \tau_1^3 \\ \tau_2^1 \tau_2^1 & \tau_2^1 \tau_2^2 + \tau_2^2 \tau_2^1 & \tau_2^1 \tau_2^3 + \tau_2^3 \tau_2^1 & \tau_2^2 \tau_2^2 & \tau_2^2 \tau_2^3 + \tau_2^3 \tau_2^2 & \tau_2^3 \tau_2^3 \\ \tau_1^1 \tau_2^1 & \tau_1^1 \tau_2^2 + \tau_1^2 \tau_2^1 & \tau_1^1 \tau_2^3 + \tau_1^3 \tau_2^1 & \tau_1^2 \tau_2^2 & \tau_1^2 \tau_2^3 + \tau_1^3 \tau_2^2 & \tau_1^3 \tau_2^3 \\ \tau_1^1 n^1 & \tau_1^1 n^2 + \tau_1^2 n^1 & \tau_1^1 n^3 + \tau_1^3 n^1 & \tau_1^2 n^2 & \tau_1^2 n^3 + \tau_1^3 n^2 & \tau_1^3 n^3 \\ \tau_2^1 n^1 & \tau_2^1 n^2 + \tau_2^2 n^1 & \tau_2^1 n^3 + \tau_2^3 n^1 & \tau_2^2 n^2 & \tau_2^2 n^3 + \tau_2^3 n^2 & \tau_2^3 n^3 \\ 1 & 0 & 0 & 1 & 0 & 1 \end{pmatrix}
$$

$$
(5.9)\qquad := \begin{pmatrix} C_2^{11} & C_2^{12} & C_2^{13} & C_2^{14} & C_2^{15} & C_2^{16} \\ C_2^{21} & C_2^{22} & C_2^{23} & C_2^{24} & C_2^{25} & C_2^{26} \\ C_2^{31} & C_2^{32} & C_2^{33} & C_2^{34} & C_2^{35} & C_2^{36} \\ C_2^{41} & C_2^{42} & C_2^{43} & C_2^{44} & C_2^{45} & C_2^{46} \\ C_2^{51} & C_2^{52} & C_2^{53} & C_2^{54} & C_2^{55} & C_2^{56} \\ C_2^{61} & C_2^{62} & C_2^{63} & C_2^{64} & C_2^{65} & C_2^{66} \end{pmatrix}.
$$

Nonsingularity of coefficient matrix C_2 is proved in Appendix B, and the right-hand side of (5.8) is known from (5.3). Thus, from (5.8), jump conditions of all second-order velocity derivatives can be found.

With the use of Poisson equation (2.3) for the pressure, jump conditions of all second-order pressure derivatives can be found in the similar way to satisfy

$$
C_2 \begin{bmatrix} \frac{\partial^2 p}{\partial x^1 \partial x^1} \\ \frac{\partial^2 p}{\partial x^1 \partial x^2} \\ \frac{\partial^2 p}{\partial x^1 \partial x^3} \\ \frac{\partial^2 p}{\partial x^2 \partial x^2} \\ \frac{\partial^2 p}{\partial x^2 \partial x^3} \\ \frac{\partial^2 p}{\partial x^3 \partial x^3} \end{bmatrix} = \begin{pmatrix} \frac{\partial^2}{\partial \alpha^1 \partial \alpha^1}\left(\frac{f^i n_i}{\mathcal{J}}\right) \\ \frac{\partial^2}{\partial \alpha^2 \partial \alpha^2}\left(\frac{f^i n_i}{\mathcal{J}}\right) \\ \frac{\partial^2}{\partial \alpha^1 \partial \alpha^2}\left(\frac{f^i n_i}{\mathcal{J}}\right) \\ \frac{\partial}{\partial \alpha^1}\left(\frac{1}{\mathcal{J}}\frac{\partial \tilde{f}^1}{\partial \alpha^1} + \frac{1}{\mathcal{J}}\frac{\partial \tilde{f}^2}{\partial \alpha^2}\right) \\ \frac{\partial}{\partial \alpha^2}\left(\frac{1}{\mathcal{J}}\frac{\partial \tilde{f}^1}{\partial \alpha^1} + \frac{1}{\mathcal{J}}\frac{\partial \tilde{f}^2}{\partial \alpha^2}\right) \\ -\rho\left[\frac{\partial u^i}{\partial x^j}\frac{\partial u^j}{\partial x^i}\right] \end{pmatrix}
$$
$$
- \begin{pmatrix} \frac{\partial^2 X^1}{\partial \alpha^1 \partial \alpha^1} & \frac{\partial^2 X^2}{\partial \alpha^1 \partial \alpha^1} & \frac{\partial^2 X^3}{\partial \alpha^1 \partial \alpha^1} \\ \frac{\partial^2 X^1}{\partial \alpha^2 \partial \alpha^2} & \frac{\partial^2 X^2}{\partial \alpha^2 \partial \alpha^2} & \frac{\partial^2 X^3}{\partial \alpha^2 \partial \alpha^2} \\ \frac{\partial^2 X^1}{\partial \alpha^1 \partial \alpha^2} & \frac{\partial^2 X^2}{\partial \alpha^1 \partial \alpha^2} & \frac{\partial^2 X^3}{\partial \alpha^1 \partial \alpha^2} \\ \frac{\partial n^1}{\partial \alpha^1} & \frac{\partial n^2}{\partial \alpha^1} & \frac{\partial n^3}{\partial \alpha^1} \\ \frac{\partial n^1}{\partial \alpha^2} & \frac{\partial n^2}{\partial \alpha^2} & \frac{\partial n^3}{\partial \alpha^2} \\ 0 & 0 & 0 \end{pmatrix} \begin{bmatrix} \frac{\partial p}{\partial x^1} \\ \frac{\partial p}{\partial x^2} \\ \frac{\partial p}{\partial x^3} \end{bmatrix}. \tag{5.10}
$$

Note that

$$
\begin{aligned}
\left[\frac{\partial u^i}{\partial x^j}\frac{\partial u^j}{\partial x^i}\right] &= 2\left(\frac{\partial u^i}{\partial x^j}\right)_{\mathcal{S}^+}\left[\frac{\partial u^j}{\partial x^i}\right] - \left[\frac{\partial u^i}{\partial x^j}\right]\left[\frac{\partial u^j}{\partial x^i}\right] \\
&= 2\left(\frac{\partial u^i}{\partial x^j}\right)_{\mathcal{S}^-}\left[\frac{\partial u^j}{\partial x^i}\right] + \left[\frac{\partial u^i}{\partial x^j}\right]\left[\frac{\partial u^j}{\partial x^i}\right],
\end{aligned}
$$

and $\left(\frac{\partial u^i}{\partial x^j}\right)_{\mathcal{S}^+}$ or $\left(\frac{\partial u^i}{\partial x^j}\right)_{\mathcal{S}^-}$ can be interpolated from the known velocity field which is used to solve pressure Poisson equation (2.3). The interpolation scheme is given in Xu and Wang [36]. From (5.10), jump conditions of all second-order pressure derivatives can be solved.

5.3. Jump conditions $\left[\frac{\partial^3 \mathbf{u}}{\partial x^i \partial x^j \partial x^k}\right]$ and $\left[\frac{\partial^3 p}{\partial x^i \partial x^j \partial x^k}\right]$. Differentiate (5.5) with respect to α^l $(l = 1, 2)$ and obtain

$$\frac{\partial^3[\mathbf{u}]}{\partial\alpha^l\partial\alpha^m\partial\alpha^n} = 0 \Rightarrow \frac{\partial X^j}{\partial\alpha^n}\frac{\partial X^i}{\partial\alpha^l}\frac{\partial X^k}{\partial\alpha^m}\left[\frac{\partial^3\mathbf{u}}{\partial x^i\partial x^j\partial x^k}\right]$$

$$= -\frac{\partial^3 X^i}{\partial\alpha^l\partial\alpha^m\partial\alpha^n}\left[\frac{\partial\mathbf{u}}{\partial x^i}\right] \tag{5.11}$$

$$-\left(\frac{\partial^2 X^k}{\partial\alpha^l\partial\alpha^m}\frac{\partial X^j}{\partial\alpha^n} + \frac{\partial^2 X^k}{\partial\alpha^l\partial\alpha^n}\frac{\partial X^j}{\partial\alpha^m} + \frac{\partial^2 X^j}{\partial\alpha^m\partial\alpha^n}\frac{\partial X^k}{\partial\alpha^l}\right)\left[\frac{\partial^2\mathbf{u}}{\partial x^j\partial x^k}\right].$$

As $(l, m, n) = (1, 1, 1), (2, 2, 2), (1, 1, 2)$, or $(2, 2, 1)$, we have four equations for $\left[\frac{\partial^3\mathbf{u}}{\partial x^i\partial x^j\partial x^k}\right]$. Differentiate (5.6) with respect to α^l $(l = 1, 2)$ and obtain

$$\frac{\partial^2}{\partial\alpha^l\partial\alpha^m}\left[\frac{\partial\mathbf{u}}{\partial\mathbf{n}}\right] = \frac{\partial^2}{\partial\alpha^l\partial\alpha^m}\frac{(f^k n_k)\mathbf{n} - \mathbf{f}}{\mu\mathcal{J}} \Rightarrow n^j\frac{\partial X^i}{\partial\alpha^l}\frac{\partial X^k}{\partial\alpha^m}\left[\frac{\partial^3\mathbf{u}}{\partial x^i\partial x^j\partial x^k}\right]$$

$$= \frac{\partial^2}{\partial\alpha^l\partial\alpha^m}\frac{(f^k n_k)\mathbf{n} - \mathbf{f}}{\mu\mathcal{J}} - \frac{\partial^2 n^j}{\partial\alpha^l\partial\alpha^m}\left[\frac{\partial\mathbf{u}}{\partial x^j}\right] \tag{5.12}$$

$$-\left(\frac{\partial n^j}{\partial\alpha^m}\frac{\partial X^k}{\partial\alpha^l} + n^j\frac{\partial^2 X^k}{\partial\alpha^l\partial\alpha^m} + \frac{\partial n^j}{\partial\alpha^l}\frac{\partial X^k}{\partial\alpha^m}\right)\left[\frac{\partial^2\mathbf{u}}{\partial x^j\partial x^k}\right].$$

As $(l, m) = (1, 1), (2, 2)$, or $(1, 2)$, we have three other equations. Since a superscript (i, j, or k) can take values 1, 2, or 3, the number of unknowns $\left[\frac{\partial^3\mathbf{u}}{\partial x^i\partial x^j\partial x^k}\right]$ is ten. Three additional equations need to be found. Differentiating (2.1) with respect to x^k $(k = 1, 2, 3)$ and applying (3.2), we find

$$\left[\frac{\partial^3\mathbf{u}}{\partial x^j\partial x^j\partial x^k}\right] = \frac{1}{\mu}\left[\frac{\partial}{\partial x^k}\nabla p\right] + \frac{\rho}{\mu}\left(\left[\frac{\partial^2\mathbf{u}}{\partial t\partial x^k}\right] + \left[\frac{\partial u^j}{\partial x^k}\frac{\partial\mathbf{u}}{\partial x^j}\right] + u^j\left[\frac{\partial^2\mathbf{u}}{\partial x^j\partial x^k}\right]\right), \tag{5.13}$$

which provides three additional equations if $\left[\frac{\partial^2\mathbf{u}}{\partial t\partial x^k}\right]$ is known. Jump conditions $\left[\frac{\partial^2\mathbf{u}}{\partial t\partial x^k}\right]$ are derived in section 6.1. Thus, combining (5.11), (5.12), and (5.13) gives a 10×10 system for ten unknowns $\left[\frac{\partial^3\mathbf{u}}{\partial x^i\partial x^j\partial x^k}\right]$.

To simplify the system, let

$$\mathbf{d}^j_{lm} = \frac{\partial X^i}{\partial\alpha^l}\frac{\partial X^k}{\partial\alpha^m}\left[\frac{\partial^3\mathbf{u}}{\partial x^i\partial x^j\partial x^k}\right], \tag{5.14}$$

and denote the right-hand sides of (5.11) and (5.12) as $\mathbf{r}^n_{lm}$ $(n = 1, 2)$ and $\mathbf{r}^3_{lm}$, respectively. Then, we can rewrite (5.11) and (5.12) as

$$C_1\begin{pmatrix}\mathbf{d}^1_{lm}\\ \mathbf{d}^2_{lm}\\ \mathbf{d}^3_{lm}\end{pmatrix} = \begin{pmatrix}\mathbf{r}^1_{lm}\\ \mathbf{r}^2_{lm}\\ \mathbf{r}^3_{lm}\end{pmatrix},$$

from which we can solve $\mathbf{d}^j_{lm}$ $(j = 1, 2, 3)$ since C_1 is nonsingular. After $\mathbf{d}^j_{lm}$ is solved, we can combine (5.14) and (5.13) to obtain a simplified system as

$$C_3 \begin{bmatrix} \frac{\partial^3 \mathbf{u}}{\partial x^1 \partial x^1 \partial x^1} \\ \frac{\partial^3 \mathbf{u}}{\partial x^1 \partial x^1 \partial x^2} \\ \frac{\partial^3 \mathbf{u}}{\partial x^1 \partial x^1 \partial x^3} \\ \frac{\partial^3 \mathbf{u}}{\partial x^1 \partial x^2 \partial x^2} \\ \frac{\partial^3 \mathbf{u}}{\partial x^1 \partial x^2 \partial x^3} \\ \frac{\partial^3 \mathbf{u}}{\partial x^1 \partial x^3 \partial x^3} \\ \frac{\partial^3 \mathbf{u}}{\partial x^2 \partial x^2 \partial x^2} \\ \frac{\partial^3 \mathbf{u}}{\partial x^2 \partial x^2 \partial x^3} \\ \frac{\partial^3 \mathbf{u}}{\partial x^2 \partial x^3 \partial x^3} \\ \frac{\partial^3 \mathbf{u}}{\partial x^3 \partial x^3 \partial x^3} \end{bmatrix} = \begin{pmatrix} \mathbf{d}^1_{11} \\ \mathbf{d}^1_{22} \\ \mathbf{d}^1_{12} \\ \mathbf{d}^2_{11} \\ \mathbf{d}^2_{22} \\ \mathbf{d}^2_{12} \\ \mathbf{d}^3_{11} \\ \mathbf{d}^3_{22} \\ \mathbf{d}^3_{12} \\ \mathbf{d}^4_{11} \\ \mathbf{d}^4_{22} \\ \mathbf{d}^4_{12} \end{pmatrix}, \tag{5.15}$$

where

$$C_3 = \begin{pmatrix} C_2^{11} & C_2^{12} & C_2^{13} & C_2^{14} & C_2^{15} & C_2^{16} & 0 & 0 & 0 & 0 \\ C_2^{21} & C_2^{22} & C_2^{23} & C_2^{24} & C_2^{25} & C_2^{26} & 0 & 0 & 0 & 0 \\ C_2^{31} & C_2^{32} & C_2^{33} & C_2^{34} & C_2^{35} & C_2^{36} & 0 & 0 & 0 & 0 \\ 0 & C_2^{11} & 0 & C_2^{12} & C_2^{13} & 0 & C_2^{14} & C_2^{15} & C_2^{16} & 0 \\ 0 & C_2^{21} & 0 & C_2^{22} & C_2^{23} & 0 & C_2^{24} & C_2^{25} & C_2^{26} & 0 \\ 0 & C_2^{31} & 0 & C_2^{32} & C_2^{33} & 0 & C_2^{34} & C_2^{35} & C_2^{36} & 0 \\ 0 & 0 & C_2^{11} & 0 & C_2^{12} & C_2^{13} & 0 & C_2^{14} & C_2^{15} & C_2^{16} \\ 0 & 0 & C_2^{21} & 0 & C_2^{22} & C_2^{23} & 0 & C_2^{24} & C_2^{25} & C_2^{26} \\ 0 & 0 & C_2^{31} & 0 & C_2^{32} & C_2^{33} & 0 & C_2^{34} & C_2^{35} & C_2^{36} \\ 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \end{pmatrix}, \tag{5.16}$$

with matrix elements defined in (5.9) and

$$\mathbf{d}^4_{11} = \frac{1}{\mu}\left[\frac{\partial}{\partial x^1}\nabla p\right] + \frac{\rho}{\mu}\left(\left[\frac{\partial^2 \mathbf{u}}{\partial t \partial x^1}\right] + \left[\frac{\partial u^j}{\partial x^1}\frac{\partial \mathbf{u}}{\partial x^j}\right] + u^j\left[\frac{\partial^2 \mathbf{u}}{\partial x^j \partial x^1}\right]\right),$$

$$\mathbf{d}^4_{22} = \frac{1}{\mu}\left[\frac{\partial}{\partial x^2}\nabla p\right] + \frac{\rho}{\mu}\left(\left[\frac{\partial^2 \mathbf{u}}{\partial t \partial x^2}\right] + \left[\frac{\partial u^j}{\partial x^2}\frac{\partial \mathbf{u}}{\partial x^j}\right] + u^j\left[\frac{\partial^2 \mathbf{u}}{\partial x^j \partial x^2}\right]\right),$$

$$\mathbf{d}^4_{12} = \frac{1}{\mu}\left[\frac{\partial}{\partial x^3}\nabla p\right] + \frac{\rho}{\mu}\left(\left[\frac{\partial^2 \mathbf{u}}{\partial t \partial x^3}\right] + \left[\frac{\partial u^j}{\partial x^3}\frac{\partial \mathbf{u}}{\partial x^j}\right] + u^j\left[\frac{\partial^2 \mathbf{u}}{\partial x^j \partial x^3}\right]\right).$$

To obtain a unique solution to (5.15) we need to show $\operatorname{rank}(C_3) = 10$, which is done in Appendix C.

Similarly, with the use of Poisson equation (2.3), we can obtain an equation system for $\left[\frac{\partial^3 p}{\partial x^i \partial x^j \partial x^k}\right]$ as

$$C_3 \begin{bmatrix} \dfrac{\partial^3 p}{\partial x^1 \partial x^1 \partial x^1} \\ \dfrac{\partial^3 p}{\partial x^1 \partial x^1 \partial x^2} \\ \dfrac{\partial^3 p}{\partial x^1 \partial x^1 \partial x^3} \\ \dfrac{\partial^3 p}{\partial x^1 \partial x^2 \partial x^2} \\ \dfrac{\partial^3 p}{\partial x^1 \partial x^2 \partial x^3} \\ \dfrac{\partial^3 p}{\partial x^1 \partial x^3 \partial x^3} \\ \dfrac{\partial^3 p}{\partial x^2 \partial x^2 \partial x^2} \\ \dfrac{\partial^3 p}{\partial x^2 \partial x^2 \partial x^3} \\ \dfrac{\partial^3 p}{\partial x^2 \partial x^3 \partial x^3} \\ \dfrac{\partial^3 p}{\partial x^3 \partial x^3 \partial x^3} \end{bmatrix} = \begin{pmatrix} D^1_{11} \\ D^1_{22} \\ D^1_{12} \\ D^2_{11} \\ D^2_{22} \\ D^2_{12} \\ D^3_{11} \\ D^3_{22} \\ D^3_{12} \\ D^4_{11} \\ D^4_{22} \\ D^4_{12} \end{pmatrix}, \tag{5.17}$$

in which

$$D^j_{lm} = \frac{\partial X^i}{\partial \alpha^l} \frac{\partial X^k}{\partial \alpha^m} \left[\frac{\partial^3 p}{\partial x^i \partial x^j \partial x^k}\right], \tag{5.18}$$

with $j = 1, 2, 3$ and $(l, m) = (1, 1), (2, 2)$, or $(1, 2)$, and

$$\begin{aligned} D^4_{11} = &- \frac{\partial \rho}{\partial x^i} \left(\left[\frac{\partial^2 u^i}{\partial t \partial x^1}\right] + u^j \left[\frac{\partial^2 u^i}{\partial x^j \partial x^1}\right] + \left[\frac{\partial u^j}{\partial x^1} \frac{\partial u^i}{\partial x^j}\right] \right) \\ &- \frac{\partial \rho}{\partial x^1} \left[\frac{\partial u^j}{\partial x^i} \frac{\partial u^i}{\partial x^j}\right] - 2\rho \left[\frac{\partial u^j}{\partial x^i} \frac{\partial^2 u^i}{\partial x^j \partial x^1}\right], \\ D^4_{22} = &- \frac{\partial \rho}{\partial x^i} \left(\left[\frac{\partial^2 u^i}{\partial t \partial x^2}\right] + u^j \left[\frac{\partial^2 u^i}{\partial x^j \partial x^2}\right] + \left[\frac{\partial u^j}{\partial x^2} \frac{\partial u^i}{\partial x^j}\right] \right) \\ &- \frac{\partial \rho}{\partial x^2} \left[\frac{\partial u^j}{\partial x^i} \frac{\partial u^i}{\partial x^j}\right] - 2\rho \left[\frac{\partial u^j}{\partial x^i} \frac{\partial^2 u^i}{\partial x^j \partial x^2}\right], \\ D^4_{12} = &- \frac{\partial \rho}{\partial x^i} \left(\left[\frac{\partial^2 u^i}{\partial t \partial x^3}\right] + u^j \left[\frac{\partial^2 u^i}{\partial x^j \partial x^3}\right] + \left[\frac{\partial u^j}{\partial x^3} \frac{\partial u^i}{\partial x^j}\right] \right) \\ &- \frac{\partial \rho}{\partial x^3} \left[\frac{\partial u^j}{\partial x^i} \frac{\partial u^i}{\partial x^j}\right] - 2\rho \left[\frac{\partial u^j}{\partial x^i} \frac{\partial^2 u^i}{\partial x^j \partial x^3}\right]. \end{aligned}$$

If fluid density ρ is a constant, we have

$$D^4_{11} = -\,2\rho\left[\frac{\partial u^j}{\partial x^i}\frac{\partial^2 u^i}{\partial x^j\partial x^1}\right],$$

$$D^4_{22} = -\,2\rho\left[\frac{\partial u^j}{\partial x^i}\frac{\partial^2 u^i}{\partial x^j\partial x^2}\right],$$

$$D^4_{12} = -\,2\rho\left[\frac{\partial u^j}{\partial x^i}\frac{\partial^2 u^i}{\partial x^j\partial x^3}\right].$$

We solve D^j_{lm} from

$$C_1\begin{pmatrix} D^1_{lm} \\ D^2_{lm} \\ D^3_{lm}\end{pmatrix} = \begin{pmatrix} R^1_{lm} \\ R^2_{lm} \\ R^3_{lm}\end{pmatrix},$$

where R^n_{lm} $(n = 1, 2)$ and R^3_{lm} are

$$\begin{aligned} R^n_{lm} = {} & \frac{\partial^3}{\partial\alpha^l\partial\alpha^m\partial\alpha^n}\left(\frac{f^i n_i}{\mathcal{J}}\right) - \frac{\partial^3 X^i}{\partial\alpha^l\partial\alpha^m\partial\alpha^n}\left[\frac{\partial p}{\partial x^i}\right] \\ & - \left(\frac{\partial^2 X^k}{\partial\alpha^l\partial\alpha^m}\frac{\partial X^j}{\partial\alpha^n} + \frac{\partial^2 X^k}{\partial\alpha^l\partial\alpha^n}\frac{\partial X^j}{\partial\alpha^m} + \frac{\partial^2 X^j}{\partial\alpha^m\partial\alpha^n}\frac{\partial X^k}{\partial\alpha^l}\right)\left[\frac{\partial^2 p}{\partial x^j\partial x^k}\right], \\ R^3_{lm} = {} & \frac{\partial^2}{\partial\alpha^l\partial\alpha^m}\left(\frac{1}{\mathcal{J}}\frac{\partial\tilde{f}^1}{\partial\alpha^1} + \frac{1}{\mathcal{J}}\frac{\partial\tilde{f}^2}{\partial\alpha^2}\right) - \frac{\partial^2 n^j}{\partial\alpha^l\partial\alpha^m}\left[\frac{\partial p}{\partial x^j}\right] \\ & - \left(\frac{\partial n^j}{\partial\alpha^m}\frac{\partial X^k}{\partial\alpha^l} + n^j\frac{\partial^2 X^k}{\partial\alpha^l\partial\alpha^m} + \frac{\partial n^j}{\partial\alpha^l}\frac{\partial X^k}{\partial\alpha^m}\right)\left[\frac{\partial^2 p}{\partial x^j\partial x^k}\right]. \end{aligned}$$

6. Jump conditions of temporal derivatives. When singular surface $\mathcal{S}$ is passing a fixed point $\mathbf{x}^*$ in space at time t^*, using $\mathbf{X}^*$ to denote the point on $\mathcal{S}$ which coincides with the point $\mathbf{x}^*$, for flow quantity ψ, we have the following relation between $[\![\psi(\mathbf{X}^*, t^*)]\!] = (\psi)_{t^{*+}} - (\psi)_{t^{*-}}$ and $[\psi(\mathbf{X}^*, t^*)] = (\psi)_{\mathcal{S}^+} - (\psi)_{\mathcal{S}^-}$:

$$[\![\psi]\!] = \begin{cases} [\psi], & \mathbf{u}(\mathbf{x}^*)\cdot\mathbf{n}(\mathbf{X}^*) < 0, \\ -[\psi], & \mathbf{u}(\mathbf{x}^*)\cdot\mathbf{n}(\mathbf{X}^*) > 0. \end{cases} \tag{6.1}$$

If $\mathbf{u}(\mathbf{x}^*)\cdot\mathbf{n}(\mathbf{X}^*) = 0$, we can approximate temporal derivatives at $\mathbf{x}^*$ by those at $\mathbf{X}^*|_{\mathcal{S}^+}$ or $\mathbf{X}^*|_{\mathcal{S}^-}$ with $[\![\cdot]\!] = 0$. Thus, instead of deriving $[\![u^i]\!]$, $[\![\frac{\partial u^i}{\partial t}]\!]$, and $[\![\frac{\partial^2 u^i}{\partial t^2}]\!]$, we turn to deriving $[u^i]$, $[\frac{\partial u^i}{\partial t}]$, and $[\frac{\partial^2 u^i}{\partial t^2}]$. We also need to derive $[\frac{\partial^2 u^i}{\partial t\partial x^k}]$, which appears in the right-hand side of (5.13). The spatial jump condition of velocity, $[u^i]$, is already given in (3.10).

Note that we need interpolation to obtain t^* and jump conditions evaluated at t^* in simulation practice. The interpolation procedures are given in Xu and Wang [36].

6.1. Jump conditions $[\frac{\partial u^i}{\partial t}]$, $[\frac{\partial^2 u^i}{\partial t^2}]$, and $[\frac{\partial^2 u^i}{\partial t\partial x^k}]$. From (3.2), we directly have

$$\left[\frac{\partial u^i}{\partial t}\right] = -u^j\left[\frac{\partial u^i}{\partial x^j}\right]. \tag{6.2}$$

Differentiating (6.2) with respect to t, we have

$$\left[\frac{\partial^2 u^i}{\partial t^2}\right] = -2u^j \left[\frac{\partial^2 u^i}{\partial t \partial x^j}\right] - \left[\frac{\partial u^j}{\partial t}\frac{\partial u^i}{\partial x^j}\right] - u^j u^k \left[\frac{\partial^2 u^i}{\partial x^j \partial x^k}\right] - u^k \left[\frac{\partial u^j}{\partial x^k}\frac{\partial u^i}{\partial x^j}\right], \tag{6.3}$$

where $\left[\frac{\partial^2 u^i}{\partial t \partial x^k}\right]$ also appears on the right-hand side of (5.13).

Differentiating (6.2) with respect to α^l $(l = 1, 2)$, we obtain

$$\frac{\partial X^k}{\partial \alpha^l}\left[\frac{\partial^2 u^i}{\partial t \partial x^k}\right] = \frac{\partial X^k}{\partial \alpha^l}\left(\left[\frac{\partial u^i}{\partial x^j}\frac{\partial u^j}{\partial x^k}\right] + u^j \left[\frac{\partial^2 u^i}{\partial x^j \partial x^k}\right]\right). \tag{6.4}$$

Differentiating (3.5) with respect to time t, we obtain

$$n^k \left[\frac{\partial^2 u^i}{\partial t \partial x^k}\right] = \frac{\mathrm{d}}{\mathrm{d}t}\frac{(f^k n_k)n^i - f^i}{\mu \mathcal{J}} - \frac{\mathrm{d}n^k}{\mathrm{d}t}\left[\frac{\partial u^i}{\partial x^k}\right] - n^k u^j \left[\frac{\partial^2 u^i}{\partial x^k \partial x^j}\right]. \tag{6.5}$$

Denote the right-hand side of (6.4) by r_t^{il} $(l = 1, 2)$ and the right-hand side of (6.5) by r_t^{i3}. Combining (6.4) and (6.5) then gives

$$C_1 \begin{bmatrix} \frac{\partial^2 u^i}{\partial t \partial x^1} \\ \frac{\partial^2 u^i}{\partial t \partial x^2} \\ \frac{\partial^2 u^i}{\partial t \partial x^3} \end{bmatrix} = \begin{pmatrix} r_t^{i1} \\ r_t^{i2} \\ r_t^{i3} \end{pmatrix}, \tag{6.6}$$

from which we can solve $\left[\frac{\partial^2 u^i}{\partial t \partial x^k}\right]$.

6.2. A numerical example. Here, we implement the IIM in a one-dimensional linear wave equation with a moving singular source. This simple example is designed to illustrate the necessity of including temporal jump conditions in the discretization of temporal derivatives. We also examine the convergence property of the IIM in this example.

The system has the following form:

$$\begin{cases} \frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} = c(t)\delta(x - X(t)), \quad X(t) = bt, \quad L \le x \le R, \\ u(x, 0) = g(x), \\ \frac{\partial u(L, t)}{\partial x} = 0, \end{cases}$$

where a and b are constants satisfying $a > 0$ and $a \neq b$. The analytical solution to the problem is

$$u(x, t) = g(x - at) + \frac{h(x - bt) + h(at - x)}{2(a - b)} c\left(\frac{x - at}{b - a}\right),$$

where $h(x)$ is the step function.

Integrating the wave equation with respect to x from $X^-(t)$ to $X^+(t)$, the jump condition of u at $x = X(t)$ can be obtained as

$$[u] = \frac{c(t)}{a - b},$$

where $[\cdot] = (\cdot)_{X^+} - (\cdot)_{X^-}$. Following the procedure described in sections 5 and 6.1, we have derived the jump conditions at the singular source $x = X(t)$:

$$[u] = \frac{c(t)}{a-b}, \quad \left[\frac{\partial u}{\partial x}\right] = -\frac{1}{(a-b)^2}\frac{\mathrm{d}c(t)}{\mathrm{d}t}, \quad \left[\frac{\partial^2 u}{\partial t \partial x}\right] = 0, \quad \left[\frac{\partial^2 u}{\partial x^2}\right] = 0,$$

$$[\![u]\!] = -\frac{c(t)}{a-b}, \quad \left[\!\!\left[\frac{\partial u}{\partial t}\right]\!\!\right] = -\frac{a}{(a-b)^2}\frac{\mathrm{d}c(t)}{\mathrm{d}t}, \quad \left[\!\!\left[\frac{\partial^2 u}{\partial x \partial t}\right]\!\!\right] = 0, \quad \left[\!\!\left[\frac{\partial^2 u}{\partial t^2}\right]\!\!\right] = 0,$$

where $[\![\cdot]\!] = (\cdot)_{t^+} - (\cdot)_{t^-}$. At the wave front generated by the singular source, i.e., at $x = X(0) + at$, the solution is continuous but unsmooth, and the corresponding jump conditions are

$$[u] = 0, \quad \left[\frac{\partial u}{\partial x}\right] = \frac{1}{(a-b)^2}\frac{\mathrm{d}c(t)}{\mathrm{d}t}, \quad \left[\frac{\partial^2 u}{\partial t \partial x}\right] = 0, \quad \left[\frac{\partial^2 u}{\partial x^2}\right] = 0,$$

$$[\![u]\!] = 0, \quad \left[\!\!\left[\frac{\partial u}{\partial t}\right]\!\!\right] = \frac{a}{(a-b)^2}\frac{\mathrm{d}c(t)}{\mathrm{d}t}, \quad \left[\!\!\left[\frac{\partial^2 u}{\partial x \partial t}\right]\!\!\right] = 0, \quad \left[\!\!\left[\frac{\partial^2 u}{\partial t^2}\right]\!\!\right] = 0.$$

The Crank–Nicolson method is used to discretize the problem on a uniform mesh:

$$\left(\frac{u_i^{n+1} - u_i^n}{\Delta t} + CT_i^n\right) + \frac{a}{2}\left(\left(\frac{u_{i+1}^n - u_{i-1}^n}{2\Delta x} + CX_i^n\right) + \left(\frac{u_{i+1}^{n+1} - u_{i-1}^{n+1}}{2\Delta x} + CX_i^{n+1} + CXT_i^n\right)\right) = 0,$$

where n denotes a time layer, i an interior grid point, Δt a time step, Δx a space step, CT the jump contribution in the finite difference of the temporal derivative, CX the jump contribution in the finite difference of the spatial derivative, and CXT the jump contribution to keep the second-order temporal accuracy of the Crank–Nicolson scheme. The jump contributions CT, CX, and CXT are caused by the unsmoothness of the solution. We calculate CT and CX according to generalized Taylor expansion to achieve $\mathcal{O}((\Delta t)^2)$ and $\mathcal{O}((\Delta x)^2)$ accuracy of the finite differences. The calculation of CXT originates from the following generalized Taylor expansion:

$$\left(\frac{\partial u}{\partial x}\right)^{n+1} = \left(\frac{\partial u}{\partial x}\right)^n + \left(\frac{\partial^2 u}{\partial x \partial t}\right)^n \Delta t + \left(\left[\!\!\left[\frac{\partial u}{\partial x}\right]\!\!\right] + \left[\!\!\left[\frac{\partial^2 u}{\partial x \partial t}\right]\!\!\right]\Delta t\right) + \mathcal{O}(\Delta t)^2.$$

If there is a grid point I satisfying $X^n < x_I \leq X^{n+1}$ from time layer n to time layer $(n+1)$, then we let $x_I = X((n+\beta)\Delta t), 0 < \beta \leq 1$, and compute CT_i^n and CXT_i^n as

$$\begin{cases} CT_I^n = -\dfrac{1}{\Delta t}\left([\![u]\!] + \left[\!\!\left[\dfrac{\partial u}{\partial t}\right]\!\!\right](1-\beta)\Delta t + \dfrac{1}{2}\left[\!\!\left[\dfrac{\partial^2 u}{\partial t^2}\right]\!\!\right](1-\beta)^2(\Delta t)^2\right), \\ CT_i^n = 0, \quad i \neq I. \end{cases}$$

and

$$\begin{cases} CXT_I^n = -\left(\left[\!\!\left[\dfrac{\partial u}{\partial x}\right]\!\!\right] + \left[\!\!\left[\dfrac{\partial^2 u}{\partial x \partial t}\right]\!\!\right]\Delta t\right), \\ CXT_i^n = 0, \quad i \neq I. \end{cases}$$

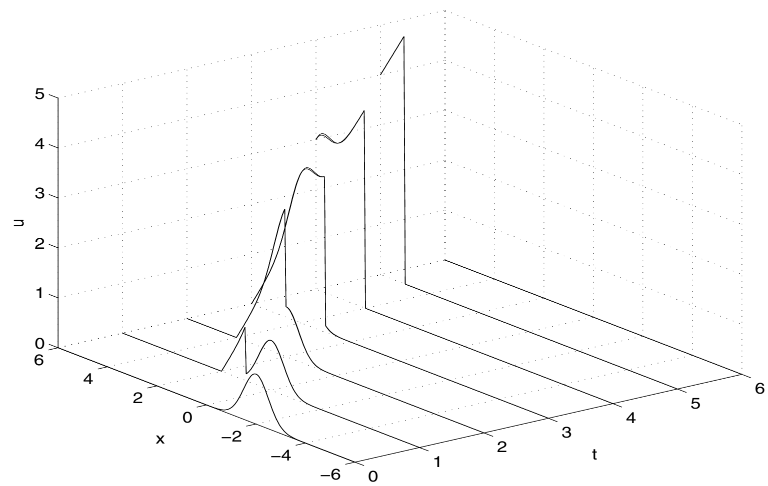

FIG. 6.1. *Comparison between the numerical solution (solid lines) and the analytical solution (dashed lines, hidden by solid lines). For the numerical solution,* $\Delta x = 0.02$ *and* $CFL = \frac{2\Delta t}{\Delta x} = 0.4$.

Otherwise, we have $CT_i^n = 0$ and $CXT_i^n = 0$ for any grid point i. If $x_{I-1} \le X^n < x_I$ at time n, then we can compute CX_i^n as

$$\begin{cases} CX_I^n = -\dfrac{1}{2\Delta x}\left([u] + \left[\dfrac{\partial u}{\partial x}\right](x_{I-1} - X^n) + \dfrac{1}{2}\left[\dfrac{\partial^2 u}{\partial x^2}\right](x_{I-1} - X^n)^2\right), \\ CX_{I-1}^n = -\dfrac{1}{2\Delta x}\left([u] + \left[\dfrac{\partial u}{\partial x}\right](x_I - X^n) + \dfrac{1}{2}\left[\dfrac{\partial^2 u}{\partial x^2}\right](x_I - X^n)^2\right), \\ CT_i^n = 0, \quad i \neq I, I-1. \end{cases}$$

Similarly, we can compute CX_i^{n+1}. At the outlet $x = R$ of the domain, an upwind finite difference scheme is used to approximate the spatial derivative, and the correction CX takes a different form. We also treat the effect of the discontinuity at $x = X(0) + at$ in the same way.

The results presented below are for $a = 2$, $b = 1$, $c(t) = t$, $L = -6$, $R = 6$, and $g(x) = \mathrm{e}^{-2(x+2)^2}$. Figure 6.1 shows the numerical solution and the analytical solution at times $t = 0, 1, 2, \ldots, 6$. After time $t = 6$, the initial wave, the singular source, and the wave generated by the singular source have all exited from the domain. The IIM produces the sharp jumps in the numerical solution and the correct wave-source interaction.

To check the accuracy of the numerical scheme near the singular source, we look at the solutions at time $t = 2$, when the position of the singular source is at $x = 2$. As spatial resolution changes, Figure 6.2(a) shows the change of the infinity norm of the error based on the analytical solution. For each spatial resolution, a very small time step corresponding to $CFL = \frac{2\Delta t}{\Delta x} = 0.002$ is used to ensure that the temporal discretization error is negligible compared with the spatial one. Second-order accuracy in space is indicated in Figure 6.2(a), as expected.

In order to check temporal accuracy, we obtain an accurate reference solution by using a very small time step corresponding to $CFL = 0.002$. We compute the numerical solution using different time steps with the same spatial resolution $\Delta x = 0.05$ as a reference. By subtracting the reference solution from a numerical solution

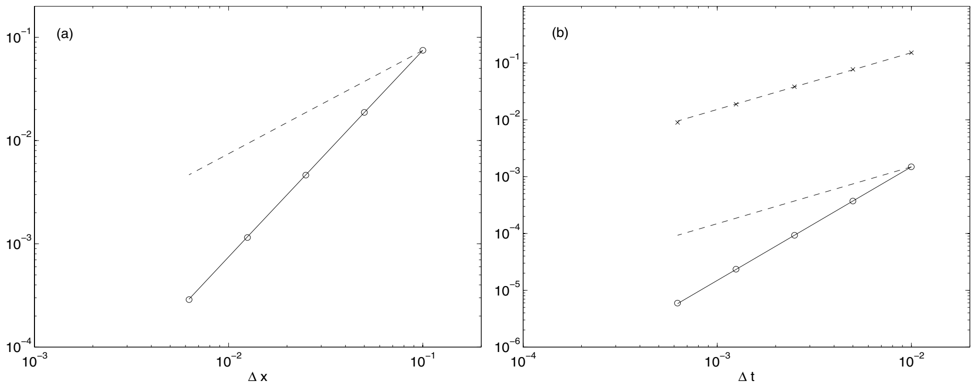

FIG. 6.2. *The infinity norm of the error as a function of* (a) *spatial resolution with* $CFL = 0.002$ *and* (b) *temporal resolution with* $\Delta x = 0.05$*. Open circles: numerical error with the jump contribution* CXT*, x-marks: numerical error without the jump contribution* CXT*, solid lines: second-order accuracy, dashed lines: first-order accuracy.*

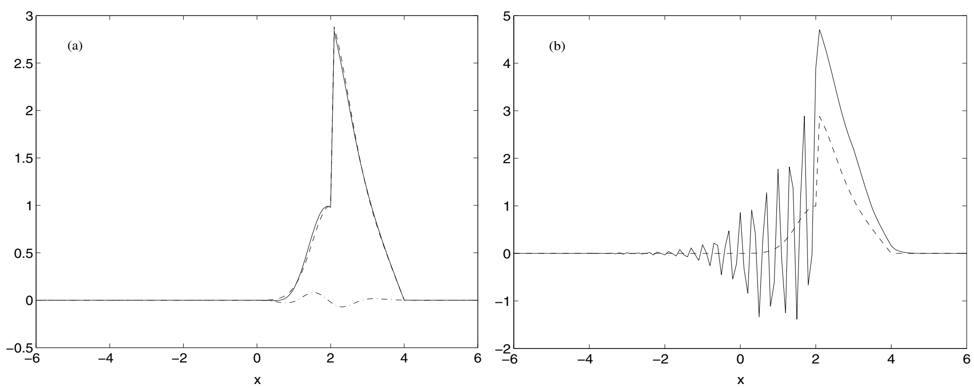

FIG. 6.3. *The effect of the jump contributions in the temporal discretization. Results* (a) *with temporal jump contributions,* (b) *without temporal jump contributions. The results are for* $t = 2$ *with* $\Delta x = 0.1$ *and* $CFL = 0.4$*. Solid lines: numerical results, dashed lines: analytical results, dash-dotted lines: difference between numerical and analytical results.*

calculated using the same space step but a different time step, we cancel out the spatial discretization error and obtain the temporal error. The results are plotted in Figure 6.2(b). Second-order accuracy in time is seen. If the jump contribution CXT is not included, only first-order accuracy in time can be achieved, as seen in Figure 6.2(b).

Figure 6.3(a) compares the numerical and analytical results at time $t = 2$ with $\Delta x = 0.1$ and $CFL = 0.4$. The amplitude of the jump contribution CT has the same order as the jump contribution CX when Δt is of the same order as Δx. If the jump contribution CT is not included, a numerical result can be totally wrong, as also shown in Figure 6.3(b).

As the velocity can be piecewise smooth across a singular surface in a viscous flow, jump contributions in the discretizations of the temporal terms in Navier–Stokes equations can be nonzero. If they are of the same order as the leading ones in spatial discretizations, the inclusion of the jump contributions in the temporal discretizations may be necessary. The moving interface problem simulated by Li and Lai [20] is an exception. They considered the relaxation of a perturbed two-dimensional balloon immersed in an incompressible viscous fluid. The tangential force along the balloon surface is always zero in their case, so the velocity is smooth in both space and time, though the pressure is not. The jump contributions in temporal discretizations are always zero in this case. Recently, we designed an oscillating Taylor–Couette flow to look at the effect of jump contributions for temporal discretization on temporal convergence rate and temporal accuracy [36]. We found that the effect is very small. Whether this is true in general remains to be investigated.

7. Discussion. Naturally, we ask whether we can improve discretization accuracy in the IIM to arbitrarily high order for three-dimensional incompressible Navier–Stokes flows. We can derive equations for jump conditions by differentiating the principal jump conditions with respect to the Lagrangian parameters and differentiating the governing equations with respect to the Cartesian coordinates, as presented in sections 5 and 6.1. To examine this possibility, we need to know whether the number of derived equations is enough for unknown jump conditions and whether the equation system has a unique solution.

Regarding the number of equations and unknowns, we introduce a lemma.

LEMMA 7.1. *Paint m indistinctive balls with n different colors. Each ball has to be painted with one and only one color. The number of different nonordered color combinations for the m balls is*

$$\mathcal{H}_m^n = \sum_{k_1=0}^{m} \sum_{k_2=0}^{k_1} \sum_{k_3=0}^{k_2} \cdots \sum_{k_{n-2}=0}^{k_{n-3}} \sum_{k_{n-1}=0}^{k_{n-2}} \mathcal{H}_{k_{n-1}}^1, \tag{7.1}$$

with $\mathcal{H}_0^i = 1$ and $\mathcal{H}_i^1 = 1$, where i is a positive integer.

Proof. Paint k_1 $(k_1 = 0, 1, 2, \ldots, m)$ balls using the same color. There are $n-1$ colors and $m - k_1$ balls left. Therefore, for a particular value of k_1, the number of different outcomes is $\mathcal{H}_{m-k_1}^{n-1}$. Summing $\mathcal{H}_{m-k_1}^{n-1}$ over all possible values of k_1 yields

$$\mathcal{H}_m^n = \sum_{k_1=0}^{m} \mathcal{H}_{m-k_1}^{n-1} = \sum_{k_1=0}^{m} \mathcal{H}_{k_1}^{n-1}. \tag{7.2}$$

Using recursion (7.2) repeatedly, the lemma follows. ∎

If we want the discretization accuracy of the second-order velocity and pressure derivatives to be order of $m-1$ $(m \geq 2)$ in Navier–Stokes flow simulation, we need jump conditions of all velocity and pressure derivatives of order m. In a simulation of space dimension n, the number of unknown jump conditions of the velocity derivatives of order m is $\mathcal{H}_m^n$. Differentiating (3.10) with respect to the Lagrangian parameters can provide $\mathcal{H}_m^{n-1}$ equations for the unknowns, differentiating (3.12) can provide $\mathcal{H}_{m-1}^{n-1}$, and differentiating (2.1) with respect to the Cartesian coordinates and then subtracting the resulting equation at $\mathcal{S}^-$ from that at $\mathcal{S}^+$ can give $\mathcal{H}_{m-2}^n$ equations. For closure, we require

$$\mathcal{H}_m^{n-1} + \mathcal{H}_{m-1}^{n-1} + \mathcal{H}_{m-2}^n \geq \mathcal{H}_m^n.$$

The same requirement applies to the unknown jump conditions of the pressure derivatives of order m.

In three-dimensional simulations, $n = 3$, $\mathcal{H}_m^2 = m + 1$, and $\mathcal{H}_m^3 = \frac{(m+1)(m+2)}{2}$ according to (7.1). Hence, we have

$$\mathcal{H}_m^2 + \mathcal{H}_{m-1}^2 + \mathcal{H}_{m-2}^3 = \mathcal{H}_m^3;$$

that is, the number of equations is equal to the number of unknowns, supposing $\mathcal{H}^n_{m-2}$ jump conditions for temporal derivatives in the equations (for example, $\left[\frac{\partial^2 u^i}{\partial t \partial x^k}\right]$ in (5.13) in the case when $m = 3$ and $n = 3$) are known. The equation system may have a unique solution, which may be verified by induction with the use of the method given in Appendix C. However, the $\mathcal{H}^n_{m-2}$ jump conditions for the temporal derivatives are not directly available. We need to find equations for them too. As shown in section 6.1, a unique solution to the required temporal-derivative jump conditions can be obtained when $m = 3$. When $m > 3$, it is not clear whether a unique solution can be found. Thus with the method that we present in this paper for deriving jump conditions, ascending the discretization accuracy to arbitrarily high order is not achievable in the IIM for three-dimensional Navier–Stokes flows.

Appendix A. Proof of Theorem 3.3. First consider the situation in which G^i has finite jumps at $\mathcal{S}$. If necessary, smoothly extend $\mathcal{S}$ so that it cuts $\mathcal{V}$ into two separated regions. Form a banded region $\mathcal{V}_s$ which encloses $\mathcal{S}$ with surface $\mathcal{S}^+_{\epsilon/2}$ at one side of $\mathcal{S}$ and $\mathcal{S}^-_{\epsilon/2}$ at the other, where $\mathcal{S}^+_{\epsilon/2}$ and $\mathcal{S}^-_{\epsilon/2}$ are away from $\mathcal{S}$ by a small distance $\frac{\epsilon}{2}$. Thus,

$$\int_{\mathcal{V}_s} \frac{\partial G^i}{\partial x^i} \mathrm{d}\mathcal{V} = \int_{\mathcal{S}} \int_{-\frac{\epsilon}{2}}^{+\frac{\epsilon}{2}} \frac{\partial G^i}{\partial x^i} \mathrm{d}r \mathrm{d}\mathcal{S},$$

where $r = (\mathbf{x} - \mathbf{X}_{\mathrm{d}\mathcal{S}}) \cdot \mathbf{n}$ with $\mathbf{X}_{\mathrm{d}\mathcal{S}}$ representing a fixed point at the infinitesimal surface $\mathrm{d}\mathcal{S}$ and $\mathbf{n}$ normal at $\mathbf{X}_{\mathrm{d}\mathcal{S}}$ pointing to side $\mathcal{S}^+_{\epsilon/2}$. Denote the region between $\mathcal{S}^+_{\epsilon/2}$ and $\mathcal{S}$ as $\mathcal{V}_1$ and the region between $\mathcal{S}^-_{\epsilon/2}$ and $\mathcal{S}$ as $\mathcal{V}_2$. Analytically extend G^i in region $\mathcal{V}_1$ to region $\mathcal{V}_2$, and name the extended function G^i_1. Analytically extend G^i in region $\mathcal{V}_2$ to region $\mathcal{V}_1$, and name the extended function G^i_2. Introduce step function $h(r)$ as

$$h(r) = \begin{cases} 0, & r < 0, \\ 1, & r > 0. \end{cases}$$

Then,

$$\begin{aligned}
\int_{\mathcal{S}} \int_{-\frac{\epsilon}{2}}^{+\frac{\epsilon}{2}} \frac{\partial G^i}{\partial x^i} \mathrm{d}r \mathrm{d}\mathcal{S} &= \int_{\mathcal{S}} \int_{-\frac{\epsilon}{2}}^{+\frac{\epsilon}{2}} \frac{\partial (h(r) G^i_1 + h(-r) G^i_2)}{\partial x^i} \mathrm{d}r \mathrm{d}\mathcal{S} \\
&= \int_{\mathcal{S}} \int_{-\frac{\epsilon}{2}}^{+\frac{\epsilon}{2}} \frac{\partial (G^i_2 + h(r)(G^i_1 - G^i_2))}{\partial x^i} \mathrm{d}r \mathrm{d}\mathcal{S} \\
&= \int_{\mathcal{S}} \int_{-\frac{\epsilon}{2}}^{+\frac{\epsilon}{2}} \left(h(r) \frac{\partial G^i_1}{\partial x^i} + h(-r) \frac{\partial G^i_2}{\partial x^i} + \frac{\partial h(r)}{\partial x^i} (G^i_1 - G^i_2) \right) \mathrm{d}r \mathrm{d}\mathcal{S} \\
&= \int_{\mathcal{V}_1} \frac{\partial G^i_1}{\partial x^i} \mathrm{d}\mathcal{V} + \int_{\mathcal{V}_2} \frac{\partial G^i_2}{\partial x^i} \mathrm{d}\mathcal{V} + \int_{\mathcal{S}} \int_{-\frac{\epsilon}{2}}^{+\frac{\epsilon}{2}} \frac{\partial h(r)}{\partial x^i} (G^i_1 - G^i_2) \mathrm{d}r \mathrm{d}\mathcal{S}.
\end{aligned}$$

With $\frac{\partial h(r)}{\partial x^i} = \delta(r) n^i$, where $\delta(r)$ is the Dirac δ function, we thus have

$$\lim_{\epsilon \to 0} \int_{\mathcal{V}_s} \frac{\partial G^i}{\partial x^i} \mathrm{d}\mathcal{V} = \int_{\mathcal{S}} \int_{-\frac{\epsilon}{2}}^{+\frac{\epsilon}{2}} \delta(r) n^i [G^i] \mathrm{d}r \mathrm{d}\mathcal{S} = \int_{\mathcal{S}} n^i [G^i] \mathrm{d}\mathcal{S},$$

where $[\cdot]$ denotes a jump calculated by $(\cdot)_{\mathcal{S}^+} - (\cdot)_{\mathcal{S}^-}$. Here $\mathcal{S}^+$ represents $\mathcal{S}^+_{\epsilon/2}(\epsilon \to 0)$, and $\mathcal{S}^-$ represents $\mathcal{S}^-_{\epsilon/2}(\epsilon \to 0)$.

$\mathcal{V}_s$ divides $\mathcal{V}$ into three regions: $\mathcal{V}_s$ itself, $\mathcal{V}^+$, and $\mathcal{V}^-$, where $\mathcal{V}^+$ and $\mathcal{S}^+$ are at the same side of $\mathcal{S}$ and $\mathcal{V}^-$ and $\mathcal{S}^-$ are at the same side. Hence, as $\epsilon \to 0$,

$$\begin{aligned}\int_{\mathcal{V}} \frac{\partial G^i}{\partial x^i} \mathrm{d}\mathcal{V} &= \int_{\mathcal{V}^+} \frac{\partial G^i}{\partial x^i} \mathrm{d}\mathcal{V} + \int_{\mathcal{V}^-} \frac{\partial G^i}{\partial x^i} \mathrm{d}\mathcal{V} + \int_{\mathcal{V}_s} \frac{\partial G^i}{\partial x^i} \mathrm{d}\mathcal{V} \\ &= \oint_{\mathcal{A}} G^i N_i \mathrm{d}\mathcal{A} + \int_{\mathcal{S}^+} (-n_i G^i) \mathrm{d}\mathcal{S} + \int_{\mathcal{S}^-} n_i G^i \mathrm{d}\mathcal{S} + \int_{\mathcal{S}} n_i [G^i] \mathrm{d}\mathcal{S} = \oint_{\mathcal{A}} G^i N_i \mathrm{d}\mathcal{A},\end{aligned}$$

which completes the proof for the first situation that G^i has finite jumps at $\mathcal{S}$.

The second situation, in which G^i has a singularity of Dirac δ function type, can be proved by treating the Dirac δ function as a weak limit of a hat function and then using the result from the first situation.

Appendix B. Nonsingularity of C_2. By writing the third row of C_2 twice and then rearranging rows, we expand C_2 to C_2^e as follows:

$$C_2^e = \begin{pmatrix} \tau_1^1\tau_1^1 & \tau_1^1\tau_1^2 + \tau_1^2\tau_1^1 & \tau_1^1\tau_1^3 + \tau_1^3\tau_1^1 & \tau_1^2\tau_1^2 & \tau_1^2\tau_1^3 + \tau_1^3\tau_1^2 & \tau_1^3\tau_1^3 \\ \tau_1^1\tau_2^1 & \tau_1^1\tau_2^2 + \tau_1^2\tau_2^1 & \tau_1^1\tau_2^3 + \tau_1^3\tau_2^1 & \tau_1^2\tau_2^2 & \tau_1^2\tau_2^3 + \tau_1^3\tau_2^2 & \tau_1^3\tau_2^3 \\ \tau_1^1 n^1 & \tau_1^1 n^2 + \tau_1^2 n^1 & \tau_1^1 n^3 + \tau_1^3 n^1 & \tau_1^2 n^2 & \tau_1^2 n^3 + \tau_1^3 n^2 & \tau_1^3 n^3 \\ \tau_1^1\tau_2^1 & \tau_1^1\tau_2^2 + \tau_1^2\tau_2^1 & \tau_1^1\tau_2^3 + \tau_1^3\tau_2^1 & \tau_1^2\tau_2^2 & \tau_1^2\tau_2^3 + \tau_1^3\tau_2^2 & \tau_1^3\tau_2^3 \\ \tau_2^1\tau_2^1 & \tau_2^1\tau_2^2 + \tau_2^2\tau_2^1 & \tau_2^1\tau_2^3 + \tau_2^3\tau_2^1 & \tau_2^2\tau_2^2 & \tau_2^2\tau_2^3 + \tau_2^3\tau_2^2 & \tau_2^3\tau_2^3 \\ \tau_2^1 n^1 & \tau_2^1 n^2 + \tau_2^2 n^1 & \tau_2^1 n^3 + \tau_2^3 n^1 & \tau_2^2 n^2 & \tau_2^2 n^3 + \tau_2^3 n^2 & \tau_2^3 n^3 \\ 1 & 0 & 0 & 1 & 0 & 1 \end{pmatrix}.$$

We can rewrite C_2^e as

$$C_2^e = C_1^e C_2^* = \begin{pmatrix} C_1 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & C_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & 1 \end{pmatrix} \begin{pmatrix} \tau_1^1 & \tau_1^2 & \tau_1^3 & 0 & 0 & 0 \\ 0 & \tau_1^1 & 0 & \tau_1^2 & \tau_1^3 & 0 \\ 0 & 0 & \tau_1^1 & 0 & \tau_1^2 & \tau_1^3 \\ \tau_2^1 & \tau_2^2 & \tau_2^3 & 0 & 0 & 0 \\ 0 & \tau_2^1 & 0 & \tau_2^2 & \tau_2^3 & 0 \\ 0 & 0 & \tau_2^1 & 0 & \tau_2^2 & \tau_2^3 \\ 1 & 0 & 0 & 1 & 0 & 1 \end{pmatrix}.$$

As C_1 is nonsingular, C_1^e is also nonsingular with inverse

$$(C_1^e)^{-1} = \begin{pmatrix} C_1^{-1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & C_1^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & 1 \end{pmatrix}.$$

Thus, if $\operatorname{rank}(C_2^*) = 6$, then C_2 is nonsingular. We present two methods to show $\operatorname{rank}(C_2^*) = 6$.

B.1. Method I. As $\boldsymbol{\tau_1} \neq 0$, one of τ_1^1, τ_1^2, or τ_1^3 must be nonzero. Through row and column permutations, C_2^* can be transformed to one of the following two matrices:

$$\begin{pmatrix} \tau_1^2 & \tau_1^3 & \tau_1^1 & 0 & 0 & 0 \\ 0 & \tau_1^2 & 0 & \tau_1^3 & \tau_1^1 & 0 \\ 0 & 0 & \tau_1^2 & 0 & \tau_1^3 & \tau_1^1 \\ \tau_2^2 & \tau_2^3 & \tau_2^1 & 0 & 0 & 0 \\ 0 & \tau_2^2 & 0 & \tau_2^3 & \tau_2^1 & 0 \\ 0 & 0 & \tau_2^2 & 0 & \tau_2^3 & \tau_2^1 \\ 1 & 0 & 0 & 1 & 0 & 1 \end{pmatrix}, \qquad \begin{pmatrix} \tau_1^3 & \tau_1^2 & \tau_1^1 & 0 & 0 & 0 \\ 0 & \tau_1^3 & 0 & \tau_1^2 & \tau_1^1 & 0 \\ 0 & 0 & \tau_1^3 & 0 & \tau_1^2 & \tau_1^1 \\ \tau_2^3 & \tau_2^2 & \tau_2^1 & 0 & 0 & 0 \\ 0 & \tau_2^3 & 0 & \tau_2^2 & \tau_2^1 & 0 \\ 0 & 0 & \tau_2^3 & 0 & \tau_2^2 & \tau_2^1 \\ 1 & 0 & 0 & 1 & 0 & 1 \end{pmatrix}.$$

Both have the same structure and similar element ordering as C_2^*. Therefore, we suppose $\tau_1^1 \neq 0$ and need to work on only this case.

Using elementary operations, we can transform C_2^* to C_2^{**} as

$$C_2^* \to C_2^{**} = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & \tau_1^1 & 0 & \tau_1^2 & \tau_1^3 & 0 \\ 0 & 0 & \tau_1^1 & 0 & \tau_1^2 & \tau_1^3 \\ 0 & 0 & 0 & -s_3 & -2\tau_1^2\tau_1^3 & -s_2 \\ 0 & 0 & 0 & 0 & e & f \\ 0 & 0 & 0 & 0 & g & h \\ 0 & 0 & 0 & 0 & N^3 & -N^2 \end{pmatrix},$$

with the following definitions:

$$\begin{aligned} s_2 &= \tau_1^1\tau_1^1 + \tau_1^3\tau_1^3, \\ s_3 &= \tau_1^1\tau_1^1 + \tau_1^2\tau_1^2, \\ N^2 &= \tau_1^3\tau_2^1 - \tau_2^3\tau_1^1, \\ N^3 &= \tau_1^1\tau_2^2 - \tau_2^1\tau_1^2, \\ e &= (\tau_2^2\tau_1^3 + \tau_1^2\tau_2^3)s_3 - 2\tau_1^2\tau_1^3 d_3, \\ f &= d_2 s_3 - d_3 s_2, \\ g &= N^2 s_3 + 2\tau_1^2\tau_1^3 N^3, \\ h &= s_2 N^3, \end{aligned}$$

where

$$\begin{aligned} d_2 &= \tau_1^1\tau_2^1 + \tau_1^3\tau_2^3, \\ d_3 &= \tau_1^1\tau_2^1 + \tau_1^2\tau_2^2. \end{aligned}$$

If $N^2 = N^3 = 0$, then as $\tau_1^1 \neq 0$,

$$\boldsymbol{\tau_1} \cdot (\boldsymbol{\tau_1} \times \boldsymbol{\tau_2}) = \tau_1^1 N^1 + \tau_1^2 N^2 + \tau_1^3 N^3 = 0 \Rightarrow N^1 = 0 \Rightarrow \boldsymbol{\tau_1} \times \boldsymbol{\tau_2} = 0,$$

where $N^1 = \tau_1^2\tau_2^3 - \tau_2^2\tau_1^3$. This is impossible. Thus, one of N^2 and N^3 must be nonzero. Suppose $N^3 \neq 0$. By elementary operations, we can transform C_2^{**} to

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & \tau_1^1 & 0 & \tau_1^2 & \tau_1^3 & 0 \\ 0 & 0 & \tau_1^1 & 0 & \tau_1^2 & \tau_1^3 \\ 0 & 0 & 0 & -s_3 & -2\tau_1^2\tau_1^3 & -s_2 \\ 0 & 0 & 0 & 0 & N^3 & -N^2 \\ 0 & 0 & 0 & 0 & 0 & N^3 f + N^2 e \\ 0 & 0 & 0 & 0 & 0 & N^3 h + N^2 g \end{pmatrix}.$$

Since

$$N^3 h + N^2 g = \tau_1^1\tau_1^1 \|\boldsymbol{\tau_1} \times \boldsymbol{\tau_2}\|^2 \neq 0,$$

we can conclude that $\text{rank}(C_2^{**}) \geq 6$. Supposing $N^2 \neq 0$, we can similarly show $\text{rank}(C_2^{**}) \geq 6$. C_2^e is expanded from C_2, and thus $\text{rank}(C_2^e) \leq 6$. Therefore, $\text{rank}(C_2^e) = \text{rank}(C_2^*) = \text{rank}(C_2^{**}) = 6$. The proof is completed. This proof gives a way to solve (5.8).

B.2. Method II. As $\tau_\mathbf{1} \times \tau_\mathbf{2} \neq 0$, there exist matrices P_r and P_c such that

$$P_r \begin{pmatrix} \tau_1^1 & \tau_1^2 & \tau_1^3 \\ \tau_2^1 & \tau_2^2 & \tau_2^3 \end{pmatrix} P_c = T,$$

where P_r is a 2×2 elementary matrix, P_c is a 3×3 elementary matrix without permutation operations, and $T = \begin{pmatrix} e_t^1 & e_t^2 & e_t^3 \\ e_b^1 & e_b^2 & e_b^3 \end{pmatrix}$ is one of

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}, \quad \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Construct elementary matrices E_r and E_c as

$$E_r = \begin{pmatrix} P_r & \mathbf{0} \\ \mathbf{0} & I_{5\times 5} \end{pmatrix}, \quad E_c = \begin{pmatrix} P_c & \mathbf{0} \\ \mathbf{0} & I_{3\times 3} \end{pmatrix},$$

where $I_{n\times n}$ denotes an $n \times n$ unit matrix. With a series of actions by permutations and E_r and E_c, C_2^* can be transformed as follows:

$$\begin{aligned} C_2^* \to E_r &\begin{pmatrix} \tau_1^1 & \tau_1^2 & \tau_1^3 & 0 & 0 & 0 \\ \tau_2^1 & \tau_2^2 & \tau_2^3 & 0 & 0 & 0 \\ 0 & \tau_1^1 & 0 & \tau_1^2 & \tau_1^3 & 0 \\ 0 & \tau_2^1 & 0 & \tau_2^2 & \tau_2^3 & 0 \\ 0 & 0 & \tau_1^1 & 0 & \tau_1^2 & \tau_1^3 \\ 0 & 0 & \tau_2^1 & 0 & \tau_2^2 & \tau_2^3 \\ 1 & 0 & 0 & 1 & 0 & 1 \end{pmatrix} E_c = \begin{pmatrix} e_t^1 & e_t^2 & e_t^3 & 0 & 0 & 0 \\ e_b^1 & e_b^2 & e_b^3 & 0 & 0 & 0 \\ 0 & \tau_1^1 & 0 & \tau_1^2 & \tau_1^3 & 0 \\ 0 & \tau_2^1 & 0 & \tau_2^2 & \tau_2^3 & 0 \\ 0 & 0 & \tau_1^1 & 0 & \tau_1^2 & \tau_1^3 \\ 0 & 0 & \tau_2^1 & 0 & \tau_2^2 & \tau_2^3 \\ 1 & 0 & 0 & 1 & 0 & 1 \end{pmatrix} \\ \to E_r &\begin{pmatrix} \tau_1^1 & \tau_1^2 & \tau_1^3 & 0 & 0 & 0 \\ \tau_2^1 & \tau_2^2 & \tau_2^3 & 0 & 0 & 0 \\ e_t^2 & 0 & 0 & e_t^1 & e_t^3 & 0 \\ e_b^2 & 0 & 0 & e_b^1 & e_b^3 & 0 \\ 0 & 0 & \tau_1^2 & 0 & \tau_1^1 & \tau_1^3 \\ 0 & 0 & \tau_2^2 & 0 & \tau_2^1 & \tau_2^3 \\ 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix} E_b = \begin{pmatrix} e_t^1 & e_t^2 & e_t^3 & 0 & 0 & 0 \\ e_b^1 & e_b^2 & e_b^3 & 0 & 0 & 0 \\ e_t^2 & 0 & 0 & e_t^1 & e_t^3 & 0 \\ e_b^2 & 0 & 0 & e_b^1 & e_b^3 & 0 \\ 0 & 0 & \tau_1^2 & 0 & \tau_1^1 & \tau_1^3 \\ 0 & 0 & \tau_2^2 & 0 & \tau_2^1 & \tau_2^3 \\ 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix} \\ \to E_r &\begin{pmatrix} \tau_1^1 & \tau_1^2 & \tau_1^3 & 0 & 0 & 0 \\ \tau_2^1 & \tau_2^2 & \tau_2^3 & 0 & 0 & 0 \\ 0 & e_t^3 & 0 & e_t^1 & e_t^2 & 0 \\ 0 & e_b^3 & 0 & e_b^1 & e_b^2 & 0 \\ e_t^3 & 0 & 0 & e_t^2 & 0 & e_t^1 \\ e_b^3 & 0 & 0 & e_b^2 & 0 & e_b^1 \\ 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix} E_c = \begin{pmatrix} e_t^1 & e_t^2 & e_t^3 & 0 & 0 & 0 \\ e_b^1 & e_b^2 & e_b^3 & 0 & 0 & 0 \\ 0 & e_t^3 & 0 & e_t^1 & e_t^2 & 0 \\ 0 & e_b^3 & 0 & e_b^1 & e_b^2 & 0 \\ e_t^3 & 0 & 0 & e_t^2 & 0 & e_t^1 \\ e_b^3 & 0 & 0 & e_b^2 & 0 & e_b^1 \\ 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix}. \end{aligned}$$

As matrix T takes different choices, the last matrix above correspondingly takes

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix}, \quad \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix}, \quad \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix}.$$

Thus, $\text{rank}(C_2^*) = 6$.

Appendix C. rank(C_3) = 10. Expand C_3 to C_3^e as

$$C_3^e = \begin{pmatrix} C_2^{11} & C_2^{12} & C_2^{13} & C_2^{14} & C_2^{15} & C_2^{16} & 0 & 0 & 0 & 0 \\ C_2^{31} & C_2^{32} & C_2^{33} & C_2^{34} & C_2^{35} & C_2^{36} & 0 & 0 & 0 & 0 \\ C_2^{31} & C_2^{32} & C_2^{33} & C_2^{34} & C_2^{35} & C_2^{36} & 0 & 0 & 0 & 0 \\ C_2^{21} & C_2^{22} & C_2^{23} & C_2^{24} & C_2^{25} & C_2^{26} & 0 & 0 & 0 & 0 \\ 0 & C_2^{11} & 0 & C_2^{12} & C_2^{13} & 0 & C_2^{14} & C_2^{15} & C_2^{16} & 0 \\ 0 & C_2^{31} & 0 & C_2^{32} & C_2^{33} & 0 & C_2^{34} & C_2^{35} & C_2^{36} & 0 \\ 0 & C_2^{31} & 0 & C_2^{32} & C_2^{33} & 0 & C_2^{34} & C_2^{35} & C_2^{36} & 0 \\ 0 & C_2^{21} & 0 & C_2^{22} & C_2^{23} & 0 & C_2^{24} & C_2^{25} & C_2^{26} & 0 \\ 0 & 0 & C_2^{11} & 0 & C_2^{12} & C_2^{13} & 0 & C_2^{14} & C_2^{15} & C_2^{16} \\ 0 & 0 & C_2^{31} & 0 & C_2^{32} & C_2^{33} & 0 & C_2^{34} & C_2^{35} & C_2^{36} \\ 0 & 0 & C_2^{31} & 0 & C_2^{32} & C_2^{33} & 0 & C_2^{34} & C_2^{35} & C_2^{36} \\ 0 & 0 & C_2^{21} & 0 & C_2^{22} & C_2^{23} & 0 & C_2^{24} & C_2^{25} & C_2^{26} \\ 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \end{pmatrix}.$$

If $\text{rank}(C_3^e) = 10$, then $\text{rank}(C_3) = 10$.

Define a corner block

$$C_3^c := \begin{pmatrix} C_2^{11} & C_2^{12} & C_2^{13} & C_2^{14} & C_2^{15} & C_2^{16} \\ C_2^{31} & C_2^{32} & C_2^{33} & C_2^{34} & C_2^{35} & C_2^{36} \\ C_2^{31} & C_2^{32} & C_2^{33} & C_2^{34} & C_2^{35} & C_2^{36} \\ C_2^{21} & C_2^{22} & C_2^{23} & C_2^{24} & C_2^{25} & C_2^{26} \end{pmatrix},$$

and decompose C_3^c as

$$C_3^c = \begin{pmatrix} \tau_1^1\tau_1^1 & \tau_1^1\tau_1^2 + \tau_1^2\tau_1^1 & \tau_1^1\tau_1^3 + \tau_1^3\tau_1^1 & \tau_1^2\tau_1^2 & \tau_1^2\tau_1^3 + \tau_1^3\tau_1^2 & \tau_1^3\tau_1^3 \\ \tau_1^1\tau_2^1 & \tau_1^1\tau_2^2 + \tau_1^2\tau_2^1 & \tau_1^1\tau_2^3 + \tau_1^3\tau_2^1 & \tau_1^2\tau_2^2 & \tau_1^2\tau_2^3 + \tau_1^3\tau_2^2 & \tau_1^3\tau_2^3 \\ \tau_1^1\tau_2^1 & \tau_1^1\tau_2^2 + \tau_1^2\tau_2^1 & \tau_1^1\tau_2^3 + \tau_1^3\tau_2^1 & \tau_1^2\tau_2^2 & \tau_1^2\tau_2^3 + \tau_1^3\tau_2^2 & \tau_1^3\tau_2^3 \\ \tau_2^1\tau_2^1 & \tau_2^1\tau_2^2 + \tau_2^2\tau_2^1 & \tau_2^1\tau_2^3 + \tau_2^3\tau_2^1 & \tau_2^2\tau_2^2 & \tau_2^2\tau_2^3 + \tau_2^3\tau_2^2 & \tau_2^3\tau_2^3 \end{pmatrix}$$

$$:= L_1 R_1 = \begin{pmatrix} \tau_1^1 & 0 & \tau_1^2 & 0 & \tau_1^3 & 0 \\ \tau_2^1 & 0 & \tau_2^2 & 0 & \tau_2^3 & 0 \\ 0 & \tau_1^1 & 0 & \tau_1^2 & 0 & \tau_1^3 \\ 0 & \tau_2^1 & 0 & \tau_2^2 & 0 & \tau_2^3 \end{pmatrix} \begin{pmatrix} \tau_1^1 & \tau_1^2 & \tau_1^3 & 0 & 0 & 0 \\ \tau_2^1 & \tau_2^2 & \tau_2^3 & 0 & 0 & 0 \\ 0 & \tau_1^1 & 0 & \tau_1^2 & \tau_1^3 & 0 \\ 0 & \tau_2^1 & 0 & \tau_2^2 & \tau_2^3 & 0 \\ 0 & 0 & \tau_1^1 & 0 & \tau_1^2 & \tau_1^3 \\ 0 & 0 & \tau_2^1 & 0 & \tau_2^2 & \tau_2^3 \end{pmatrix}.$$

Through permutations, we can transform C_3^c as

$$C_3^c \to L_1^* R_1^* := \begin{pmatrix} \tau_1^2 & 0 & \tau_1^3 & 0 & \tau_1^1 & 0 \\ \tau_2^2 & 0 & \tau_2^3 & 0 & \tau_2^1 & 0 \\ 0 & \tau_1^2 & 0 & \tau_1^3 & 0 & \tau_1^1 \\ 0 & \tau_2^2 & 0 & \tau_2^3 & 0 & \tau_2^1 \end{pmatrix} \begin{pmatrix} \tau_1^2 & \tau_1^3 & \tau_1^1 & 0 & 0 & 0 \\ \tau_2^2 & \tau_2^3 & \tau_2^1 & 0 & 0 & 0 \\ 0 & \tau_1^2 & 0 & \tau_1^3 & \tau_1^1 & 0 \\ 0 & \tau_2^2 & 0 & \tau_2^3 & \tau_2^1 & 0 \\ 0 & 0 & \tau_1^2 & 0 & \tau_1^3 & \tau_1^1 \\ 0 & 0 & \tau_2^2 & 0 & \tau_2^3 & \tau_2^1 \end{pmatrix}$$

or

$$C_3^c \to L_1^{**}R_1^{**} := \begin{pmatrix} \tau_1^3 & 0 & \tau_1^2 & 0 & \tau_1^1 & 0 \\ \tau_2^3 & 0 & \tau_2^2 & 0 & \tau_2^1 & 0 \\ 0 & \tau_1^3 & 0 & \tau_1^2 & 0 & \tau_1^1 \\ 0 & \tau_2^3 & 0 & \tau_2^2 & 0 & \tau_2^1 \end{pmatrix} \begin{pmatrix} \tau_1^3 & \tau_1^2 & \tau_1^1 & 0 & 0 & 0 \\ \tau_2^3 & \tau_2^2 & \tau_2^1 & 0 & 0 & 0 \\ 0 & \tau_1^3 & 0 & \tau_1^2 & \tau_1^1 & 0 \\ 0 & \tau_2^3 & 0 & \tau_2^2 & \tau_2^1 & 0 \\ 0 & 0 & \tau_1^3 & 0 & \tau_1^2 & \tau_1^1 \\ 0 & 0 & \tau_2^3 & 0 & \tau_2^2 & \tau_2^1 \end{pmatrix}.$$

Since L_1, L_1^*, and L_1^{**} have the same structure and similar element ordering, and so do R_1, R_1^*, and R_1^{**}, we suppose $\tau_1^1 \neq 0$ and need to work on only this case.

As $\tau_1^1 \neq 0$ and $\boldsymbol{\tau_1} \times \boldsymbol{\tau_2} \neq 0$, there exists matrix P such that

$$\begin{pmatrix} \tau_1^1 & \tau_1^2 & \tau_1^3 \\ \tau_2^1 & \tau_2^2 & \tau_2^3 \end{pmatrix} P = T,$$

where P is a 3×3 elementary matrix without permutation operations and $T = \begin{pmatrix} e_t^1 & e_t^2 & e_t^3 \\ e_b^1 & e_b^2 & e_b^3 \end{pmatrix}$ is one of

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}, \quad \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Construct elementary matrix E as

$$E = \begin{pmatrix} P & \mathbf{0} \\ \mathbf{0} & I_{3\times 3} \end{pmatrix}.$$

With a series of actions by permutations and E, C_3^c can be transformed as follows:

$$C_3^c = L_1R_1 \to L_1R_1E = L_1 \begin{pmatrix} e_t^1 & e_t^2 & e_t^3 & 0 & 0 & 0 \\ e_b^1 & e_b^2 & e_b^3 & 0 & 0 & 0 \\ 0 & \tau_1^1 & 0 & \tau_1^2 & \tau_1^3 & 0 \\ 0 & \tau_2^1 & 0 & \tau_2^2 & \tau_2^3 & 0 \\ 0 & 0 & \tau_1^1 & 0 & \tau_1^2 & \tau_1^3 \\ 0 & 0 & \tau_2^1 & 0 & \tau_2^2 & \tau_2^3 \end{pmatrix}$$

$$\to \begin{pmatrix} \tau_1^2 & 0 & \tau_1^1 & 0 & \tau_1^3 & 0 \\ \tau_2^2 & 0 & \tau_2^1 & 0 & \tau_2^3 & 0 \\ 0 & \tau_1^2 & 0 & \tau_1^1 & 0 & \tau_1^3 \\ 0 & \tau_2^2 & 0 & \tau_2^1 & 0 & \tau_2^3 \end{pmatrix} \begin{pmatrix} \tau_1^1 & \tau_1^2 & \tau_1^3 & 0 & 0 & 0 \\ \tau_2^1 & \tau_2^2 & \tau_2^3 & 0 & 0 & 0 \\ e_t^2 & 0 & 0 & e_t^1 & e_t^3 & 0 \\ e_b^2 & 0 & 0 & e_b^1 & e_b^3 & 0 \\ 0 & 0 & \tau_1^2 & 0 & \tau_1^1 & \tau_1^3 \\ 0 & 0 & \tau_2^2 & 0 & \tau_2^1 & \tau_2^3 \end{pmatrix}$$

$$:= L_2R_2 \to L_2R_2E = L_2 \begin{pmatrix} e_t^1 & e_t^2 & e_t^3 & 0 & 0 & 0 \\ e_b^1 & e_b^2 & e_b^3 & 0 & 0 & 0 \\ e_t^2 & 0 & 0 & e_t^1 & e_t^3 & 0 \\ e_b^2 & 0 & 0 & e_b^1 & e_b^3 & 0 \\ 0 & 0 & \tau_1^2 & 0 & \tau_1^1 & \tau_1^3 \\ 0 & 0 & \tau_2^2 & 0 & \tau_2^1 & \tau_2^3 \end{pmatrix}$$

$$\to \begin{pmatrix} \tau_1^3 & 0 & \tau_1^1 & 0 & \tau_1^2 & 0 \\ \tau_2^3 & 0 & \tau_2^1 & 0 & \tau_2^2 & 0 \\ 0 & \tau_1^3 & 0 & \tau_1^1 & 0 & \tau_1^2 \\ 0 & \tau_2^3 & 0 & \tau_2^1 & 0 & \tau_2^2 \end{pmatrix} \begin{pmatrix} \tau_1^1 & \tau_1^2 & \tau_1^3 & 0 & 0 & 0 \\ \tau_2^1 & \tau_2^2 & \tau_2^3 & 0 & 0 & 0 \\ e_t^3 & 0 & 0 & e_t^2 & 0 & e_t^1 \\ e_b^3 & 0 & 0 & e_b^2 & 0 & e_b^1 \\ 0 & e_t^3 & 0 & e_t^1 & e_t^2 & 0 \\ 0 & e_b^3 & 0 & e_b^1 & e_b^2 & 0 \end{pmatrix}$$

$$:= L_3R_3 \to L_3R_3E = L_3 \begin{pmatrix} e_t^1 & e_t^2 & e_t^3 & 0 & 0 & 0 \\ e_b^1 & e_b^2 & e_b^3 & 0 & 0 & 0 \\ e_t^3 & 0 & 0 & e_t^2 & 0 & e_t^1 \\ e_b^3 & 0 & 0 & e_b^2 & 0 & e_b^1 \\ 0 & e_t^3 & 0 & e_t^1 & e_t^2 & 0 \\ 0 & e_b^3 & 0 & e_b^1 & e_b^2 & 0 \end{pmatrix} := L_4R_4.$$

When

$$T = \begin{pmatrix} e_t^1 & e_t^2 & e_t^3 \\ e_b^1 & e_b^2 & e_b^3 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix},$$

we continue to transform C_3^c as

$$\begin{aligned} L_4R_4 &= \begin{pmatrix} \tau_1^3 & 0 & 0 & \tau_1^2 & 0 & \tau_1^1 \\ \tau_2^3 & 0 & 0 & \tau_2^2 & 0 & \tau_2^1 \\ 0 & \tau_1^3 & 0 & \tau_1^1 & \tau_1^2 & 0 \\ 0 & \tau_2^3 & 0 & \tau_2^1 & \tau_2^2 & 0 \end{pmatrix} \to \begin{pmatrix} \tau_1^1 & \tau_1^2 & \tau_1^3 & 0 & 0 & 0 \\ \tau_2^1 & \tau_2^2 & \tau_2^3 & 0 & 0 & 0 \\ 0 & \tau_1^1 & 0 & \tau_1^2 & \tau_1^3 & 0 \\ 0 & \tau_2^1 & 0 & \tau_2^2 & \tau_2^3 & 0 \end{pmatrix} E \\ &= \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & \tau_1^1 & 0 & \tau_1^2 & \tau_1^3 & 0 \\ 0 & \tau_2^1 & 0 & \tau_2^2 & \tau_2^3 & 0 \end{pmatrix} \to \begin{pmatrix} \tau_1^1 & \tau_1^2 & \tau_1^3 & 0 & 0 & 0 \\ \tau_2^1 & \tau_2^2 & \tau_2^3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} E \\ &= \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \to \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}. \end{aligned}$$

When

$$T = \begin{pmatrix} e_t^1 & e_t^2 & e_t^3 \\ e_b^1 & e_b^2 & e_b^3 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix},$$

we continue to transform C_3^c as

$$\begin{aligned} L_4R_4 &= \begin{pmatrix} \tau_1^3 & 0 & 0 & \tau_1^2 & 0 & \tau_1^1 \\ \tau_2^3 & 0 & 0 & \tau_2^2 & 0 & \tau_2^1 \\ \tau_1^1 & \tau_1^2 & \tau_1^3 & 0 & 0 & 0 \\ \tau_2^1 & \tau_2^2 & \tau_2^3 & 0 & 0 & 0 \end{pmatrix} \to \begin{pmatrix} \tau_1^1 & \tau_1^2 & \tau_1^3 & 0 & 0 & 0 \\ \tau_2^1 & \tau_2^2 & \tau_2^3 & 0 & 0 & 0 \\ 0 & 0 & \tau_1^1 & \tau_1^2 & \tau_1^3 & 0 \\ 0 & 0 & \tau_2^1 & \tau_2^2 & \tau_2^3 & 0 \end{pmatrix} E \\ &= \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & \tau_1^1 & \tau_1^2 & \tau_1^3 & 0 \\ 0 & 0 & \tau_2^1 & \tau_2^2 & \tau_2^3 & 0 \end{pmatrix} \to \begin{pmatrix} \tau_1^1 & \tau_1^2 & \tau_1^3 & 0 & 0 & 0 \\ \tau_2^1 & \tau_2^2 & \tau_2^3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} E \\ &= \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \to \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}. \end{aligned}$$

Combining all the row-operating elementary matrices as a 4×4 elementary matrix E_r^c and all the column-operating elementary matrices as a 6×6 elementary matrix E_c^c, we can write transformations to C_3^c as

$$C_3^c \to E_r^c C_3^c E_c^c = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & \gamma^1 & \gamma^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix},$$

where (γ^1, γ^2) equals $(1, 0)$ or $(0, 1)$.

Construct elementary matrices Q_r and Q_c as

$$Q_r = \begin{pmatrix} E_r^c & \mathbf{0} \\ \mathbf{0} & I_{11\times 11} \end{pmatrix}, \quad Q_c = \begin{pmatrix} E_c^c & \mathbf{0} \\ \mathbf{0} & I_{4\times 4} \end{pmatrix}.$$

With a series of actions by permutations, Q_r and Q_c, C_3^e can be transformed as follows:

$$C_3^e \to Q_r C_3^e Q_c = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \gamma^1 & \gamma^2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & C_2^{11} & 0 & C_2^{12} & C_2^{13} & 0 & C_2^{14} & C_2^{15} & C_2^{16} & 0 \\ 0 & C_2^{31} & 0 & C_2^{32} & C_2^{33} & 0 & C_2^{34} & C_2^{35} & C_2^{36} & 0 \\ 0 & C_2^{31} & 0 & C_2^{32} & C_2^{33} & 0 & C_2^{34} & C_2^{35} & C_2^{36} & 0 \\ 0 & C_2^{21} & 0 & C_2^{22} & C_2^{23} & 0 & C_2^{24} & C_2^{25} & C_2^{26} & 0 \\ 0 & 0 & C_2^{11} & 0 & C_2^{12} & C_2^{13} & 0 & C_2^{14} & C_2^{15} & C_2^{16} \\ 0 & 0 & C_2^{31} & 0 & C_2^{32} & C_2^{33} & 0 & C_2^{34} & C_2^{35} & C_2^{36} \\ 0 & 0 & C_2^{31} & 0 & C_2^{32} & C_2^{33} & 0 & C_2^{34} & C_2^{35} & C_2^{36} \\ 0 & 0 & C_2^{21} & 0 & C_2^{22} & C_2^{23} & 0 & C_2^{24} & C_2^{25} & C_2^{26} \\ 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \end{pmatrix}$$

$$\to \cdots \to \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \gamma^1 & \gamma^2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & \gamma^2 & 0 & 0 & 0 & 0 & 0 & \gamma^1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ \gamma^2 & 0 & 0 & 0 & 0 & 0 & 0 & \gamma^1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

When (γ^1, γ^2) equals $(1, 0)$ or $(0, 1)$, it can be seen that the rank of the last matrix above is 10. Thus $\text{rank}(C_3) = \text{rank}(C_3^e) = 10$.

REFERENCES

[1] K. M. Arthurs, L. C. Moore, C. S. Peskin, E. B. Pitman, and H. E. Layton, *Modeling arteriolar flow and mass transport using the immersed boundary method*, J. Comput. Phys., 147 (1998), pp. 402–440.

[2] R. P. Beyer and R. J. Leveque, *Analysis of a one-dimensional model for the immersed boundary method*, SIAM J. Numer. Anal., 29 (1992), pp. 332–364.

[3] J. M. Birch and M. H. Dickinson, *Spanwise flow and the attachment of the leading-edge vortex on insect wings*, Nature, 412 (2001), pp. 729–733.

[4] D. M. Bushnell, J. N. Hefner, and R. L. Ash, *Effects of compliant wall motion on turbulent boundary layers*, Phys. Fluids A, 20 (1977), pp. s31–48.

[5] D. Calhoun, *A Cartesian grid method for solving the two-dimensional streamfunction-vorticity equations in irregular regions*, J. Comput. Phys., 176 (2002), pp. 231–275.

[6] P. W. Carpenter, C. Davies, and A. D. Lucey, *Hydrodynamics and compliant walls: Does the dolphin have a secret?*, Current Science, 79 (2000), pp. 758–765.

[7] R. Cortez and M. Minion, *The blob projection method for immersed boundary problems*, J. Comput. Phys., 161 (2000), pp. 428–453.

[8] M. H. Dickinson, F. O. Lehmann, and S. P. Sane, *Wing rotation and the aerodynamic basis of insect flight*, Science, 284 (1999), pp. 1954–1960.

[9] C. P. Ellington, C. van den Berg, A. P. Willmott, and A. L. R. Thomas, *Leading-edge vortices in insect flight*, Nature, 384 (1996), pp. 626–630.

[10] L. J. Fauci and C. S. Peskin, *A computational model of aquatic animal locomotion*, J. Comput. Phys., 77 (1988), pp. 85–108.

[11] A. L. Fogelson and C. S. Peskin, *A fast numerical method for solving the three-dimensional Stokes' equations in the presence of suspended particles*, J. Comput. Phys., 79 (1988), pp. 50–69.

[12] A. L. Fogelson and J. P. Keener, *Immersed interface methods for Neumann and related problems in two and three dimensions*, SIAM J. Sci. Comput., 22 (2000), pp. 1630–1654.

[13] S. N. Fry, R. Sayaman, and M. H. Dickinson, *The aerodynamics of free-flight maneuvers in Drosophila*, Science, 300 (2003), pp. 495–498.

[14] M. Gad-El-Hak, *Compliant coatings: The simpler alternative*, Experimental Thermal and Fluid Science, 16 (1998), pp. 141–156.

[15] M. O. Kramer, *The dolphin's secret*, Journal of American Society for Naval Engineering, 73 (1961), pp. 103–107.

[16] M.-C. Lai and C. S. Peskin, *An immersed boundary method with formal second-order accuracy and reduced numerical viscosity*, J. Comput. Phys., 160 (2000), pp. 705–719.

[17] M.-C. Lai and Z. Li, *A remark on jump conditions for the three-dimensional Navier–Stokes equations involving an immersed moving membrane*, Appl. Math. Lett., 14 (2001), pp. 149–154.

[18] R. J. LeVeque and Z. Li, *The immersed interface method for elliptic equations with discontinuous coefficients and singular sources*, SIAM J. Numer. Anal., 31 (1994), pp. 1019–1044.

[19] R. J. LeVeque and Z. Li, *Immersed interface methods for Stokes flow with elastic boundaries or surface tension*, SIAM J. Sci. Comput., 18 (1997), pp. 709–735.

[20] Z. Li and M.-C. Lai, *The immersed interface method for the Navier–Stokes equations with singular forces*, J. Comput. Phys., 171 (2001), pp. 822–842.

[21] A. Mayo, *The fast solution of Poisson's and the biharmonic equations on irregular regions*, SIAM J. Numer. Anal., 21 (1984), pp. 285–299.

[22] A. McKenney, L. Greengard, and A. Mayo, *A fast Poisson solver for complex geometries*, J. Comput. Phys., 118 (1995), pp. 348–355.

[23] C. S. Peskin, *Flow patterns around heart valves: A numerical method*, J. Comput. Phys., 10 (1972), pp. 252–271.

[24] C. S. Peskin, *Numerical analysis of blood flow in the heart*, J. Comput. Phys., 25 (1977), pp. 220–252.

[25] C. S. Peskin and D. M. McQueen, *A three-dimensional computational method for blood flow in the heart* I. *Immersed elastic fibers in a viscous incompressible fluid*, J. Comput. Phys., 81 (1989), pp. 372–405.

[26] C. S. Peskin and B. F. Printz, *Improved volume conservation in the computation of flows with immersed elastic boundaries*, J. Comput. Phys., 105 (1993), pp. 33–46.

[27] A. M. Roma, C. S. Peskin, and M. J. Berger, *An adaptive version of the immersed boundary method*, J. Comput. Phys., 153 (1999), pp. 509–534.

[28] D. Russell and Z. J. Wang, *A Cartesian grid method for modeling multiple moving objects in 2D incompressible viscous flow*, J. Comput. Phys., 191 (2003), pp. 177–205.

[29] C. Tu and C. S. Peskin, *Stability and instability in the computation of flows with moving immersed boundaries: A comparison of three methods*, SIAM J. Statist. Comput., 13 (1992), pp. 1361–1376.

[30] 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.

[31] Z. J. Wang, *Vortex shedding and frequency selection in flapping flight*, J. Fluid Mech., 410 (2000), pp. 323–341.

[32] Z. J. Wang, *Two dimensional mechanism for insect hovering*, Phys. Rev. Lett., 85 (2000), pp. 2216–2219.

[33] A. Wiegmann and K. P. Bube, *The immersed interface method for nonlinear differential equations with discontinuous coefficients and singular sources*, SIAM J. Numer. Anal., 35 (1998), pp. 177–200.

[34] A. Wiegmann and K. P. Bube, *The explicit-jump immersed interface method: Finite difference methods for PDEs with piecewise smooth solutions*, SIAM J. Numer. Anal., 37 (2000), pp. 827–862.

[35] S. Xu, D. Rempfer, and J. Lumley, *Turbulence over a compliant surface: Numerical simulation and analysis*, J. Fluid Mech., 478 (2003), pp. 11–34.

[36] S. Xu and Z. J. Wang, *An immersed interface method for simulating the interaction of a fluid with moving boundaries*, J. Comput. Phys., to appear.

[37] L. Zhu and C. S. Peskin, *Simulation of a flapping flexible filament in a flowing soap film by the immersed boundary method*, J. Comput. Phys., 179 (2002), pp. 452–468.