On Various Techniques for Computer Simulation of Boundaries with Mass

Yongsam Kim †, Luoding Zhu ‡, Xiaodong Wang *, and Charles Peskin †

Abstract

In this short paper, we present three distinct numerical treatments for immersed boundaries with different mass density than the surrounding fluid. We start with some background information on the immersed boundary method and its application in the modeling of biological fluids. The selected numerical example demonstrates the potential in modeling complex biological fluids involving charged particles under both viscous fluid and electrokinetic forces.

† Courant Institute of Mathematical Sciences, New York University, 251 Mercer Street, New York, NY 10012

‡ Department of Computer Science, University of California at Santa Barbara, Santa Barbara, CA 93106

* Department of Mechanical Engineering, Polytechnic University, Six MetroTech Center, Brooklyn, NY 11201, the corresponding author, xwang@poly.edu

Submitted: 2nd MIT Conference on Computational Fluid and Solid Mechanics

Keywords: Biological Fluid; Immersed Boundary Method; Computer Modeling; Computational Mechanics; Particulate flow.

1 Introduction

The coupling of fluids and solids is the central feature in the study of the mechanics of the heart, arteries, veins, microcirculation, and pulmonary blood flow. Nevertheless, the modeling of strong hemodynamic interaction with flexible structures is often limited by severe fluid mesh distortions around structures experiencing large deformations. These computations often require adaptive meshing which is impractical in the modeling of deformable cells interacting with viscous fluid and flexible vessels.

The immersed boundary method was originally developed by Peskin [5] for the computation of blood flows interacting with the heart and heart valves. It has since then been extended to three-dimensional heart-flow interactions and a variety of other biomechanics problems, which included the design of prosthetic cardiac valves (McQueen and Peskin, 1985 [4]), swimming motions of marine worms (Fauci and Peskin, 1988 [2]), wood pulp fiber dynamics (Stockie and Wetton, 1999 [9]), wave propagation in cochlea (Beyer, 1992 [8]), and biofilm processes (Fogelson *et al.*, 1996 [1]). Recently, the extended immersed boundary method (EIBM) was also proposed to model the *entire* submerged elastic structure as a collection of nodal points connected through a regular finite element method (FEM) or meshless reproducing kernel particle method (RKPM) (Wang and Liu, 2002 [10]).

2 Problem Description

Consider an immersed flexible structure contained in a viscous incompressible fluid with periodic boundary conditions. The immersed elastic structure is assumed to have a density ρ that is the same as the fluid.

Let $\mathbf{F}(\mathbf{r}, t)$ and $\mathbf{f}(\mathbf{x}, t)$ stand for the elastic fiber point force and the corresponding effective body force in the fluid domain at time t where the submerged point and the fluid domain are represented by a Lagrangian parametric coordinate $\mathbf{r} = (r_1, r_2, r_3)$ and the Eulerian coordinate $\mathbf{x}$, respectively. Denote Ω as the domain of influence or support attached to every immersed elastic nodal point $\mathbf{X}(\mathbf{r}, t)$, along a single smooth submerged elastic fiber Γ_s, the elastic fiber point force $\mathbf{F}(\mathbf{r}, t)$ is evaluated as

$$\mathbf{F}(\mathbf{r}, t) = \frac{\partial (T\mathbf{t})}{\partial s}, \tag{1}$$

where T represents the tension within the fiber, $\mathbf{t}$ is the unit tangent vector, and s stands for the arc length.

Notice that at a typical point A, the resultant force $\mathbf{F}_A$ is determined by the fiber configuration and material property. Furthermore, if there are I arcs passing through the point A, assuming that each fiber has a constant tension T, the nodal force $\mathbf{F}_A(\mathbf{x},t)$ is calculated as

$$\mathbf{F}_A(\mathbf{x},t) = \sum_{i=1}^{I} \mathbf{F}_A^i(\mathbf{r},t) = \sum_{i=1}^{I} \frac{\partial (T\mathbf{t}_A^i)}{\partial s}, \tag{2}$$

where $\mathbf{t}_A^i$ represents the unit tangent vector of the i^{th} arc at the point A.

Therefore, the governing equations of the fluid-structure interaction system involving a single smooth submerged elastic fiber Γ_s can be stated as

$$\begin{aligned}
\rho(\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v}\cdot\boldsymbol{\nabla}\mathbf{v}) &= -\boldsymbol{\nabla} p + \mu\nabla^2\mathbf{v} + \mathbf{f}, && (3)\\
\boldsymbol{\nabla}\cdot\mathbf{v} &= 0, && (4)\\
\mathbf{f}(\mathbf{x},t) &= \int_{\Gamma_s} \mathbf{F}(\mathbf{r},t)\delta(\mathbf{x}-\mathbf{X}(\mathbf{r},t))ds, && (5)\\
\mathbf{v}(\mathbf{X}(\mathbf{r},t),t) &= \int_{\Omega} \mathbf{v}(\mathbf{x},t)\delta(\mathbf{x}-\mathbf{X}(\mathbf{r},t))d\mathbf{x}, && (6)
\end{aligned}$$

where $\mathbf{v}(\mathbf{x},t)$, $p(\mathbf{x},t)$, ρ, and μ represent the fluid velocity, pressure, density, and dynamic viscosity of the fluid.

Note that the line integrals are performed for all the submerged elastic fiber lines and the corresponding nodal forces are distributed as body forces in the fluid domain. Although Eqs. (3) and (4) can be solved with various Navier-Stokes solvers, if the fluid equations are discretized on a regular Eulerian lattice, a discretized Fast Fourier Transform is employed, which can greatly improve the computation efficiency. It is clear that Eq. (6) represents the equation of motion for immersed boundary points, and also implies that the immersed boundary points move at the local fluid velocity interpolated through the same discretized delta function, which is equivalent to the no-slip condition along the fluid-structure interface [6] [7].

3 Numerical Procedures

Consider two typical fibers $k-1$ and k connected at the point k, the nodal force at the point k is determined by tensions within the two adjacent fibers $k-1$ and k that are linked to this point. The submerged elastic point k is subjected to the fluid force $\mathbf{R}_k$, which will balance the resultant elastic nodal force $\mathbf{F}_k$. The fluid force $\mathbf{R}_k$ is entirely dependent upon the tension T_{k-1} and T_k, if the elastic structure has the same density as the fluid. However, if the elastic structure has a different density, the surrounding fluid force $\mathbf{R}_k$ must achieve a dynamic equilibrium with the difference of the resultant nodal force $\mathbf{F}_k$ and the D'Alembert force $\mathbf{I}_k$. If the lumped mass and the acceleration of the point k are denoted as M and $\mathbf{a}_k$, respectively, the D'Alembert force $\mathbf{I}_k$ can be expressed as $-M\mathbf{a}_k$. Therefore, the force exerted on the fluid by the submerged structure at the point k is equal to the nodal force $\mathbf{F}_k$ if the structure has the same density as the fluid, or $\mathbf{F}_k - \mathbf{I}_k$ if the structure has a different density. Naturally, the distribution of the nodal force in Eq. (5) represents the distribution of the resultant fluid forces at nodal points as the equivalent body forces.

There are three different treatments of the inertia effects of the submerged points if the immersed boundary has different density from the surrounding fluid. *The first method* is the straightforward employment of the D'Alembert force $-M\mathbf{a}$, where M and $\mathbf{a}$ stand for the concentrated mass at that submerged point and the corresponding acceleration. In the actual numerical implementation, the n^{th} step acceleration at the immersed boundary point k is denoted as

$$\mathbf{a}_k^n = \frac{\mathbf{v}_k^n - \mathbf{v}_k^{n-1}}{\Delta t}, \tag{7}$$

where $\mathbf{v}_k^n$ represents the immersed boundary point velocity at the time step n which is directly calculated with Eq. 6, and Δt is the time step.

The second method in dealing with the mass of immersed boundaries employs a penalty idea [3]. To do this, we first introduce $\mathbf{Y}_k$ of a massive boundary as a copy of $\mathbf{X}_k$ and then assume that there is no additional mass attached to $\mathbf{X}_k$. In addition, we assume that the massive boundary $\mathbf{Y}_k$ has a mass M which has no direct interaction with the fluid. On the other hand, the massless boundary $\mathbf{X}_k$ does interact with the fluid, namely, it moves with the local fluid velocity and exerts force locally on the fluid. The couple of the massless and massive boundary points occupy the exactly same positions initially and are linked through stiff springs. Therefore, if a pair of corresponding points move apart, the following restoring force $\mathbf{G}$ ensues

$$\mathbf{G}_k = C(\mathbf{Y}_k - \mathbf{X}_k), \tag{8}$$

where C stands for a large spring constant.

Since the submerged elastic boundary $\mathbf{X}_k$ is neutrally buoyant, $M\mathbf{a}_k$ has no meaning for $\mathbf{X}_k$. Instead, the massive boundary $\mathbf{Y}_k$ moves by the equation $M\mathbf{a}_k = -\mathbf{G}_k$. Here $\mathbf{a}_k$ can be calculated by the velocities of $\mathbf{Y}_k$ in two time steps: $\mathbf{a}_k = (\mathbf{v}_k^n - \mathbf{v}_k^{n-1})/\Delta t$, where $\mathbf{v}_k$ is the velocity of $\mathbf{Y}_k$, and different from the above definition. The mass effect is now given to the fluid equation by changing the resultant nodal force $\mathbf{F}_k$ into $\mathbf{F}_k + \mathbf{G}_k$.

Note that the restoring force $-\mathbf{G}_k$ acting on the massive boundary is also acting on the fluid in the opposite direction, and that the massive boundary does not interact with the fluid directly. Consider the case in which the coefficient C goes to the infinity. Then the massless boundary $\mathbf{X}_k$ and the massive boundary $\mathbf{Y}_k$ should be the same and the resultant force of the surrounding fluid $\mathbf{R}_k$ becomes exactly $\mathbf{F}_k + M\mathbf{a}_k$. Of course, in practice, C cannot be infinite but we can still keep $\mathbf{X}_k$ and $\mathbf{Y}_k$ close by choosing a sufficiently large constant C.

The third method in treating immersed boundary with mass is to use the conventional IB method with inhomogeneous fluid mass density [11] [4] [7], where the mass of the immersed boundary is assigned to the nearby surrounding fluid by the Dirac δ function, in exactly the same way the Lagrangian force defined at the immersed boundary is spread to the ambient fluid. The Eulerian mass density is defined by

$$\rho(\mathbf{x}, t) = \rho_f + \int_{V^s} (\rho_s - \rho_f)\delta(\mathbf{x} - \mathbf{X}(\mathbf{r}, t))dV, \tag{9}$$

where ρ_f is the fluid mass density, ρ_s is the mass density of the immersed boundary, $\mathbf{x}$ is the Eulerian coordinate, V^s represents the solid domain, and $\mathbf{X}$ is the parametric representation of the immersed boundary location.

The discrete form of the above expression is as follows,

$$\rho^n(\mathbf{x}) = \rho_f + \sum_k (\rho_s - \rho_f)\delta_h(\mathbf{x} - \mathbf{X}_k^n)\Delta V^s, \tag{10}$$

where δ_h is the same discretized delta function employed for the distribution of the nodal force at the immersed boundary, k stands for the discretized solid nodal points, and the incremental volume ΔV^s is often denoted as h^3.

Note that after the additional mass of the immersed boundary is transfered to the surrounding fluid, the composite material (fluid plus immersed boundary) is still incompressible, even though the Eulerian mass density $\rho(\mathbf{x}, t)$ is no longer a constant. However, because of this nonuniform mass density, we can no longer employ the discretized Fast Fourier Transform solver. Therefore, in the third method, we employ an iterative procedure, e.g., the multigrid method for the solution of the discretized incompressible Navier-Stokes equations. In the multigrid method, a correct way to spread the mass of immersed boundary to a series of gradually coarsened meshes is as follows:

$$\rho_{lh}^n(\mathbf{x}) = \rho_f + \int_{V^s} (\rho_s - \rho_f)\delta_{lh}(\mathbf{x} - \mathbf{X}^n)dV, \tag{11}$$

here lh stands for the grid whose meshwidth is $l \times h$, where h is the meshwidth of the finest grid, δ_{lh} means the discretized delta function on a grid with spacing $l \times h$, and $\rho_{lh}^n(\mathbf{x})$ is the

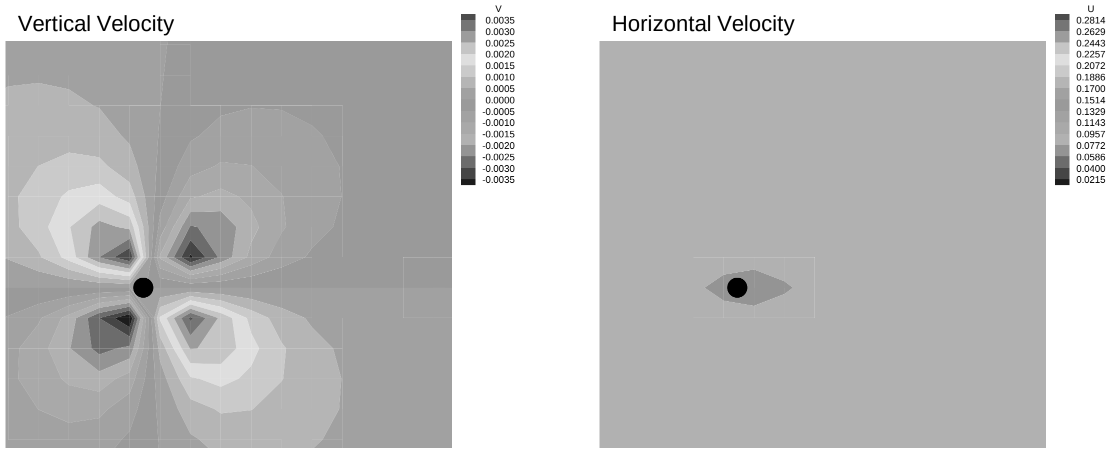

Figure 1: Velocity bandplots of a particle accelerating in a viscous fluid.

density defined on a mesh with spatial width $l \times h$ at the n^{th} time step.

4 Numerical Examples

Consider a two-dimensional straight channel with a viscous fluid and a moving particle. The particle with a mass of 0.003 g is released from a stationary position as the uniform flow is driven from one side of the channel to the other. To avoid the sudden change of velocity, the inlet velocity ramps up from 0 to 0.3 cm/s within 5 s. In our simulations, we compare results within 10 s.

If the particle has no mass, it moves like a passive particle and the motion coincides with the streamline trajectory passing through the initial location. The massive particle, however, needs some time to start up and reach to the speed of the surrounding flow. In this process, the inertia force exerts an effect on the surrounding fluid as shown in Fig. 1.

Obviously, this transient time delay depends on the mass of the particle: the bigger the mass, the longer the delay is. After the speed of the particle reaches its steady state, the particle will move with the fluid at a constant velocity and then its mass will not play any further role. Figs. 2 and 3 show the particle position and speed in terms of time. Note

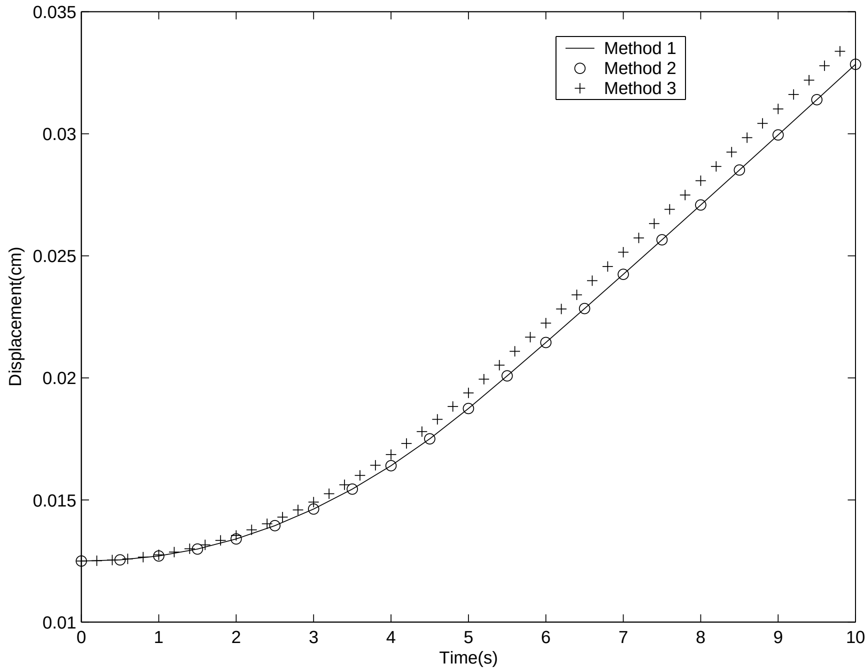

Figure 2: Comparison of the particle position.

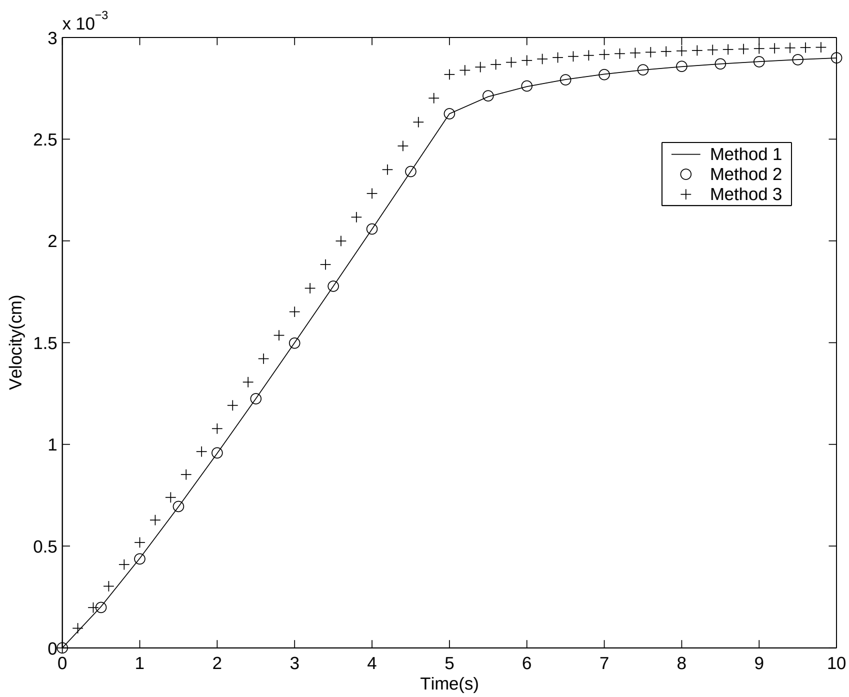

Figure 3: Comparison of the particle velocity.

that three methods which we explained above give very similar results which match with the physical intuition. In particular, the first and second methods (direct D'Alembert force and penalty method) yield virtually identical results. Because of the fact that the third method uses an iterative solver and a fairly different approach, there exist discernible numerical discrepancies of both position and velocity results.

5 Conclusions

To summarize, this work demonstrates the feasibility of three numerical treatments for immersed boundaries with mass. It is found that the first method, namely, a direct D'Alembert force approach, is most straightforward, yet it requires small time step for large mass; whereas the second method, namely, a penalty approach, yields virtually the same result as the first method, yet it is not feasible in handling relatively small mass and there also exist high frequency oscillations due to the attached stiff spring-mass system. Finally, the third method, namely, a distribution of boundary mass to the fluid along with a multigrid fluid solver, provides similar results, yet the required solver is not as efficient as the Fast Fourier Transform solver employed in the first two methods.

6 Acknowledgment

The support of a short-term faculty seed grant from the Othmer Institute at Polytechnic University in Brooklyn, NY for Xiaodong Wang is acknowledged.

References

[1] R. Dillon, L. Fauci, A. Fogelson, and D. Gaver III. Modeling biofilm processes using the immersed boundary method. *Journal of Computational Physics*, 129:57–73, 1996.

[2] L.J. Fauci and C.S. Peskin. A computational model of aquatic animal locomotion. *Journal of Computational Physics*, 77:85–108, 1988.

[3] Y. Kim and C.S. Peskin. Some applications of the immersed boundary method with massive boundary. *Journal of Computational Physics*, 2002. Submitted.

[4] D.M. McQueen and C.S. Peskin. Computer-assisted design of butterfly bileaflet valves for the mitral position. *Scand. J. Thor. Cardiovasc. Surg.*, 19:139–148, 1985.

[5] C.S. Peskin. Numerical analysis of blood flow in the heart. *Journal of Computational Physics*, 25:220–252, 1977.

[6] C.S. Peskin. Immersed boundary method. *Courant Lecture Notes*, 2000.

[7] C.S. Peskin. The immersed boundary method. *Acta Numerica*, 11:479–517, 2002.

[8] Jr. R.P. Beyer. A computational model of the cochlea using the immersed boundary method. *Journal of Computational Physics*, 98:145–162, 1992.

[9] J.M. Stockie and B.R. Wetton. Analysis of stiffness in the immersed boundary method and implications for time-stepping schemes. *Journal of Computational Physics*, 154:41–64, 1999.

[10] X. Wang and W. K. Liu. Extended immersed boundary method using FEM and RKPM. *Journal of Computational Physics*, 2002. Submitted.

[11] L. Zhu and C.S. Peskin. Simulation of a flexible flapping filament in a flowing soap film by the immersed boundary method. *Journal of Computational Physics*, 179(2):452–468, 2002.