A Numerically Efficient Scheme for Elastic Immersed Boundaries

F. Pacull[1] and M. Garbey[2]

[1] Fluorem - Ecully 69134 France - fpacull@fluorem.com
[2] Dept. of Computer Science, University of Houston, Houston, TX 77204, USA

1 Introduction

The main approaches to simulate fluid flows in complex moving geometries, use either moving-grid or immersed boundary techniques [5, 6, 7]. This former type of methods imply re-meshing, which are expensive computationally in the fluid/elastic-structure interaction cases that involve large structure deformations. In contrast, in the immersed boundary techniques, the effect of the boundary is applied remotely to the fluid by a constraint/penalty on the governing equations or a locally modified discretization/stencil: the fluid mesh is then globally independent of the moving interface, described by Lagrangian coordinates, and the effect of the interaction is introduced into the fluid variables at the Eulerian grid points next to the interface.

Many applications of fluid/flexible-body interaction simulations with large deformation are in bio-engineering. The accuracy of the input data in such a problem is not very high and one may prefer to emphasis the robustness of the numerical method over high accuracy of the solution process. A major advantage of the Immersed Boundary Method (IBM), pioneered by C.S. Peskin [10], is the high level of uniformity of mesh and stencil, avoiding the critical interpolation processes of the cut-cell/direct methods. Based on the standard finite-difference method, the IBM allows highly efficient domain decomposition techniques to be implemented. In other words, the difficulty of simulating dynamical interaction phenomena with complex geometries can be overcome by implementing, in a fast and easy way, large fine grid parallel computations that takes full advantage of a uniform stencil on an extended regular domain, as described in [3, 4], for blood flow applications. We are first going to recall the IBM formulation.

2 Discretization of the IBM

A complete and accurate introduction to the IBM can be found in [10]. Here is a brief description of the fluid/elastic interface model unified into a set of coupled PDEs. The incompressible Navier-Stokes system is written as:

$$\rho\left[\frac{\partial V}{\partial t}+(V\cdot\nabla)V\right] = -\nabla P+\mu\Delta V+F \tag{1}$$

$$\nabla\cdot V = 0 \tag{2}$$

The IBM requires the extrapolation of the Lagrangian vector f into the Eulerian vector field F from the RHS of (1). In the IBM of Peskin, we use a distribution of Dirac delta functions δ for that purpose:

$$F(x,t)=\int_{\Gamma} f(s,t)\delta\left(x-X(s,t)\right)ds=\begin{cases} f(s,t) & \text{if } x=X(s,t)\\ 0 & \text{otherwise}\end{cases} \tag{3}$$

The motion of the immersed boundary should match the motion of the neighboring fluid particles because of a no-slip boundary condition. Eq.(4) approximates this no-slip boundary condition using the Dirac delta function as an interpolating tool for V, from Ω to Γ:

$$\frac{\partial X(s,t)}{\partial t}=\int_{\Omega} V(x,t)\delta(x-X(s,t))dx=\begin{cases} V(X(s,t),t) & \text{if } x=X(s,t)\\ 0 & \text{otherwise}\end{cases} \tag{4}$$

The immersed boundary obeys a linear elastic model. We use Hooke's law of elasticity, i.e. the tension $\mathcal{T}$ of the immersed boundary is a linear function of the strain. For a one-dimensional boundary, we have:

$$\mathcal{T}(s,t)=\sigma\left|\frac{\partial X(s,t)}{\partial s}\right|, \tag{5}$$

where σ is the boundary elasticity coefficient. The local elastic force density f is defined as:

$$f(s,t)=\frac{\partial(\mathcal{T}(s,t)\tau(s,t))}{\partial s}, \quad \tau(s,t)=\frac{\partial X(s,t)/\partial s}{|\partial X(s,t)/\partial s|}. \tag{6}$$

τ is the unit tangent vector to Γ. Finally, by plugging (5) into the set of equations in (6), we get:

$$f(s,t)=\sigma\frac{\partial^2 X(s,t)}{\partial s^2} \tag{7}$$

The practical implementation of the IBM of Peskin offers dozens of different possibilities regarding the choice of the temporal scheme, the space discretization, the discrete approximation of the Dirac function and so on. There is clearly a compromise between the stability of the scheme that suffers from sharp numerical interfaces for the pressure, that should be discontinuous, and accuracy that needs this numerical feature. We refer to the thesis of the first author [9] and its bibliography for an extensive comparison of possible implementations for standard benchmark problems such as the oscillation relaxation of stretched bubble toward its equilibrium or the motion of the bubble in a cavity flow test case. These benchmark problems show clearly that the IBM method does not preserve the volume of the bubble. The "numerical porosity" of the IBM is a drawback of this immersed boundary technique. We are going to present a volume conservation method that fixes this problem.

3 Volume Conservation Method Based on Constrained Optimization

As described in [8], most of the existing methods to improve volume conservation use a local change of stencil on the Eulerian grid, or require the computation of the normal to the boundary at each of its points to build a constrained local interpolation operator for the boundary velocity. The following method is global and uses the initial volume enclosed by the immersed boundary as a control objective. We represent the position vector of the immersed boundary with a global polynomial. In the 2D bubble test case, it is natural to use a Fourier expansion. Initially the discretization points on the immersed boundary are equally spaced in the curvilinear space. The motion of these discrete points follows the fluid flow because of the no-slip boundary condition. At all time, therefore, these discretization points are a regular transformation of the original distribution and can be used in the Fourier representation. For convenience, we are going to define the global volume conservation problem in the case of a closed immersed boundary like in our benchmark problems above.

After discretization, the immersed boundary is represented by a finite set of grid points:

$$\{X_i\}_{0\leq i\leq M-1} = \{X_{1,i}, X_{2,i}\}_{0\leq i\leq M-1}. \tag{8}$$

$\{X_{1,i}\}_{0\leq i\leq M-1}$ is the vector of the horizontal components of the moving points and $\{X_{2,i}\}_{0\leq i\leq M-1}$ that of their vertical components. We assume that M is even and then define $K \equiv \frac{M}{2}$. The Fourier expansion of X is, for $j = 1,2$ and $0 \leq i \leq M-1$:

$$\widehat{X}_{j,i}(\tilde{\alpha}) = \frac{1}{M}\left[\alpha^A_{j,0} + 2\sum_{k=1}^{K-1}\left(\alpha^A_{j,k}\cos\left(2\pi k\frac{i}{M}\right) + \alpha^B_{j,k}\sin\left(2\pi k\frac{i}{M}\right)\right) + \alpha^A_{j,K}(-1)^i\right] \tag{9}$$

where

$$\alpha^A_{j,k} = \sum_{i=0}^{M-1} X_{j,i}\cos\left(2\pi k\frac{i}{M}\right), \quad j=1,2,\ 0\leq k\leq K,$$
$$\alpha^B_{j,k} = \sum_{i=0}^{M-1} X_{j,i}\sin\left(2\pi k\frac{i}{M}\right), \quad j=1,2,\ 1\leq k\leq K-1,$$
$$\alpha^B_{j,0} = \alpha^B_{j,K} = 0, \quad j=1,2.$$

Let us introduce the notation:

$$\tilde{\alpha} = (\alpha^A_{1,0}..\alpha^A_{1,K}\alpha^B_{1,1}..\alpha^B_{1,K-1}\alpha^A_{2,0}..\alpha^A_{2,K}\alpha^B_{2,1}..\alpha^B_{2,K-1}) \tag{10}$$

and

$$\alpha = (\alpha^A_{1,1}..\alpha^A_{1,K-1}\alpha^B_{1,1}..\alpha^B_{1,K-1}\alpha^A_{2,1}..\alpha^A_{2,K-1}\alpha^B_{2,1}..\alpha^B_{2,K-1}). \tag{11}$$

It is easy to compute analytically the area of the bubble using (9) and Green's theorem:

$$\text{Area}(\alpha) = \frac{4\pi}{M^2}\sum_{k=1}^{K-1} k(\alpha^A_{1,k}\alpha^B_{2,k} - \alpha^A_{2,k}\alpha^B_{1,k}) \tag{12}$$

If $\text{Area}(\alpha) = V_0$ at time zero, our constraints writes $c(\alpha) = \text{Area}(\alpha) - V_0 = 0$.

We perform a least square minimization of the position change, constrained with the area preservation. The function to minimize is:

$$\begin{aligned} F(\tilde{\alpha}) &= \|\{X_i - \widehat{X}_i(\tilde{\alpha})\}_{0\le i\le M-1}\|_2^2 \\ &= \sum_{i=0}^{M-1}\left((X_{1,i} - \widehat{X}_{1,i}(\tilde{\alpha}))^2 + (X_{2,i} - \widehat{X}_{2,i}(\tilde{\alpha}))^2\right) \end{aligned} \tag{13}$$

Since the input variable of the constraint $c(\alpha) = 0$ is α and not $\tilde{\alpha}$, the search space of the minimization is $\mathbb{R}^{4(K-1)}$ and not $\mathbb{R}^{4K}$. The coefficients

$$(\alpha^A_{1,0}, \alpha^A_{1,K}, \alpha^A_{2,0}, \alpha^A_{2,K})$$

are fixed. These coefficients are not related to the area, but control the global position of the immersed boundary in the domain.

We can express the constrained minimization problem as follows:

$$\min_{\alpha/c(\alpha)=0} F(\alpha) \quad \text{with } F(\alpha) = F(\tilde{\alpha})|_{(\alpha^A_{1,0},\alpha^A_{1,K},\alpha^A_{2,0},\alpha^A_{2,K})}. \tag{14}$$

We are going to show that this optimization problem has a unique solution.

We define the Lagrangian $L(\alpha,\lambda)$ with $\alpha \in \mathbb{R}^{4(K-1)}$, $\lambda \in \mathbb{R}$:

$$L(\alpha,\lambda) = F(\alpha) + \lambda c(\alpha) \tag{15}$$

F and c are both twice continuously differentiable with respect to α. A nice property of the Hessian $\nabla_{\alpha\alpha}F$ is that it is independent of α and diagonal:

$$\nabla_{\alpha\alpha}F = \frac{4}{M} I_{4(K-1)} \tag{16}$$

This is also true for $\nabla_{\alpha\alpha}c$ in this particular test-case with area formula (12):

$$\nabla_{\alpha\alpha}c = \frac{4\pi}{M^2}\begin{bmatrix} 0 & & & +B_K \\ & & -B_K & \\ & -B_K & & \\ +B_K & & & 0 \end{bmatrix}, \tag{17}$$

$$B_K = \begin{bmatrix} 1 & & & 0 \\ & 2 & & \\ & & \ddots & \\ 0 & & & K-1 \end{bmatrix}. \tag{18}$$

So, for a given pair (α^*,λ^*) and for all α in the neighborhood of α^*, we have:

$$\alpha^T \nabla_{\alpha\alpha} L(\alpha^*, \lambda^*) \alpha = \alpha^T \nabla_{\alpha\alpha} F \alpha + \lambda^* \alpha^T \nabla_{\alpha\alpha} c \alpha \tag{19}$$

$$= \frac{4}{M} \alpha^T \alpha + \frac{8\pi\lambda^*}{M^2} \sum_{k=1}^{K-1} k(\alpha^A_{1,k} \alpha^B_{2,k} - \alpha^A_{2,k} \alpha^B_{1,k}) \tag{20}$$

$$= \frac{4}{M} \alpha^2 + 2\lambda^* \text{Area}(\alpha) \tag{21}$$

Hence, $\lambda^* > 0$ implies:

$$\alpha^T \nabla_{\alpha\alpha} L(\alpha^*, \lambda^*) \alpha > 0 \tag{22}$$

Whenever we find a pair (α^*, λ^*) such that $\nabla_\alpha F(\alpha^*) = 0$, $c(\alpha^*) = 0$ and $\lambda^* > 0$, the first-order Lagrangian sufficiency condition is satisfied. This implies that α^* is a strict local minimum.

We note here that $\nabla_{\alpha\alpha} F$ has the same form for any geometry of Γ in any dimension, since it simply represents the norm of the correction on X. We also have here an elegant analytical formulation for the constraint in the "Bubble" test case. The functions $F(\alpha)$ and $c(\alpha)$ are easy to compute as well as $\nabla_\alpha F(\alpha)$, $\nabla_\alpha c(\alpha)$ and the Hessians of F and c, respectively $\nabla_{\alpha\alpha} F$ and $\nabla_{\alpha\alpha} c$, are constant. In the general situation, one may have to use a more complicated numerical method to get the elements of the minimization problem (14) regarding the constraint. In the numerical implementation for the "Bubble" test case, we solve:

$$\begin{pmatrix} \nabla_\alpha L(\alpha, \lambda) \\ \nabla_\lambda L(\alpha, \lambda) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{23}$$

using the classical Newton-Raphson algorithm. A good initial solution for the iterative Newton algorithm is the α coefficients corresponding to the actual position of the boundary X before the correction. In most cases, the volume of the immersed boundary is found to evolve relatively slowly with respect to time. Then it is not necessary to perform this minimization at every time step.

Compared to the traditional method, the Fourier expansion allows a fairly compact representation of the interface, without any loss of accuracy.

To reduce the computational load of the minimization process, one may want to restrict the search space of the α coefficients to a finite space of dimension smaller than M the number of discrete points that support the immersed interface. Because of the high order accuracy of the Fourier expansion, we work with the first $\frac{K}{4}$ of the Fourier expansion coefficients $(\alpha_{j,k}, \beta_{j,k}),\ j = 1, 2$. This corresponds roughly to the idea that one needs at least 4 mesh points to represent a wave period. This compact representation of the immersed closed boundary has several advantages. First, it filters out the high wave frequency components of the position vector, and removes most of the noise in the force term. Second, it can also drastically speed up an implicit IBM scheme, by reducing the search space for the Newton algorithm. In [9], we implemented the Inexact Newton Backtracking Method of M. Pernice and H.F. Walker in an implicit IBM scheme, in which the unknown is the position of the moving boundary at the next time step. We used a Krylov method to solve the inexact Newton condition and an associated Jacobian matrix approached by finite-differences. Without a Fourier representation of the boundary position, the size of the Newton search space in two space dimensions is $2M$, while here it becomes $\frac{M}{2}$.

For the "bubble" test cases discussed earlier we observe a perfect global volume preservation. Most of the minimization work is done in the very first step and we observe then a regular and fast evolution of $\nabla_\alpha L(\alpha,\lambda)$ while $c(\alpha) = \nabla_\lambda L(\alpha,\lambda)$ requires only two or three steps to be negligible. The high Fourier modes should be cut off so that the volume correction does not make the system numerically unstable: the small corrections can originate high frequencies oscillations that will be self-exciting in the interaction.

We will now to present an application of our IBM implementation that is non trivial and can take advantage of the Fourier representation of the immersed interface.

4 Application of the IBM and Conclusion

Let us consider a single bubble in a long rectangular cavity at rest. The flow velocity at the initial time is null and the bubble is a circle. We equip the moving bubble with a membrane that can contract or dilate periodically by forcing the boundary elasticity coefficient in the Hooke law. We set:

$$\sigma(\theta) = \sigma_0\left(1+\sigma_1\left(1+\sin\left(2\pi\frac{t}{P}\right)\right)(\cos(\theta)+1)\right), \tag{24}$$

where $\theta \in (0,2\pi)$ is the angle in the polar moving coordinate system attached to the bubble. To be more specific if X is a point attached to the membrane, the corresponding θ stays invariant with respect to X, no matter the X motion. $\theta \in (\frac{\pi}{2}, 3\frac{\pi}{2})$ corresponds to the anterior side of the bubble, and the posterior side is the opposite side of the bubble.

The elasticity coefficient increases and decreases periodically in time with period P. The variation of the elasticity coefficient in time is most pronounced around $\theta = 0$, and has less variation around $\theta = \pi$. The largest contraction and relaxation move of the membrane happens around $\theta = 0$.

During the contraction phase, the posterior side of the membrane projects the liquid inside the bubble toward the forward side. While this mass of liquid travels forward, the posterior side of the membrane relaxes. It results overall in a motion of the bubble forward. We observe from the numerical simulation that this motion is maintained by two nice symmetric vortices that companion the bubble motion. Fig. 1 shows an established forward motion of the bubble with the bubble starting at rest near the right side of the cavity. In this simulation, we have $(\sigma_0,\sigma_1) = (200,5)$, and the elasticity coefficient varies in the interval $200 \le \sigma(\theta) \le 4200$. The center of the bubble is centered on a symmetry axis of the rectangular cavity at $t = 0$. The solution conserves this symmetry as time goes on. However, if the position of the bubble is slightly shifted away of this axis at time $t = 0$, the no-slip boundary flow condition rapidly breaks the symmetry. The direction of motion becomes unstable and the bubble starts a (slow) chaotic motion. In practice a tail attached at the bubble at the $\theta = 0$

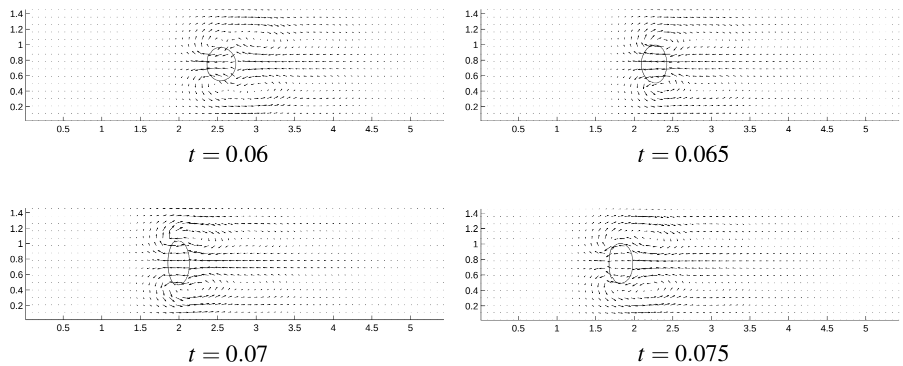

Fig. 1. Active bubble motion with time period 0.02

angle position should stabilize this motion because of viscosity forces. Alternatively, a tandem of two "active" bubbles can pilot a larger body. We are currently studying the optimum design of this setup using our simulation tool.

This motion of the "active bubble" with no flapping fins nor flagella, or helicoidal motion [1] is an amazing example of fluid dynamic. We found particularly fascinating that the IBM technique, that is relatively simple to implement, gives access to such complex fluid flow problems.

References

[1] Cortez, R., Cowen, N., Dillon, R., Fauci, L.: Simulation of Swimming Organisms; Couling Internal Mechanics with External Fluid Dynamics, *Comput. Sci. Engrg.*, 6(3): 38–45, 2004.

[2] Francois, M., Shyy, W.: Computations of Drop Dynamics with the Immersed Boundary Method, part 1: Numerical Algorithm and Buoyancy-Induced Effect, *Numer. Heat Transfer B*, 44:101–118, 2003.

[3] Garbey, M., Pacull, F.: Toward A Matlab MPI Parallelized Immersed Boundary Method. In *Parallel Computational Fluid Dynamics, Theory and Applications*, pages 397–404. A. Deane et al., eds. Elsevier, 2006.

[4] Garbey, M., Pacull, F.: A Versatile Incompressible Navier-Stokes Solver for Blood Flow Application, *Internat. J. Numer. Methods Fluids* 54(5):473–496, 2007.

[5] Glowinski, R.: A Fictitious Domain Approach to the Direct Numerical Simulation of Incompressible Flow Past Moving Rigid Bodies: Application to Particle Flow, *J. Comput. Phys.*, 162:363–426, 2001.

[6] Lee, L., LeVeque, R.: Immersed Interface Methods for Incompressible Navier-Stokes Equations, *SIAM J. Sci. Comput.*, 25:832–856, 2003.

[7] Mittal, R., Iaccarino, G.: Immersed Boundary Methods, *Ann. Rev. Fluid. Mech.*, 37:239–261, 2005.

[8] Newren, E.P.: *Enhancing The Immersed Boundary Method Stability, Volume Conservation, And Implicit Solvers*, PhD thesis, Department of Mathematics, University of Utah, 2007.
[9] Pacull, F.: *A Numerical Study of The Immersed Boundary Method and Application to Blood Flow*, PhD thesis, Department of Mathematics, University of Houston, 2006.
[10] Peskin, C.S.: The Immersed Boundary Method, *Acta Numer.*, 11:479–517, 2002.