

Simulation of Biological Flow and Transport in Complex Geometries using Embedded Boundary / Volume-of-Fluid Methods

David Trebotich

Center for Applied Scientific Computing, Lawrence Livermore National Laboratory,
Livermore, CA

E-mail: trebotich1@llnl.gov

Abstract. We have developed a simulation capability to model multiscale flow and transport in complex biological systems based on algorithms and software infrastructure developed under the SciDAC APDEC CET. The foundation of this work is a new hybrid fluid-particle method for modeling polymer fluids in irregular microscale geometries that enables long-time simulation of validation experiments. Both continuum viscoelastic and discrete particle representations have been used to model the constitutive behavior of polymer fluids. Complex flow environment geometries are represented on Cartesian grids using an implicit function. Direct simulation of flow in the irregular geometry is then possible using embedded boundary / volume-of-fluid methods without loss of geometric detail. This capability has been used to simulate biological flows in a variety of application geometries including biomedical microdevices, anatomical structures and porous media.

1. Introduction

Biological flow is complex and not well-understood due to the presence of macromolecules in the fluid whose molecular lengths are comparable to the flow geometries of microscale systems. This is inherently a multi-scale problem as flow, transport, molecular recognition, and reaction in these types of flows require description of multi-species processes in complex three-dimensional geometries. For example, in a biomedical microdevice the length scales range from about 1 nanometer (the scale at which the surface of a small globular protein is discretized), to tens or hundreds of microns for typical fluidic processors. The time scales range from tens of nanoseconds (a characteristic time for protein reorientation) to a few minutes (a typical time for a complete bioassay). Currently it is not possible to computationally resolve the phenomena of interest over this range of length and time scales in a computational design or modeling tool. Additionally, available software packages do not contain advanced numerical algorithms to physically model such complex fluids.

Additional complexity arises from the need to represent realistic geometries in biological flows. Geometric details in device or anatomical flow domains can affect system level behavior. Therefore, it is necessary to work within a self-consistent framework that includes capabilities from surface extraction of image data (e.g., CAD, MRI, CT) to compatible gridding and direct simulation in the derived geometry without loss in geometric details. However, methods of surface extraction are limited in their abilities to recover the geometry of complex objects in real

time. Moreover, it is necessary to convert the surface extracted by image processing techniques into a computational domain appropriate for the CFD solver, involving the construction of a mesh on the surface and in the 3D domain that it encloses. Valuable information can be lost during this construction due to limitations of the CFD solvers with respect to the properties of the surface mesh. It is often the case, for example, that the surface needs to be smoothed in order to build a finite-element mesh necessary to some CFD codes. Failing that, irregular surfaces must be approximated by a large number of small mesh elements, pushing the limits of computer memory. The result in this and other cases is a compromise in the level of accuracy of the surface extraction method.

In this paper we discuss computational solutions to particle-laden flow problems which leverage solver capabilities and infrastructure developed in *An Algorithmic and Software Framework for Applied Partial Differential Equations* (APDEC), a SciDAC Center for Enabling Technology. The fundamental CFD algorithm is based on a new fluid-particle coupling scheme for polymer fluids. Irregular geometry is handled by the embedded boundary / volume-of-fluid method.

2. Particle-laden Fluidic Systems

We use the Navier-Stokes equations to model the solvent as a continuum on domain Ω :

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} + \frac{1}{\rho} \nabla P = \nu \Delta \mathbf{u} + \frac{1}{\rho} \mathbf{F} \quad (1)$$

$$\nabla \cdot \mathbf{u} = 0. \quad (2)$$

These equations describe an incompressible fluid of density ρ , pressure P , velocity \mathbf{u} , and Newtonian viscosity ν , subject to an additional body force \mathbf{F} . On the domain boundary $\delta\Omega$ we have the no-slip boundary condition $\mathbf{u} = 0$.

The polymer solute is represented as a collection of point masses each subject to Newton's second law of motion

$$m_\alpha \frac{d^2 \mathbf{x}_\alpha}{dt^2} = m_\alpha \frac{d\mathbf{v}_\alpha}{dt} = \mathbf{f}_\alpha. \quad (3)$$

Here m_α is the mass of the α^{th} particle, \mathbf{x}_α is its coordinate, and \mathbf{v}_α is its velocity. The particle is subject to a force \mathbf{f}_α which combines a Stokes drag term with a stochastic (Brownian) perturbation,

$$\mathbf{f}_\alpha = m_\alpha \gamma (\mathbf{u}(\mathbf{x}_\alpha) - \mathbf{v}_\alpha) + \mathcal{F}_{\mathbf{B}\alpha}. \quad (4)$$

Here, $1/\gamma$ is a phenomenological relaxation time ($m\gamma = 6\pi\mu b$ for a Stokes sphere of radius b), and $\mathcal{F}_{\mathbf{B}}$ is the stochastic force

$$\langle \mathcal{F}_{\mathbf{B}\alpha}(t) \rangle = 0 \quad (5)$$

$$\langle \mathcal{F}_{\mathbf{B}\alpha}(t) \mathcal{F}_{\mathbf{B}\alpha}(t') \rangle = \sigma_\alpha^2 I \delta(t - t') \quad (6)$$

$$\sigma_\alpha = \sqrt{2m_\alpha \gamma k_B T}, \quad (7)$$

with k_B being Boltzmann's constant and T the temperature.

The force \mathbf{F} acting on the fluid is

$$\mathbf{F}(\mathbf{x}) = - \sum_{\alpha} \mathbf{f}_\alpha \delta_\epsilon(\mathbf{x} - \mathbf{x}_\alpha), \quad (8)$$

where δ_ϵ represents a smoothed Dirac delta function with length scale ϵ .

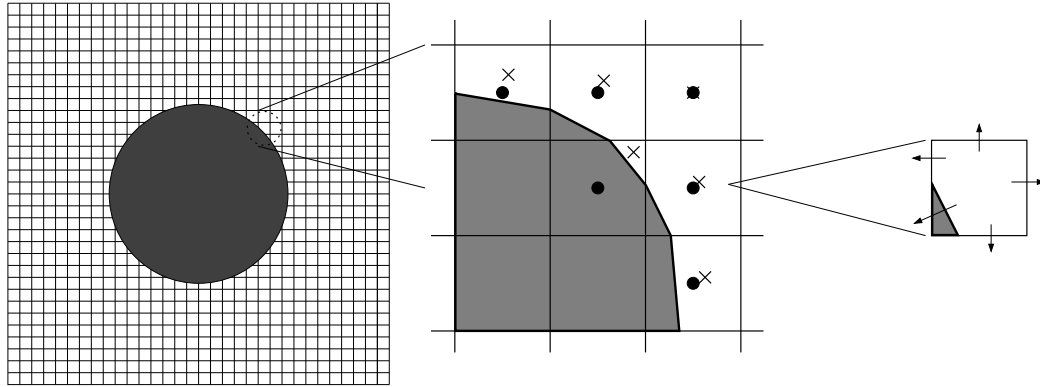


Figure 1. Example of an irregular geometry on a Cartesian grid (left). Close-up view of embedded boundaries “cutting” regular cells (middle). Single irregular cut cell showing boundary fluxes (right). Shaded area represents volume of cells excluded from domain. Dots represent cell-centers. X’s represent centroids.

In addition to the incompressibility condition (2) we have three additional constraints on the particles: (i) interparticle spacing is constant

$$\|\mathbf{x}_\alpha - \mathbf{x}_\beta\| = a \quad (9)$$

if particles α and β represent adjacent nodes in a bead-rod polymer representation; (ii) particles cannot pass through a physical boundary,

$$\mathbf{x}_\alpha \in \Omega; \quad (10)$$

and (iii) rods cannot cross.

3. Embedded Boundary / Volume-of-Fluid Methods

We use a Cartesian grid embedded boundary method to discretize the fluid equations in the presence of irregular boundaries. In this approach, the irregular domain is discretized as a collection of control volumes formed by the intersection of the problem domain with the cubic Cartesian grid cells as in a “cookie-cutter” (see Figure 1). The various operators – the discrete divergence $\nabla \cdot$, discrete gradient ∇ , and discrete Laplacian Δ – are approximated using finite volume differences on the irregular control volumes, with the fluxes computed using the primary discretized dependent variables, which approximate the solution evaluated at the centers of the original Cartesian cells.

For example, the Laplacian operator, $\Delta\phi = \nabla \cdot \nabla\phi$, is a divergence of a flux and can be calculated in a finite volume (such as the cut cell in Figure 1c) by applying the divergence theorem which converts a volume integral to a surface integral so that fluxes can be simply summed around the perimeter of the cut cell along normals:

$$\nabla \cdot \vec{F} \approx \frac{1}{\kappa h^d} \int_{\Omega} \nabla \cdot \vec{F} d\Omega = \frac{1}{\kappa h} \int_{\partial\Omega} \vec{F} \cdot \vec{n} dS = \frac{1}{\kappa h} \sum_s \alpha_s \vec{F}_s + \alpha_b \vec{F}_b, \quad (11)$$

where κ is the volume fraction of the cell, α is the area fraction of a cell edge, h is the grid spacing and subscripts s and b indicate cell edges and the embedded boundary, respectively.

To obtain a flux at an embedded boundary when only cell-centered data exists, which is the case when a homogeneous Dirichlet boundary condition is needed for no-slip of the fluid velocity

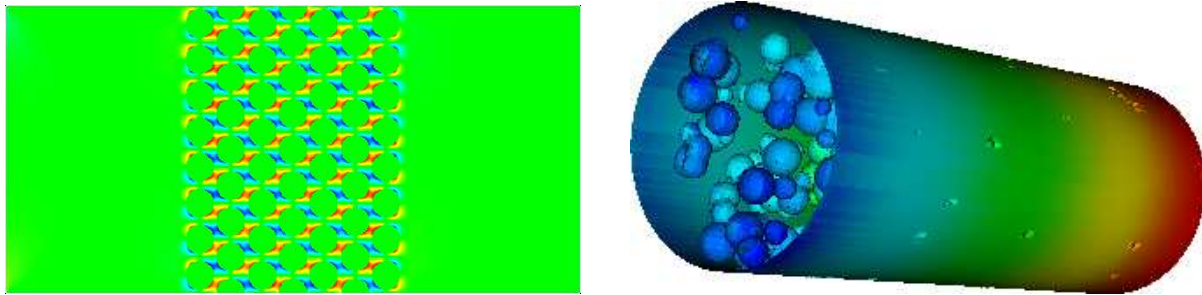


Figure 2. Direct simulation of flow in geometries obtained from implicit function on grid. (a) Computed transverse pressure gradient in a 2D idealized pore geometry representation of a post-array microchannel. Flow is left to right. Posts are $20\mu\text{m}$ in diameter. (b) Computed pressure in 3D idealized pore geometry representation of a microtube randomly packed with glass beads (packed bed). Spheres range in diameter from 10 to $30\mu\text{m}$.

at solid walls, a special extrapolation procedure is needed. Instead of applying the advanced stencil for Dirichlet boundary conditions described in [4] we assume a lower-order truncation error stencil based on least squares [8, 7], but one that maintains second-order solution error, to interpolate the flux φ at an irregular boundary, B . The gradient $\nabla\varphi$ is obtained from the system:

$$\mathcal{A} \cdot \nabla\varphi = \delta\varphi, \quad (12)$$

where

$$\mathcal{A} = (\delta\vec{x}_1, \delta\vec{x}_2, \dots, \delta\vec{x}_p)^T \quad (13)$$

$$\delta\varphi = (\delta\varphi_1, \delta\varphi_2, \dots, \delta\varphi_p)^T \quad (14)$$

$$\delta\vec{x}_m = \vec{x}_m - \vec{x}_B \quad (15)$$

$$\delta\varphi_m = \varphi_m - \varphi_B. \quad (16)$$

The stencil ($m = 1, 2, \dots, p$) is determined by the normal of the embedded boundary. In 2D the stencil includes up, side and corner cells (see Figure 1b), with $p = 3$; in 3D, the normal points to an octant, where $p = 7$. In 2D, for example, there are two equations and three unknowns; a least squares fit is applied to obtain the gradients $\nabla\varphi$.

The embedded boundary approach to complex geometry is compatible with a fast and accurate technique for surface extraction from CAD and image data [2, 9]. In this technique fast marching level sets methods are used to obtain a surface rendering from the image. The surface is then represented on a Cartesian grid with implicit functions, making possible direct simulation in highly packed, irregular microscale geometries (e.g., porous media) with algorithms based on embedded boundaries. We have demonstrated the implicit function technique for ideal pore scale geometries in 2D and 3D. Figure 2a is an example of 2D flow in a microchannel with a structured array of posts or cylinders. The transverse pressure gradient is shown which is non-zero due to the geometry. Figure 2b is an example of 3D flow in a cylinder randomly packed with microspheres resembling a packed bed column. The following implicit function is used to represent these geometries on the grid for both the 2D and 3D examples:

$$\phi(x) = \min_k (|\vec{x} - \vec{x}_k|^2 - r_k^2), \quad (17)$$

where \vec{x}_k = center of k^{th} sphere, r_k = radius of k^{th} sphere and $\vec{x} : \phi(\vec{x}) = 0$ on the boundary.

4. Particle Interactions in Hybrid Fluid-Particle Systems

Our previous versions of the particle method have included varying degrees of fidelity in particle interactions. In [12] we elastically bounced particles off surfaces but ignored the rod crossing constraint which is common in other implementations [3]; the particle timestep was two orders of magnitude less than the stable fluid timestep. In [10] we used a soft potential for rod-rod and bead-surface interactions in 2D, and explained a hard constraint algorithm with fluid coupling in [11]. Currently we use a hierarchical approach to enforce the constraints of the particle system as in [5]:

- (i) Calculate unconstrained particle motion due to Newton's Second Law (3).
- (ii) Calculate motion subject to rod length constraint (9).
- (iii) Calculate motion subject to rod-rod crossing constraint (e.g., [10, 11]).
- (iv) Calculate motion subject to bead-surface crossing constraint (10).

The rod length constraint, which is similar to the molecular dynamics SHAKE algorithm [6], can be differentiated to obtain another constraint, on velocity, similar to molecular dynamics RATTLE algorithm [1]. Also, the rod-rod and bead-surface crossing constraints can be enforced simultaneously to obtain a larger timestep on the order of a stable fluid timestep determined by advection alone (CFL condition), enabling system-level modeling through long-time, experimental-scale simulation [5].

Acknowledgment

This work was performed under the auspices of the U.S. Department of Energy by the University of California, Lawrence Livermore National Laboratory under contract No. W-7405-Eng-48.

References

- [1] H. C. Anderson. Rattle: A "velocity" version of the Shake algorithm for molecular dynamics calculations. *J. Comp. Phys.*, 52:24–34, 1983.
- [2] T. Deschamps, P. Schwartz, D. Trebotich, P. Colella, D. Saloner, and R. Malladi. Vessel segmentation and blood flow simulation using level-sets and embedded boundary methods. In *International Congress Series*, volume 1268, pages 75–80, June 2004.
- [3] J. S. Hur, E. S. G. Shaqfeh, H. P. Babcock, and S. Chu. Dynamics and configurational fluctuations of single DNA molecules in linear mixed flows. *Phys. Rev. E*, 66, 2002.
- [4] H. Johansen and P. Colella. A Cartesian grid embedded boundary method for Poisson's equation on irregular domains. *J. Comp. Phys.*, 147(2):60–85, December 1998.
- [5] G. H. Miller and D. Trebotich. Toward a mesoscale model for the dynamics of polymer solutions. *J. Comput. Theor. Nanosci.*, 4(4):797–801, 2007.
- [6] J.-P. Ryckaert, G. Ciccotti, and H. J. C. Berendsen. Numerical integration of the Cartesian equations of motion of a system with constraints: Molecular dynamics of n-alkanes. *J. Comp. Phys.*, 23:327–341, 1977.
- [7] P. O. Schwartz, M. Barad, P. Colella, and T. J. Ligocki. A Cartesian grid embedded boundary method for the heat equation and Poisson's equation in three dimensions. *J. Comput. Phys.*, 211:531–550, 2006.
- [8] D. Trebotich, P. Colella, G. H. Miller, A. Nonaka, T. Marshall, S. Gulati, and D. Liepmann. A numerical algorithm for complex biological flow in irregular microdevice geometries. In *Technical Proceedings of the 2004 Nanotechnology Conference and Trade Show*, volume 2, pages 470–473, 2004.
- [9] D. Trebotich, T. Deschamps, and P. Schwartz. Air-flow simulation in realistic models of the trachea. In *Technical Proceedings of the 26th Annual International Conference IEEE Engineering in Medicine and Biology Society*, pages 3933–3936, September 2004.
- [10] D. Trebotich, G. H. Miller, and M. D. Bybee. A hard constraint algorithm to model particle interactions in DNA-laden flows. *Nano. Micro. Thermophys. Engr.*, 11(1-2):121–128, 2007.
- [11] D. Trebotich, G. H. Miller, and M. D. Bybee. A penalty method to model particle interactions in DNA-laden flows. *J. Nanosci. Nanotech.*, 2007. accepted to appear. Also available as LLNL Technical Report UCRL-JRNL-223318.
- [12] D. Trebotich, G. H. Miller, P. Colella, D. T. Graves, D. F. Martin, and P. O. Schwartz. A tightly couple particle-fluid model for DNA-laden flows in complex microscale geometries. *Computational Fluid and Solid Mechanics*, pages 1018–1022, 2005.