Sarah Ritchey Patterson

Research Statement

I am a numerical analyst who works on computational fluid dynamic problems. My interests lie in the realm of fluid-structure interaction problems with applications in mathematical biology and physiology. To overcome difficulties in capturing complex interactions in the circulatory system, I devise and implement numerical methods.

In particular, I am interested in the mechanisms that prevent structures in the kidneys from rupturing when blood pressure rises. A vessel called the afferent arteriole actively contracts and dilates to regulate blood pressure and protect delicate downstream structures. This defensive action is called the myogenic response and it requires accurately computing the blood pressure, blood velocity, and the movement of the afferent arterial vessel.

General Research Goal: Develop sophisticated mathematical models to help researchers understand the myogenic response and the hemodynamics in the kidneys.

Immersed Boundary Problems

A natural way to model the afferent arteriole and the blood flow is to frame it as an immersed boundary problem [8]. Typically in these problems, the blood vessel is represented as an infinitely thin interface, Γ, immersed in a fluid. Let $X(s,t)$ for $0 < s < L$ be a parametrization of the position of the interface at time t where L is the length of the interface. The interface supplies a force F that is singularly supported on the interface such as

$$F(x,t) = \int_\Gamma f(X(s,t),t)\delta(X(s,t)-x)ds$$

where f is the strength of the interfacial force and δ is the Dirac delta function. Incompressible Navier Stokes equations is used to model the motion of the internal blood and external fluid.

$$\rho\left(\frac{\partial u}{\partial t} + u\cdot\nabla u\right) = -\nabla p + \mu\Delta u + F$$

$$\nabla\cdot u = 0$$

where u is the velocity of the fluid, p is the pressure, ρ is the density, and μ is the viscosity. Typically the interface moves at the speed of the local fluid or the no slip condition

$$\frac{dX}{dt} = u(X).$$

Numerical Immersed Boundary Formulation

The fluid solutions are computed on a fixed orthogonal grid while the interface is allowed to move freely without restriction from the underlying fluid grid as seen in figure 1. This allows intricate interfaces to be implemented without the need for complex body-fitted meshes.

The singular force, F, may cause ∇u and p to be discontinuous across the interface Γ. Standard finite difference methods cannot accurately find discontinuous solutions, so Peskin developed the Immersed Boundary method in which the Dirac delta function in the force F is replaced with a smooth approximate delta function [8]. The immersed boundary method generally only has first order accuracy and results in smooth solutions. So ∇u and p cannot not accurate when discontinuities occur near the immersed interface. These inaccuracies cause the movement of the interface to deteriorate over time and loose volume. This is unacceptable for the modeling applications I am interested in since the vessel movement is determined from the local fluid

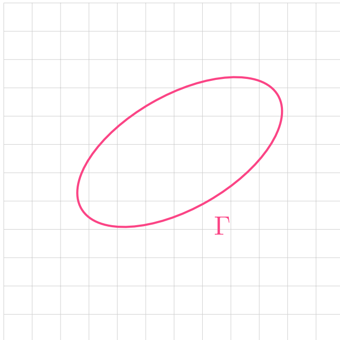

Figure 1: Underlying orthogonal fluid grid and immersed interface Γ.

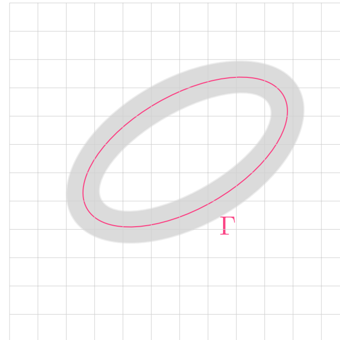

Figure 2: Immersed boundary method spreads force to the underlying fluid grid.

velocity. Therefore, I use the immersed interface method to overcome these challenges found in immersed boundary problems.

Immersed Interface Method

The Immersed Interface Method (IIM) was developed by Leveque and Li to sharply capture the discontinuities of fluid solutions and the movement of the immersed structures [5]. Denote the magnitude of the discontinuities by

$$[g](s,t) = \lim_{h\to 0} g(X(s)+hn,t) - g(X(s)-hn,t).$$

The magnitudes of the discontinuities can be computed at any point $X(s)$ on the closed immersed interface, Γ from the interfacial force and position

$$\begin{aligned} [p](s) &= f(X(s),t)\cdot n \\ [p_n](s) &= \frac{\partial}{\partial s}\left(f(X(s)),t)\cdot\tau\right) \\ [u](s) &= 0 \\ [u_n](s) &= -(f(X(s),t)\cdot\tau)\tau \end{aligned}$$

where s is the arc-length parameter of the interface, τ is the unit tangent, n is the unit normal, and f is the force strength [7].

The finite difference approximations of partial derivatives can then be corrected at grid points near the interface. In the original derivation of the jump conditions, the interface was a closed surface. Therefore in order to use this method, blood vessel have been modeled as a tube with capped ends and the flow is created by adding a fluid source and sink to opposing ends as seen in figure 3 [1, 6]. These modifications create unrealistic flow in biological models of blood vessels especially near the source and sink. I am currently working on novel extensions of the IIM to open interfaces which can provide a more natural fluid profile as can be seen in figure 4.

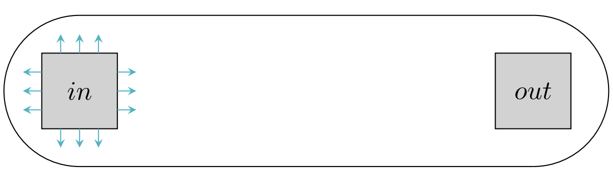

Figure 3: Closed interface model for blood flow in a vessel with source and sink terms to drive blood flow.

Figure 4: Open interface model for blood flow in a vessel with Poiseuille flow at inlet boundary.

Current Research

I began by implementing IIM for Incompressible Stokes equations in Matlab.

$$\mu \Delta \boldsymbol{u} = \nabla p + F$$

$$\nabla \cdot \boldsymbol{u} = 0$$

Stokes equation is ideal for testing my open tube immersed interface method since it is linear, steady, and has several exact solutions to use as test problems. I was able to achieve second-order accuracy in space for benchmark problems and problems with exact solutions. I proved that jump conditions for the IIM for both closed and open tubes could be computed in the same manner from the force strength function. Finally, I implemented the novel immersed interface method for an open tube for stokes equation in both rectangular and axial-symmetric cylindrical coordinates.

I am currently implementing IIM-OT for Incompressible Navier Stokes equation. Navier Stokes is appropriate for modeling more applications than Stokes equation including fluids with large Reynolds numbers. However, Navier Stokes equation has additional numerical challenges, namely nonlinear and time-dependent terms. I discretize these terms using a semi-Lagrangian method. The time-dependent and convective terms are thought of as a material derivative which computed the change of a quantity along a streamline of the fluid. This also avoids the need for correction terms when a grid point changes from one side of the interface to the other at time steps since a particle will stay on the same side of the interface taking a derivative along a streamline will not cross the interface.

Although the IIM can be applied directly to Navier-Stokes, this is not ideal since there are a cumbersome amount of correction terms to add [7]. This can be avoided by using the velocity decomposition method by Layton and Beale [2]. This method leverages the fact that the jumps in the solution for Stokes and Navier-Stokes are identical. Therefore, the solution can be broken into two parts: a part that satisfies Stokes equation with discontinuities from the interface and a regular part that satisfies Navier Stokes equation with a body force computed from the Stokes part. The Navier-Stokes solution will be smooth and can be solved with a Projection Method, which is a time splitting method that projects the fluid solution into divergent free space [3, 4]. These two solutions can then be added together to give the full solution to Navier-Stokes with an immersed interface. This builds on my previous work implementing the immersed interface method for Stokes Equation.

I now have all of the tools developed to accurately model blood flow. I plan to replace my simple interface vessel walls with walls that behave like smooth muscle cells. Ideally, these cells would exhibit rhythmic contractions as well as respond to changes in pressure similar to a myogenic response. Currently, this is a phenomenological model of the afferent arteriole without concern for many of the biological systems that control these mechanisms. For example, the vessel walls contract and dilate in response to concentrations of certain substances in the body. I would like to add terms that represent the concentrations of these vasodilators to my model.

Currently, this model assumes that the afferent arteriole has radial symmetry. Extending these models to non-symmetric models would allow me to add additional details including multiple upstream branches and downstream structures like the nephron. This would require using a mesh surface to represent the interface.

Collaborations

Due to my background in computer science, I write modular code. This encapsulation allows for collaborators with various backgrounds to participate in this research. In particular, many of the numerical methods that I program could be accessible to students with an elementary background in numerical analysis and programming. Additionally, I use Matlab for many of my models, which has a low learning curve for undergraduate students. I am also interested in interdisciplinary collaborations with personnel in biological or medical fields.

*

References

[1] Kayne M Arthurs et al. “Modeling arteriolar flow and mass transport using the immersed boundary method”. In: *Journal of Computational Physics* 147.2 (1998), pp. 402–440.

[2] J Thomas Beale and Anita T Layton. “A velocity decomposition approach for moving interfaces in viscous fluids”. In: *Journal of Computational Physics* 228.9 (2009), pp. 3358–3367.

[3] David L Brown, Ricardo Cortez, and Michael L Minion. “Accurate projection methods for the incompressible Navier–Stokes equations”. In: *Journal of computational physics* 168.2 (2001), pp. 464–499.

[4] Alexandre Joel Chorin. “Numerical solution of the Navier-Stokes equations”. In: *Mathematics of computation* 22.104 (1968), pp. 745–762.

[5] Randall J Leveque and Zhilin Li. “The immersed interface method for elliptic equations with discontinuous coefficients and singular sources”. In: *SIAM Journal on Numerical Analysis* 31.4 (1994), pp. 1019–1044.

[6] Yi Li, Ioannis Sgouralis, and Anita T Layton. “Computing Viscous Flow in an Elastic Tube.” In: *Numerical Mathematics: Theory, Methods & Applications* 7.4 (2014).

[7] Zhilin Li and Ming-Chih Lai. “The immersed interface method for the Navier–Stokes equations with singular forces”. In: *Journal of Computational Physics* 171.2 (2001), pp. 822–842.

[8] Charles S Peskin. “The immersed boundary method”. In: *Acta numerica* 11 (2002), pp. 479–517.