Numer. Math. Theor. Meth. Appl. Vol. 7, No. 4, pp. 555-574
doi: 10.4208/nmtma.2014.1303si November 2014

Computing Viscous Flow in an Elastic Tube

Yi Li, Ioannis Sgouralis and Anita T. Layton[*]

Department of Mathematics, Duke University, Durham, NC, USA.

Received 17 April 2013; Accepted 14 February 2014

Available online 11 November 2014

Abstract. We have developed a numerical method for simulating viscous flow through a compliant closed tube, driven by a pair of fluid source and sink. As is natural for tubular flow simulations, the problem is formulated in axisymmetric cylindrical coordinates, with fluid flow described by the Navier-Stokes equations. Because the tubular walls are assumed to be elastic, when stretched or compressed they exert forces on the fluid. Since these forces are singularly supported along the boundaries, the fluid velocity and pressure fields become unsmooth. To accurately compute the solution, we use the velocity decomposition approach, according to which pressure and velocity are decomposed into a singular part and a remainder part. The singular part satisfies the Stokes equations with singular boundary forces. Because the Stokes solution is unsmooth, it is computed to second-order accuracy using the immersed interface method, which incorporates known jump discontinuities in the solution and derivatives into the finite difference stencils. The remainder part, which satisfies the Navier-Stokes equations with a continuous body force, is regular. The equations describing the remainder part are discretized in time using the semi-Lagrangian approach, and then solved using a pressure-free projection method. Numerical results indicate that the computed overall solution is second-order accurate in space, and the velocity is second-order accurate in time.

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

Key words: Navier-Stokes flow, axisymmetry, immersed boundary, fluid-structure interaction.

1. Introduction

Many biological systems involve viscous flow through contracting or compliant tubes. Such examples include blood flow, food mixing and chyme movement in intestine, transport of spermatozoa in cervical canal, transport of bile in bile ducts, fluid flow through the kidney's collecting duct that is undergoing peristaltic contracts, etc. The interactions between the moving tubular walls and the luminal fluid can give rise

[*]Corresponding author. *Email addresses:* yili@math.duke.edu (Y. Li), io-sgou@math.duke.edu (I. Sgouralis), alayton@math.duke.edu (A. T. Layton)

to complex fluid dynamics. It is natural to formulate such problems in cylindrical coordinates. To lower computational cost while still capturing the 3D flow features, one may assume axisymmetry.

Numerical approaches for simulating fluid-structure interactions can be grouped into two main categories: conforming approaches, such as the Arbitrary Lagrangian Eulerian (ALE) method [5–7, 18, 19, 23], and non-conforming approaches. A popular non-conforming grid method is the immersed boundary method, which was originally developed in [21] for solving the full incompressible Navier-Stokes equations with moving boundaries, originally for studying blood flow through a beating heart [20]. Moving boundaries are represented by means of Lagrangian markers, at which boundary forces are computed. These forces are then transferred to an underlying Cartesian fluid grid via a regularization using discrete (smoothed) delta functions, and the Navier-Stokes equations are solved on the Cartesian grid.

The immersed boundary method is generally first-order accurate. To improve solution accuracy, especially near the immersed boundary, Li and co-workers developed the immersed interface method [13, 14], which captures jump discontinuities in the solution and its derivatives sharply, and generates approximations with second-order accuracy. The higher accuracy is achieved by incorporating jumps in the solution or its derivatives, which can be computed from boundary forces, into the finite difference schemes. The immersed interface method has been developed for Stokes [11, 14] and Navier-Stokes [16, 24] flows, in Cartesian [14, 16] and polar [10, 17] coordinates.

Recently we developed a numerical method for simulating driven Stokes flows through a compliant closed tube, driven by a pair of internal fluid source and sink [15]. Motivated by models of blood flow through vessels, the problem is formulated in axisymmetric cylindrical coordinates. The method decomposes the pressure and velocity fields into parts due to the tubular boundary force, which is singular, and due to the source and sink, which have compact support. Each part is computed by means of an appropriate method that is second-order accurate, and also efficient given that characteristics of that part of the solution. The singular tubular boundary force induces jump discontinuities in the solution and its derivatives. To compute this part of the solution, we use the immersed interface method, which has not been previously applied to cylindrical coordinates, and for which we derived the jump conditions in axisymmetric cylindrical coordinates. In contrast, the solution due to the fluid source and sink is smooth. If one is interested only in tracking the tubular wall movements, then the source/sink-driven solution can be efficiently calculated along the tubular surface via a boundary integral. The method is second-order accurate and robustly captures the jump discontinuities in the overall solution and its derivatives.

The method developed in [15] is limited to creeping flows or fluid with sufficiently high viscosity. However, many biofluid applications, such as blood flow in arteries and arterioles, have medium to high Reynolds number and are thus more appropriately described as Navier-Stokes flow. In the present study, we aim to extend the method to the Navier-Stokes equations, using the velocity decomposition approach [2]. That approach is motivated by two observations: first, the jump conditions in the solution and some derivatives, arising from singular forces, are the same for Stokes and Navier-Stokes equations; second, the immersed interface method is much easier to implement for the Stokes than for the Navier-Stokes equations. Given these observations, the velocity decomposition approach decomposes the overall solution into a singular piece that satisfies the Stokes equations with singular forces, and a remainder piece given by the Navier-Stokes equations with a continuous body force [2]. The singular piece can be computed accurately using the immersed interface method or boundary integrals. In the present study, the version of the immersed interface method we developed for the axisymmetric cylindrical coordinates [15] is used to compute the singular piece. The remainder piece, which is regular, is computed using standard finite difference methods and the projection method.

Numerical tests indicate that the method developed in the present study is second-order accurate in space, and second-order in time for the velocity. The sharp jumps in the solution and its derivatives are preserved robustly.

2. Model equations

We describe the model formulation and a computational method for simulating driven Navier-Stokes flow in a three-dimensional elastic tube. The computational method is based, in part, on the immersed interface method [14], which requires that the immersed boundary or surface be closed. Thus, we model the tubular wall, denoted by Γ, as a closed surface, e.g., an ellipsoid or a closed tube. To represent inflow and outflow conditions in this closed domain, we incorporate a pair of internal fluid source and sink, located near the two ends of the tube. The tube is immersed in a fluid which fills the domain $\Omega = [-R, R] \times [0, 2\pi] \times [-L, L]$, given in cylindrical coordinates. The tube is assumed to be impermeable; that is, there is no transmural fluid flux. Fluid flow is assumed to be axisymmetric, and fluid characteristics (i.e., density and viscosity) are assumed to be identical throughout Ω, inside and outside of the tube. (A model tube, configured using parameters for Example 2, is shown in Fig. 1.)

In axisymmetric cylindrical coordinates (which implicitly assumes that the velocity in the θ-direction is 0), the Navier-Stokes equations are

$$\frac{\partial u}{\partial t} + \vec{v} \cdot \nabla u = -\frac{\partial p}{\partial r} + \mu \left(\nabla^2 - \frac{1}{r^2} \right) u + F_1, \tag{2.1}$$

$$\frac{\partial w}{\partial t} + \vec{v} \cdot \nabla w = -\frac{\partial p}{\partial z} + \mu \nabla^2 w + F_3, \tag{2.2}$$

$$\nabla \cdot \vec{v} = g, \tag{2.3}$$

where $\vec{v} = (u, w)$ is the velocity, with u and w denoting the r- and z-components, respectively; p is the pressure, and μ is the viscosity. The force $\vec{F} = (F_1, F_3)$ is given by a surface integral over Γ,

$$F_i = \int_\Gamma f_i(s, \theta, t) \delta(\vec{x} - \vec{X}(s, \theta, t))\, dS$$

for $i = 1, 3$, where $\vec{x} = (r, z)$, and $\vec{f} = (f_1, f_3)$ denotes the body force density (singularly supported on Γ). $\vec{X}$ denotes the position of the tubular boundary, and (s, θ) are material coordinates that parameterize the tubular surface. g represents the sink/source strength (see below).

The vector operator notation refers to the following standard definitions

$$\begin{aligned}
\nabla p &= \left(\frac{\partial p}{\partial r}, \frac{1}{r}\frac{\partial p}{\partial \theta}, \frac{\partial p}{\partial z}\right),\\
\nabla \cdot (u, v, w) &= \left(\frac{u}{r} + \frac{\partial u}{\partial r}\right) + \left(\frac{1}{r}\frac{\partial v}{\partial \theta}\right) + \frac{\partial w}{\partial z},\\
\nabla^2 u &= \frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial u}{\partial r}\right) + \frac{1}{r^2}\frac{\partial^2 u}{\partial \theta^2} + \frac{\partial^2 u}{\partial z^2}.
\end{aligned}$$

Given the axisymmetry assumption, the θ-terms drop out.

Homogeneous Dirichlet boundary conditions are assumed for the velocity field along $\partial\Omega$. No-slip boundary conditions are imposed along Γ, so that the tubular walls move at the velocity of the local fluid, i.e.,

$$\frac{\partial \vec{X}}{\partial t} = \vec{v}\left(\vec{X}, t\right). \tag{2.4}$$

2.1. Definition of source, sink and external source/sink

Rather than enforcing incompressibility everywhere, Eq. (2.3) includes the source/sink term g that drives flow through the tube. The function g is compactly supported in three separate regions: one simulates a fluid source and another simulates a fluid sink. These two regions are located inside the tube, close to its two ends. A third region, located outside the tube, is used to maintain overall fluid incompressibility in Ω.

We first describe the two interior fluid sink/source terms. The source term, denoted g_1, is compactly supported in a cylindrical region denoted by S_{source}, which is centered at the point (r_1, z_1), such that $|r - r_1| \leq \kappa$ and $|z - z_1| \leq \kappa$, for fixed width κ. The sink term, denoted g_2, is compactly supported in S_{sink}, which is centered at (r_2, z_2), and is similarly defined.

The source g_1 and sink g_2 are defined as

$$g_1(r, z, t) = A_1(t)\gamma_1(r, z), \tag{2.5}$$

$$g_2(r, z, t) = A_2(t)\gamma_2(r, z), \tag{2.6}$$

$$\gamma_1(r, z) = \begin{cases} h_1(r)k_1(z), & (r, z) \in S_{\text{source}},\\ 0, & \text{otherwise}, \end{cases} \tag{2.7}$$

$$\gamma_2(r, z) = \begin{cases} h_2(r)k_2(z), & (r, z) \in S_{\text{sink}},\\ 0, & \text{otherwise}, \end{cases} \tag{2.8}$$

where, for $i = 1, 2$, $h_i(r)$ and $k_i(z)$ satisfy the following properties

$$\int_0^{2\pi} \int_{r_i}^{r_i+\kappa} h_i(r) \cdot r \, dr \, d\theta = 1, \tag{2.9}$$

$$\int_{-\kappa}^{\kappa} k_i(z) \, dz = 1. \tag{2.10}$$

Eqs. (2.9) and (2.10) do not uniquely define h_i and k_i. We assume that h_i and k_i are scaled cosine functions

$$h_i(r) = \beta \left(1 + \cos \left(\frac{\pi(r - r_i)}{\kappa} \right) \right), \tag{2.11}$$

$$k_i(z) = \frac{1}{2\kappa} \left(1 + \cos \left(\frac{\pi(z - z_i)}{\kappa} \right) \right), \tag{2.12}$$

where the constant β is chosen such that Eq. (2.9) is satisfied.

If the source/sink strengths A_1 and A_2 cancel, i.e., if $A_1(t) + A_2(t) = 0$ for all time t, then overall fluid volume is conserved. To impose overall incompressibility for the case where $A_1 + A_2 \neq 0$, we follow the approach in Ref. [1] and add an external source/sink outside of the tube, such that its strength is given by

$$A_3(t) = -A_1(t) - A_2(t). \tag{2.13}$$

Similar to the internal source and sink, the external source/sink is assumed to be compactly supported on S_{ext}, which is a region centered at (r_3, z_3) with radius κ_3. The external source/sink term g_3 is defined by

$$g_3(r, z, t) = A_3(t)\gamma_3(r, z), \tag{2.14}$$

$$\gamma_3(r, z) = \begin{cases} h_3(r)k_3(z), & (r, z) \in S_{\text{ext}}, \\ 0, & \text{otherwise}, \end{cases} \tag{2.15}$$

with $h_3(r)$ and $k_3(z)$ satisfying

$$\int_0^{2\pi} \int_{r_3}^{r_3+\kappa_3} h_3(r) \cdot r \, dr \, d\theta = 1, \tag{2.16}$$

$$\int_{-\kappa_3}^{\kappa_3} k_3(z) \, dz = 1, \tag{2.17}$$

where $h_3(r)$ and $k_3(z)$ are assumed to be scaled cosine functions analogous to Eqs. (2.11) and (2.12).

In the current model, the strength $A_1(t)$ is prescribed, whereas $A_2(t)$ can be obtained by the Hagen-Poiseuille equation

$$A_2(t) = \frac{p_2 - \bar{p}_2(t)}{R_2}. \tag{2.18}$$

This relation assumes that the tube is attached to a downstream resistor R_2, at the end of which pressure is fixed at p_2. The resistance R_2 and end pressure p_2 are assumed known. $\bar{p}_2$ is the difference between average pressure over S_{sink} and over S_{ext}, i.e.,

$$\bar{p}_2(t) = \int_\Omega p(r,z,t)\big(\gamma_2(r,z) - \gamma_3(r,z)\big)\, dV. \tag{2.19}$$

Taken together, the sink/source term g is given by

$$g(r,z,t) = g_1(r,z,t) + g_2(r,z,t) + g_3(r,z,t). \tag{2.20}$$

3. Computational method

We aim to compute fluid velocity and pressure on a fixed, Eulerian grid, and to use a moving, Lagrangian frame of reference to track the location of the interface Γ over time. Owing to the assumption of axisymmetry, we compute the solution field only for the right half of the domain Ω, i.e., $r \geq 0$; this subdomain is denoted $\hat{\Omega}$. And we enforce symmetry for the left half, i.e., $r < 0$.

To discretize the model equations on $\hat{\Omega}$, we impose the following boundary conditions along $r = 0$ as implied by symmetry

$$u(0,z) = 0, \quad \frac{\partial w}{\partial r}(0,z) = 0, \quad \frac{\partial p}{\partial r}(0,z) = 0. \tag{3.1}$$

The axisymmetric assumption also allows us to represent the surface Γ by its cross section at $\theta = 0$ (or any given θ). We will call the boundary (curve) associated with this cross section Γ as well, with the distinction between curve Γ and surface Γ made clear by the context. Even though we only compute solution in the right half $\hat{\Omega}$ of the domain, we keep Γ as a closed curve so we can impose periodicity. Thus, Γ spans over the entire cross-section of Ω. The curve Γ is represented by markers that move independently of the fixed fluid grid, interpolated by periodic cubic splines [14].

To compute the solution field, we follow the velocity decomposition approach of [2]. Recall that the velocity decomposition approach is motivated by the key observation that the jump conditions for the solution and its derivatives, for given singular forces, are the same for Stokes and Navier-Stokes equations (albeit the actual solutions differ). Thus, we decompose the overall solution into a “Stokes” (or “singular”) part and a “remainder” (or “regular”) part

$$\vec{v} = \vec{v}_s + \vec{v}_{\text{reg}}, \tag{3.2}$$

$$p = p_s + p_{\text{reg}}. \tag{3.3}$$

Here the subscripts “s” and “reg” denote the Stokes and remainder parts, respectively. The Stokes solution $\vec{v}_s$ and p_s satisfy the Stokes equation with the singular forces. Thus, the jump discontinuities are “captured” by the Stokes solution. The remainder solution $\vec{v}_{\text{reg}}$ and p_{reg} satisfy the Navier-Stokes equations with a continuous body force. More details are given below. This decomposition allows us to extend the method that we previously developed for Stokes [15] to Navier-Stokes flow.

3.1. Stokes part

As previously noted, the Stokes part of the solution satisfies the Stokes equations with the boundary force and divergence-free condition

$$\mu\left(\nabla^2-\frac{1}{r^2}\right)u_s-\frac{\partial p_s}{\partial r}+F_1=0, \tag{3.4}$$

$$\mu\nabla^2 w_s-\frac{\partial p_s}{\partial z}+F_3=0, \tag{3.5}$$

$$\nabla\cdot\vec{v}_s=0. \tag{3.6}$$

Note that Eq. (3.6) does not contain the source/sink term g. Along $r=0$, as implied by symmetry, homogeneous Neumann boundary conditions are imposed on p_s, w_s, and homogeneous Dirichlet boundary conditions on u_s. These boundary conditions are analogous to Eq. (3.1). Across the other three sides of $\hat{\Omega}$, homogeneous Dirichlet boundary conditions are assumed for all three variables.

The jump conditions for p_s and $\vec{v}_s$, which have been derived in [15], are similar to those in Cartesian coordinates

$$[p_s]=f_n,\qquad\left[\frac{\partial p_s}{\partial n}\right]=\frac{1}{r}\frac{\partial(f_\tau r)}{\partial s}, \tag{3.7}$$

$$\left[\mu\frac{\partial u_s}{\partial n}\right]=f_\tau\sin\alpha,\qquad\left[\mu\frac{\partial w_s}{\partial n}\right]=-f_\tau\cos\alpha. \tag{3.8}$$

Here f_n and f_τ are the normal and tangential boundary force densities, respectively, and α denotes the angle between the normal- and r-directions. The Stokes solution can be computed by means of the immersed interface method described in [15]. In that approach, Eqs. (3.4)-(3.6) are reduced into a sequence of three Poisson problems, one for each variable. For the pressure,

$$\nabla^2 p=\nabla\cdot\vec{F}, \tag{3.9}$$

which can be solved by incorporating the jump conditions for pressure and its derivatives (Eq. (3.7)) into the finite difference stencils, and then neglecting the dipole source and solving the homogeneous version of the above equation. Once p is computed, Eqs. (3.4) and (3.5) are independent Poisson problems for u and w, respectively.

3.2. Remainder part

Taking the difference of Eqs. (2.1)-(2.3) and Eqs. (3.4)-(3.6), one obtains the equation for the regular part of the velocity and pressure

$$\frac{\partial u_{\rm reg}}{\partial t}+\vec{v}\cdot\nabla u_{\rm reg}=-\frac{\partial p_{\rm reg}}{\partial r}+\mu\left(\nabla^2-\frac{1}{r^2}\right)u_{\rm reg}+F_{b1}, \tag{3.10}$$

$$\frac{\partial w_{\rm reg}}{\partial t}+\vec{v}\cdot\nabla w_{\rm reg}=-\frac{\partial p_{\rm reg}}{\partial z}+\mu\nabla^2 w_{\rm reg}+F_{b3}, \tag{3.11}$$

$$\nabla\cdot\vec{v}_{\rm reg}=g, \tag{3.12}$$

where F_{b1} and F_{b3} are the body forces given by the material derivative of the Stokes velocity

$$F_{b1} = -\frac{\partial u_s}{\partial t} - \vec{v} \cdot \nabla u_s, \tag{3.13}$$

$$F_{b3} = -\frac{\partial w_s}{\partial t} - \vec{v} \cdot \nabla w_s. \tag{3.14}$$

Note that the advection terms in Eqs. (3.10), (3.11), (3.13), and (3.14) involve the overall velocity $\vec{v}$, not the remainder or Stokes velocity.

Given the boundary conditions we have chosen for $\vec{v}_s$, the boundary conditions imposed on $\vec{v}_{\rm reg}$ are the same as those of $\vec{v}$.

To avoid discretizing $\nabla \cdot \vec{v}_{\rm reg}$, we solve Eqs. (3.10) and (3.11) using the semi-Lagrangian method [3, 22], which integrates backward along fluid trajectories and computes function values at mesh points for each time level. In the semi-Lagrangian discretization, the advection term is incorporated into the material derivative, which yields

$$\frac{du_{\rm reg}}{dt} = -\frac{\partial p_{\rm reg}}{\partial r} + \mu \left(\nabla^2 - \frac{1}{r^2}\right) u_{\rm reg} + F_{b1}, \tag{3.15}$$

$$\frac{dw_{\rm reg}}{dt} = -\frac{\partial p_{\rm reg}}{\partial z} + \mu \nabla^2 w_{\rm reg} + F_{b3}. \tag{3.16}$$

The body forces are also similarly written as material derivatives

$$F_{b1} = -\frac{du_s}{dt}, \quad F_{b3} = -\frac{dw_s}{dt}. \tag{3.17}$$

To discretize Eqs. (3.15) and (3.16), we use the second-order backward difference formula,

$$\frac{3u^{n+1}_{\rm reg} - 4\tilde{u}^n_{\rm reg} + \tilde{u}^{n-1}_{\rm reg}}{2\Delta t} = -\frac{\partial p^{n+1}_{\rm reg}}{\partial r} + \mu \left(\nabla^2 - \frac{1}{r^2}\right) u^{n+1}_{\rm reg} + F^{n+1}_{b1}, \tag{3.18}$$

$$\frac{3w^{n+1}_{\rm reg} - 4\tilde{w}^n_{\rm reg} + \tilde{w}^{n-1}_{\rm reg}}{2\Delta t} = -\frac{\partial p^{n+1}_{\rm reg}}{\partial z} + \mu \nabla^2 w^{n+1}_{\rm reg} + F^{n+1}_{b3}. \tag{3.19}$$

In Eq. (3.18), $\tilde{u}^n_{reg}$ and $\tilde{u}^{n-1}_{reg}$ denote the r-component of the velocity at the upstream positions $(\vec{x}^n, t_n)$ and $(\vec{x}^{n-1}, t_{n-1})$, respectively. Analogous notation is used in Eq. (3.19) for w. The same discretization is also applied to evaluate the body forces F_{b1} and F_{b3}.

To evaluate $\tilde{u}^n_{\rm reg}$ and $\tilde{u}^{n-1}_{\rm reg}$, we first estimate $\vec{x}^n$ and $\vec{x}^{n-1}$, given by the initial value problem

$$\frac{d\vec{x}(t)}{dt} = \vec{v}\big(\vec{x}(t), t\big), \quad \vec{x}(t_{n+1}) = \vec{x}_0, \tag{3.20}$$

where the initial points $\vec{x}_0$ is a regular mesh. We estimate the upstream points $\vec{x}^n$ and $\vec{x}^{n-1}$ by integrating Eq. (3.20) backward in time over the integral $[t_{n+1}, t_n]$ and

$[t_{n+1}, t_{n-1}]$, respectively, using the midpoint method

$$\vec{x}_* = \vec{x}_0 - \frac{\Delta t}{2}\vec{v}\left(\vec{x}_0 - \frac{\Delta t}{2}\vec{u}^{n+\frac{1}{2}}, t_{n+\frac{1}{2}}\right), \quad \vec{x}^n = \vec{x}_0 - \Delta t\,\vec{v}\left(\vec{x}_*, t_{n+\frac{1}{2}}\right), \tag{3.21}$$

$$\vec{x}_{**} = \vec{x}_0 - \Delta t\,\vec{v}(\vec{x}_0 - \Delta t\,\vec{v}^n, t_n), \qquad \vec{x}^{n-1} = \vec{x}_0 - 2\Delta t\,\vec{v}(\vec{x}_{**}, t_n). \tag{3.22}$$

Because $\vec{v}$ values are required at off-mesh points, spatial interpolations are required (explained below).

Eqs. (3.12), (3.18), and (3.19) are solved by means of a second-order pressure-free projection method [9]. We first solve the Helmholtz-type equations for the intermediate velocity $\vec{v}_*$

$$\left(\frac{3}{2\Delta t} - \mu\left(\nabla^2 - \frac{1}{r^2}\right)\right)u_* = \frac{-4\tilde{u}^n_{\text{reg}} + \tilde{u}^{n-1}_{\text{reg}}}{2\Delta t} + F^{n+1}_{b1}, \tag{3.23}$$

$$\left(\frac{3}{2\Delta t} - \mu\nabla^2\right)w_* = \frac{-4\tilde{w}^n_{\text{reg}} + \tilde{w}^{n-1}_{\text{reg}}}{2\Delta t} + F^{n+1}_{b3}, \tag{3.24}$$

with the following boundary conditions

$$u_*(0,z) = 0, \qquad \frac{\partial w_*}{\partial r}(0,z) = 0,$$

and along the other three sides of $\hat{\Omega}$ [4]

$$\vec{v}_* \cdot \hat{n} = 0, \qquad \vec{v}_* \cdot \hat{t} = \Delta t \nabla\phi^n \cdot \hat{t},$$

where $\hat{n}$ and $\hat{t}$ denote the normal and tangential directions on $\partial\hat{\Omega}$.

To project $\vec{v}_*$ onto the divergence-free space, we compute ϕ^{n+1} which satisfies

$$\vec{v}^{n+1}_{\text{reg}} = \vec{v}_* - \Delta t \nabla\phi^{n+1}. \tag{3.25}$$

Taking the divergence of Eq. (3.25), we get

$$\Delta t \nabla^2\phi^{n+1} = \nabla\cdot\vec{v}_* - g^{n+1}. \tag{3.26}$$

Note that from Eq. (2.20), g^{n+1} depends on g_2^{n+1} (and thus A_2^{n+1}), which is a function of p^{n+1}, see Eq. (2.18). Because p^{n+1} has yet to be computed, the right-hand-side of Eq. (3.26) is unknown. To compute ϕ^{n+1}, instead of solving Eq. (3.26) directly, we decompose ϕ^{n+1} into three parts

$$\phi^{n+1} = \phi_1^{n+1} + A_1^{n+1}\phi_2 + A_2^{n+1}\phi_3, \tag{3.27}$$

where $A_1^{n+1}\phi_2$ and $A_2^{n+1}\phi_3$ are due to the source and sink forcings, respectively, and ϕ_1^{n+1} is the remainder part. From Eq. (2.13), g^{n+1} can be rewritten as

$$g^{n+1} = A_1^{n+1}(\gamma_1(r,z) - \gamma_3(r,z)) + A_2^{n+1}(\gamma_2(r,z) - \gamma_3(r,z)). \tag{3.28}$$

Therefore, ϕ_1^{n+1}, ϕ_2, and ϕ_3 satisfy the following equations

$$\Delta t \nabla^2 \phi_1^{n+1} = \nabla \cdot \vec{v}_*, \tag{3.29}$$

$$\Delta t \nabla^2 \phi_2 = \gamma_1(r,z) - \gamma_3(r,z), \tag{3.30}$$

$$\Delta t \nabla^2 \phi_3 = \gamma_2(r,z) - \gamma_3(r,z). \tag{3.31}$$

Homogeneous Neumann boundary conditions are imposed on ϕ_1, ϕ_2, and ϕ_3. Because ϕ_2 and ϕ_3 depend only on time-independent functions δ_1, δ_2, and δ_3, those two ϕ's are independent of time. This implies that Eqs. (3.30) and (3.31) only need to be solved once.

After evaluating ϕ_1^{n+1}, the regular pressure at t^{n+1} is obtained by

$$p_{\text{reg}}^{n+1} = \frac{3}{2}\phi^{n+1} - \mu \Delta t \nabla^2 \phi^{n+1} \tag{3.32}$$

$$= \frac{3}{2}\left(\phi_1^{n+1} + A_1^{n+1}\phi_2 + A_2^{n+1}\phi_3\right) - \mu(\nabla \cdot \vec{v}_* - g^{n+1}). \tag{3.33}$$

Note that whenever A_2^{n+1} depends on p_{reg}^{n+1}, the above equation defines p_{reg}^{n+1} implicitly, and needs to be solved for p_{reg}^{n+1}.

3.3. Advancing boundary position

To advance the boundary configuration Γ from t_n to t_{n+1}, we update the boundary markers as follows

$$\vec{X}^{n+1} = \vec{X}^n + \Delta t \left(\frac{3}{2}\vec{v}^n(\vec{X}^n) - \frac{1}{2}\vec{v}^{n-1}(\vec{X}^{n-1}) \right). \tag{3.34}$$

The boundary velocity values $\vec{v}(\vec{X})$ are approximated by means of bilinear spatial interpolation, with care taken near Γ, switching to extrapolation so that the interpolation stencils never cross the boundary.

4. Numerical examples

4.1. Convergence test 1: A relaxing ellipsoid

To demonstrate the second-order spatial accuracy of our method, we first consider a simple example in which a spherical, elastic surface is deformed into an ellipsoidal shape. This example does not include source and sink; i.e., $\nabla \cdot \vec{v} = 0$ throughout Ω. At a given plane angle θ, the boundary of the ellipsoid is described parametrically by $(0.8\cos(\theta), 0.6\sin(\theta))$. The equilibrium shape for this cross-section boundary is a circle of radius 0.6, centered at the origin. The fluid is initialized to be at rest. Fluid viscosity μ is 0.1.

The ellipsoid boundary Γ is assumed to be elastic. As it is stretched or compressed, an elastic tension force $\vec{f}$ is induced

$$\vec{f} = \frac{\partial T}{\partial \hat{t}} \hat{t} - 2T\kappa\hat{n}, \tag{4.1}$$

where $T(s,t)$ is the tension

$$T(s,t) = T_0 \left(\left| \frac{\partial \vec{X}}{\partial s_0} \right| - 1 \right),$$

where s_0 denotes the material coordinate. We use elasticity constant $T_0 = 0.2$. A derivation for the tension force in cylindrical coordinates can be found in Ref. [12].

In this example, the fluid computational domain is $\hat{\Omega} = [0,1.5] \times [-1.5,1.5]$. The solution field is computed with uniform grid spacing $h = 1.5/N$, for $N =$ 160, 320, and 640. A time-step of $\Delta t = 0.001$ was used to integrate the system from $t = 0$ to $t = 0.3$. The immersed boundary is discretized using 88 markers. We use the solution computed on a high-resolution grid ($N =$ 1280) as the reference solution to estimate errors. Key convergence results are displayed in Table 1, where we estimate the dependence of solution errors on spatial discretization by computing the solution errors, and then dividing the errors by h^2. Results in Table 1 show that the resulting ratio is approximately constant (in particular, non-increasing), which indicates second-order spatial accuracy.

Table 1: Convergence results for pressure and velocity, obtained for the relaxing ellipsoid. Approximations exhibit second-order convergence. (err$_1$ and err$_\infty$ denote the L_1 and L_∞ errors, respectively.)

	p		u		w	
N	err$_1/h^2$	err$_\infty/h^2$	err$_1/h^2$	err$_\infty/h^2$	err$_1/h^2$	err$_\infty/h^2$
160	4.082e-1	2.571	1.243e-2	3.160e-1	2.937e-2	3.184e-1
320	2.516e-1	1.903	1.204e-2	3.224e-1	1.556e-2	2.727e-1
640	1.598e-1	1.378	1.013e-2	2.351e-1	1.596e-2	2.350e-1

4.2. Convergence test 2: Flow through an elastic tube

In the previous example, the ellipsoid relaxes to its equilibrium shape of a sphere. Because after a sufficiently long simulation, the fluid settles to rest, that problem is not ideal for studying the temporal error of the method. To more thoroughly understand the accuracy of our method, we perform a second convergence test using an elastic tube with oscillating source and sink. Fig. 1 shows the configuration of the tube. The source and sink are centered at $(0.0, 2.0)$ and $(0.0, -2.0)$, respectively, with radius parameter $\kappa = 0.0225$.

At equilibrium, the top of the tube is the upper half of the circle with radius 0.2 centered at the source, and similarly, the bottom of the tube is the lower half of the

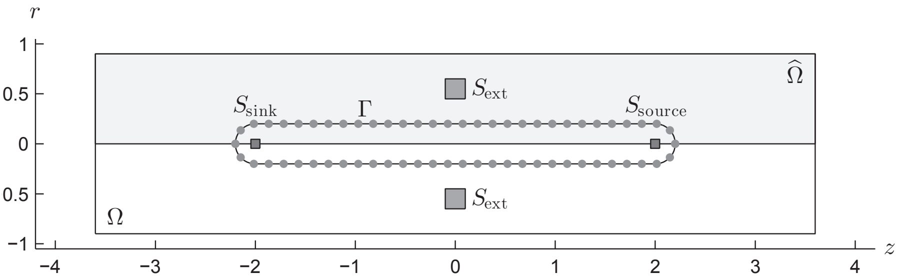

Figure 1: Tube configuration. A cross section of the fluid region Ω is shown, which corresponds to the larger rectangle. Darker region denotes the computational domain $\hat{\Omega}$. Fluid source, S_{source}; sink, S_{sink}; external source/sink, S_{ext}.

circle with radius 0.2 centered at the sink. The left and right walls of the tube are the line segments $r = 0.2$ and $r = -0.2$ between $z = -2.0$ and $z = 2.0$.

The boundary force along Γ consists of two parts

$$\vec{f} = \vec{f}_e + \vec{f}_t.$$

The elastic tension force $\vec{f}_e$ has two components. The first component $\vec{f}_{e1}$ arises from tension along the z-direction and is given by Eq. (4.1), with the elasticity constant T_0 set to 0.2. The second component $\vec{f}_{e2}$ points in the radial direction and arises from circumference tension and is defined as

$$f^r_{e2} = m\,(r(s,t) - L_T)\,\hat{r}, \tag{4.2}$$

where $\hat{r}$ denotes the unit vector in the r-direction. Here L_T is the equilibrium width of the tube, and m is the force constant. In this example, $L_T = 0.4$ and $m = 100$.

To prevent the tube from moving along with the flow, we add the tether force $\vec{f}_t$

$$\vec{f}_t = -k(\vec{X} - \vec{X}_{eq}),$$

with spring force $k = 100$, and $\vec{X}_{eq}$ the equilibrium position described above.

The strength of the driving source is given by the periodic function

$$A_1(t) = 0.005(1.0 + \sin(\omega t)). \tag{4.3}$$

In this example we assume that the sink strength is prescribed to be opposite to the source, i.e. $A_2(t) = -A_1(t)$. Because the sink and source cancel, an external sink/source term is not necessary, i.e., $A_3(t) = 0$. The driving frequency ω is set to 30. The system is integrated from $t = 0$ to $t = 0.3$, which approximately corresponds to ten periods. The fluid is initialized to be at rest, and the tube at the equilibrium shape. Fluid viscosity μ is 0.1.

The fluid domain is $\hat{\Omega} = [0, 0.9] \times [-3.6, 3.6]$. Fluid solution is computed with uniform grid spacing $h = 3.6/N$, for $N = 160$, 320, and 640. The immersed boundary is discretized with 320 markers. We use the solution computed on a high-resolution grid ($N = 1280$) as the reference solution to estimate errors. A time-step $\Delta t = 0.001$ was used. The ratios of solution errors over h^2 are displayed in Table 2. Those ratios, which are approximately constant (mostly non-increasing), indicate second-order spatial accuracy.

Table 2: Results for pressure and velocity for spatial convergence test 2. Approximations exhibit second-order convergence. (err_1 and err_∞ denote the L_1 and L_∞ errors, respectively.)

	p		u		w	
N	err_1/h^2	err_∞/h^2	err_1/h^2	err_∞/h^2	err_1/h^2	err_∞/h^2
160	19.32	214.3	1.941e-1	4.043	3.031e-1	6.643
320	24.72	253.2	9.740e-1	3.752	3.784e-1	5.990
640	20.71	217.9	6.260e-2	5.532	2.615e-1	5.674

We also conduct a time convergence test. A fixed spatial grid of $N = 320$ is used; solutions are computed for Δt = 0.004, 0.002, and 0.001. The error is assessed at time $t = 0.2$. We use the solution computed using Δt =0.0005 as the reference solution. The ratios of solutions errors over Δt^2 are displayed in Table 3. The ratios corresponding to u and w are approximately constant, which indicate second-order temporal accuracy. However, the ratios corresponding p exhibits graduate increase, although they don't double as Δt is doubled. These results suggest super-linear temporal convergence for pressure.

Table 3: Results for pressure and velocity for temporal convergence test 2. Velocity exhibits second-order convergence in time whereas pressure exhibits super-linear convergence. (err_1 and err_∞ denote the L_1 and L_∞ errors, respectively.)

	p		u		w	
Δt	$\mathrm{err}_1/\Delta t^2$	$\mathrm{err}_\infty/\Delta t^2$	$\mathrm{err}_1/\Delta t^2$	$\mathrm{err}_\infty/\Delta t^2$	$\mathrm{err}_1/\Delta t^2$	$\mathrm{err}_\infty/\Delta t^2$
0.004	1.564e-1	1.328	1.836e-3	4.344e-1	4.165e-3	2.187e-1
0.002	2.749e-1	1.834	1.854e-3	4.422e-1	4.355e-3	2.705e-1
0.001	3.727e-1	2.164	1.542e-3	3.288e-1	4.183e-3	2.663e-1

4.3. Flow characteristics study

We conduct a set of simulations to study the characteristics of flow along the elastic tube. In the previous example, the sink strength is assumed known *a priori*. In this example, we assume the same source strength $A_1(t)$ (Eq. (4.3)), but compute the sink strength using Eq. (2.18). Because the sink and source strengths do not necessarily cancel, an external source/sink, given by Eq. (2.13), is applied outside of the tube to ensure incompressibility. The external source/sink is centered at $(0.55, 0.0)$ with radius parameter $\kappa_3 = 0.1$. The downstream resistance and pressure used to compute the sink strength A_2 are set to $R_2 = 8.2542$ and $p_2 = -2$.

4.3.1. Poiseuille characteristics

We first study the flow characteristics of our model. In particular, we assess the extent to which the tubular flow generated by the oscillating sink can be approximated by Poiseuille flow. We perform a simulation in which we set the driving frequency of the oscillating source to $\omega = 20$, and access the velocity field and shear stress after the system reaches equilibrium. Cross sections, at $z = 0.99$, of the z-component of the velocity field and shear stress are shown in Figs. 2 and 3. Results are qualitatively similar at the other cross sections. The approximately parabolic shape of the z-velocity (Fig. 2) and the V-shaped shear stress (Fig. 3) are both consistent with Poiseuille flow, a result that is expected for laminar flow through an approximately circular pipe. Note also that the jump discontinuity in the velocity derivative is captured robustly by our numerical method.

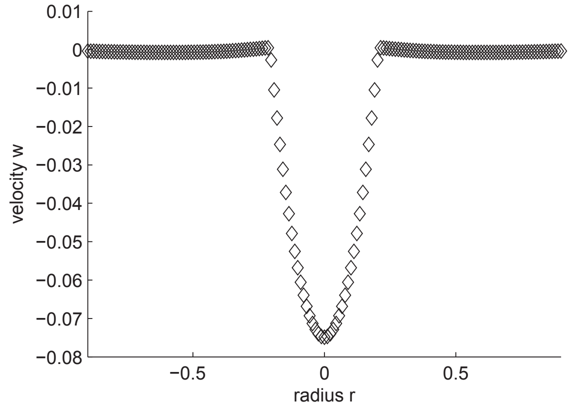

Figure 2: Velocity in the z-direction at $z = 0.99$. The approximately parabolic profile is characteristic of Poiseuille flow.

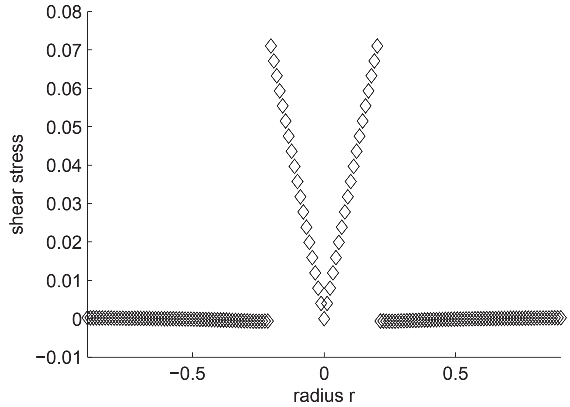

Figure 3: Shear stress at $z = 0.99$. V-shape is characteristic of Poiseuille flow.

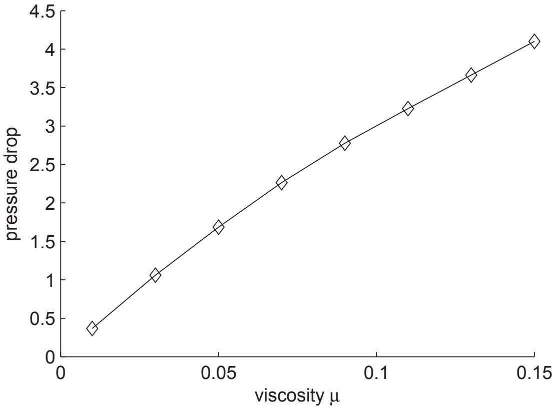

Figure 4: Pressure drop from $z = 1.8$ to $z = -1.8$ for different μ. An approximately linear relation, characteristic of Poiseuille flow, is observed.

We also perform a set of simulations using the same parameters as above, but varied fluid viscosity μ over the range 0.01–0.15. For each value of μ, we computed the pressure drop along the tube, from position $(0, 1.8)$ to $(0, -1.8)$. Results, shown in Fig. 4, indicate that pressure drop is an approximately linear function of fluid viscosity, a prediction that is again consistent with Poiseuille flow.

4.3.2. Frequency-amplitude study

In the next set of simulations, we study the response of the fluid at driving frequencies ω over the range 5–80 and measure the pressure and velocity of the generated flow. Due to the oscillations of the driving force, fluid pressure and flow oscillate in time as well. Fig. 5 shows the maximum and minimum pressure values recorded over one period $(1/\omega)$ at position $(0, 1.8)$ as a function of driving frequency ω. Solution at other locations exhibit qualitatively similar responses (not shown). Our model predicts a nonlinear dependence of pressure amplitude on frequency: pressure amplitude decreases as ω increases from 5 to between 10–20, and then increases as ω further increases. Note that the amplitude of the forcing oscillations is fixed.

Fig. 6 shows the maximum and minimum flow velocity values, obtained by integrating w at $(0, 0.99)$ in time over one oscillation period $(1/\omega)$. For all driving frequencies, fluid flows in the $-z$ direction, i.e., no flow reversal. The flow amplitude in Fig. 6 exhibits similar (nonlinear) trends as the pressure amplitude, although the flow amplitude increase is not as substantial as that for pressure.

4.3.3. Viscosity-amplitude study

In the last set of numerical simulations, we consider the effect of fluid viscosity on flow characteristics. For a fixed driving frequency $\omega = 20$, we vary viscosity over the range

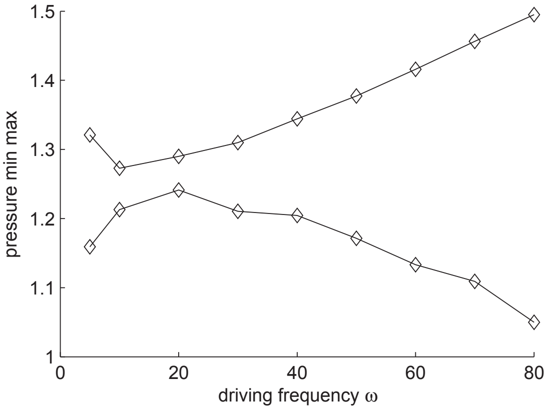

Figure 5: Minimum and maximum of pressure cycle at $(r,z)=(0,1.8)$ for driving frequency $5\le\omega\le 80$.

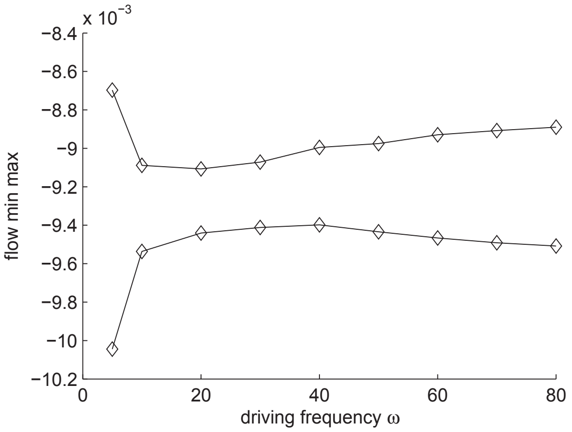

Figure 6: Minimum and maximum of flow across $z=0.99$ for driving frequency $5\le\omega\le 80$.

0.01–0.15, and compute average pressure and velocity values at $(0,1.8)$. Average velocity is measured by integrating the z-component at $(0,0.99)$ over one period $(1/\omega)$. Average pressure is computed analogously. Again, the model predicts that both pressure and velocity amplitudes depend nonlinearly on fluid viscosity. As shown in Figs. 7 and 8, average pressure and flow values initially decrease as μ increases from $\mu=0.01$ to $\mu\sim 0.04$–0.06. Afterward, as μ increases, so do pressure and velocity.

5. Discussion

We present a numerical method for computing fluid flow through an elastic tube. The method is an extension of our previous work on Stokes flow [15] and is based on the velocity decomposition approach [2]. The method yields approximations that exhibit approximately second-order accuracy (except for temporal accuracy for pressure, which is super-linear), as suggested by the numerical examples, even though grid

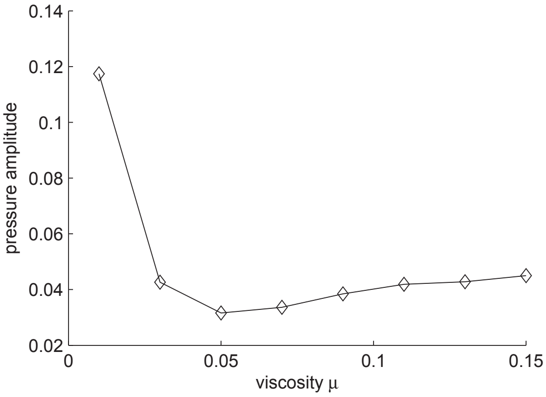

Figure 7: Pressure amplitude at $(r, z) = (0, 1.8)$ for viscosity $0.01 \le \mu \le 0.15$.

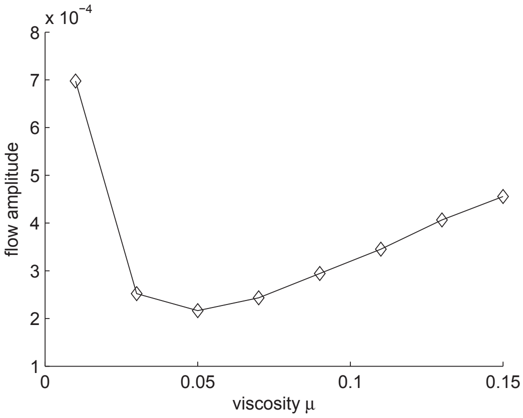

Figure 8: Amplitude of flow across $z = 0.99$ for viscosity $0.01 \le \mu \le 0.15$.

values of the fluid quantities are corrected near the interface only for the Stokes part of the solution. We consider this a great advantage, since these corrections are much simpler to make than those for the full problem. It is noteworthy that even though the velocity decomposition approach essentially "extends" a Stokes solution to one that satisfies the Navier-Stokes equations, it does not require the Stokes and Navier-Stokes solutions to be sufficiently similar (or, that the remainder solution be sufficiently small). Consequently, the approach is applicable even at high Reynolds numbers.

Our method is computationally efficient: at each time step, the most costly computations are the solution of the three Poisson problems associated with the Stokes problem (e.g., Eq. (3.9)), and the solution of the Helmholtz-type equations (3.23) and (3.24) and the projection step (Eq. (3.25)) in the Navier-Stokes problem. Those solutions are computed using fast sine or cosine transforms (depending on the boundary conditions), and thus cost $\mathcal{O}(N^2 \log N)$.

It is assumed in this work that the fluid is the same inside and outside of the interface. If instead the interface separates two Navier-Stokes fluids with different viscosities and densities, the corresponding velocity component in the r-direction will no longer be regular, although the right hand side of its evolution equation will be less singular than the interfacial force. It remains to be seen whether the decomposition would reduce the difficulty of the two-fluid problem.

How the flow fluctuation amplitude varies as a function of the driving force depends on a number of factors, including fluid properties, forcing frequency, how the amplitude, elastic properties of the tube, etc. In our previous simulation study of Stokes flows [15], the model predicted pressure and flow fields that settle (i.e., fluctuation amplitudes decrease) as the pumping speed increases. In contrast, the Navier-Stokes model of the present study predicts a nonlinear depends on flow oscillations on forcing frequency, such that at sufficiently high forcing frequency, the amplitude of pressure and flow velocity oscillations increases as forcing frequency increases (see Figs. 5 and 6). These contrasting results highlight the qualitatively different behaviors of Stokes and Navier-Stokes flows.

Simulation studies by other groups (e.g., [8]) have demonstrated pumped flow that changes nonlinearly with forcing function frequency, even reversing direction for certain frequency ranges. In contrast, while our model predicts a nonlinear dependence, none of our simulations generate reversed flow. The discrepancy between the results of the present study and Ref. [8] may be attributed to the different means by which fluid flow is driven: in Ref. [8] the active contractions of the elastic tube pumps fluid along, whereas the present model drives flow via a pair of internal fluid sink and source.

Our principal goal of developing the present method is to simulate water transport along compliant biological tubules, such as the contracting collecting ducts in the kidney, which have been mentioned in the Introduction. Because the epithelial walls of these tubules are water permeable, and because transepithelial water fluxes can be driven by hydrostatic pressure, it is crucial that pressure jumps across the tubular wall be computed accurately. Indeed, that requirement is our motivation for developing the present method, which computes second-order accurate solution and, perhaps more importantly, captures their jumps sharply. In contrast, the immersed boundary method smears out the pressure jumps and exhibits $\mathcal{O}(1)$ errors for pressure near the interface.

Another contribution of the present study is to show that fluid flow along an elastic tube can be reasonably approximated by Poiseuille flow. That is true provided that the tube does not deviate substantially from a cylinder. That condition is likely satisfied by some biological tubes, such as those in the kidney, which are tightly packed and held in place by the interstitial matrix, so that tubular dilation or constriction is limited. This observation implies that for a complex model of a biological system, tubular flows may be described by Poiseuille flow, rather than Navier-Stokes flow in an elastic tube. The former, which is much cheaper computationally but yields flow characteristics similar to a more sophisticated immersed boundary problem, will allow the overall model to be computationally tractable. It is important for mathematical models to have reasonable computational cost, inasmuch as in modeling studies one frequently conducts parameter sensitivity studies that involve computing model solutions for many different sets of model parameters.

Acknowledgments This work was supported in part by the National Institutes of Health, National Institute of Diabetes and Digestive and Kidney Diseases, through grant DK089066 (to Layton).

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.*, 147:402–440, 1998.

[2] J. T. Beale and A. T. Layton. A velocity decomposition approach for moving interfaces in viscous fluids. *J. Comput. Phys.*, 228:3358–3367, 2009.

[3] G. Biros, L. Ying, and D. Zorin. An embedded boundary integral solver for the unsteady incompressible navier-stokes equations. *SIAM J. Sci. Comput.*, submitted for publication.

[4] D. L. Brown, R. Cortez, and M. L. Minion. Accurate projection methods for the incompressible navier-stokes equations. *J. Comput. Phys.*, 168:464–494, 2001.

[5] H.H. Hu, N.A. Patankar, and M.Y. Zhu. Direct numerical simulations of fluid-solid systems using the arbitrary Lagrangian-Eulerian technique. *Journal of Computational Physics*, 169:427–462, 2001.

[6] A. Huerta and W.K. Liu. Viscous flow with large free surface motion. *Computer Methods in Applied Mechanics and Engineering*, 69:277–324, 1988.

[7] T.J.R. Hughes, W.K. Liu, and T.K. Zimmermann. Lagrangian-Eulerian finite element formulation for incompressible viscous flows. *Computer Methods in Applied Mechanics and Engineering*, 29:329–349, 1981.

[8] E. Jung, S. Lim, W. Lee, and S. Lee. Computational models of valveless pumping using the immersed boundary method. *Comput. Methods Appl. Mech. Engin.*, 197:2329–2339, 2008.

[9] J. Kim and P. Moin. Application of a fractional-step method to incompressible navier-stokes equations. *J. Comput. Phys.*, 59:308, 1985.

[10] M.-C. Lai. Fourth-order finite difference scheme for the incompressible Navier-Stokes equations in a disk. *Int J Numer Meth Fluids*, 42:909–922, 2003.

[11] M.-C. Lai and H.-C. Tseng. A simple implementation of the immersed interface methods for stokes flows with singular forces. *Comput Fluids*, 37:99–106, 2008.

[12] MC Lai, CY Huang, and YM Huang. Simulating the axisymmetric interfacial flows with insoluble surfactant by immsered boundary method. *Int J Numer Anal Model*, 8:105–117, 2011.

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

[14] R. J. LeVeque and Z. Li. Immersed interface methods for Stokes flow with elastic boundaries or surface tension. *SIAM J. Sci. Comput.*, 18(3):709–735, 1997.

[15] Y Li, S Williams, and AT Layton. A hybrid immersed interface method for driven stokes flow in an elastic tube. *Numer Math* in press, 2012.

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

[17] Z. Li, W.-C. Wang, I.-L. Chern, and M.-C. Lai. New formulations for interface problems in polar coordinates. *SIAM J. Sci. Comput.*, 25:224–245, 2003.

[18] W.K. Liu, H. Chang, J. Chen, and T. Belytschko. Arbitrary Lagrangian-Eulerian Petrov-Galerkin finite elements for nonlinear continua. *Computer Methods in Applied Mechanics and Engineering*, 68:259–310, 1988.

[19] W.K. Liu and D.C. Ma. Computer implementation aspects for fluid-structure interaction problems. *Comput Meth Appl Mech Eng*, 31:129–148, 1982.

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

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

[22] D. Xiu and G. Karniadakis. A semi-Lagrangian high-order method for Navier-Stokes equations. *J. Comput. Phys.*, 172:658–684, 2001.

[23] L.T. Zhang, G.Wagner, and W.K. Liu. Modeling and simulation of fluid structure interaction by meshfree and fem. *Communications in Numerical Methods in Engineering*, 19:615–621, 2003.

[24] T. Zhijun, D.V. Le, K.M. Lim, and B.C. Khoo. An immersed interface method for the incompressible Navier-Stokes equations with discontinuous viscosity across the interface. *SIAM J. Sci. Comput.*, 31(3):1798–1819, 2009.