Numerical Methods for Simulating Fluid Motion Driven By Immersed Interfaces

by

Yi Li

Department of Mathematics
Duke University

Date: _____

Approved:

_____
Anita Layton, Supervisor

_____
J.Thomas Beale

_____
John Dolbow

_____
Jianguo Liu

Dissertation submitted in partial fulfillment of the requirements for the degree of
Doctor of Philosophy in the Department of Mathematics
in the Graduate School of Duke University
2012

Abstract

(Mathematical science)

Numerical Methods for Simulating Fluid Motion Driven By Immersed Interfaces

by

Yi Li

Department of Mathematics
Duke University

Date: _____

Approved:

_____
Anita Layton, Supervisor

_____
J.Thomas Beale

_____
John Dolbow

_____
Jianguo Liu

An abstract of a dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Mathematics in the Graduate School of Duke University
2012

Abstract

This dissertation introduces the new computational methods for two major topics. The first topic is computing the Stokes flow driven by an open immersed interface. The other topic is the simulation of the Stokes and Navier-Stokes fluid through an elastic tube driven by an internal source and sink.

For the first topic, we developed two second-order accurate method. One is for accurately evaluating boundary integral solutions at a point, and the other is for computing Stokes solution values on a rectangular mesh. We first describe a method for computing singular or nearly singular integrals, evaluated at a point on or near the curve. To improve accuracy of the numerical quadrature, we add corrections for the errors arising from discretization, which are found by asymptotic analysis. When used to solve the Stokes equations with sources on an open, immersed interface, the method generates second-order approximations, for both the pressure and the velocity, and preserves the jumps in the solutions and their derivatives across the boundary. We then combine the method with a mesh-based solver to yield a hybrid method for computing Stokes solutions at N^2 grid points on a rectangular grid. Numerical results are presented which exhibit second-order accuracy. To demonstrate the applicability of the method, we use the method to simulate fluid dynamics induced by the beating motion of a cilium.

For the second topic, we present numerical method for simulating both Stokes and Navier Stokes fluid flow through a compliant, closed tube, driven by an internal source

and sink. The governing equations are implemented in axisymmetric cylindrical coordinates, which capture 3D flow dynamics with only 2D computations.

In the Stokes fluid flow simulations, we solve the model equations using a hybrid approach: we decompose the pressure and velocity fields into parts due to the surface force and due to the source and sink, with each part handled separately by means of an appropriate method. Because the singularly-supported surface force yields an unsmooth solution, that part of the solution is computed by using the immersed interface method with the jump conditions for the axisymmetric cylindrical coordinates. The velocity due to the source and sink is calculated along the tubular surface using boundary integrals. The source and sink are prescribed in the simulation. From the convergence test and oscillating frequency-amplitude study, we can demonstrate second-order accuracy and applicability of the method.

In the Navier-Stokes flow simulations, we adopt the velocity decomposition approach developed by Beale and Layton [5]. The total velocity is decomposed into the Stokes part and the regular part. The Stokes part satisfies the Stokes equation and includes the boundary force. The regular part satisfies the modified Navier-Stokes equation that incorporate the source and sink terms, with the latter computed using the Hagen-Poiseuille equation. Convergence test, oscillating frequency-amplitude study and fluid viscosity-amplitude study are presented that demonstrate the accuracy of the method.

To My Family

Contents

List of Tables

List of Figures

Acknowledgements

It is a pleasure to acknowledge with gratitude many people who made this thesis possible.

First of all, I would like to express my deep and sincere gratitude to my Ph.D. advisor Dr. Anita Layton, for her enthusiasm, her inspiration, her patience and her great efforts in supervising me. Secondly, I would like thank to my other committee members, Dr. Thomas Beale, Dr. John Dolbow and Dr. Jianguo Liu for their advice, suggestions and encouragement.

Next, I wish to thank my family for their supports. My special gratitude is due to my parents and my parents in law. Most of all, I wish to thank my wife, Hongxia Yang, for her endless support and love. To them I dedicate this thesis.

1

Introduction

The fluid structure interaction problems in the Stokes and Navier-Stokes flow have been studied for several years due to its wide application in physical and biological systems. Several advanced methods have been developed and applied to scientific and engineering problems.

One of the most widely used method is the immersed boundary method, which is first proposed by Peskin for studying blood flow through a beating heart [31], and has since been applied successfully to many other biological problems, such as flow and transport in a rental arteriole [1], the mechanism of valveless pumping [21, 20] and flow in a collapsible tube [34]. This method employs a set of lagrangian markers to track the motion of interface and fluid variables are solved on the fixed Eulerian fluid grid. The interpolation and spreading operation links the variables between Eulerian and Lagrangian grid. When simulating the motion of a singular elastic interface in the Navier-Stokes flow, the equation of motion is as follows:

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = \mu \Delta \mathbf{u} - \nabla p + \mathbf{F}, \tag{1.1}$$

$$\nabla \cdot \mathbf{u} = 0, \tag{1.2}$$

$$\mathbf{F}(\mathbf{y}) = \int_{\gamma} \mathbf{f}(\tau) \delta\big(\mathbf{y} - \mathbf{x}(\tau)\big)\, d\mathbf{x}(\tau) \tag{1.3}$$

where p is the pressure, $\mathbf{u}$ is the velocity, μ is the viscosity, and $\mathbf{F}$ is the force that is singularly supported along a boundary γ immersed in a two-dimensional fluid. $\mathbf{x}(\tau)$ denotes a point on γ, which is parametrized by τ, and δ is the Dirac delta function.

The immersed boundary method discretize the interface to be a set of Langragian markers$(X(\tau_k), Y(\tau_k))$, k=1,2..n. and replace the boundary integral in Eq. (1.3) by a discrete summation. Then this method transfers the singular boundary forces onto the neighborhood Eulerian fluid grid points by using approximate (smooth) Dirac delta functions typically with $\mathcal{O}(h)$ support. A common choice for the approximate delta function is the cosine function, which has the following form:

$$\delta_h(x - x_0) = \frac{1}{2h}\left(1 + \cos\left(\frac{\pi(x - x_0)}{h}\right)\right) \tag{1.4}$$

With Peskin's discrete delta function approach, the discretized form of the force at grid (i,j) is

$$F_{ij} = \sum_{k=1}^{n} f_k \delta_h(x_i - X_k)\delta_h(y_j - Y_k)\Delta\tau \tag{1.5}$$

Because the delta function is smoothed, this approach does not capture the jump discontinuity in the solution (e.g., pressure) at the immersed boundary, but rather approximates the solution as a continuous function with a large gradient. In general, the immersed boundary method computes approximations with first-order spatial accuracy.

When simulating Stokes or Navier-Stokes flows driven by a closed boundary, one may use the immersed interface method developed by LeVeque and Li [26, 27], which captures the jumps in the solution and its derivatives sharply, and which generates approximations with second-order accuracy. The key idea in the immersed interface method is the incorporation of known jumps in the solution or its derivatives into the finite difference schemes. The necessary jump condition has been derived for Stokes and Navier-Stokes flow in [27] and [25]. The jump condition for interface in Stokes flow for 2D Cartesian coordinate is the following [27]:

$$[p] = f_n \tag{1.6}$$

$$\left[\frac{\partial p}{\partial n}\right] = \frac{\partial(f_\tau)}{\partial s}. \tag{1.7}$$

$$\left[\mu\frac{\partial u}{\partial n}\right] = f_\tau \sin\alpha. \tag{1.8}$$

$$\left[\mu\frac{\partial w}{\partial n}\right] = -f_\tau \cos\alpha. \tag{1.9}$$

For the Stokes equation, the pressure jump, the first and second order derivative of pressure and velocity jump are incorporated into the finite difference discretized equations when the stencil cross the interface to achieve the second order convergence. For example, in 2D Cartesian coordinate, by taking the divergence on Eq. (1.1) and discretizing the elliptic equation by using the standard five point stencil, one obtains:

$$\frac{1}{h^2}(p_{i+1,j} + p_{i-1,j} + p_{i,j+1} + p_{i,j-1} - 4p_{i,j}) = C_{i,j}. \tag{1.10}$$

When grid stencils are away from the interface, the right hand side $C_{i,j} = 0$. When grid stencils cross the interface, corrections need to be derived. However, the limitation of this method is that the jump conditions can only be derived for closed boundaries.

An alternative method, which employs the finite element approach for simulating

the free and moving boundary problem, is the Arbitrary Lagrangian-Eulerian method [19, 13, 23]. This method combines the advantage of pure Lagrangian type method and Eulerian type method. In Lagrangian type algorithms, the computational mesh moves with fluid material particle, therefore the free surface or interface can be easily tracked. However, large deformation may cause the mesh entanglement and cease the computation. In Eulerian type algorithm, the computational mesh is fixed, therefore the large deformation of material can be handled relatively easily. However, high resolution grid and precise definition of the interface may be needed to track the interface accurately. In the Arbitrary Lagrangian-Eulerian method, after updating the fluid mesh by the fluid particle velocity, a rezoning process can be included so that the mesh can be updated to optimal position, avoiding the mesh collapsing. Rezoning techniques have been developed over years by researchers [39, 24].

An important Eulerian type method, the Volume of fluid method, which was firstly proposed by Hirt and Nichols [12], can be applied to track the interface between immiscible fluid. Rather than marking the interface, this method tracks the fluid surface by marking the fluid region through fraction of volume, thus can handle the complicated surface change, such as surface crossing or surface breaking. Denoting the fraction of volume by F, the equation F satisfies is:

$$\frac{\partial F}{\partial t} + \mathbf{u} \cdot \frac{\partial F}{\partial \mathbf{x}} = 0 \tag{1.11}$$

Here $\mathbf{u}$ is the velocity vector and the above equation is an advection equation. The interface lies in the cell that has F with value $0 < F < 1$. Over the years researchers have proposed several advection techniques that can keep sharp interface and maintain stability, such as the Donor Acceptor schemes by Hirt and Nichols [12], flux corrected transport by Boris [9].

Another important interface tracking algorithm is the level set method, first proposed by Osher and Sethian [30]. Without discrete parametrization of the interface,

this method models the interface by using a zero level set function ϕ. With the curvature K along the interface and the normal velocity speed vector $\mathbf{F}(K)$, ϕ satisfies the Hamilton-Jacobi type equation:

$$\phi_t - \mathbf{F}(K)|\nabla\phi| = 0 \tag{1.12}$$

The appealing part of this method is it can handle the topological change of the surface, such as surface merging or breaking, however, with the computational cost of solving the Hamilton-Jacobi type equation by sophisticated numerical techniques.

The methods introduced above can be applied for simulating both the Navier-Stokes flow and Stokes flow. When simulating Stokes flows with singularly supported interface, one may use the method of regularized Stokeslets [11], which is a boundary integral type method, first proposed by Cortez. A Stokeslet is the fundamental solution of Eqs. (1.1)–(1.2), and it represents fluid motions due to a concentrated external force acting on the fluid at a single point [2, 33]. In two dimensions, the solutions to the Stokes equations in free space (1.1)–(1.2) can be written as the boundary integrals

$$p(\mathbf{y}) = -\frac{1}{2\pi}\int_\gamma \nabla G(\mathbf{x}(\tau)-\mathbf{y})\cdot\mathbf{f}(\tau)\,d\mathbf{x}(\tau), \quad \mathbf{u}(\mathbf{y}) = \frac{1}{4\pi\mu}\int_\gamma V(\mathbf{x}(\tau)-\mathbf{y})\mathbf{f}(\tau)\,d\mathbf{x}(\tau), \tag{1.13}$$

where

$$\nabla G(\mathbf{x}-\mathbf{y}) = \frac{\mathbf{x}-\mathbf{y}}{|\mathbf{x}-\mathbf{y}|^2}, \tag{1.14}$$

$$V_{ij}(\mathbf{x}) = -\delta_{ij}\log|\mathbf{x}| + \frac{x_i x_j}{|\mathbf{x}|^2}, \quad i,j = 1,2, \tag{1.15}$$

where $\delta_{i,j}$ is the Kronecker delta. The gradient of Green's function in Eq. (1.14) is singular. The first term in the velocity integrand (1.15) has a logarithmic singularity; the second term has a smooth integrand on the interface but not at neighboring points. In the method of regularized stokeslet, the singular force on the interface can

be spread over a small region through blobs function ϕ_ϵ. Then a regularized Green's function G_ϵ can be derived from $\nabla^2 G_\epsilon = \phi_\varepsilon$. With the blob function to be:

$$\phi_\epsilon(\mathbf{x}) = \frac{\epsilon^{\mathbf{3}}}{\mathbf{2}\pi(|\mathbf{x}|^{\mathbf{2}} + \epsilon^{\mathbf{2}})^{\frac{\mathbf{5}}{\mathbf{2}}}} \tag{1.16}$$

The regularized Green's function has the form:

$$G_\epsilon(\mathbf{x}-\mathbf{y}) = \frac{1}{2\pi}[\log(\sqrt{(\mathbf{x}-\mathbf{y})^2 + \epsilon^2})] - \frac{\epsilon}{\sqrt{(\mathbf{x}-\mathbf{y})^2 + \epsilon^2}} \tag{1.17}$$

Correspondingly, one can obtains the regularized integral solution of pressure and velocity. For points near the interface, the integrals for the pressure and velocity are nearly singular, and standard quadrature (e.g., the trapezoidal rule) would lead to substantial errors. In the case of a closed and smooth boundary, corrections terms were derived in [3] so that the third order regularized error and second order quadrature error can be achieved. However, such correction can only be applied for the smooth and closed interface. The correction for nearly singularity problem for open or non-smooth interface will be derived in the thesis by following Surles' work [37].

The first part of the thesis is to develop numerical methods that accurately compute solution to Stokes flow driven by forces singularly supported along an immersed, open interface. Solution values may be sought at an arbitrary point in two-dimensional space, or at grid points on a rectangular grid. Specifically, we aim to develop methods that generate second-order approximations. The numerical challenges in achieving that goal arise from the singularities in the Stokes solution introduced by the forces on the open interface: as previously noted, p and the normal derivatives of $\mathbf{u}$ have jump discontinuities, which should be preserved if the computed solutions are to be accurate. But because those jump conditions are not known for open interfaces, the immersed interface method cannot be applied. Another difficulty

is that at the end-points of the immersed interface, p and the normal derivatives of $\mathbf{u}$ are unbounded, which may render mesh-based finite-difference schemes inaccurate.

Two numerical methods are developed. We first present a method, which we call "the method of corrected boundary integrals" for computing Stokes solution at an arbitrary location. The method is based on an accurate computation of the boundary integrals involving the singular Stokeslets. Those integrals are corrected to reduce discretization errors, to yield second-order accuracy, and to preserve the jumps in the Stokes solutions and their derivatives. We then combine the method of corrected boundary integrals with a mesh-based solver to yield a hybrid method for computing Stokes solutions at a rectangular mesh with N^2 grid points. Numerical examples are presented to evaluate the accuracy of these methods.

In addition, we formulate the methods for a quasi-dynamic Stokes problem. The model equations are based on the steady-state Stokes equations, but the fluid motion evolves in time, as the immersed boundary moves with the fluid and the boundary forces are updated based on the boundary configuration. We apply the methods to simulate the hydrodynamic flows generated by the beating motion of a pulmonary cilium.

In the second part of the thesis, we consider incompressible Stokes or Navier-Stokes flow through a compliant, closed tube, driven by an internal source and sink. The internal source and sink allow us to represent inflow and outflow conditions while using techniques that apply to closed boundaries. Motivated by applications to blood flow through vessels, we take the structure and flow to be axisymmetric.

In the simulation study for Stokes flow, our approach is to decompose the pressure and velocity fields into parts due to the tube boundary and due to the source and sink, so that each may be treated with an appropriate method. The tube surface creates a singularly-supported force on the fluid, resulting in an unsmooth solution. We find this part of the solution using the immersed interface method. Meanwhile,

the smooth solution due to the source and sink is efficiently calculated along the tube surface via a boundary integral.

In the Navier-Stokes flow simulation study, we adopt the velocity decomposition approach developed by Beale and Layton [5]. We decompose the velocity and pressure into the Stokes part and the regular part. The Stokes part satisfies the Stokes equation with the singular force on the boundary. Then the body force for the regular part equation is the material derivative of the Stokes velocity, which is continuous along the interface. Therefore, we can solve the regular part fluid variables without jump condition along the interface.

Rather than prescribing the sink strength in this problem, we compute the sink strength by using the Hagen-Poiseuille equation. Besides doing the convergence test to demonstrate the accuracy of method, we explore the trend between the oscillation frequency and the oscillation amplitude, flow viscosity and the oscillation amplitude of pressure and velocity.

2

Accurate Computation of Stokes Flow Driven By An Open Immersed Interface

2.1 The method of corrected boundary integrals

In this section, we derive the method of corrected boundary integrals, which computes the integrals in Eq. (1.13) to second-order accuracy, even for a point $\mathbf{y}$ that is sufficiently near the boundary so that the integrals are nearly singular. Along the boundary, the pressure is singular, but second-order accurate approximations can be computed for the velocity. We parametrize the curve γ using τ, such that $\tau \in [0, 1]$, which we discretize using M grid points with grid spacing $\Delta\tau$.

2.1.1 Computing pressure integral

According to Eq.1.13, the boundary integral for p can be written in terms of its tangential and normal components

$$p(\mathbf{y}) = -\frac{1}{2\pi} I(\mathbf{y}) \equiv -\frac{1}{2\pi}\big(I_t(\mathbf{y}) + I_n(\mathbf{y})\big), \tag{2.1}$$

where the subscripts t and n denote the tangential and normal components, respectively. $t = (x_1'(\tau), x_2'(\tau))$ and $n = (x_2'(\tau), -x_1'(\tau))$. I_t, I_n are

$$I_t(\mathbf{y}) = \int_0^1 \partial_t G(\mathbf{x}(\tau) - \mathbf{y}) f_t(\tau)\, d\tau, \tag{2.2}$$

$$I_n(\mathbf{y}) = \int_0^1 \partial_n G(\mathbf{x}(\tau) - \mathbf{y}) f_n(\tau)\, d\tau. \tag{2.3}$$

Both I_t and I_n are nearly singular as $\mathbf{y}$ approaches $\mathbf{x}$. Below we present the method of corrected boundary integrals which yields second-order accurate approximations to $I(\mathbf{y})$, or $p(\mathbf{y})$, including $\mathbf{y}$ near (but not on) the boundary. The corrections for the normal part integral I_n was originally derived in Matthew Surles' thesis [37]. Here we follow Surles' method to derive correction for tangential part integral. Corrections for quadrature errors are derived separately for three sets of points: one set in which the points are near one of the end-points of γ (i.e., $\mathbf{x}(0)$ or $\mathbf{x}(1)$) and exterior to γ; another set in which the points are interior to γ; and a third set in which the points are away from γ. We consider a point $\mathbf{y}$ *interior* to the curve if there exists a line normal to the curve that passes through $\mathbf{y}$ and some point $\mathbf{x}$ on the curve. Otherwise $\mathbf{y}$ is considered *exterior* to the curve.

Points near an end-point and exterior to the boundary

We first consider a point $\mathbf{y}$ that is near one of the end-points $\mathbf{x}(0)$ or $\mathbf{x}(1)$ but exterior to the boundary. A point $\mathbf{y}$ is considered near an end-point if the distance between the two points is $< C\Delta\tau^k$, for some small constant C and for $k > 0$; see Fig. 2.1, panel A. (In our implementation, C is chosen to be $1 \leqslant C \leqslant 3$ and $k = 1/2$. Our numerical experience suggests that solution accuracy is insensitive to a substantial range of C values. The choice of $k = 1/2$ is justified below in Sect. 2.1.1.) Because $|\mathbf{y} - \mathbf{x}(\tau)|$ is small for $\tau = 0$ or $\tau = 1$, I_t and I_n are nearly singular. We first consider I_t, and we assume that $\mathbf{y}$ is close to $\mathbf{x}(0)$. We will subsequently discuss I_n and the

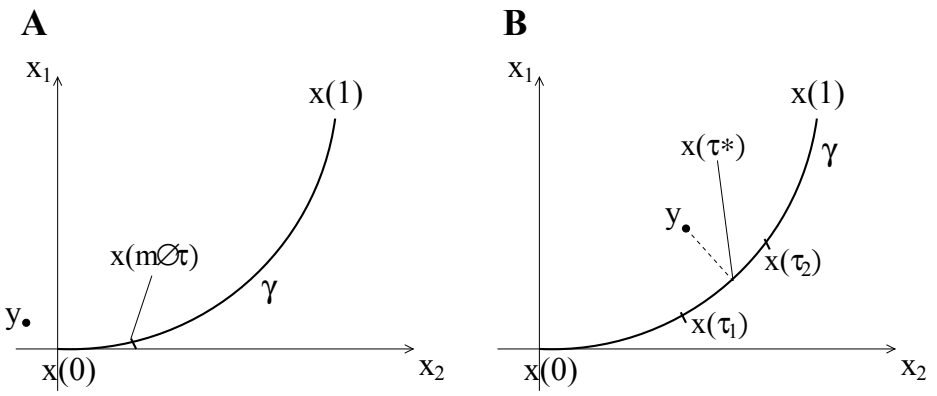

FIGURE 2.1: A: $\mathbf{y}$ denotes a point close to end-point $x(0)$ and exterior to the curve γ. The segment between $x(0)$ and $x(m\Delta\tau)$ is the "short segment". B: $\mathbf{y}$ is close to γ and interior to the curve. $\mathbf{x}(\tau^*)$ is the intersection between γ and the normal line through $\mathbf{y}$. The segment between $\mathbf{x}(\tau_1)$ and $\mathbf{x}(\tau_2)$ is the short segment.

case where $\mathbf{y}$ is close to the other end point $\mathbf{x}(1)$.

Let $\mathbf{x} \equiv (x_1(\tau), x_2(\tau))$. To derive corrections for the trapezoidal approximation of I_t, we begin by applying a translation and a coordinate rotation of the boundary, such that we obtain $(x_1(0), x_2(0)) = (0, 0)$ and $(x_1'(0), x_2'(0)) = (1, 0)$. To reduce the singularity, we rewrite Eq. (2.2) as

$$I_t = I_{t,1} + I_{t,2}, \tag{2.4}$$

where

$$I_{t,1} = \int_0^1 \partial_t G(\mathbf{x}(\tau) - \mathbf{y})(f_t(\tau) - f_t(0))\, d\tau, \tag{2.5}$$

$$I_{t,2} = \int_0^1 \partial_t G(\mathbf{x}(\tau) - \mathbf{y}) f_t(0)\, d\tau. \tag{2.6}$$

$I_{t,2}$ can be computed analytically to yield $\log \frac{|x(1)-y|}{|x(0)-y|} f_t(0)$. Corrections will be derived for the trapezoidal approximation of $I_{t,1}$.

Similarly, we decompose I_n into two parts, given by

$$I_n = I_{n,1} + I_{n,2}, \tag{2.7}$$

where

$$I_{n,1} = \int_0^1 \partial_n G(\mathbf{x}(\tau) - \mathbf{y})(f_n(\tau) - f_n(0))\, d\tau, \tag{2.8}$$

$$I_{n,2} = \int_0^1 \partial_n G(\mathbf{x}(\tau) - \mathbf{y}) f_n(0)\, d\tau. \tag{2.9}$$

$I_{n,2}$ can be integrated explicitly to yield

$$I_{n,2} = f_n(0)\big(\arg(\mathbf{z}(1) - \mathbf{z}_0) - \arg(\mathbf{z}(0) - \mathbf{z}_0)\big), \tag{2.10}$$

where $\mathbf{z}(\tau)$ and $\mathbf{z}_0$ denote the points $\mathbf{x}(\tau)$ and $\mathbf{y}$, respectively, in the complex domain [37]. $I_{n,1}$ will be approximated by means of the trapezoidal rule with appropriate corrections; see below.

To approximate $I_{t,1}$ and $I_{n,1}$ accurately, we decompose each of the integrals into two terms, one along a short segment of the boundary near one of the end-points, and the other along the remaining longer segment of the boundary. We will first consider $I_{t,1}$. For notational simplicity, we denote the integrand in Eq. (2.5) by

$$\alpha(\tau) \equiv \partial_t G(\mathbf{x}(\tau) - \mathbf{y})\,(f_t(\tau) - f_t(0))\,. \tag{2.11}$$

Using this notation, we write $I_{t,1}$ as

$$I_{t,1} = I_{t,S} + I_{t,L}, \tag{2.12}$$

where $I_{t,S}$ and $I_{t,L}$ denote the short and long parts of the integral. The two integrals are given by

$$I_{t,S} = \int_0^{m\Delta\tau} \alpha(\tau)\, d\tau, \quad I_{t,L} = \int_{m\Delta\tau}^1 \alpha(\tau)\, d\tau, \tag{2.13}$$

where m is the number of grid points along the short segment, chosen such that $m = \mathcal{O}(\Delta\tau^{\frac{-1}{3}})$; a rationale of this choice of m is given below. Thus, for a sufficiently refined discretization of γ with M grid points, $m \ll M$. Both $I_{t,S}$ and $I_{t,L}$ are nearly

singular. An analogous decomposition can be obtained for $I_{n,1}$. Below we will derive the corrections necessary for evaluating these integrals with second-order accuracy using the trapezoidal rule.

Corrected integrals for the long segments. We first consider integrals for the long segments. We approximate $I_{t,L}$ by

$$I_{t,L} \approx T_{t,h}(\alpha(\tau))|_{m\Delta\tau}^{1} + \frac{B_2}{2}\Delta\tau^2\alpha'(m\Delta\tau), \tag{2.14}$$

where $T_{t,h}(\alpha(\tau))|_{m\Delta\tau}^{1}$ denotes the trapezoidal approximation of $I_{t,L}$ and B_2 denotes the second Bernoulli number. The second term in Eq. (2.14) corrects for the discretization error in $T_{t,h}(\alpha(\tau))|_{m\Delta\tau}^{1}$. Below we show that the above approximation is $\mathcal{O}(\Delta\tau^2)$. To that end, we rewrite $I_{t,L}$ as

$$I_{t,L} = \int_{m\Delta\tau}^{1} \frac{\big(\mathbf{x}(\tau) - \mathbf{y}\big)\cdot\mathbf{t}(\tau)}{|\mathbf{x}(\tau) - \mathbf{y}|^2}\big(f_t(\tau) - f_t(0)\big)\,d\tau. \tag{2.15}$$

We first consider the case where $\mathbf{y}$ is close to $\mathbf{x}(0)$. At $\tau = 0$, we have assumed that, after the translation and rotation, $x_1(0) = x_2(0) = 0$ and $x_1'(0) = 1$, $x_2'(0) = 0$; thus, the Taylor expansion of the tangent vector $\mathbf{t}(\tau)$ around $\tau = 0$ is

$$\mathbf{t}(\tau) = \big(x_1^{'}(\tau), x_2^{'}(\tau)\big) = \big(1 + x_1^{''}(0)\tau + \mathcal{O}(\tau^2), x_2^{''}(0)\tau + \mathcal{O}(\tau^2)\big). \tag{2.16}$$

Let the coordinates of $\mathbf{y}$ after the translation and rotation be (a, b). Then the Taylor expansion of $\mathbf{x}(\tau) - \mathbf{y}$ is given by

$$\big(\mathbf{x}(\tau) - \mathbf{y}\big) = \big(x_1(\tau) - a, x_2(\tau) - b\big) = \big(-a + \tau + \mathcal{O}(\tau^2), -b + \mathcal{O}(\tau^2)\big). \tag{2.17}$$

Taking the dot product of Eqs. (2.16) and (2.17), we obtain that

$$\big(\mathbf{x}(\tau) - \mathbf{y}\big)\cdot\mathbf{t}(\tau) = -a + \tau + \mathcal{O}(\epsilon\tau) + \mathcal{O}(\tau^2), \tag{2.18}$$

where $\epsilon \equiv \sqrt{a^2+b^2}$. Substituting the above expression into Eq. (2.15), and applying Taylor expansion, one can approximate $I_{t,L}$ as

$$\begin{aligned} I_{t,L} &= \int_{m\Delta\tau}^{1} \frac{-a+\tau+\mathcal{O}(\epsilon\tau)+\mathcal{O}(\tau^2)}{r^2}\big(f_t'(0)\tau+\mathcal{O}(\tau^2)\big)\,d\tau, \\ &= \int_{m\Delta\tau}^{1} \frac{1}{r^2}\Big((-a+\tau)f_t'(0)\tau+\mathcal{O}(\epsilon\tau^2)f_t'(0)+f_t'(0)\mathcal{O}(\tau^3)+(-a+\tau)\mathcal{O}(\tau^2) \\ &\qquad +\mathcal{O}(\epsilon\tau^3)+\mathcal{O}(\tau^4)\Big)\;d\tau. \end{aligned} \tag{2.19}$$

where $r = |\mathbf{x}(\tau)-\mathbf{y}|$.

We first analyze the term $\frac{(-a+\tau)f_t'(0)\tau}{r^2}$, which we denote by $\psi(\tau)$; the other terms can be treated analogously. By means of the Euler Maclaurin summation, we obtain that

$$\begin{aligned} \int_{m\Delta\tau}^{1}\psi(\tau)\,d\tau =& T_{t,h}\big(\psi(\tau)\big)+\frac{B_2}{2!}\Delta\tau^2\big(\psi'(m\Delta\tau)-\psi'(1)\big) \\ &+\frac{B_4}{4!}\Delta\tau^4\big(\psi'''(m\Delta\tau)-\psi'''(1)\big) \\ &-\Delta\tau^5\int_{m\Delta\tau}^{1}P_5\left(M\frac{\tau-m\Delta\tau}{1-m\Delta\tau}\right)\psi^5(\tau)\,d\tau, \end{aligned} \tag{2.20}$$

where B_k denotes the kth Bernoulli number, and P_k denotes the corresponding function.

To approximate $I_{t,L}$ to $\mathcal{O}(\Delta\tau^2)$, we must understand the size of each of the term on the right of $T_{t,h}$ in Eq. (2.20). First consider $\frac{B_2}{2}\Delta\tau^2\psi'(m\Delta\tau)$, which we call $\mathcal{E}_1$. Here we apply the following lemma [37]:

lemma 2.1.1. *Suppose $Q(\epsilon,a,b)$ is a degree q homogeneous polynomial, $w(\tau) = \tau^k\phi(\tau)$, $g = 1/r^p$, and $p+1 \geqslant q+k$. Then we have*

$$\left|\frac{\partial^l}{\partial\tau^l}\left(\frac{Q\tau^k\phi(\tau)}{r^p}\right)\right| = \left|\frac{\partial^l}{\partial\tau^l}(Qgw(\tau))\right| \leqslant \frac{K_l}{r^{(p+l)-(q+k)}} \leqslant \frac{C_l}{\tau^{(p+l)-(q+k)}}$$

for some constants K_l and C_l.

From Lemma 2.1.1, one can show that $\psi'(\tau) \leqslant c_0/\tau$ for some constant c_0 using $p = 2$, $l = 1$, and $q + k = 2$. If one chooses m to be $\Delta\tau^{-1/3}$ (this choice will be justified below), then $\psi'(m\Delta\tau) = \mathcal{O}(\Delta\tau^{-2/3})$. Thus,

$$\mathcal{E}_1 \equiv \frac{B_2}{2}\Delta\tau^2\psi'(m\Delta\tau) = \mathcal{O}(\Delta\tau^{4/3}). \tag{2.21}$$

Consequently, to attain overall $\mathcal{O}(\Delta\tau^2)$ accuracy, $\mathcal{E}_1$ must be included as a correction.

We then consider the other terms,

$$\mathcal{E}_2 \equiv -\frac{B_2}{2}\Delta\tau^2\psi'(1),$$

$$\mathcal{E}_3 \equiv \frac{B_4}{4!}\Delta\tau^4\big(\psi'''(m\Delta\tau) - \psi'''(1)\big),$$

$$\mathcal{E}_4 \equiv \Delta\tau^5 \int_{m\Delta\tau}^{1} P_5\big(M\frac{\tau - m\Delta\tau}{1 - m\Delta\tau}\big)\psi^5(\tau)\, d\tau.$$

Because $\mathbf{y}$ is close to $\mathbf{x}(0)$, $\psi'(1)$ is $\mathcal{O}(1)$; thus $\mathcal{E}_2 = \mathcal{O}(\Delta\tau^2)$. Again by applying Lemma 2.1.1, we obtain the following bounds on the derivatives of ψ:

$$\psi'''(\tau) \leqslant \frac{c_1}{\tau^3}, \quad \psi^5(\tau) \leqslant \frac{c_2}{\tau^5}, \tag{2.22}$$

where c_1 and c_2 are constants. These above bounds imply that $\mathcal{E}_3 = \mathcal{O}(m^{-3}\Delta\tau)$ and $\mathcal{E}_4 = \mathcal{O}(m^{-4}\Delta\tau)$. Thus, for both terms to be $\mathcal{O}(\Delta\tau^2)$, we choose m to be $O(\Delta\tau^{\frac{-1}{3}})$.

A similar approach is used to derive corrections for $I_{n,L}$

$$I_{n,L} = T_{n,h}(\beta(\tau))|_{m\Delta\tau}^{1} + \frac{B_2}{2}\Delta\tau^2\beta'(m\Delta\tau) + \mathcal{O}(\Delta\tau^2), \tag{2.23}$$

where $T_{n,h}$ denotes the trapezoidal approximation of $I_{n,L}$, $\beta = \frac{(\mathbf{x}(\tau)-\mathbf{y})\cdot\mathbf{n}(\tau)}{|\mathbf{x}(\tau)-\mathbf{y}|^2}\big(f_n(\tau) - f_n(0)\big)$.

If $\mathbf{y}$ is close to $\mathbf{x}(1)$, then the integrals $I_{t,L}$ and $I_{n,L}$ can be approximated by

$$I_{t,L} = T_{t,h}(\alpha(\tau))|_{0}^{1-m\Delta\tau} - \frac{B_2}{2}\Delta\tau^2\alpha'(1 - m\Delta\tau) + \mathcal{O}(\Delta\tau^2), \tag{2.24}$$

$$I_{n,L} = T_{n,h}(\beta(\tau))|_{0}^{1-m\Delta\tau} - \frac{B_2}{2}\Delta\tau^2\beta'(1 - m\Delta\tau) + \mathcal{O}(\Delta\tau^2). \tag{2.25}$$

Corrected integrals for the short segments. Next we derive corrections for approximating $I_{t,S}$ and $I_{n,S}$ to second-order accuracy. We first consider $I_{t,S}$ for the case in which $\mathbf{y}$ is close to $\mathbf{x}(0)$. We begin by linearizing both $\mathbf{x}(\tau)$ and $f(\tau)$. Let $\tilde{\alpha}(\tau)$ be the linearized form of $\alpha(\tau)$, given by

$$\tilde{\alpha}(\tau) = \frac{(\tau - a)}{(\tau - a)^2 + b^2} f_t'(0)\tau, \tag{2.26}$$

We approximate $I_{t,S}$ by

$$I_{t,S} \approx T_{t,h}(\alpha - \tilde{\alpha})|_0^{m\Delta\tau} + \int_0^{m\Delta\tau} \tilde{\alpha}(\tau)\, d\tau, \tag{2.27}$$

where $T_{t,h}$ denotes the trapezoidal approximation of $I_{t,S}$. The term $\int_0^{m\Delta\tau} \tilde{\alpha}(\tau)\, d\tau$ can be evaluated analytically to yield

$$\int_0^{m\Delta\tau} \tilde{\alpha}(\tau)\, d\tau = \left(\frac{1}{2} a \log((a-\tau)^2 + b^2) + b \tan^{-1}\left(\frac{a-\tau}{b}\right) + \tau \right) \Big|_0^{m\Delta\tau}. \tag{2.28}$$

We will show that the approximation (2.27) is $\mathcal{O}(\Delta\tau^2)$. To that end, we consider the difference between the two sides of Eq. (2.27), which can be written as

$$\int_0^{m\Delta\tau} \alpha_2(\tau)\, d\tau - T_{t,h}(\alpha_2)|_0^{m\Delta\tau},$$

where we let $\alpha_2(\tau) \equiv \alpha(\tau) - \tilde{\alpha}(\tau)$. We will show that this difference is $\mathcal{O}(\Delta\tau^2)$.

Consider the numerator and denominator of $\alpha(\tau)$ and $\tilde{\alpha}(\tau)$, denoted by α^P, α^Q

and $\tilde{\alpha}^P$,$\tilde{\alpha}^Q$, respectively:

$$\begin{aligned}
\alpha^P(\tau) =&\big(\mathbf{x}(\tau)-\mathbf{y}\big)\cdot\big(x_1^{'}(\tau),x_2^{'}(\tau)\big)\big(f_t(\tau)-f_t(0)\big)\\
=&\Big(\big(\tau-a+\mathcal{O}(\tau^2)\big)\big(1+\tau w_1(\tau)\big)+(\mathcal{O}(\tau^2)-b)\tau w_2(\tau)\Big)\big(f_t^{'}(0)\tau+\mathcal{O}(\tau^2)\big),
\end{aligned} \tag{2.29}$$

$$\begin{aligned}
\alpha^Q(\tau) =&|\mathbf{x}(\tau)-\mathbf{y}|^2=\big(x_1^{'}(\tau)-a\big)^2+\big(x_2^{'}(\tau)-b\big)^2,\\
=&\big(\tau-a+\mathcal{O}(\tau^2)\big)^2+\big(\mathcal{O}(\tau^2)-b\big)^2,
\end{aligned} \tag{2.30}$$

$$\begin{aligned}
\tilde{\alpha}^P(\tau) =&(\tau-a)f_t^{'}(0)\tau,\\
\tilde{\alpha}^Q(\tau) =&(\tau-a)^2+b^2.
\end{aligned}$$

for some functions $w_1(\tau)$ and $w_2(\tau)$.

Then we have

$$\begin{aligned}
\alpha_2(\tau)=&\frac{\alpha^P}{\alpha^Q}-\frac{\tilde{\alpha}^P}{\tilde{\alpha}^Q}=\frac{\alpha^P\tilde{\alpha}^Q-\tilde{\alpha}^P\alpha^Q}{\alpha^Q\tilde{\alpha}^Q}=\frac{\alpha^P-\tilde{\alpha}^P}{\tilde{\alpha}^Q}+\frac{\alpha^P(\tilde{\alpha}^Q-\alpha^Q)}{\alpha^Q\tilde{\alpha}^Q},\\
=&\frac{\mathcal{O}\big(\tau^2(a+b)\big)}{(\alpha^Q)^2}+\frac{\mathcal{O}(\tau^3)}{(\alpha^Q)^2}-\frac{\mathcal{O}\big((\tau-a)\tau^4\big)}{(\alpha^Q)^2(\tilde{\alpha}^Q)^2}-\frac{\mathcal{O}\big((\tau-a)(a+b)\tau^3\big)}{(\alpha^Q)^2(\tilde{\alpha}^Q)^2},
\end{aligned} \tag{2.31}$$

after simplification. Applying Lemma 2.1.1 again to the first term on the right side of the above equation, one can show that

$$\int_0^{m\Delta\tau}\alpha_2(\tau)\,d\tau-T_{t,h}(\alpha_2)|_0^{m\Delta\tau}=\mathcal{O}(\Delta\tau^2). \tag{2.32}$$

Thus, the corrected boundary integral approximation given in Eq. (2.27) is second order.

The correction for trapezoidal sum for $I_{n,S}$ is [37]:

$$I_{n,S}=T_{n,h}(\beta-\tilde{\beta})|_0^{m\Delta\tau}+\int_0^{m\Delta\tau}\tilde{\beta}(\tau)\,d\tau+\mathcal{O}(\Delta\tau^2), \tag{2.33}$$

where $T_{n,h}$ denotes the trapezoidal approximation of $I_{n,S}$, and $\tilde{\beta}(\tau)$ is the linearized

form of $\beta(\tau)$. The second term can be integrated analytically to yield

$$\int_0^{m\Delta\tau} \tilde{\beta}(\tau)\, d\tau = f_n'(0)\left(\frac{b}{2}\log((\tau-a)^2+b^2) + a\arctan\left(\frac{x-a}{b}\right)\right)\Big|_0^{m\Delta\tau}. \tag{2.34}$$

If $\mathbf{y}$ is close to $\mathbf{x}(1)$, then the short-segment integrals are approximated by

$$I_{t,S} = T_{t,h}(\alpha - \tilde{\alpha})\big|_{1-m\Delta\tau}^{1} + \int_{1-m\Delta\tau}^{1} \tilde{\alpha}(\tau)\, d\tau + \mathcal{O}(\Delta\tau^2), \tag{2.35}$$

$$I_{n,S} \approx T_{n,h}(\beta - \tilde{\beta})\big|_{1-m\Delta\tau}^{1} + \int_{1-m\Delta\tau}^{1} \tilde{\beta}(\tau)\, d\tau + \mathcal{O}(\Delta\tau^2). \tag{2.36}$$

Points interior and close to the curve

Next, we consider a point located sufficiently close (within a distance of $C\Delta\tau^k$) and interior to the curve. To derive corrections for the approximation of $I_n + I_t$ by the trapezoidal rule, we proceed as before and begin by reducing the singularity of I_n and I_t by rewriting those integrals as:

$$I_t = I_{t,1} + I_{t,2}, \quad I_n = I_{n,1} + I_{n,2}, \tag{2.37}$$

where

$$I_{d,1} = \int_0^1 \partial_d G(\mathbf{x}(\tau) - \mathbf{y})\big(f_d(\tau) - f_d(\tau^*)\big)\, d\tau, \tag{2.38}$$

$$I_{d,2} = \int_0^1 \partial_d G(\mathbf{x}(\tau) - \mathbf{y}) f_d(\tau^*)\, d\tau, \tag{2.39}$$

for $d = t$ or n, and where $\mathbf{x}(\tau^*)$ is the intersection point between the curve γ and the line normal to γ and through $\mathbf{y}$. Similar to Eqs. (2.6) and (2.9), the second integrals $I_{t,2}$ and $I_{n,2}$ can be computed analytically to be

$$I_{t,2} = \log\left|\frac{\mathbf{x}(1)-\mathbf{y}}{\mathbf{x}(0)-\mathbf{y}}\right| f_t(\mathbf{x}(\tau^*)), \tag{2.40}$$

$$I_{n,2} = f_n(\mathbf{x}(\tau^*))(\arg(\mathbf{z}(1)-\mathbf{z}_0) - \arg(\mathbf{z}(0)-\mathbf{z}_0)), \tag{2.41}$$

where $\mathbf{z}$ and $\mathbf{z}_0$ denote the points $\mathbf{x}(\tau)$ and $\mathbf{y}$, respectively, in complex domain.

We will describe the procedures for approximating $I_{t,1}$ and $I_{n,1}$ to second order. The approach is similar to the previous case where $\mathbf{y}$ is close to one of the end-points but exterior to the curve. The difference is the boundary between the long and short segments. We first consider the case where $\mathbf{y}$ is close to the curve but away from both end-points. For this case, the short segment is chosen to be near the intersection point $\mathbf{x}(\tau^*)$. We let τ_2' be associated with the grid point that is closest to $\mathbf{x}(\tau^*)$ and such that $\tau_2' > \tau^*$, and we let τ_1' to be the grid point that is closest to $\mathbf{x}(\tau^*)$ and such that $\tau_1' < \tau^*$. Then we pick

$$\tau_2 = \tau_2' + m\Delta\tau, \quad \tau_1 = \tau_1' - m\Delta\tau, \tag{2.42}$$

where $m = \mathcal{O}(\Delta\tau^{\frac{1}{3}})$. With this notation, (τ_1, τ_2) is the interval for the short segment, and the remainder of the curve are the two long segments. See Fig. 2.1, panel B. Second-order approximations for $I_{t,L}$ and $I_{t,S}$ are given by

$$I_{t,L} = T_{t,h,L}(\alpha(\tau))|_0^{\tau_1} + T_h(\alpha(\tau))|_{\tau_2}^1 + \frac{B_2}{2!}\Delta\tau^2\left(\alpha'(\tau_1) - \alpha'(\tau_2)\right) + \mathcal{O}(\Delta\tau^2), \tag{2.43}$$

$$I_{t,S} = T_{t,h,S}(\alpha - \tilde{\alpha})|_{\tau_1}^{\tau_2} + \int_{\tau_1}^{\tau_2} \tilde{\alpha}(\tau)\, d\tau + \mathcal{O}(\Delta\tau^2), \tag{2.44}$$

where $T_{t,h,L}$ and $T_{t,h,S}$ denote the trapezoidal approximations of the associated integrals. Second-order approximations for $I_{n,L}$ and $I_{n,S}$ are similar, with α and $\tilde{\alpha}$ replaced by β and $\tilde{\beta}$, respectively.

If $\mathbf{y}$ is interior to the curve, but lies close to $\mathbf{x}(1)$, then $\tau_2 \geqslant 1$. For this case, we integrate I_t by setting τ_2 in Eqs. (2.43) and (2.44) to 1. Similarly, if $\mathbf{y}$ is close to $\mathbf{x}(0)$ such that $\tau_1 \leqslant 0$, then we set $\tau_1 = 0$ in Eqs. (2.43) and (2.44). Analogous procedures are used to integrate I_n

Points away from the curve

Recall that a point is considered away from γ if its distance from the curve is greater than $C\sqrt{\Delta\tau}$ (we chose the exponent of $\Delta\tau$, k, to be $1/2$; that choice is made to ensure that the error is $\mathcal{O}(\Delta\tau^2)$, see below.). We will derive corrections for errors arising in the discretization of I_t in Eq. (2.2); the derivation of the corrections for I_n is analogous. To proceed, we let

$$\phi(\tau) = \frac{(\mathbf{x}(\tau) - \mathbf{y}) \cdot \mathbf{t}(\tau)}{|\mathbf{x}(\tau) - \mathbf{y}|^2} f_t(\tau) \tag{2.45}$$

for notational simplicity. After applying the Euler Maclaurin summation formula, we obtain the following expression, similar to Eq. (2.20):

$$\begin{aligned} I_t = \int_0^1 \phi(\tau)\, d\tau =& T_h(\phi(\tau)) + \frac{B_2}{2}\Delta\tau^2\big(\phi'(0) - \phi'(1)\big) \\ &+ \frac{B_4}{4!}\Delta\tau^4\big(\phi'''(0) - \phi'''(1)\big) - \Delta\tau^5 \int_0^1 P_5(M\tau)\phi^5(\tau)\, d\tau. \end{aligned} \tag{2.46}$$

We approximate I_t by

$$I_t \approx T_h(\phi(\tau)) + \frac{B_2}{2}\Delta\tau^2\big(\phi'(0) - \phi'(1)\big). \tag{2.47}$$

The remainder terms,

$$\frac{B_4}{4!}\Delta\tau^4\big(\phi'''(0) - \phi'''(1)\big) - \Delta\tau^5 \int_0^1 P_5(M\tau)\phi^5(\tau), \tag{2.48}$$

constitute error. We will show that the error is $\mathcal{O}(\Delta\tau^2)$. To that end, we apply Taylor expansion to $\phi(\tau)$ at $\tau = \tau^*$, where $\mathbf{x}(\tau^*)$ is the intersect between γ and the normal line that passes through $\mathbf{y}$. We obtain the following expression for I_t

$$I_t = \int_0^1 \frac{\hat{\tau} + \mathcal{O}(b\hat{\tau}) + \mathcal{O}(\hat{\tau}^2) + \mathcal{O}(b\hat{\tau}^2) + \mathcal{O}(\hat{\tau}^3) + \mathcal{O}(\hat{\tau}^4)\big(f(\tau^*) + \hat{\tau} f'(\tau^*) + \mathcal{O}(\hat{\tau}^2)\big)}{r^2}\, d\tau,$$

where $\hat{\tau} \equiv \tau - \tau^*$ and $r \equiv |\mathbf{x}(\tau) - \mathbf{y}|$. Below we analyze the term $\frac{\hat{\tau} f(\tau^*)}{r^2}$, which we denote by $\phi_1(\tau)$. The rest of the terms can be treated analogously. The analysis is based on the following corollary [37]:

corollary 2.1.2. *There exists a constant c such that $a \leqslant cr(\tau)$, $b \leqslant cr(\tau)$, $\tau \leqslant cr(\tau)$, and $\epsilon = \sqrt{a^2 + b^2}$ for all $\tau \in [0, 1]$.*

From Corollary 2.1.2 , we have the following inequality

$$|\phi_1(\tau)| = \frac{|\hat{\tau} f(\tau^*)|}{r^2} \leqslant \frac{C_1 r}{r^2} = \frac{C_1}{r}, \tag{2.49}$$

where C_1 is a constant. We then apply Lemma 2.1.1 for $k = m = 0$, $p = 1$, $l = 3$ and 5, and obtain that

$$|\phi_1'''(\tau)| \leqslant \frac{C_3}{r^4}, \quad |\phi_1^5(\tau)| \leqslant \frac{C_5}{r^6}. \tag{2.50}$$

Since $\mathbf{y}$ is assumed to be far away from γ, $r \geqslant C\sqrt{\Delta\tau}$. With this assumption, we derive the following bounds for the error terms

$$\Delta\tau^4\big(\phi_1'''(0) - \phi_1'''(1)\big) \leqslant \Delta\tau^4 \frac{K_3}{r^4} \leqslant \Delta\tau^4 \frac{K_3}{(C\Delta\tau^{\frac{1}{2}})^4} = \mathcal{O}(\Delta\tau^2),$$

$$\Delta\tau^5 \int_0^1 \phi_1^5(\tau)\, d\tau \leqslant \Delta\tau^5 \int_0^1 \frac{K_5}{r^6}\, d\tau \leqslant \Delta\tau^5 \int_0^1 \frac{K_5}{(C\Delta\tau^{\frac{1}{2}})^6}\, d\tau = \mathcal{O}(\Delta\tau^2), \tag{2.51}$$

for some constants K_3, and K_5. Because the error terms are $\mathcal{O}(\Delta\tau^2)$, the approximation for I_t given in Eq. (2.47) can achieve second-order accuracy.

Technically speaking, when $\mathbf{y}$ is sufficiently far from γ, the integral is not nearly singular, and corrections are not needed to attain second-order accuracy when the trapezoidal rule is used. Thus, the corrections in Eq. (2.47) need only be included when $\mathbf{y}$ lies between a distance of $C\sqrt{\Delta\tau}$ and some constant d from γ. However, for simplicity (to limit the number of cases), we include the corrections for all $\mathbf{y}$ farther than $C\sqrt{\Delta\tau}$ from γ.

2.1.2 Computing velocity integral

We now derive corrections needed for accurately evaluating the Stokes velocity integral (1.13), which, as previously noted, is singular when evaluated on the curve, and nearly singular close to the curve.

Points on the interface

We first discuss the evaluation of the part of the velocity integral (1.13) that involves the logarithmic term, which is singular when $\mathbf{y}$ is on the interface. Correction terms will be derived for the velocity integral in the x-direction (i.e., u); the integral for v can be computed analogously. Given that $\mathbf{y}$ lies on γ, let $\mathbf{y} = \mathbf{x}(\tau^*)$. We consider the logarithmic integral

$$I'_u \equiv \int_\gamma -f_x(\mathbf{x}) \log|\mathbf{x}-\mathbf{y}|\, d\mathbf{x} = \int_\gamma k(\tau) \log|\mathbf{x}(\tau)-\mathbf{x}(\tau^*)|\, d\tau, \tag{2.52}$$

where $k(\tau) \equiv -f_x(\tau)|\mathbf{x}'(\tau)|$. Applying the Taylor expansion at $\tau = \tau^*$, $k(\tau)$ and $\mathbf{x}(\tau)$ can be expressed as:

$$k(\tau) = k(\tau^*) + (\tau-\tau^*)k'(\tau^*) + \mathcal{O}((\tau-\tau^*)^2), \tag{2.53}$$

$$\mathbf{x}(\tau) = \mathbf{x}(\tau^*) + (\tau-\tau^*)\mathbf{x}'(\tau^*) + (\tau-\tau^*)^2 H(\tau), \tag{2.54}$$

for some function $H(\tau)$. Rearranging Eq. (2.54) yields

$$|\mathbf{x}(\tau)-\mathbf{x}(\tau^*)|^2 = (\tau-\tau^*)^2(|\mathbf{x}'(\tau^*)|^2 + (\tau-\tau^*)h_1(\tau) + (\tau-\tau^*)^2 h_2(\tau)), \tag{2.55}$$

for some functions $h_1(\tau)$ and $h_2(\tau)$. Then, taking the logarithm of both sides, one obtains

$$\log|\mathbf{x}(\tau)-\mathbf{x}(\tau^*)| = \log|\tau-\tau^*| + \frac{1}{2}\log\left(|\mathbf{x}'(\tau^*)|^2 + (\tau-\tau^*)h_1(\tau) + (\tau-\tau^*)^2 h_2(\tau)\right). \tag{2.56}$$

Substituting Eq. (2.53) and (2.56) into Eq. (2.52) yields

$$I'_u = \int_\gamma \big(k(\tau^*) + (\tau-\tau^*)k'(\tau^*)\big)\log|\tau-\tau^*|\,d\tau + \int_\gamma \mathcal{O}\big((\tau-\tau^*)^2\big)\log|\tau-\tau^*|\,d\tau$$

$$+ \int_\gamma k(\tau)\frac{1}{2}\log\big(|\mathbf{x}'(\tau^*)|^2 + (\tau-\tau^*)h_1(\tau) + (\tau-\tau^*)^2 h_2(\tau)\big)\,d\tau. \tag{2.57}$$

The first integral on the right-side of Eq. (2.57) can be integrally analytically to yield

$$\int_\gamma \big(k(\tau^*) + (\tau-\tau^*)k'(\tau^*)\big)\log|\tau-\tau^*|\,d\tau$$

$$= k(\tau^*)\big(\tau^*\log(\tau^*) + (1-\tau^*)\log(1-\tau^*) - 1\big)$$

$$+ k'(\tau^*)\frac{1}{4}\Big(2(\tau^*-1)^2\log(1-\tau^*) - (\tau^*-1)^2 - 2\tau^{*2}\log(\tau^*) + \tau^{*2}\Big)$$

The second integral on the right-side of Eq. (2.57) can be approximated by trapezoidal summation. For the third integral term, note that $|\mathbf{x}'(\tau^*)|^2$ is positive; thus, there is no singularity in $\log\big(|\mathbf{x}'(\tau^*)|^2 + (\tau-\tau^*)h_1(\tau) + (\tau-\tau^*)^2 h_2(\tau)\big)$. Therefore, trapezoidal summation can be also used to approximate the third integral on the right-side of Eq. (2.57) to second-order accuracy.

Points near the interface

When $\mathbf{y}$ is not on γ, but close to it, the boundary integral for $\mathbf{u}$ is nearly singular. To derive corrections for the quadrature approximation, we first apply a translation and a coordinate rotation to γ and $\mathbf{y}$, such that $(x_1(0), x_2(0)) = (0,0)$ and $(x_1'(0), x_2'(0)) = (1,0)$. Then, we rewrite the velocity integral (1.13) in the transformed coordinates as

$$I_u \equiv \int_\gamma k(\tau)\log|\mathbf{x}(\tau)-\mathbf{y}|\,d\tau + \int_\gamma \mathbf{K}(\tau)\cdot(\mathbf{x}-\mathbf{y})\frac{(x_1(\tau)-a)}{|\mathbf{x}(\tau)-\mathbf{y}|^2}\,d\tau, \tag{2.58}$$

where x_1 and a are the transformed x-coordinates of $\mathbf{x}$ and $\mathbf{y}$, respectively, $\mathbf{K}(\tau) \equiv (k(\tau), k_1(\tau))$, and $k_1(\tau) \equiv -f_y(\tau)|x'(\tau)|$. By a Taylor expansion of $k(\tau)$ around τ^*,

we can express I_u as the sum of three integrals

$$I_u = I_{u,1} + I_{u,2} + I_{u,3}, \tag{2.59}$$

where

$$I_{u,1} = \int_\gamma (k(\tau^*) + (\tau - \tau^*)k'(\tau^*)) \log|\mathbf{x}(\tau) - \mathbf{y}|\, d\tau, \tag{2.60}$$

$$I_{u,2} = \int_\gamma \mathcal{O}((\tau - \tau^*)^2) \log|\mathbf{x}(\tau) - \mathbf{y}|\, d\tau, \tag{2.61}$$

$$I_{u,3} = \int_\gamma \mathbf{K}(\tau) \cdot (\mathbf{x} - \mathbf{y}) \frac{(x_1(\tau) - a)}{|\mathbf{x}(\tau) - \mathbf{y}|^2}\, d\tau, \tag{2.62}$$

where τ^* is the intersection point between curve γ and the line normal to γ and through $\mathbf{y}$. When $\mathbf{y}$ is exterior to the curve and close to $\mathbf{x}(0)$ or $\mathbf{x}(1)$, we take $\tau^* = 0$ or 1, respectively. Note that when $\mathbf{y}$ is close to γ, $I_{u,1}$ and $I_{u,3}$ become nearly singular, but not $I_{u,2}$. Thus, $I_{u,2}$ can be computed accurately by means of the standard trapezoidal rule.

To derive an accurate evaluation of $I_{u,1}$, we integrate by parts to obtain

$$\begin{aligned} I_{u,1} = &\left(k(\tau^*)\tau + \frac{1}{2}(\tau - \tau^*)^2 k'(\tau^*) \right) \log|\mathbf{x}(\tau) - \mathbf{y}| \Big|_0^1 \\ &- \int_\gamma \left(k(\tau^*)\tau + \frac{1}{2}(\tau - \tau^*)^2 k'(\tau^*) \right) \frac{\partial}{\partial \tau} \log|\mathbf{x}(\tau) - \mathbf{y}|\, d\tau. \end{aligned} \tag{2.63}$$

Note that the integral in (2.63) has the form $\int_\gamma f(\tau)\partial_t G\, d\tau$. Therefore, $I_{u,1}$ can be approximated by the trapezoidal rule, with correction terms given in Eqs. (2.14), (2.24), (2.27), and (2.35).

To derive an accurate evaluation of $I_{u,3}$, we denote $\mathbf{K}(\tau)(x_1(\tau) - a)$ by $\mathbf{Q}$, which allows us to write $I_{u,3}$ in a form similar to the boundary integral of pressure. Thus, we can evaluate $I_{u,3}$ by means of the corrected integral approximations given in Eqs. (2.14), (2.23), (2.24), (2.25), (2.27), (2.33), (2.35), and (2.36).

2.2 The hybrid approach

The method of corrected boundary integrals described above can be used to evaluate the free space velocity and pressure at any point. However, if values are needed at grid points over the entire computational domain, this method, in a standard implementation without acceleration (see below), would be too expensive inasmuch as it requires $\mathcal{O}(MN^2)$ computations, where M and N denote the numbers of spatial subintervals along the immersed boundary and along one dimension in the computational domain, respectively. (A fast multipole-based method may yield substantially faster computations [17].) If M scales as N, then the computational cost is $\mathcal{O}(N^3)$. Another difficulty is that p and the derivatives of the fluid variables are unbounded near the end-points of the interface, which introduce large errors in standard mesh-based finite-difference schemes. Motivated by these challenges, we propose a hybrid approach for accurately evaluating Stokes solutions everywhere by combining the integral method with a mesh-based method. Briefly, we use corrected boundary integrals to compute the solution accurately near the interface and in the neighborhood of the end-points, then form an accurate value for the right hand side of a discrete grid problem, extend away from the interface, and finally solve the entire grid problem. This approach was suggested in [29] and used in [3].

To solve the Stokes equations, we reduce (1.1)–(1.2) to a sequence of Poisson problems, first for pressure, then for velocity, as in [27],

$$\Delta p = \nabla \cdot \mathbf{F}, \qquad \mu \Delta \mathbf{u} = \nabla p. \tag{2.64}$$

The Poisson problems are solved on a rectangular computational domain Ω. For the free space problem, the boundary values for p and $\mathbf{u}$ along the domain boundary $\partial\Omega$ are computed using (1.13), and specified as Dirichlet boundary conditions along $\partial\Omega$.

In the numerical discretization, let h denote the meshsize, assumed to be the same along the x- and y-directions. For an arbitrary function $\psi(x, y)$, let $\psi_{i,j}$ denote

$\psi(ih, jh)$. To solve for p using the hybrid approach, we first label the *irregular* grid points, defined as the grid points for which the five-point stencil of the discrete Laplacian crosses the interface. We then evaluate p, using the method of corrected boundary integrals as above, at all grid points needed to form the stencil at the irregular points. Thus p is computed as a boundary integral at irregular grid points as well as their four nearest neighbors. We then form the discrete Laplacian at the irregular grid points,

$$\Delta^h p = \frac{p_{i-1,j} + p_{i,j-1} - 4p_{i,j} + p_{i+1,j} + p_{i,j+1}}{h^2}, \tag{2.65}$$

where Δ^h denotes the discrete Laplacian. As in [29] we set $\Delta^h p$ to zero at the *regular* points where the stencil does not cross the interface, with $O(h^2)$ accuracy.

In previous studies [29, 3], the linear system for grid values of p is then solved to yield p that is uniformly $\mathcal{O}(h^2)$ accurate [4]. However, this approach cannot be applied directly to the present problem, owing to a complication inherent in using the hybrid approach to solve the open-curve immersed boundary problem: p is unbounded at the end-points. To compute uniformly $\mathcal{O}(h^2)$ approximations, we enclose each of the two end-points in a circle of radius R. How R should be chosen will be discussed later. We assume that $p = 0$ within the circles, and we solve the following problem instead:

$$\tilde{p}(\mathbf{x}) = \begin{cases} 0, & |\mathbf{x} - \mathbf{x}(0)| < R, \text{ or } |\mathbf{x} - \mathbf{x}(1)| < R, \\ p(\mathbf{x}), & \text{otherwise.} \end{cases} \tag{2.66}$$

The new problem (2.66) is solved using the standard hybrid approach. The irregular points in (2.66) include those near the open curve γ as well as the boundaries of the two circles. Unlike the original Poisson problem, $\tilde{p}$ is bounded everywhere; thus, the hybrid approach yields uniformly $\mathcal{O}(h^2)$ grid-point values for $\tilde{p}$. Then, outside of the two circles, we set $p = \tilde{p}$. Within the circles, values of p are computed using the method of corrected boundary integrals.

Next we compute the velocity $\mathbf{u}$. Because the derivatives of $\mathbf{u}$ are unbounded at the end-points, we again enclose each of the two end-points in a circle, and assume that $\tilde{\mathbf{u}} = 0$ within the circles. We then approximate ∇p at the regular points outside the circles by the second-order centered difference $\nabla^h p$ and set $\Delta^h \mathbf{u} = \nabla^h p/\mu$. As above, we compute $\mathbf{u}$ from the integral (1.13) at grid points that are required to form the discrete Laplacian $\Delta^h \mathbf{u}$ at the irregular grid points, and then we form the discrete Laplacian. With $\Delta^h \mathbf{u}$ prescribed, we then solve the Poisson equations.The method of corrected boundary integrals is used to compute grid-point $\mathbf{u}$ values within the circles.

In this approach, the method of corrected boundary integrals is used to evaluate solution values at irregular grid points, and at grid points within the two circles enclosing the end-points of γ. There are $\mathcal{O}(N)$ irregular grid points. Assuming that the size of the two circles does not scale with N, then there are $\mathcal{O}(N^2)$ grid points within those circles. That gives $\mathcal{O}(N^2)$ total solution values, which incur a computational cost of $\mathcal{O}(N^3)$. Applying fast cosine transforms to the linear system arising from a finite-difference discretization of the two-dimensional Poisson problem has a cost of $\mathcal{O}(N^2 \log N)$. Thus, the computational complexity of the hybrid approach does not seem to be an improvement over using the method of corrected boundary integrals to compute solution values everywhere. However, because the size of the circles can be chosen to be much smaller than the computational domain, the actual computational times incurred by the hybrid approach should be substantially lower.

To attain $\mathcal{O}(h^2)$ accuracy, R should be taken to be sufficiently large that the values of p outside the circles are not too large to reduce solution accuracy. However, because the computations within the circles ($\mathcal{O}(N^3)$) are more expensive than outside ($\mathcal{O}(N^2 \log N)$), a R value that is too large will require large computational times. Thus, what value of R should be chosen depends on one's balance of time and accuracy, and may be unclear *a priori*. In practice, we take a trial-and-error approach

by picking a R and checking the values of p at the boundaries of the circles. If any p value exceeds a maximum tolerance, R is increased.

2.3 Numerical tests

In this section, we present numerical examples to assess the accuracy of the methods presented in this study. We first use boundary integral examples to evaluate the accuracy of the the method of corrected boundary integrals. We then use a steady-state Stokes example to test the hybrid approach. In all examples, fluid viscosity μ is taken to be 0.1.

2.3.1 Boundary integral examples

Example 1

In this example, the boundary is the horizontal line segment parametrized by $(\tau, 0)$, for $\tau \in [0, 1]$. The boundary forces are given by $f_t(\tau) = f_n(\tau) = \tau + \tau^2$, where $f_t(\tau)$ and $f_n(\tau)$ are the forces in the tangential and normal directions, respectively. The analytic solutions are given by

$$\begin{aligned} I_t(\mathbf{y}) =& \frac{1}{2}(a^2 + a - b^2) \log\left(\frac{(1-a)^2 + b^2}{a^2 + b^2}\right) \\ &+ (2a+1)b\left(\arctan\left(\frac{a-1}{b}\right) - \arctan\left(\frac{a}{b}\right)\right) + a + 1.5, \\ I_n(\mathbf{y}) =& b\left(a + \frac{1}{2}\right) \log\left(\frac{(a-1)^2 + b^2}{a^2 + b^2}\right) \\ &+ (a^2 - b^2 + a)\left(\arctan\left(\frac{1-a}{b}\right) - \arctan\left(\frac{-a}{b}\right)\right) + b. \end{aligned} \tag{2.67}$$

where $\mathbf{y} = (a, b)$.

The integrals are evaluated at five points: $\mathbf{y} = \{(-\Delta\tau^{\frac{1}{2}}, \Delta\tau^{\frac{1}{2}}),\ (\Delta\tau^{\frac{1}{2}}, \Delta\tau^{\frac{1}{2}}),\ (0.5, \Delta\tau^{\frac{1}{2}}),\ (1 - \Delta\tau^{\frac{1}{2}}, \Delta\tau^{\frac{1}{2}}),\ (1 + \Delta\tau^{\frac{1}{2}}, \Delta\tau^{\frac{1}{2}})\}$. The points $(-\Delta\tau^{\frac{1}{2}}, \Delta\tau^{\frac{1}{2}})$ and $(1 + \Delta\tau^{\frac{1}{2}}, \Delta\tau^{\frac{1}{2}})$ are close to an end-point and exterior to the curve. The points $(\Delta\tau^{\frac{1}{2}}, \Delta\tau^{\frac{1}{2}})$

Table 2.1: Convergence results for Example 1. Approximations computed using the method of corrected boundary integrals exhibit second-order convergence and are much more accurate than the trapezoidal rule, which suffers order reduction.

	Trapezoidal rule		Corrected boundary integrals	
M	L_1 error/$\Delta\tau^2$	L_{inf} error/$\Delta\tau^2$	L_1 error/$\Delta\tau^2$	L_{inf} error/$\Delta\tau^2$
		I_t		
10^3	2.105e0	4.028e0	5.215e-2	1.105e-1
11^3	2.425e0	4.637e0	5.054e-2	1.091e-1
12^3	2.771e0	5.272e0	4.981e-2	1.080e-1
13^3	3.123e0	5.938e0	4.843e-2	1.069e-1
14^3	3.493e0	6.629e0	4.838e-2	1.060e-1
15^3	3.870e0	7.339e0	4.772e-2	1.052e-1
		I_n		
10^3	1.074e0	3.958e0	6.101e-2	9.466e-2
11^3	1.232e0	4.568e0	5.962e-2	9.246e-2
12^3	1.401e0	5.202e0	5.811e-2	9.006e-2
13^3	1.583e0	5.862e0	5.658e-2	8.749e-2
14^3	1.766e0	6.553e0	5.580e-2	8.678e-2
15^3	1.954e0	7.263e0	5.493e-2	8.573e-2

and $(1 - \Delta\tau^{\frac{1}{2}}, \Delta\tau^{\frac{1}{2}})$ are close to an end-point and interior to the curve. The point $(0.5 + \Delta\tau^{\frac{1}{2}}, 0)$ is close to the curve and away from the end-points. We evaluate I_t and I_n in Eqs. (2.2) and (2.3), associated with the p integral solution, using both the standard trapezoidal rule and the method of corrected boundary integrals, and we compare their accuracy. The errors for I_t and I_n are exhibited in Table 2.1. The number of grid points M are chosen so that $m = \Delta\tau^{-1/3}$, the number of grid points along the short segment, is an integer. [When smaller values of M are used such that $\Delta\tau^{1/3}$ is not an integer or such that m is too small, the convergence rates are less robust (results not shown).] For both I_t and I_n, the method of corrected boundary integrals yields approximations with $\mathcal{O}(\Delta\tau^2)$ accuracy, an accuracy that is substantially higher than the trapezoidal rule.

Example 2

In the second example, the boundary is taken to be a quadratic line segment, parameterized by (τ, τ^2), $\tau \in [0, 1]$. The boundary force is given by $f_t(\tau) = f_n(\tau) = \exp(\tau)$. Because the analytical solution is not known for this example, we use a highly refined solution, computed using 40^3 grid points, as the reference solution to estimate the errors. We evaluate the pressure integrals (2.2) and (2.3) at $(-\Delta\tau^2, \Delta\tau^2)$, $(\Delta\tau^2, \Delta\tau^2)$, $(0.5 + \Delta\tau^2, 0.25 + \Delta\tau^2)$, $(1 - \Delta\tau^2, 1 - \Delta\tau^2)$, and $(1 + \Delta\tau^2, 1 + \Delta\tau^2)$. The errors are shown in Table 2.2. As in the previous example, the method of corrected boundary integrals yields approximations with $\mathcal{O}(\Delta\tau^2)$ accuracy, whereas the trapezoidal rule exhibits order reduction.

Table 2.2: Convergence results for Example 2. Approximations computed using the method of corrected boundary integrals exhibit second-order convergence and are much more accurate than the trapezoidal rule, which suffers order reduction.

	Trapezoidal rule		Corrected boundary integrals	
M	L_1 error/$\Delta\tau^2$	L_{inf} error/$\Delta\tau^2$	L_1 error/$\Delta\tau^2$	L_{inf} error/$\Delta\tau^2$
		I_t		
10^3	6.109e2	1.331e3	1.201e-1	1.988e-1
11^3	5.877e2	1.765e3	1.200e-1	1.983e-1
12^3	1.030e3	2.279e3	1.198e-1	1.980e-1
13^3	9.632e2	2.878e3	1.196e-1	1.976e-1
14^3	1.610e3	3.564e3	1.195e-1	1.973e-1
15^3	1.463e3	4.340e3	1.193e-1	1.970e-1
		I_n		
10^3	2.944e2	8.118e2	6.272e-1	1.135e-1
11^3	1.784e2	5.334e2	6.248e-1	1.134e-1
12^3	5.006e2	1.386e3	6.248e-1	1.162e-1
13^3	2.936e2	8.671e2	6.231e-1	1.134e-1
14^3	7.804e2	2.165e3	6.226e-1	1.133e-1
15^3	4.517e2	1.305e3	6.211e-1	1.131e-1

Example 3

In this example, we seek to evaluate the velocity integral (1.13) when $\mathbf{y}$ is on the interface. We use the quadratic curve (τ, τ^2) and the boundary forces in Example

2. The integral is evaluated at $\tau = 0$, $\tau = 0.3$, and $\tau = 1$. Numerical results, which exhibit second-order convergence, are shown in Table 2.3. Because $\tau = 0$ and $\tau = 1$ are natural grid points, standard trapezoidal rule cannot be applied directly.

Table 2.3: Example 3: Convergence results for $\mathbf{u}$, evaluated at points on the curve by means of the method of corrected boundary integrals. Approximations exhibit second-order convergence.

M	L_1 error/$\Delta\tau^2$	L_{inf} error/$\Delta\tau^2$
11^3	9.093e0	2.883e1
12^3	9.211e0	2.926e1
13^3	9.320e0	2.965e1
14^3	9.414e0	3.001e1
15^3	9.500e0	3.032e1
16^3	9.573e0	3.060e1

Example 4

To assess the accuracy of the method in evaluating the velocity integral at points that are close, but not on, the interface, we conduct another numerical test, where we compute $\mathbf{u}$ at the points $\mathbf{y}$ given in Example 2. The interface and boundary force are those given in Examples 2 and 3. Numerical results for the method of corrected boundary integrals indicate second-order accuracy, whereas results for the trapezoidal rule show lower convergence rate and have substantially larger errors; see Table 2.4.

Example 5

Next we test the method on an example with a buckled curve, which is composed of two straight lines, parameterized by $(\tau, 0)$,(τ, τ) for $\tau \in [0, 1]$. We evaluate p and $\mathbf{u}$ at five points: $(-\Delta\tau, \Delta\tau)$, $(0.5, \Delta\tau)$,$(1.0 + \Delta\tau, 0)$,$(1.0 + \Delta\tau, 1.0 + \Delta\tau)$, $(0.5, 0.5 + \Delta\tau)$. The boundary forces are those given in Example 2. The point $(-\Delta\tau, \Delta\tau)$ is sufficiently close to both lines that corrections for both lines must be included.

Table 2.4: Example 4: Convergence results for $\mathbf{u}$, evaluated at points near the curve by means of the method of corrected boundary integrals. Approximations exhibit second-order convergence.

	Trapezoidal rule		Corrected boundary integrals	
M	L_1 error/$\Delta\tau^2$	L_{inf} error/$\Delta\tau^2$	L_1 error/$\Delta\tau^2$	L_{inf} error/$\Delta\tau^2$
10^3	1.368e3	4.239e3	3.777e-1	8.235e-1
11^3	1.310e3	5.620e3	3.782e-1	8.237e-1
12^3	2.339e3	7.257e3	3.789e-1	8.237e-1
13^3	2.147e3	3.703e3	3.791e-1	8.236e-1
14^3	3.656e3	1.1347e4	3.792e-1	8.233e-1
15^3	3.263e3	1.3817e4	3.791e-1	8.227e-1

Numerical results, obtained using the method of corrected boundary integrals, which shown in Table 2.5.

Table 2.5: Example 5: Convergence results for p and $\mathbf{u}$, where the boundary is a buckled curve.

	p		$\mathbf{u}$	
N	L_1 error/$\Delta\tau^2$	L_{inf} error/$\Delta\tau^2$	L_1 error/$\Delta\tau^2$	L_{inf} error/$\Delta\tau^2$
80	2.156e-2	4.083e-2	2.723e-1	1.490
160	2.152e-2	3.994e-2	3.028e-1	1.709
320	2.075e-2	3.777e-2	3.205e-1	1.858
640	1.742e-2	3.004e-2	2.822e-1	1.670

2.3.2 Stokes examples

Example 6

To valid our method, we solve a Stokes problem with has an analytic solution. We use the hybrid approach to compute Stokes solution at each grid point on a two-dimensional rectangular domain: $[-1.0, 2.0] \times [-1.0, 2.0]$. In this example, the boundary is the horizontal line segment parametrized by $(\tau, 0)$, for $\tau \in [0, 1]$ The tangential and normal components of the boundary forces are given by $f_t(\tau) = f_n(\tau) =$

$\tau + \tau^2$, respectively. The analytic solutions are given by

$$p(\mathbf{y}) = \frac{1}{2\pi}\Big(\frac{1}{2}(-a^2 - 3a + b^2 - 1)\log\left(\frac{(1-a)^2 + b^2}{a^2 + b^2}\right) + (a^2 + a - b^2 + 2ab + b)\left(\arctan\big(\frac{1-a}{b}\big) - \arctan\big(\frac{a}{b}\big)\right) + b - a - \frac{3}{2}\Big), \tag{2.68}$$

$$\begin{aligned} u(\mathbf{y}) = \frac{1}{144\pi\mu}\Big(&- \big(15 + 18b^3 - (18a^2 + 18a)b - 6a^3 - 9a^2 + (54a + 27)b^2\big)\log((1-a)^2 + \\ & b^2) - \big(- 48b^3 + (72a + 36)b^2 + (72a^2 + 72a)b\big)\big(\arctan(\frac{1-a}{b}) - \arctan(\frac{-a}{b})\big) \\ & + 43 + 54b + 24a - 48b^2 + 36ab + 12a^2 + \big(18b^3 - (18a^2 + 18a)b + \\ & (54a + 27)b^2 - 6a^3 - 9a^2\big)\log(a^2 + b^2)\Big) \end{aligned} \tag{2.69}$$

$$\begin{aligned} v(\mathbf{y}) = \frac{1}{144\pi\mu}\Big(&\big(15 + 18b^3 + 18b^3 - 6a^3 - 9a^2 - (18a + 9)b^2 - (18a^2 + 18a)b\big) \\ & \log((1-a)^2 + b^2) + \big(24b^3 + (72b + 36)b^2\big)(\arctan(\frac{1-a}{b}) - \arctan(\frac{a}{b})) \\ & - \big(18b^3 - (18a + 9)b^2 - (18a^2 + 18a)b - 6a^3 - 9a^2\big)\log(a^2 + b^2) \\ & - 13 - 54b - 24a - 24b^2 - 36ab - 12a^2\Big) \end{aligned} \tag{2.70}$$

where $\mathbf{y} = (a, b)$.

We compute solution on 160×160 grid points, and compare it with the analytic solution (2.68)–(2.70). Table 2.6 displays the L_1 and L_∞ errors in the solutions on the N^2 grid points. Both p and $\mathbf{u}$ exhibit second-order convergence as the discretization of the boundary is increased; see Table 2.6.

Example 7

In another example, we use the hybrid approach to compute Stokes solution at each grid point on a two-dimensional rectangular domain: $[-1, 2] \times [-1, 2]$. As in Example 2, the boundary is parameterized by (τ, τ^2), with $\tau \in [0, 1]$. The boundary

Table 2.6: Example 6, Stokes equations. Convergence results for p and $\mathbf{u}$, evaluated at all grid points by means of the hybrid approach and computed using analytic solution (2.68)–(2.70). Approximations exhibit second-order convergence.

	p		$\mathbf{u}$	
M	L_1 error/$\Delta\tau^2$	L_{inf} error/$\Delta\tau^2$	L_1 error/$\Delta\tau^2$	L_{inf} error/$\Delta\tau^2$
480	1.928e-1	1.607	6.452e-1	6.337
960	1.901e-1	1.529	5.584e-1	3.760
1920	1.284e-1	1.274	3.079e-1	2.108
3840	1.054e-1	1.181	2.283e-1	1.597

force is given by $f(\tau) = \exp(\tau)$. Numerical solutions are computed for $N = 40$, 80, 160, and 320, where N denotes the number of subintervals in each spatial dimension. The immersed boundary is discretized using $M = 3N$ grid points. The two circles that we enclose the immersed boundary end-points to avoid the singularities of the solutions are taken to have radius 0.15, for all spatial resolutions. Figure 2.2 shows the fluid pressure and velocity profiles at $y \approx 0.5$ (panel A) and $y \approx 1.0$, obtained using $N = 320$. (These positions are approximate owing to the spatial discretization.) The line $y = 0.5$ intercepts the interface at $x = 0.707$. At that interception, p and the derivatives of u and v exhibit jump discontinuities, which are captured by the hybrid method. The line $y = 1.0$ touches one of the end-points of the interface, where p and the derivatives of u and v are unbounded. The computed solution clearly indicates p approaching infinity, although the velocity derivatives are more difficult to discern. To assess the accuracy of the computed solutions, we use the solution computed on a high-resolution $N = 640$ grid as the reference solution to estimate errors. Table 2.7 displays the L_1 and L_∞ errors in the solutions on the N^2 grid points. Both p and $\mathbf{u}$ exhibit second-order convergence.

2.4 Simulations of ciliary motion

To demonstrate the application of the method of corrected boundary integrals, we use the method to simulate the fluid motion driven by the beating motion of a

Table 2.7: Example 7, Stokes equations. Convergence results for p and $\mathbf{u}$, evaluated at all grid points by means of the hybrid approach. Approximations exhibit second-order convergence.

	p		$\mathbf{u}$	
N	L_1 error/$\Delta\tau^2$	L_{inf} error/$\Delta\tau^2$	L_1 error/$\Delta\tau^2$	L_{inf} error/$\Delta\tau^2$
40	1.928e-1	1.607	6.452e-1	6.337
80	1.901e-1	1.529	5.584e-1	3.760
160	1.284e-1	1.274	3.079e-1	2.108
320	1.054e-1	1.181	2.283e-1	1.597

cilium. Motile cilia, which are whiplike structures that protrude from the cell surface, are found in many eucaryotic cells. The beating motion of motile cilia generates hydrodynamic flows that are believed to play a key role during the morphogenesis of higher organisms, such as the left-right symmetry breaking of a mammalian embryo. The whip-like motion of pulmonary cilia has an important impact in lung health. The typical length of a pulmonary cilium is $\sim$6 μm, and it beats at a frequency of $\sim$15 Hz. At these scales, the Reynolds number is nearly zero, and the hydrodynamics generated by its beating motion can be described by Stokes flows.

2.4.1 Model formulation

The model is formulated as an immersed boundary problem as in [8] and represents an elastic cilium moving in Stokes flow. The cilium is modeled as an immersed boundary, i.e., an one-dimensional open curve immersed in the fluid. One end of the model cilium is assumed to be attached to a wall or floor. Thus, to compute fluid motion, one may use Green functions corresponding free-space Stokes flow with an infinite wall [33]. However, the method of corrected boundary integrals was developed for free space Stokes flow (with no wall). To simulate the presence of the floor, we represent a second immersed boundary, which is assigned boundary forces so that fluid velocity is approximately zero near the floor. Thus, the force term $\mathbf{F}$ in Eq. (1.1) arise from forces along the cilia boundary (denoted γ_c) and the floor

boundary (denoted γ_f). Thus, $\mathbf{F}$ has the form

$$\mathbf{F}(\mathbf{y}) = \mathbf{F}_c(\mathbf{y}) + \mathbf{F}_f(\mathbf{y}), \tag{2.71}$$

where

$$\mathbf{F}_c(\mathbf{y}) = \int_{\gamma_c} \mathbf{f}_c(s)\delta\big(\mathbf{y} - \mathbf{x}(s)\big)\, d\mathbf{x}(s), \tag{2.72}$$

$$\mathbf{F}_f(\mathbf{y}) = \int_{\gamma_f} \mathbf{f}_f(s')\delta\big(\mathbf{y} - \mathbf{x}(s')\big)\, d\mathbf{x}(s'), \tag{2.73}$$

where $\mathbf{x}$ denotes a point on γ_c or γ_f, which are parametrized by s and s', respectively, and δ is the Dirac delta function. Below we describe the boundary force densities $\mathbf{f}_c$ and $\mathbf{f}_f$.

The cilium is modeled as an elastic boundary that moves in a whiplike motion similar to that observed in pulmonary cilia. The force density $\mathbf{f}_c$ is assumed to consist of two components, an elastic force and a tether force (denoted $\mathbf{f}_{c,e}$ and $\mathbf{f}_{c,t}$, respectively):

$$\mathbf{f}_c = \mathbf{f}_{c,e} + \mathbf{f}_{c,t}. \tag{2.74}$$

The elastic force is given by

$$\mathbf{f}_{c,e}(s) = \frac{\partial}{\partial s}\left(T(s)\hat{\tau}(s)\right), \tag{2.75}$$

where T is the tension in the ciliary boundary and $\hat{\tau}$ is the unit tangent vector, i.e.,

$$T(s) = T_0\left(\left|\frac{\partial \mathbf{X}}{\partial s}\right| - 1\right), \quad \hat{\tau} = \frac{\partial \mathbf{X}}{\partial s} \tag{2.76}$$

The tension coefficient T_0 is the material property of the ciliary boundary. An elastic force is exerted on the fluid when the boundary is stretched or compressed.

To simulate the beating motion of a pulmonary cilium, we employ moving tether forces [15, 14, 38]. A target configuration of the model cilium is specified by $\mathbf{Y}_c(s,t)$.

The ciliary boundary is tethered to the target configuration by a set of elastic springs, such that as the target configuration is advanced in time, a force required to move the cilium boundary sufficiently close to the target configuration is generated:

$$\mathbf{f}_{c,t}(\mathbf{X}_c(s,t),t) = k_c\left(\mathbf{X}_c(s,t) - \mathbf{Y}_c(s,t)\right). \tag{2.77}$$

Both the elastic and tether forces are singular force density defined along the immersed ciliary boundary γ_c.

A tether force is exerted along the floor boundary. Let $\mathbf{Y}_f(s')$ denote the fixed target floor position. The tether force ($\mathbf{f}_f$) is given by

$$\mathbf{f}_f(\mathbf{X}_f(s',t),t) = k_f\left(\mathbf{X}_f(s',t) - \mathbf{Y}_f(s',t)\right) \tag{2.78}$$

2.4.2 Model parameters

Table 2.8: Fourier-least squares coefficients for the ciliary beat pattern. Data corresponds to Table 2(B) in [16]. The upper and lower numbers in each entry correspond to the x and y components respectively.

				A_{mn}			
n	0	1	2	3	4	5	6
$m=1$	-0.449	0.130	-0.169	0.063	-0.050	-0.040	-0.068
	2.076	-0.003	0.054	0.007	0.026	0.022	0.010
$m=2$	-0.072	-1.502	0.260	-0.123	0.011	-0.009	0.196
	-1.074	-0.230	-0.305	-0.180	-0.069	0.001	-0.080
$m=3$	0.658	0.793	-0.251	0.049	0.009	0.023	-0.111
	0.381	0.331	0.193	0.082	0.029	0.002	0.048
				B_{mn}			
n	1	2	3	4	5	6	
$m=1$	-0.030	-0.093	0.037	0.062	0.016	-0.065	
	0.080	-0.044	-0.017	0.052	0.007	0.051	
$m=2$	1.285	-0.036	-0.244	-0.093	-0.137	0.095	
	-0.298	0.513	0.004	-0.222	0.035	-0.128	
$m=3$	-1.034	0.050	0.143	0.043	0.098	-0.054	
	0.210	-0.367	0.009	0.120	-0.024	0.102	

The model's target ciliary configuration is based on data given by Fulford and Blake [16], who analyzed ciliary profiles obtained by Sanderson and Sleigh [35], and approximated those profiles by truncated Fourier series. At time t, the target configuration is approximated by

$$\mathbf{Y}_c(s,t) = \frac{1}{2}\mathbf{a}_0 + \sum_{n=1}^{N_0} \mathbf{a}_n(s)\cos(n\omega t) + \mathbf{b}_n(s)\sin(n\omega t) \tag{2.79}$$

where $\omega/2\pi$ is the angular beat frequency. In [16] s denotes the arclength; in this study, we take s to be the material coordinate, which corresponds to the arclength at initial time. The Fourier coefficients $\mathbf{a}_n$ and $\mathbf{b}_n$ are given by

$$\mathbf{a}_n = \sum_{m=1}^{M_0} \mathbf{A}_{mn} s^m, \quad \mathbf{b}_n = \sum_{m=1}^{M_0} \mathbf{B}_{mn} s^m \tag{2.80}$$

where A_{mn} and B_{mn} are constants given in Table 2.8. Time-profiles of the cilia beat cycles are displayed in Fig. 2.3.

In this study, the fluid viscosity μ is set to 0.1. The angular frequency $\omega/2\pi$ is taken to be $15/2\pi$. The tension coefficient T is set to 0.1; the spring constants for the model cilium (k_c) and for the floor (k_f) are taken to be 10 and 1000, respectively.

2.4.3 Numerical results

The model equations (1.1) and (1.2) are integrated in time using the method of corrected boundary integrals. To advance the model cilium, fluid velocity on the ciliary boundary is first computed, as described in Sect. 2.1.2. Then the boundary position is updated using a second-order Runge-Kutta method.

The time-step Δt was taken to be 0.0005. The ciliary boundary γ_c and the floor are discretized using 960 and 216 spatial subintervals, respectively. The profiles of the simulated cilia beat cycle closely approximate those shown in Fig. 2.3 for $t = iT$,

where $i = 0, 0.1, \ldots, 0.9$ and $T = 2\pi/\omega$ is the period of the beat cycle. In approximately the first 1/3 of the beat cycle, the cilium generates a forward stroke, during which the cilium is relatively extended and makes an oar-like movement towards one side. The motion of the cilium is similar to a rigid rod rotating about its base. The forward stroke is also called the effective stroke, because it is believed to be a power stroke that is responsible for transporting substances forward. In the reminder of the beat cycle, the cilium moves back by propagating a bend from base to tip. This is known as the recovery stroke. Unlike the effective stroke, here the cilium tries to move parallel to and as close as possible to the floor. During the simulation, the position of the floor remains approximately unchanged.

Owing to the forces that are singularly supported along the immersed boundaries, the fluid solution and their derivatives are expected to exhibit jump discontinuities across the boundaries. Figure 2.4 shows 1D profiles of the Stokes fluid pressure (p) and velocity (u and v) at $t = 0.3T$. The solid lines correspond to profiles along the horizontal line $y = 0.9475$, which intercepts the tip of the model cilium, and the dashed lines correspond to profiles along $y = 0.4866$, which intercepts approximately the midpoint of the model cilium. The pressure profiles in Fig. 2.4A indicate that p exhibits a jump discontinuity across the interior of the cilium, and is unbounded at the tip. The normal derivatives of the fluid velocity are expected to exhibit a jump discontinuity across the interior of the cilium, and to become unbounded at the tip. However, these jumps are not clear in the results shown in Figs. 2.4B and 2.4C. This discrepancy can be attributed to the observation that at $t = 0.3T$, the cilium is appropriately straight and unstretched, so the tension force is small. As a result, the normal component of the boundary force, most of which arises from the tether force, is substantially larger than the tangential component. And because the jump discontinuities of the velocity gradient are proportional to the tangential component of the boundary force [27, 32], those jumps are not obvious in Figs. 2.4B and 2.4C.

In contrast, because the jump in p is proportional to the normal component of the boundary force [27, 32], it can be observed in Fig. 2.4A.

Figure 2.5 shows profiles for p, u, and v obtained at $t = 0.7T$, along the vertical line $x = 0.3056$ (solid line), which intercepts the cilium at its tip, and along the horizontal line $y = 0.4725$), which intercepts the midpoint of the cilium. During the recovery stroke, the tangential component of the boundary force is sufficiently large that the jump discontinuity and unboundedness of the normal derivatives of the velocity can be seen in Figs. 2.5B and 2.5C. Note also that u and v are approximately zero near the floor ($y = 0$ along the solid lines), as expected.

To illustrate the hydrodynamics effects generated by the cilium at different phases of its beat cycle, we show p and $\mathbf{u}$ everywhere in the 2D fluid computational domain $[-2, 2] \times [-2, 2]$ during the effective stroke ($t = 0.3T$, Fig. 2.6) and during the recovery stroke ($t = 0.7T$, Fig. 2.7). The computational domain is discretized using 320 subintervals along each spatial domain. The configuration of the cilium is overlaid on the velocity plot (see Figs. 2.6 and 2.7, panels A). During the effective stroke, the cilium is approximately straight; thus, it exerts a relatively large effect on the fluid (compare the maximum pressure values in Figs. 2.6B and 2.7, and compare the size of the regions in Figs. 2.6C and 2.7 with near maximum velocity magnitude). During the recovery stroke, its effect on the surrounding fluid is reduced because the cilium has a more curved shape. Note also that, in both cases, fluid motion is larger near the tip of the cilium. Owing to the asymmetry in the effective and recovery strokes, a net propulsion effect is generated in the direction of the effective stroke.

During one period of simulation, the wall remains approximately stationary. The maximum speed and displacement of the wall were computed to be 0.01214 and 0.004005, respectively. We also compute L_1 and L_{inf} errors for the solution at $t = 0.3T$, using, as a reference solution, the approximation computed on a 320×320 grid. The errors, shown in Table 2.9, exhibit second-order convergence.

Table 2.9: Ciliary motion example. Convergence results for p and $\mathbf{u}$, evaluated at all grid points. Approximations exhibit second-order convergence.

	p		$\mathbf{u}$	
N	L_1 error/$\Delta\tau^2$	L_{inf} error/$\Delta\tau^2$	L_1 error/$\Delta\tau^2$	L_{inf} error/$\Delta\tau^2$
40	2.571e-1	2.559	8.768e-1	4.328
80	3.162e-1	3.174	9.863e-1	4.254
160	9.238e-2	2.926	2.848e-1	2.192

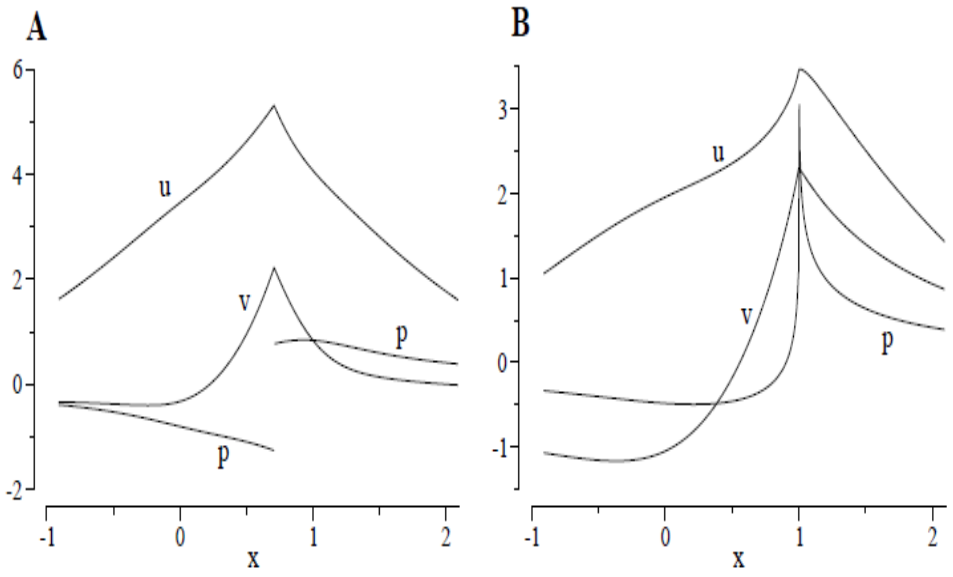

FIGURE 2.2: Pressure and fluid velocity obtained for Example 7. A, values at $y \approx 0.5$. At $x \approx 0.707$, p and the derivatives of u and v exhibit jump discontinuities. B, values at $y \approx 1.0$. At $x \approx 1.0$, p and the derivatives of u and v approach ∞.

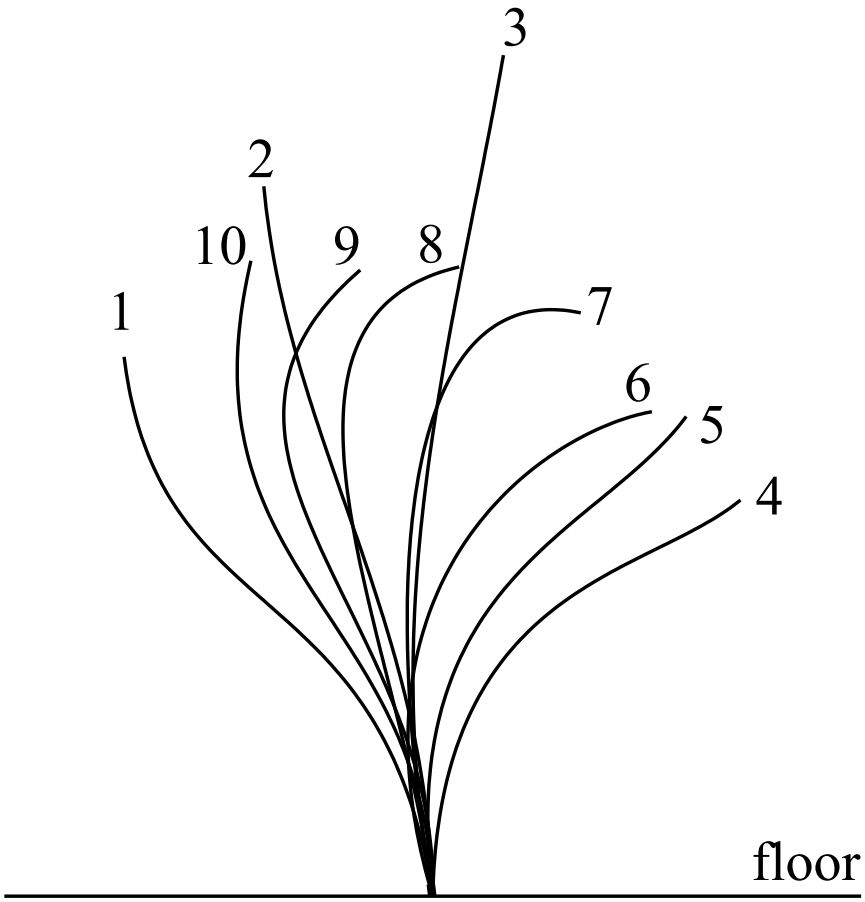

FIGURE 2.3: Target cilia configuration in one beat cycle. The label i corresponds to phase $2\pi(i-1)$ of the cycle.

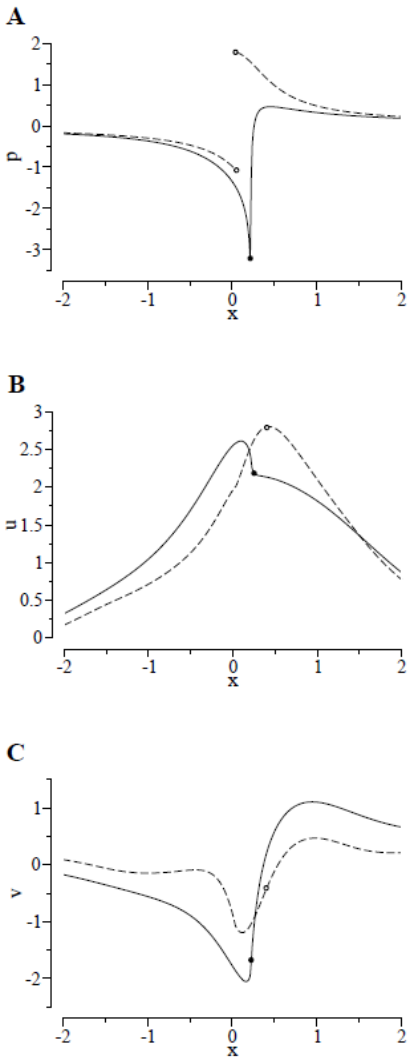

FIGURE 2.4: 1D profiles of p (A), u (B), and v (C) at $t = 0.3T$. Solid line, along $y = 0.9475$; dashed line, along $y = 0.4866$

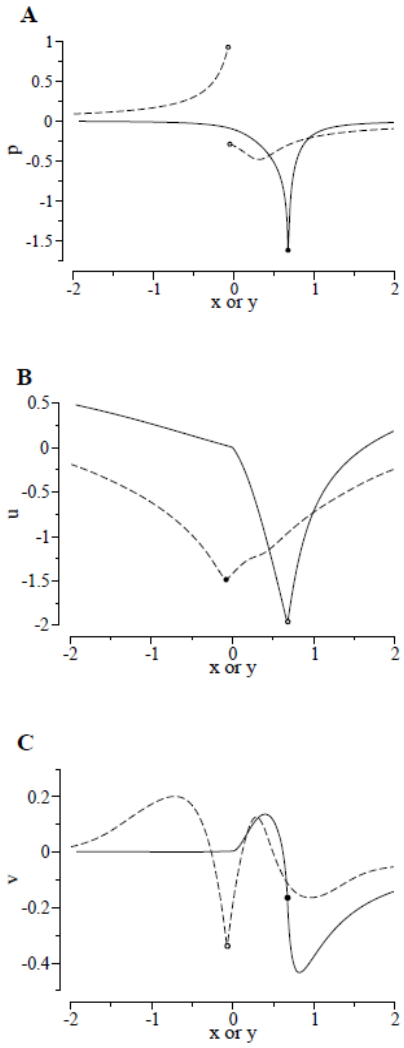

FIGURE 2.5: 1D profiles of p (A), u (B), and v (C) at $t = 0.7T$. Solid line, along $x = 0.3056$; dashed line, along $y = 0.4725$

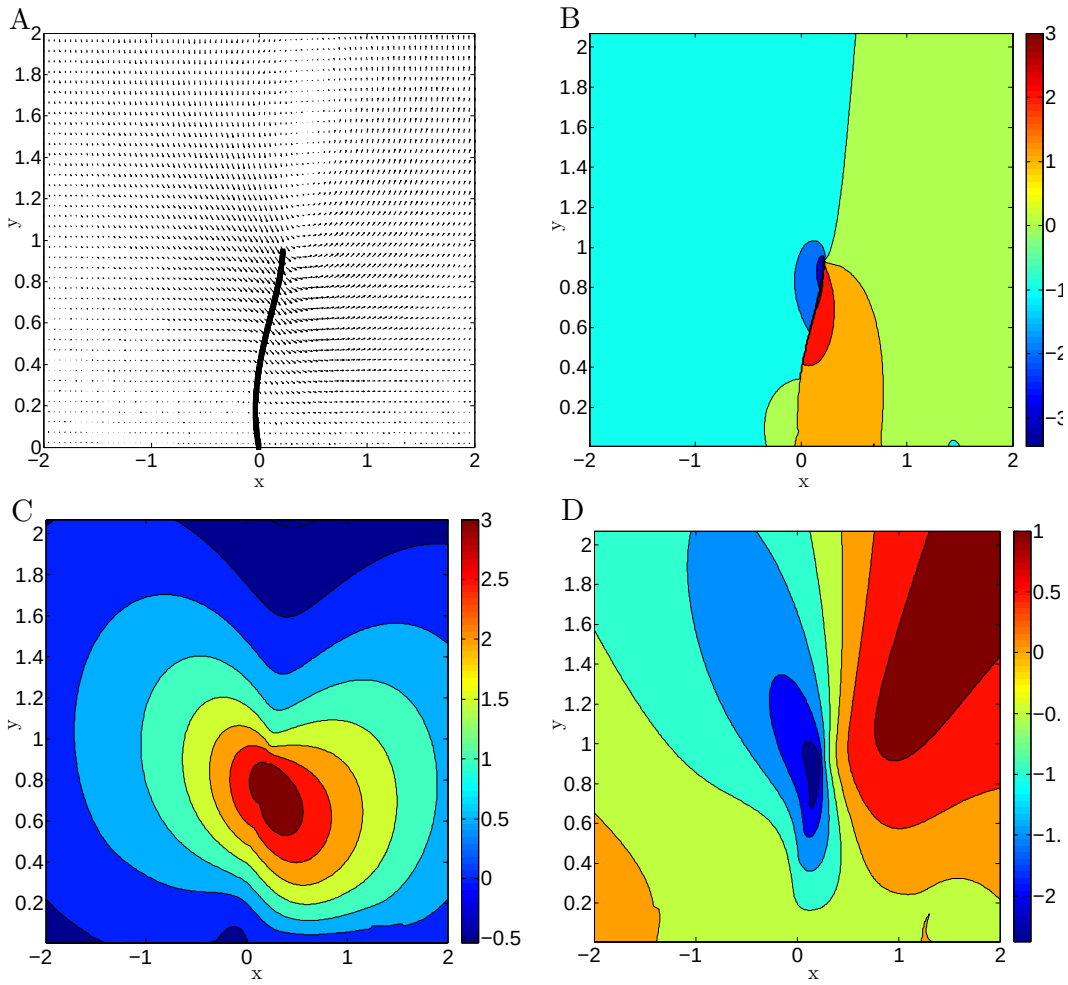

FIGURE 2.6: Plots of velocity vectors (A), pressure (B), x- and y-components of velocity (C and D) at $t = 0.3T$.

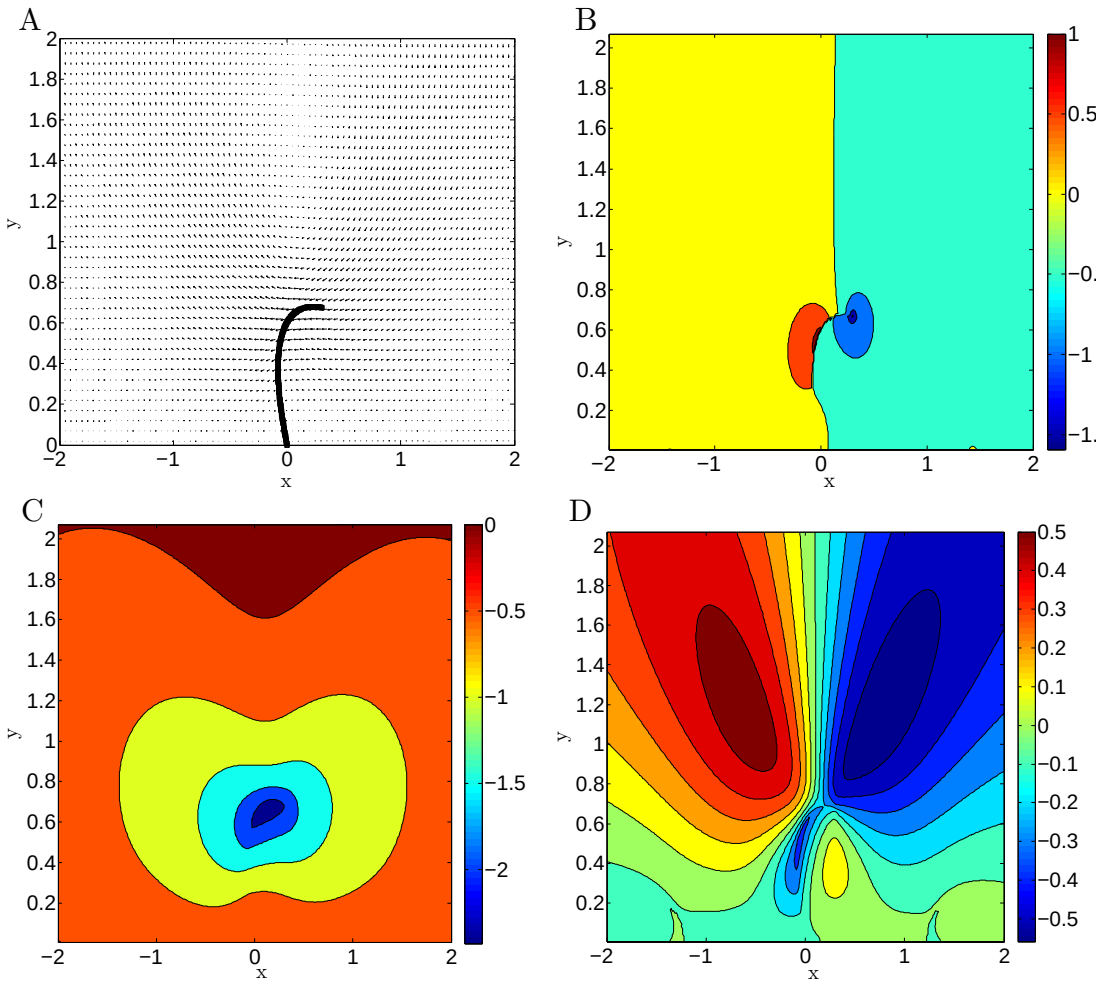

FIGURE 2.7: Plots of velocity vectors (A), pressure (B), x- and y-components of velocity (C and D) at $t = 0.7T$.

3

Simulating Driven Stokes Flow in an Elastic Tube

3.1 Governing equations

In this study, our aim is to simulate driven Stokes flow in a three-dimensional elastic tube. To take advantage of boundary integral solutions, we model the tube wall, Γ, as a closed surface, e.g., an ellipsoid or closed tube. To represent inflow and outflow conditions in this closed domain, we incorporate an internal source and sink, located

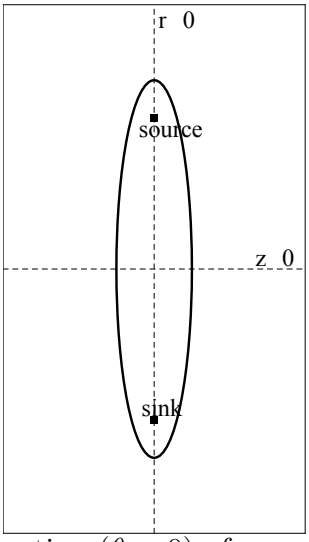

FIGURE 3.1: Cross section ($\theta = 0$) of computational domain Ω.

at the ends of the tube (see Fig. 3.1). The impermeable tube is immersed in fluid in the computational domain Ω. In this study, the characteristics of the fluid (i.e., the viscosity) are assumed to be identical inside and outside of the tube.

In the immersed interface method, the fluid velocity and pressure are computed on a fixed, Eulerian grid, while a moving, Lagrangian frame of reference is used to track the location of the interface Γ over time. Our Eulerian grid is described by cylindrical, axisymmetric coordinates. That is, we use coordinates (r, θ, z), but assume that the domain and flow are invariant with θ. In particular, we assume that velocity in the θ direction is always zero. Therefore, the fluid domain reduces to two dimensions, with $\vec{x} = (r, z)$. Furthermore, we compute variable values for the right half of the domain, $r \geqslant 0$, and enforce symmetry for the left half, $r < 0$. We denote the computed right-half of the domain by $\hat{\Omega}$.

In the axisymmetric cylindrical equations that follow, the vector operator notation refers to the following standard definitions:

$$\nabla p = \left(\frac{\partial p}{\partial r}, \frac{1}{r}\frac{\partial p}{\partial \theta}, \frac{\partial p}{\partial z} \right)$$

$$\nabla \cdot (u, v, w) = \left(\frac{u}{r} + \frac{\partial u}{\partial r} \right) + \left(\frac{1}{r}\frac{\partial v}{\partial \theta} \right) + \frac{\partial w}{\partial z}$$

$$\nabla^2 u = \frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial u}{\partial r} \right) + \frac{1}{r^2}\frac{\partial^2 u}{\partial \theta^2} + \frac{\partial^2 u}{\partial z^2}.$$

Tube surface discretization

The axisymmetric assumption implies that we can represent surface Γ by its cross section at a fixed value of θ, say, $\theta = 0$. We will call the curve found at this cross section Γ as well, with the distinction between curve Γ and surface Γ made clear by context. Curve Γ is represented by markers that move independently of the fixed fluid grid, interpolated by periodic cubic splines. Each marker is labeled by a fixed arclength value s, where s is established as the distance between the marker and an

arbitrary origin location under zero source/sink equilibrium conditions. Then $\vec{X} = (R(s,t), Z(s,t))$ is the position at time t in (axisymmetric) cylindrical coordinates of the point on Γ whose arclength label is s. We assume that the configuration described by $\vec{X}$ at any time t determines the forces supported on Γ. The use of splines in the immersed interface method is detailed by Leveque and Li in [27].

Governing equations

In axisymmetric coordinates, the governing equations for Stokes flow are

$$\mu\left(\nabla^2 - \frac{1}{r^2}\right)u - \frac{\partial p}{\partial r} + F_1 = 0 \tag{3.1}$$

$$\mu\nabla^2 w - \frac{\partial p}{\partial z} + F_3 = 0 \tag{3.2}$$

$$\nabla \cdot \vec{v} = g. \tag{3.3}$$

Here velocity $\vec{v} = (u, w)$, p is the pressure, μ is the viscosity (assumed to be constant), and $\vec{F} = (F_1, F_3)$ is given by a surface integral over Γ,

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

for $i = 1, 3$, where $\vec{f} = (f_1, f_3)$ denotes the body force (singularly supported) in the r and z directions induced by Γ. Free space boundary conditions are assumed.

Applying the divergence operator to Eqs. (3.1) and (3.2), and substituting from Eq. (3.3), we can also write

$$\nabla^2(p - \mu g) = \nabla \cdot \vec{F}. \tag{3.4}$$

Definition of source and sink terms

Rather than enforcing incompressibility everywhere, Eq. (3.3) describes the source and sink that drive flow through the tube. For compact regions $\vec{x}_{source}$ and $\vec{x}_{sink}$, we

have

$$g(\vec{x},t) \geqslant 0 \text{ for } \vec{x} \in \vec{x}_{source},$$

$$g(\vec{x},t) \leqslant 0 \text{ for } \vec{x} \in \vec{x}_{sink},$$

$$g(\vec{x},t) = 0 \text{ otherwise.}$$

We let $\vec{x}_{source}$ be the cylindrical region centered at a point (r_0, z_0) (with $r_0 = 0$ for symmetry about the z axis) such that $|r - r_0| \leqslant \kappa$ and $|z - z_0| \leqslant \kappa$ (and $0 \leqslant \theta < 2\pi$), for fixed radius parameter κ. The region $\vec{x}_{sink}$ is similarly defined.

For $(r, z) \in \vec{x}_{source}$ we define

$$g(t, r, z) = A(t)h_1(r)h_2(z) \tag{3.5}$$

with the following properties:

$$\int_0^{2\pi} \int_0^{\kappa} h_1(r) \cdot r \, dr \, d\theta = 1, \tag{3.6}$$

$$\int_{-\kappa}^{\kappa} h_2(z) \, dz = 1. \tag{3.7}$$

We use cosine functions scaled to meet these criteria:

$$h_1(r) = \beta(\kappa) \left(1 + \cos\left(\frac{\pi(r - r_0)}{\kappa}\right)\right) \tag{3.8}$$

$$h_2(z) = \frac{1}{2\kappa} \left(1 + \cos\left(\frac{\pi(z - z_0)}{\kappa}\right)\right), \tag{3.9}$$

where constant $\beta(\kappa)$ is chosen such that Eq. (3.6) is satisfied. As an example of the magnitude function $A(t)$, in section 3.4.2 we use

$$A(t) = 0.0025(1.0 + \sin(\omega t - 0.5\pi))$$

for various frequency values ω. We define $g(\vec{x}_{sink}, t)$ similarly but with opposite magnitude, so there is no net accumulation of fluid in the tube, and for other values of $\vec{x}$ we have $g(\vec{x}, t) = 0$.

Domain boundary conditions

Free space boundary conditions are assumed for the Stokes flow. To discretize the model equations, we impose the following conditions on the half-plane computational domain $\hat{\Omega}$.

Across the z axis, homogeneous Neumann boundary conditions are implemented for pressure and for velocity in the z direction,

$$\frac{\partial p}{\partial r} = 0 \text{ and } \frac{\partial w}{\partial r} = 0 \text{ at } r = 0, \tag{3.10}$$

and for velocity in the r direction we have

$$u = 0 \text{ at } r = 0. \tag{3.11}$$

The other three sides of the domain have Dirichlet boundary conditions for p, u, and w, with values obtained from boundary integrals (Eqs. (3.19),(3.20)) below..

3.2 Problem decomposition and solution methods

We decompose the pressure and velocity into parts induced by the tube boundary and induced by the source and sink,

$$p = p_{bd} + p_s, \tag{3.12}$$

$$\vec{v} = \vec{v}_{bd} + \vec{v}_s, \tag{3.13}$$

so that each may be treated with an appropriate method. The tube boundary creates a singularly-supported force; this part of the solution is addressed by the immersed interface method. The smooth solution due to the source and sink is calculated with a boundary integral method.

3.2.1 Solution induced by the tube boundary

The pressure and velocity due to the immersed tube boundary, p_{bd} and $\vec{v}_{bd}$, satisfy the equations of Stokes flow, Eqs. (3.1),(3.2), and (3.3), in the absence of source and

sink terms, i.e., with $g(\vec{x}, t) \equiv 0$. That is,

$$\mu \left(\nabla^2 - \frac{1}{r^2} \right) u_{bd} - \frac{\partial p_{bd}}{\partial r} + F_1 = 0, \tag{3.14}$$

$$\mu \nabla^2 w_{bd} - \frac{\partial p_{bd}}{\partial z} + F_3 = 0, \text{ and} \tag{3.15}$$

$$\nabla \cdot \vec{v}_{bd} = 0. \tag{3.16}$$

Forcing terms that are singularly supported on the tube boundary make this system suitable for the immersed interface method.

Immersed interface method

For the immersed interface method the Stokes equations are recast as a sequence of three Poisson problems. Each is solved by a standard finite difference method, with the discretized right-hand side of each problem determined by jump conditions.

Taking the divergence of Eqs. (3.14), (3.15) and applying Eq. (3.16), p_{bd} satisfies

$$\nabla^2 p_{bd} = \nabla \cdot \vec{F}, \tag{3.17}$$

which is discretized using a standard five-point stencil,

$$\frac{1}{h^2} \left(p_{bd_{i+1,j}} + p_{bd_{i-1,j}} + p_{bd_{i,j+1}} + p_{bd_{i,j-1}} - 4p_{bd_{i,j}} \right) + \frac{1}{2h} \left(p_{bd_{i+1,j}} - p_{bd_{i-1,j}} \right) = b_{i,j}. \tag{3.18}$$

Away from the immersed boundary, right-hand side $b_{i,j} = 0$, and when the stencil includes points from both sides of the boundary, the immersed interface method prescribes values of $b_{i,j}$ based on jump conditions, as detailed in [26] and [27]. We derive jump conditions for the axisymmetric setting in section 3.3; see Eqs. (3.38) and (3.39) below.

With values of p_{bd} given, Eqs. (3.14) and (3.15) are also Poisson problems with singularly supported right-hand side (F_1 and F_3, respectively), subject to the same

finite difference approach incorporating jump conditions. The jump conditions for velocity are given in Eqs. (3.47) and (3.49).

Boundary conditions

Boundary conditions are needed to solve the Poisson problems given above.

Eq. (3.17) has a boundary integral solution, given by

$$p_{bd} = \iint_{\Gamma} -\frac{1}{4\pi} \vec{f} \cdot \nabla \left(\frac{1}{r} \right) \, dS, \tag{3.19}$$

where Γ denotes the (two-dimensional) immersed surface and dS is the surface element.

We make use of the boundary integrals in 3-D Cartesian coordinates to compute solutions for Eqs. (3.14) and (3.15). Relating the r component of velocity in cylindrical coordinates to the x and y components of velocity in 3-D Cartesian coordinates, we have

$$u^{\text{cylindrical}} = u^{\text{Cartesian}} \cos(\theta) + v^{\text{Cartesian}} \sin(\theta).$$

When $\theta = 0$, the statement reduces to

$$u^{\text{cylindrical}} = u^{\text{Cartesian}}.$$

Therefore, we can make use of the boundary integral solution for Cartesian coordinates to obtain the the boundary value of u, as well as w (which is unchanged in Cartesian coordinates). Using the notation $\vec{x}' = (x_1', x_2', x_3')$, we have

$$\vec{v}_{bd}^{\text{Cartesian}} = \iint_{\Gamma} \frac{1}{8\pi\mu} \left(\frac{\delta_{ij}}{r} + \frac{x_i' x_j'}{r^3} \right) f_j \, dS(x'). \tag{3.20}$$

3.2.2 Solution induced by the source and sink

The part of the pressure solution due to the source and sink is determined by comparing Eqs. (3.17) and (3.4):

$$p_{bd} = p - \mu g, \tag{3.21}$$

so

$$p_s = \mu g.$$

For velocity, our decomposition produces

$$\left(\nabla^2 - \frac{1}{r^2}\right) u_s = \frac{1}{\mu}\frac{\partial p_s}{\partial r} = \frac{\partial g}{\partial r}, \tag{3.22}$$

$$\nabla^2 w_s = \frac{1}{\mu}\frac{\partial p_s}{\partial z} = \frac{\partial g}{\partial z}. \tag{3.23}$$

These equations have smooth solutions which would typically be computed with a finite difference approach. However, the $\frac{1}{r^2}$ term in Eq. (3.22) grows large as $r \to 0$. Furthermore, when g's radius parameter κ is on the order of grid spacing h, $\frac{\partial p_s}{\partial r}$ is also large near $r = 0$. This is because the maximum of g is found at $r = 0$ and $g(t, r, z) \approx 0$ for $r = \kappa$. These two large terms generate large numerical errors in a finite difference scheme; instead we use a boundary integral method to evaluate $\vec{v}_s$ on the immersed interface. We don't have the boundary integral solution for u_s directly from Eq. (3.22), but we can refer to the solution for Cartesian coordinates as we did for Eq. (3.14) above.

The relevant equation in 3-D Cartesian coordinates is

$$\nabla^2 \vec{v}_s = \nabla g, \tag{3.24}$$

which has boundary integral solution

$$\vec{v}_s^{\,\text{Cartesian}} = \iiint_{\Omega} \frac{1}{4\pi}\frac{\nabla g(x')}{r}\, dV(x'), \tag{3.25}$$

where Ω is the (three-dimensional) computational domain and dV denotes the volume element. To evaluate Eq. (3.25) we need the source/sink function g in Cartesian coordinates; in Eq. (3.5), g is given in axisymmetric coordinates. We simply use a Cartesian grid with x axis and z axis identical to the r and z axes of our axisymmetric cylindrical fluid grid.

3.2.3 Overview of hybrid computational approach

We briefly summarize the approach detailed in the previous sections. At time t^n, approximate values are known for marker locations $\vec{X}(s, t^n)$, establishing the configuration of tube surface Γ. To advance the system to time t^{n+1}:

1. *Dirichlet boundary conditions:* Compute p_{bd}, u_{bd} and w_{bd} (values due to tube boundary force) on the boundary of domain Ω using boundary integrals (3.19) and the first and third components of (3.20).

2. *Advance $\vec{v}_{bd}$ and p_{bd} via immersed interface method:*

   (a) Using a cubic spline representation of the boundary force $\vec{f}$ along Γ, calculate jump condition corrections from Eqs. (3.38) and (3.39) for pressure, and Eqs. (3.47) and (3.49) for velocity.

   (b) Incorporate jump conditions into finite difference approximations for p_{bd}^{n+1} and $\vec{v}_{bd}^{n+1}$ on the fluid grid.

3. *Find velocity due to source and sink via boundary integral:* Use Eq. (3.25) to find $\vec{v}_s$ (velocity field due to source and sink) on the tube boundary markers.

4. *Advance location of tube surface Γ:*

   (a) Obtain $\vec{v}_{bd}^{n+1}$ on boundary marker locations, using a second order interpolation scheme that incorporates the jump in velocity across the immersed boundary (see [27]).

(b) Advance tube boundary markers, i.e., find $\vec{X}(s, t^{n+1})$, according to the total velocity field, $\vec{v} = \vec{v}_{bd} + \vec{v}_s$:

$$\vec{X}(s, t^{n+1}) = \vec{X}(s, t^n) + \Delta t \left(\frac{3}{2}\vec{v}^n - \frac{1}{2}\vec{v}^{n-1} \right). \tag{3.26}$$

3.3 Derivation of jump conditions for axisymmetric coordinate system

Next we derive the jump conditions for pressure and velocity for Stokes flow in the axisymmetric cylindrical coordinate system. Note that the divergence-free flow described in this section corresponds to the part of the solution due to the immersed boundary discussed in section 3.2.1. For ease of notation, however, rather than using the variables p_{bd} and $\vec{v}_{bd}$ in this section, we use generic p, u, v, w without the subscript *bd*.

We consider momentarily the governing equations in fully three-dimensional cylindrical coordinates, then apply the assumption of invariance with θ to derive jump conditions for the axisymmetric setting.

The equations governing Stokes flow in 3-D cylindrical coordinates are

$$\mu \left(\nabla^2 - \frac{1}{r^2} \right) u - \frac{2\mu}{r^2}\frac{\partial v}{\partial \theta} - \frac{\partial p}{\partial r} + F_1 = 0 \tag{3.27}$$

$$\mu \left(\nabla^2 - \frac{1}{r^2} \right) v + \frac{2\mu}{r^2}\frac{\partial u}{\partial \theta} - \frac{1}{r}\frac{\partial p}{\partial \theta} + F_2 = 0 \tag{3.28}$$

$$\mu \nabla^2 w - \frac{\partial p}{\partial z} + F_3 = 0 \tag{3.29}$$

$$\nabla \cdot \vec{v} = 0 \tag{3.30}$$

where $\vec{x} = (r, \theta, z)$, $\vec{v} = (u, v, w)$, and $\vec{F} = (F_1, F_2, F_3)$, with

$$F_i = \iint_\Gamma f_i(s, \theta, t)\delta(\vec{x} - \vec{X}(s, \theta, t))dS \tag{3.31}$$

for $i = 1, 2, 3$. Taking the divergence of Eqs. (3.27), (3.28), (3.29) yields a Poisson equation for pressure,

$$\nabla^2 p = \nabla \cdot \vec{F}. \tag{3.32}$$

Let $\psi(r, \theta, z)$ be an arbitrary twice continuously differentiable test function. Multiplying ψ on the right hand side of Eq. (3.32), and applying Green's theorem, we have

$$\iiint_{\Omega} (\nabla \cdot \vec{F})\psi(r,\theta,z)dV = \iiint_{\Omega} \left(\iint_{\Gamma} (\nabla \cdot \vec{f}(s,\theta,t)\delta(\vec{x} - \vec{X}(s,\theta,t))dS \right) \psi(r,\theta,z)dV \tag{3.33}$$

$$= -\iint_{\Gamma} \left(f_1 \frac{\partial \psi}{\partial r} + \frac{f_2}{r}\frac{\partial \psi}{\partial \theta} + f_3 \frac{\partial \psi}{\partial z} \right) dS. \tag{3.34}$$

By using (s, θ) coordinates to parameterize surface Γ, the surface element is $dS = r\, ds\, d\theta$.

Let Γ^+ be a region that encloses the boundary surface Γ, and let the distance between Γ^+ and Γ shrink to zero. As $\Gamma^+ \to \Gamma$, multiplying $\psi(r, \theta, z)$ on the left hand side of Eq. (3.32) and applying Green's theorem again, we have

$$\iiint_{\Gamma^+} (\nabla^2 p)\psi \, dV \to 2\pi \left(\int_{\Gamma} \left[\frac{\partial p}{\partial n} \right] \psi \, r \, ds - \int_{\Gamma} [p] \psi_n r \, ds \right). \tag{3.35}$$

Here n is the outward normal direction and ds is the arclength element.

Now we make explicit our axisymmetric assumptions, that ψ and f are independent of θ. Then Eq. (3.34) can finally be written as

$$\iiint_{\Omega} (\nabla \cdot \vec{F})\psi(x,y)dV = -2\pi \int_{\Gamma} \left(f_1 \frac{\partial \psi}{\partial r} + f_3 \frac{\partial \psi}{\partial z} \right) r \, ds. \tag{3.36}$$

Here Γ denotes the boundary of the cross section of the surface for fixed θ.

Now we split the force due to the boundary into components tangent to and normal to the boundary. Because $f_1\frac{\partial\psi}{\partial r} + f_3\frac{\partial\psi}{\partial z} = f_n\frac{\partial\psi}{\partial r} + f_s\frac{\partial\psi}{\partial s}$, the right hand side of Eq. (3.36) can be written as

$$\int_\Gamma \left(f_1\frac{\partial\psi}{\partial r} + f_3\frac{\partial\psi}{\partial z}\right) r\,ds = \int_\Gamma \left(f_n\frac{\partial\psi}{\partial n} + f_s\frac{\partial\psi}{\partial s}\right) r\,ds \quad = \int_\Gamma \left(f_n r\frac{\partial\psi}{\partial n} - \frac{\partial(f_s r)}{\partial s}\psi\right) ds. \tag{3.37}$$

The second step of Eq. (3.37) is obtained by integration by parts on s. Finally, using the fact ψ is arbitrary and combining Eq. (3.36) and Eq. (3.37), the jump conditions for pressure are

$$[p] = f_n \tag{3.38}$$

$$\left[\frac{\partial p}{\partial n}\right] = \frac{1}{r}\frac{\partial(f_s r)}{\partial s}. \tag{3.39}$$

To obtain the jump condition for velocity, we can multiply test function ψ on Eq. (3.27), and integrate:

$$\iiint_{\Gamma^+} \mu(\nabla^2 u)\psi dV - \iiint_{\Gamma^+} \mu\left(\frac{u}{r^2}\right)\psi\,dV - \iiint_{\Gamma^+} \frac{\partial p}{\partial r}\psi\,dV = -\iint_\Gamma f_1\psi\,dS \tag{3.40}$$

Next we examine each term on the left-hand side of Eq. (3.40) as $\Gamma^+ \to \Gamma$. For the first term of Eq. (3.40) we have

$$\iiint_{\Gamma^+} \mu(\nabla^2 u)\psi\,dV \to \iint_\Gamma \left[\mu\frac{\partial u}{\partial n}\right]\psi\,dS. \tag{3.41}$$

The second term of Eq. (3.40) is

$$\iiint_{\Gamma^+} \mu\left(\frac{u}{r^2}\right)\psi\,dV = \iiint_{\Gamma^+} \mu\left(\frac{u}{r^2}\right)\psi\,r\,dr\,\mathrm{d}\theta\,\mathrm{dz} = \iiint_{\Gamma^+} \mu\left(\frac{u}{r}\right)\psi\,dr\,\mathrm{d}\theta\,\mathrm{dz}. \tag{3.42}$$

Now we enforce our axisymmetric assumptions and conclude that the second term vanishes as $\Gamma^+ \to \Gamma$: if $r \to 0$, then $u \to 0$ in the axisymmetric cylindrical coordinate

system. On the other hand, when r remains finite, $|\frac{u}{r}| \to |\frac{\partial u}{\partial r}|$, which remains bounded Therefore, as $\Gamma^+ \to \Gamma$, the second term of Eq. (3.40) vanishes, whether $r \to 0$ or not.

The third term of Eq. (3.40) is

$$\iiint_{\Gamma^+} \frac{\partial p}{\partial r}\psi dV = \iint_{\Gamma^+} \psi \left(\nabla \cdot [p, 0] - \frac{p}{r}\right) dV. \tag{3.43}$$

As $\Gamma^+ \to \Gamma$,

$$\iiint_{\Gamma^+} \psi \nabla \cdot [p, 0] dS \to \iint_{\Gamma} \psi [p] \cos(\alpha) dS \tag{3.44}$$

$$\iiint_{\Gamma^+} \psi \frac{p}{r} dS \to 0. \tag{3.45}$$

Here α is the angle between the normal and r direction. Therefore, we have the jump for u,

$$\left[\mu \frac{\partial u}{\partial n}\right] = [p] \cos\alpha - f_1 \tag{3.46}$$

$$= f_s \sin\alpha. \tag{3.47}$$

Similarly, we can derive the jump for w, which is

$$\left[\mu \frac{\partial w}{\partial n}\right] = [p] \sin\alpha - f_3 \tag{3.48}$$

$$= -f_s \cos\alpha. \tag{3.49}$$

3.4 Numerical results

Our method has second order spatial accuracy in the absence of source and sink, shown below in section 3.4.1, and the simulated flow is found to be approximately Poiseulle (section 4.3.3). We also study the behavior of the fluid field under oscillating inflow and outflow. Although complicated fluid behavior has previously been

observed for pumped Navier-Stokes flow, (e.g., [20]), in our Stokes flow study the amplitude of pressure and velocity fluctuations decrease as the forcing oscillations increase in frequency (section 3.4.2). We also investigate inflow/outflow oscillations of increasing frequency under controlled, steady pressure.

3.4.1 Convergence test

To demonstrate the second order spatial accuracy of our method, we consider a spherical elastic surface deformed into an ellipsoid shape. (This stationary example does not include a source and sink.) At the plane $\theta = 0$, the cross section of the ellipsoid is described parametrically as $(0.8\cos(\phi), 0.6\sin(\phi))$. The equilibrium shape for this surface cross section is a circle with radius 0.7, centered at origin. In this example, the fluid domain is $[-1.8, 1.8] \times [-1.8, 1.8]$, and the fluid field is computed for $N = 40$, 80, 160, and 320, with uniform grid spacing $h = 3.6/N$ in each case. The immersed boundary is discretized using 240 markers. We use the solution computed on a high-resolution 640×640 grid as the reference solution to estimate errors. Table 3.1 displays the convergence results, where the velocity columns average the results in the r and z directions.

Table 3.1: Convergence results for p and $\vec{v}$. Approximations exhibit second-order convergence.

	p		$\vec{v}$	
N	L_1 error/h^2	L_{inf} error/h^2	L_1 error/h^2	L_{inf} error/h^2
40	1.012e-2	8.932e-1	1.600e-2	1.362e-1
80	1.098e-2	6.470e-1	1.420e-2	1.058e-1
160	8.221e-3	9.818e-1	1.109e-2	1.045e-1
320	4.849e-3	1.405e0	9.415e-3	1.036e-1

3.4.2 Oscillating inflow

Here we simulate the motion of a compliant tube with an internal source and sink that introduce an oscillating pumping force. In this example, the tube wall is subject

to elastic tension and tether forces.

The fluid domain is $[-0.45, 0.45] \times [-3.6, 3.6]$, with the $r \geqslant 0$ half of the domain discretized on a 40×640 grid (and the $r < 0$ half reflected), so grid spacing $h = 0.01125$. The time step for the simulation is $h/20$.

Figure 3.2 shows the configuration of the tube. The source and sink are centered at $(r, z) = (0.0, 2.0)$ and $(r, z) = (0.0, -2.0)$, respectively, with radius parameter $\kappa = h = 0.0225$. At equilibrium, the top of the tube is the upper half of the circle with radius 0.2 centered at the source, described by

$$r^2 + (z - 2.0)^2 = 0.2^2.$$

Similarly, the bottom of the tube is the lower half of the circle with radius 0.2 centered at the sink:

$$r^2 + (z + 2.0)^2 = 0.2^2.$$

Each half circle is discretized by 8 markers. The left and right walls of the tube are straight lines at equilibrium, $r = 0.2$ and $r = -0.2$ between $z = -2.0$ and $z = 2.0$, with 36 markers on each side. Of the 88 markers, the positions of the 44 on the right half of the domain are computed, while the corresponding markers on the left half of the domain move according to symmetry. We use the notation $\vec{X}(s, t)$ to denote the position (r, z) of the marker associated with equilibrium arclength value s, at time t, and we use $\vec{X}(s_{opp}, t)$ to denote the position of the mirrored twin marker.

The boundary force along the tube surface is composed of two parts,

$$\vec{f} = \vec{f_e} + \vec{f_t}.$$

The elastic tension force is given by

$$\vec{f_e} = \frac{\partial}{\partial s} \left(T(s, t) \vec{\tau}(s, t) \right),$$

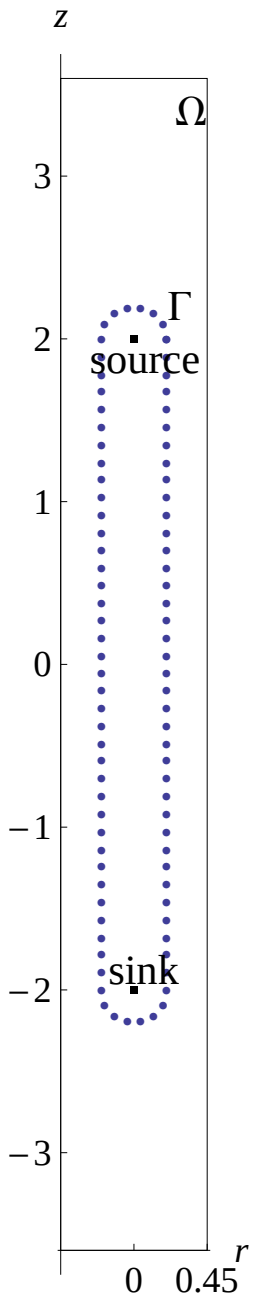

FIGURE 3.2: Markers delineate the boundary of an elastic tube.

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

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

(we use elasticity constant $T_0 = 0.2$) and $\vec{\tau}(s,t)$ is the tangent vector to the tube surface.

The tether force $\vec{f_t}$ is defined in two ways. Along the top and bottom semi-circular regions of Γ,

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

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

Along the side regions of the tube, the r and z components of the tether force are defined independently:

$$\vec{f_t} = \begin{bmatrix} f_{rt} \\ f_{zt} \end{bmatrix}, \tag{3.50}$$

where f_{zt} is the z component of the tether force defined above, and f_{rt} arises from a boundary point's elastic connection to its neighbors along the circumference of the tube. This width-restoring force is defined as

$$f_{rt} = m \left(\left| \vec{X}(s,t) - \vec{X}(s_{opp},t) \right| - L_T \right). \tag{3.51}$$

Here L_T is the equilibrium width of the tube, and m is the force constant. In this example, $L_T = 0.4$ and $m = 50$.

The fluctuating magnitude for the source is

$$A(t) = 0.0025(1.0 + \sin(\omega t - 0.5\pi)); \tag{3.52}$$

The magnitude for the sink is $-A(t)$. We examine the flow subject to varying fluctuation frequencies ω in section 3.4.2 below.

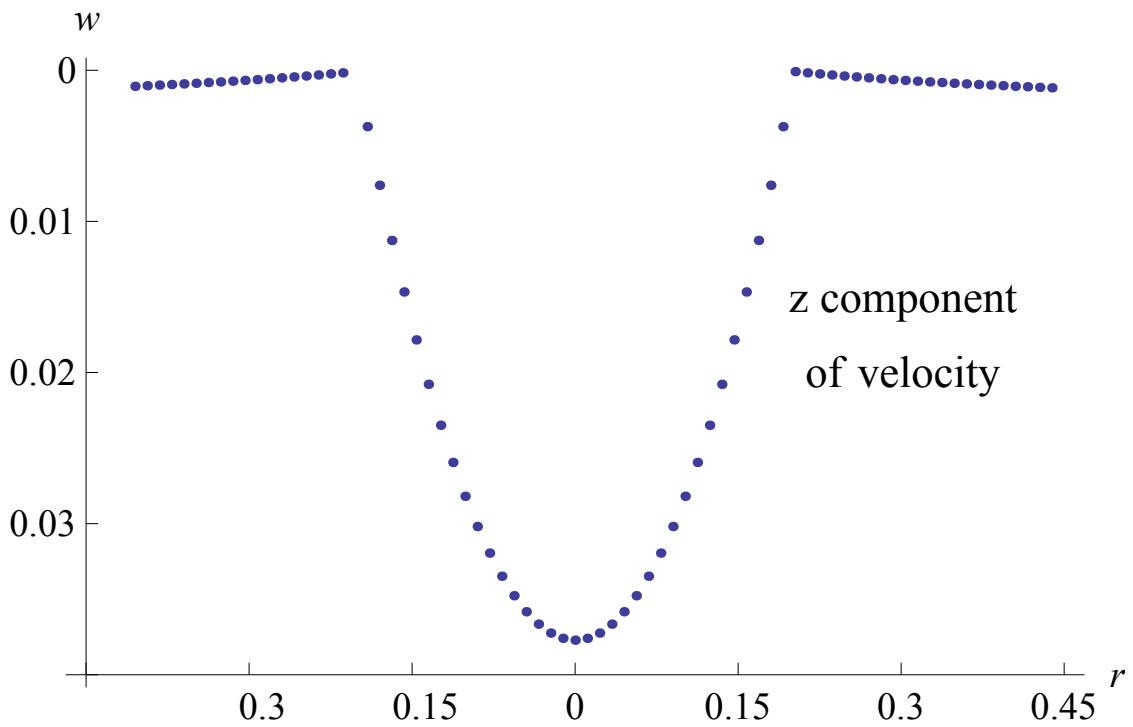

FIGURE 3.3: Velocity in the z direction at $z = 0.99$. Approximately parabolic profile is characteristic of Poiseuille flow.

Velocity and Shear Stress

In this example, we choose inflow/outflow oscillation frequency $\omega = 5$. After the system reaches equilibrium, we plot the z direction velocity and shear stress along the cross section $z = 0.99$ (results are qualitatively similar at other cross sections). From the parabolic shape of the z-velocity (Fig. 3.3), and the v-shaped shear stress (Fig. 3.4), we can see that the flow is approximately Poiseuille flow, as could be expected for laminar flow through a roughly circular pipe.

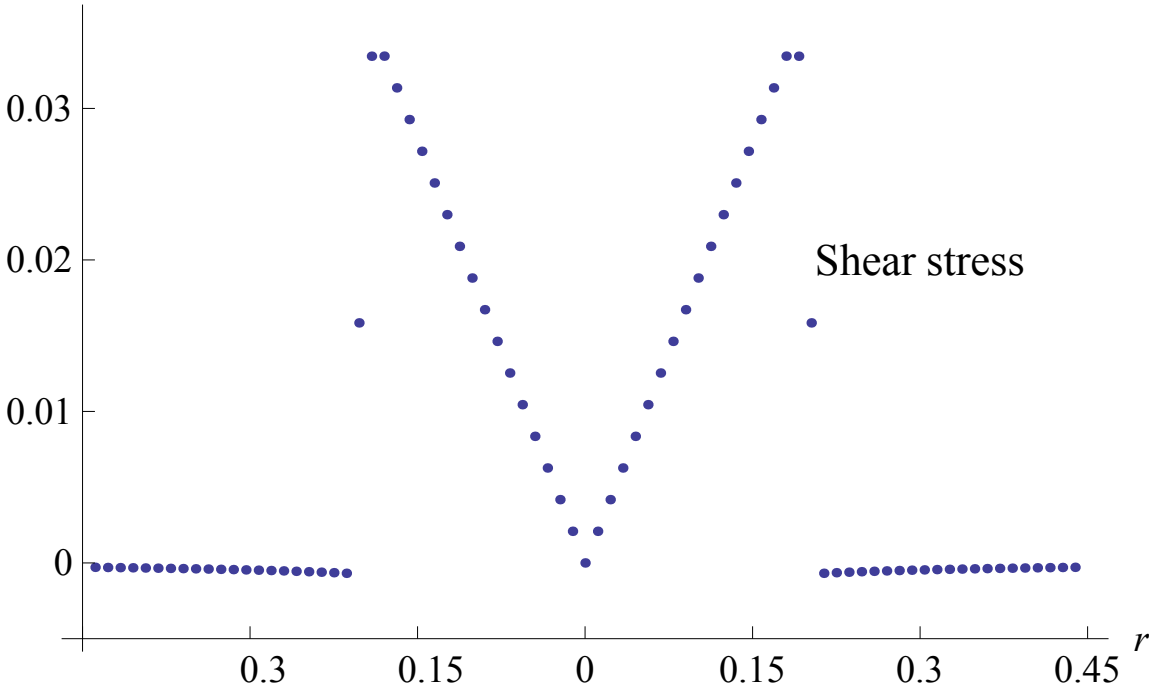

FIGURE 3.4: Shear stress at $z = 0.99$. V-shape is characteristic of Poiseuille flow.

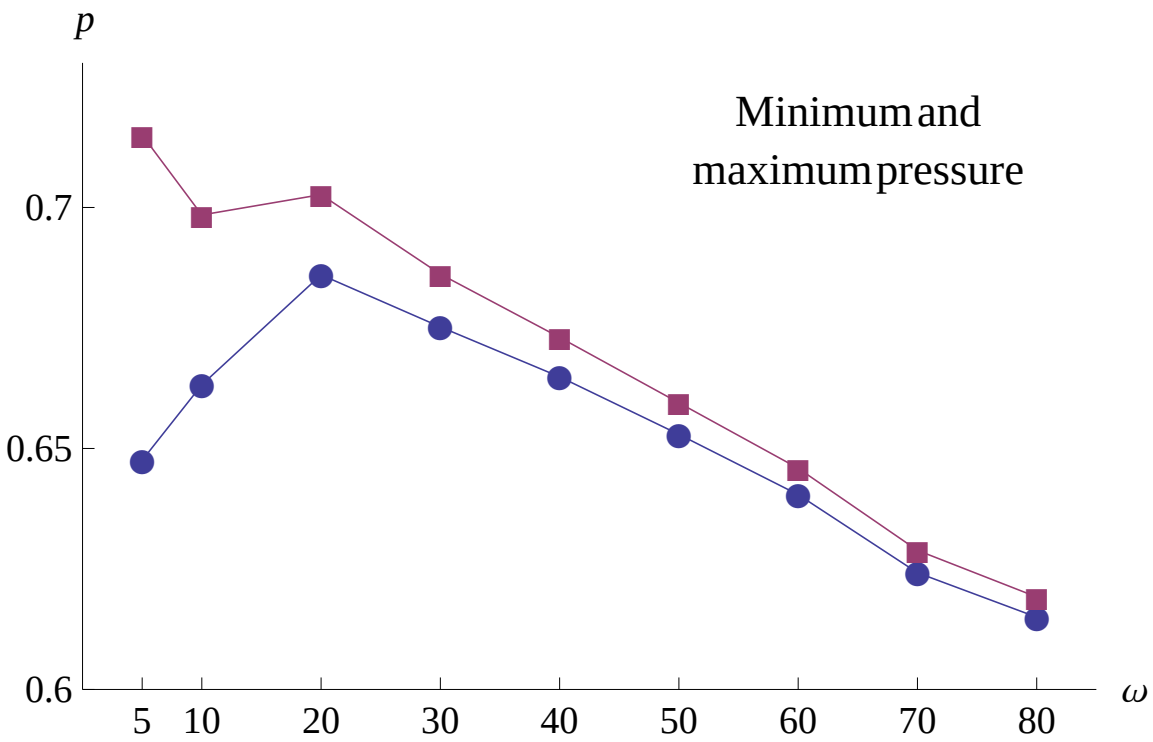

FIGURE 3.5: Minimum and maximum of pressure cycle at $(r, z) = (0, 1.8)$; pumping frequency $5 \leqslant \omega \leqslant 80$.

Frequencies and Amplitude

Other studies (e.g., [20]) have demonstrated pumped flow that changes nonlinearly with forcing function frequency, even reversing direction for certain frequency ranges. In the current, simple case of Stokes flow, however, we see pressure and flow fields that settle (fluctuation amplitudes decrease) as the pumping speed increases. The data in this set of experiments was collected at $z = 1.8$ and $z = 0.99$, but results are qualitatively similar at other locations.

Figure (3.5) shows the maximum and minimum pressure values recorded over an equilibrium temporal cycle at point $(r, z) = (0, 1.8)$, for increasing pumping frequency. For pumping frequencies $5 \leqslant \omega \leqslant 80$, the amplitude of the pressure cycle decreases monotonically.

Figure (3.6) shows the maximum and minimum flow values obtained by integrat-

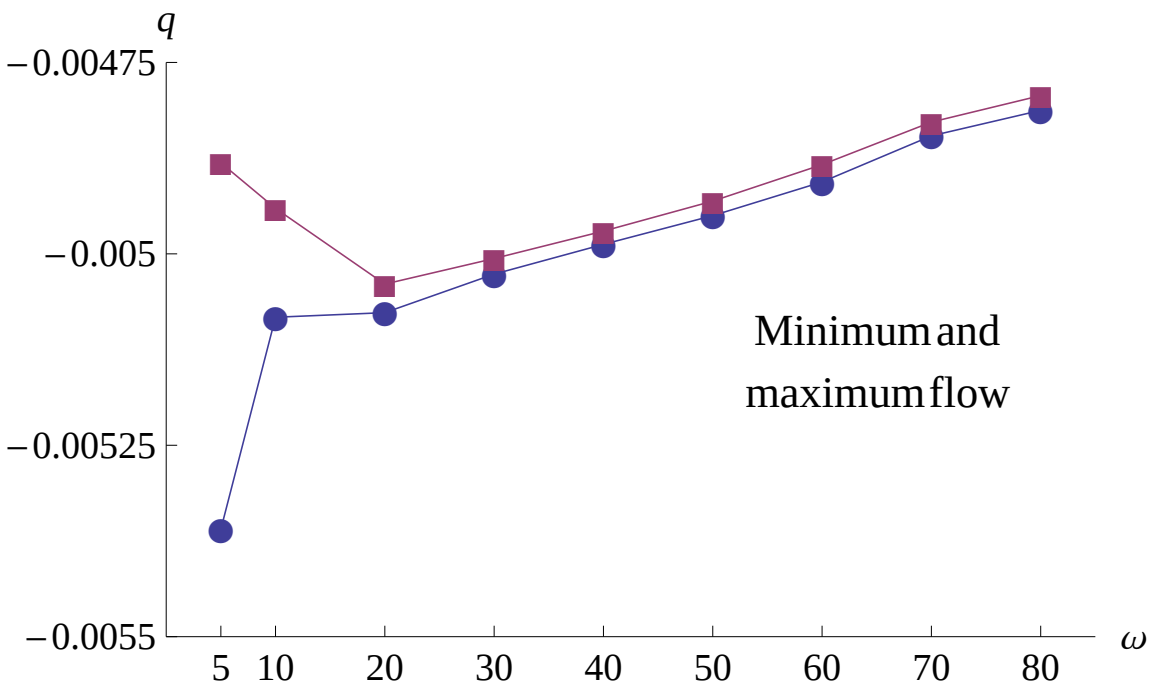

FIGURE 3.6: Minimum and maximum of flow across $z = 0.99$ for pumping frequency $5 \leqslant \omega \leqslant 80$.

ing the vertical velocity field at $z = 0.99$ throughout an equilibrium temporal cycle. For all pumping frequencies, fluid flows in the downward direction. For pumping frequencies $5 \leqslant \omega \leqslant 40$, the amplitude of the flow cycle decreases monotonically. For higher pumping frequencies, the amplitude of the flow cycle stays within 0.25% of the mean flow.

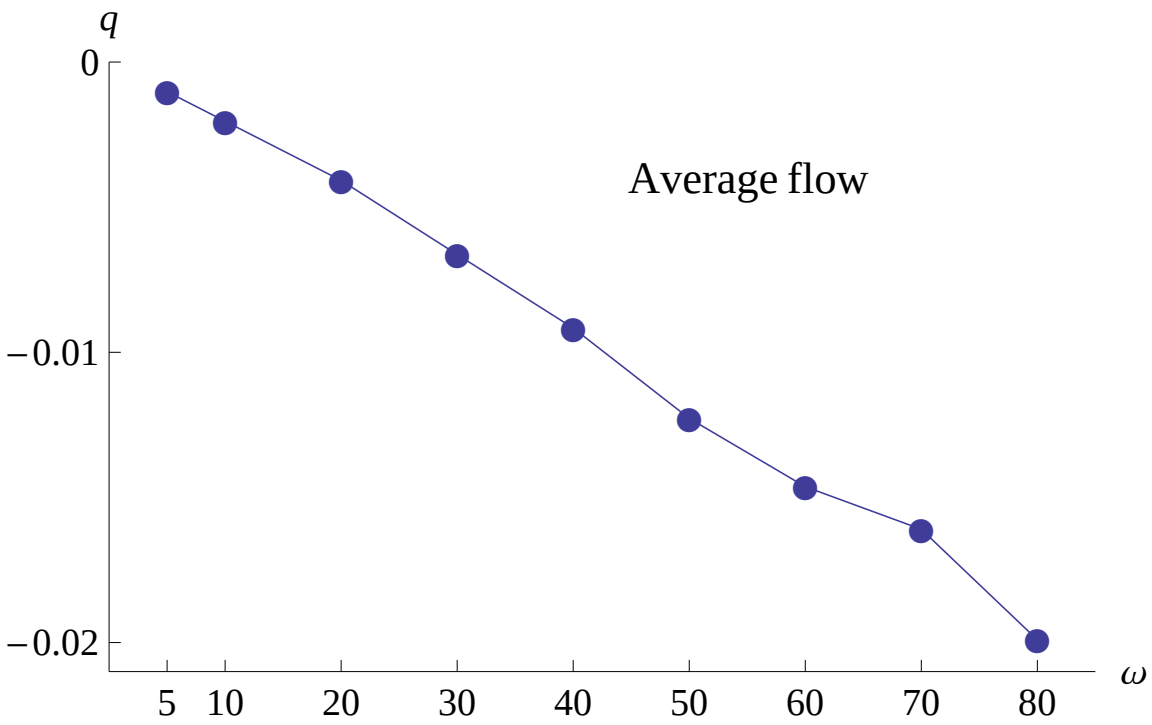

FIGURE 3.7: Under increasing pumping frequency and approximately steady pressure, the average flow at $z = 0.99$ increases in absolute value.

Finally, for a more biologically or physically relevant experiment to investigate the effect of pressure fluctuation frequency on flow rate, we varied the frequency of pumping (and therefore the frequency of pressure oscillations) while holding the pressure amplitude approximately fixed ($\approx 1.378 \times 10^{-2}$). To obtain steady pressure amplitude near the source, we adjust the magnitude of inflow/outflow function $A(t)$ (see Eq. (4.67)) as we vary ω. Figure (3.7) shows that, under increasing pumping frequency and approximately steady pressure, the average flow at $z = 0.99$ increases in absolute value.

4

Simulating Driven Navier-Stokes Flow in an Elastic Tube

4.1 Governing equations

In this study, we simulate driven Navier-Stokes flow in a three-dimensional elastic tube. The flow is driven by an oscillating internal source located at the top of the tube.

Governing equations

In axisymmetric coordinates, the governing equations for Navier-Stokes flow are

$$\frac{\partial u}{\partial t} + \vec{v} \cdot \nabla u = -\frac{\partial p}{\partial r} + \mu \left(\nabla^2 - \frac{1}{r^2} \right) u + F_1 \tag{4.1}$$

$$\frac{\partial w}{\partial t} + \vec{v} \cdot \nabla w = -\frac{\partial p}{\partial z} + \mu \nabla^2 w + F_3 \tag{4.2}$$

$$\nabla \cdot \vec{v} = g. \tag{4.3}$$

Here velocity $\vec{v} = (u, w)$, p is the pressure, μ is the viscosity (assumed to be constant), and $\vec{F} = (F_1, F_3)$ is given by a surface integral over Γ,

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

for $i = 1, 3$, where $\vec{f} = (f_1, f_3)$ denotes the body force (singularly supported) in the r and z directions induced by Γ.

Definition of source, sink and external source/sink

Rather than enforcing incompressibility everywhere, Eq. (4.3) describes the source and sink that drive flow through the tube.

We let $\vec{x}_{source}$ be the cylindrical region centered at a point (r_0, z_0) (with $r_0 = 0$ for symmetry about the z axis) such that $|r - r_0| \leqslant \kappa$ and $|z - z_0| \leqslant \kappa$ (and $0 \leqslant \theta < 2\pi$), for fixed radius parameter κ. The region $\vec{x}_{sink}$ is similarly defined.

For $(r, z) \in \vec{x}_{source}$ we define

$$g_1(t, r, z) = A_1(t)\delta_1(r, z) \tag{4.4}$$

$$\delta_1(r, z) = h_1(r)k_1(z) \tag{4.5}$$

For $(r, z) \in \vec{x}_{sink}$ we define

$$g_2(t, r, z) = A_2(t)\delta_2(r, z) \tag{4.6}$$

$$\delta_2(r, z) = h_2(r)k_2(z) \tag{4.7}$$

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

$$\int_0^{2\pi} \int_0^{\kappa} h_i(r) \cdot r \, dr \, d\theta = 1, \tag{4.8}$$

$$\int_{-\kappa}^{\kappa} k_i(z) \, dz = 1. \tag{4.9}$$

We use cosine functions scaled to meet these criteria:

$$h_i(r) = \beta(\kappa)\left(1 + \cos\left(\frac{\pi(r - r_i)}{\kappa}\right)\right) \tag{4.10}$$

$$k_i(z) = \frac{1}{2\kappa}\left(1 + \cos\left(\frac{\pi(z - z_i)}{\kappa}\right)\right), \tag{4.11}$$

where constant $\beta(\kappa)$ is chosen such that Eq. (4.8) is satisfied.

To conserve the fluid volume in the whole fluid domain, we set an external source/sink outside of the tube.(e.g.[1]) The $A_3(t)$ satisfies

$$A_3(t) = -A_1(t) - A_2(t) \tag{4.12}$$

Similarly as source and sink,

$$g_3(t, r, z) = A_3(t)\delta_3(r, z) \tag{4.13}$$

$$\delta_3(r, z) = h_3(r)k_3(z) \tag{4.14}$$

with the following property:

$$\int_0^{2\pi}\int_{-\kappa_1}^{\kappa_1} h_3(r) \cdot r \, dr \, d\theta = 1 \tag{4.15}$$

Here κ_1 is the radius of the external source/sink.

The magnitude function $A_1(t)$ is prescribed in current model. As an example of the magnitude function $A_1(t)$, in the numerical result section, we use

$$A_1(t) = 0.005(1.0 + \sin(\omega t - 0.5\pi))$$

for various frequency values ω.

$A_2(t)$ is obtained by the Hagen-Poiseuille equation (e.g.[31, 1])

$$A_2(t) = \frac{p_2 - \bar{p}_2}{R_2} \tag{4.16}$$

Here p_2 is the external pressure at the sink, R_2 is the resistance at the sink. In this study, p_2 and R_2 are both constants. $\bar{p}_2$ is the weighted average pressure over the sink region, scaled relative to the pressure over the external source and sink.

$$\bar{p}_2 = \int_\Omega p(r,z,t)(\delta_2(r,z) - \delta_3(r,z))\, dV \tag{4.17}$$

Domain boundary conditions

Homogeneous Dirichlet boundary conditions are assumed for the Navier-Stokes flow. To discretize the model equations, we impose the following conditions on the half-plane computational domain $\hat{\Omega}$.

Across the z axis, homogeneous Neumann boundary conditions are implemented for velocity in the z direction,

$$\frac{\partial w}{\partial r} = 0 \text{ at } r = 0, \tag{4.18}$$

and for velocity in the r direction we have

$$u = 0 \text{ at } r = 0. \tag{4.19}$$

The other three sides of the domain have homogeneous Dirichlet boundary conditions for u and w.

4.2 Computational methods

We follow the velocity decomposition approach in [5]. We decompose the total pressure and velocity into the regular part and the Stokes part.

$$\vec{v} = \vec{v}_{reg} + \vec{v}_s \tag{4.20}$$

$$p = p_{reg} + p_s \tag{4.21}$$

Here reg denotes the regular part and s denotes the Stokes part.

4.2.1 Stokes part

The Stokes part satisfies the Stokes equation with the boundary force and divergence free condition:

$$\mu\left(\nabla^2 - \frac{1}{r^2}\right) u_s - \frac{\partial p_s}{\partial r} + F_1 = 0 \tag{4.22}$$

$$\mu\nabla^2 w_s - \frac{\partial p_s}{\partial z} + F_3 = 0 \tag{4.23}$$

$$\nabla \cdot \vec{v} = 0. \tag{4.24}$$

The jump conditions for $\vec{v}_s$ and p_s are the following [28]:

$$[p_s] = f_n \tag{4.25}$$

$$\left[\frac{\partial p_s}{\partial n}\right] = \frac{1}{r}\frac{\partial (f_\tau r)}{\partial s}. \tag{4.26}$$

$$\left[\mu\frac{\partial u_s}{\partial n}\right] = f_\tau \sin\alpha. \tag{4.27}$$

$$\left[\mu\frac{\partial w_s}{\partial n}\right] = -f_\tau \cos\alpha. \tag{4.28}$$

Here f_n and f_τ are the normal and tangential boundary force density, respectively. α is the angle between the normal and r direction. Homogeneous Neumann boundary conditions are implemented for p_s, w_s.

$$\frac{\partial p_s}{\partial r} = 0 \text{ and } \frac{\partial w_s}{\partial r} = 0 \text{ at } r = 0, \tag{4.29}$$

and for velocity in the r direction we have

$$u_s = 0 \text{ at } r = 0. \tag{4.30}$$

The other three sides of the domain have homogeneous Dirichlet boundary conditions for p_s, u_s and w_s.

4.2.2 Regular part

Taking the difference of Eq. (4.1,4.2,4.3) and Eq. (3.1,3.2,3.3), one obtains the equation for the regular part of the velocity and pressure

$$\frac{\partial u_{reg}}{\partial t} + \vec{v} \cdot \nabla u_{reg} = -\frac{\partial p_{reg}}{\partial r} + \mu \left(\nabla^2 - \frac{1}{r^2} \right) u_{reg} + F_{b1} \tag{4.31}$$

$$\frac{\partial w_{reg}}{\partial t} + \vec{v} \cdot \nabla w_{reg} = -\frac{\partial p_{reg}}{\partial z} + \mu \nabla^2 w_{reg} + F_{b3} \tag{4.32}$$

$$\nabla \cdot \vec{v_{reg}} = g \tag{4.33}$$

F_{b1} and F_{b3} are the body force from the material derivative of Stokes velocity on r and z direction, respectively.

$$F_{b1} = -\frac{\partial u_s}{\partial t} - \vec{v} \cdot \nabla u_s \tag{4.34}$$

$$F_{b3} = -\frac{\partial w_s}{\partial t} - \vec{v} \cdot \nabla w_s \tag{4.35}$$

To avoid the discretization of $\nabla \vec{v}_{reg}$, we solve Eq. (4.31,4.32) with the semi-Lagrangian or Courant, Isaacson, Rees(CIR) method(e.g.[40, 7, 5]). In the semi-Lagrangian discretization, we incorporate advection term into material derivatives. Then Eq. (4.31,4.32) become

$$\frac{du_{reg}}{dt} = -\frac{\partial p_{reg}}{\partial r} + \mu \left(\nabla^2 - \frac{1}{r^2} \right) u_{reg} + F_{b1} \tag{4.36}$$

$$\frac{dw_{reg}}{dt} = -\frac{\partial p_{reg}}{\partial z} + \mu \nabla^2 w_{reg} + F_{b3} \tag{4.37}$$

Where

$$F_{b1} = -\frac{du_s}{dt} \tag{4.38}$$

$$F_{b3} = -\frac{dw_s}{dt} \tag{4.39}$$

To discretize Eq. (4.36,4.37), we use the second order backward difference formula, (e.g.[40, 7, 5]) the discretized equations are

$$\frac{3u_{reg}^{n+1} - 4\tilde{u}_{reg}^{n} + \tilde{u}_{reg}^{n-1}}{2\Delta t} = -\frac{\partial p_{reg}^{n+1}}{\partial r} + \mu\left(\nabla^2 - \frac{1}{r^2}\right)u_{reg}^{n+1} + F_{b1}^{n+1} \tag{4.40}$$

$$\frac{3w_{reg}^{n+1} - 4\tilde{w}_{reg}^{n} + \tilde{w}_{reg}^{n-1}}{2\Delta t} = -\frac{\partial p_{reg}^{n+1}}{\partial z} + \mu\nabla^2 w_{reg}^{n+1} + F_{b3}^{n+1} \tag{4.41}$$

In Eq. (4.40), $\tilde{u}_{reg}^{n}$, $\tilde{u}_{reg}^{n-1}$ denotes the r-velocity at upstream position $\mathbf{x^n}$ and $\mathbf{x^{n-1}}$, respectively. The notation in Eq. (4.41) has the same meaning for z-velocity.

We use the second order pressure-free projection method [22] to solve for Eq. (4.40,4.41). At time t_{n+1}, we define $\vec{v}_*$ and ϕ^{n+1} by

$$\vec{v}_{reg}^{n+1} = \vec{v}_* - \Delta t\nabla\phi^{n+1} \tag{4.42}$$

$\vec{v}_*$ is the intermediate velocity field and can be solved from the following equation

$$\frac{3u_* - 4\tilde{u}_{reg}^{n} + \tilde{u}_{reg}^{n-1}}{2\Delta t} = \mu\left(\nabla^2 - \frac{1}{r^2}\right)u_* + F_{b1}^{n+1} \tag{4.43}$$

$$\frac{3w_* - 4\tilde{w}_{reg}^{n} + \tilde{w}_{reg}^{n-1}}{2\Delta t} = \mu\nabla^2 w_* + F_{b3}^{n+1} \tag{4.44}$$

Grouping the $\vec{v}_*$ terms together for Eq. (4.43,4.44), we have the Helmholtz type equation for $\vec{v}_*$ and can solve the equation by using the fast fourier transform.

$$(\frac{3}{2\Delta t} - \mu\left(\nabla^2 - \frac{1}{r^2}\right))u_* = \frac{4\tilde{u}_{reg}^{n} - \tilde{u}_{reg}^{n-1}}{2\Delta t} + F_{b1}^{n+1} \tag{4.45}$$

$$(\frac{3}{2\Delta t} - \mu\nabla^2)w_* = \frac{4\tilde{w}_{reg}^{n} - \tilde{w}_{reg}^{n-1}}{2\Delta t} + F_{b3}^{n+1} \tag{4.46}$$

Due to the boundary condition we have chosen for $\vec{v}$ and $\vec{v}_s$, the boundary condition of $\vec{v}_{reg}$ has to be the same as the boundary condition for $\vec{v}$ or $\vec{v}_s$. Therefore, to attain such boundary condition for $\vec{v}_{reg}$, we choose the boundary condition for $\vec{v}_*$ to be

(e.g.[22, 10])

$$\vec{v}_* \cdot n = 0 \tag{4.47}$$

$$\frac{\partial w_*}{\partial r} = 0 \text{ at } r = 0 \tag{4.48}$$

$$\vec{v}_* \cdot t = \Delta t \nabla \phi^n \cdot t \text{ at the other three sides} \tag{4.49}$$

From Eq. (4.43,4.44,4.42), one can update regular pressure at t^{n+1} as

$$p_{reg}^{n+1} = \frac{3}{2}\phi^{n+1} - \mu \Delta t \nabla^2 \phi^{n+1} \tag{4.50}$$

Then ϕ^{n+1} has to be solved to update pressure. Taking the divergence on Eq. (4.42), we have

$$\Delta t \nabla^2 \phi^{n+1} = \nabla \cdot \vec{v}_* - g^{n+1} \tag{4.51}$$

To solve for ϕ^{n+1}, we decompose it into three parts:

$$\phi^{n+1} = \phi_1^{n+1} + A_1(t^{n+1})\phi_2 + A_2(t^{n+1})\phi_3 \tag{4.52}$$

ϕ_1^{n+1} is the part due to the elastic force on the tube and $A_1(t^{n+1})\phi_2$, $A_2(t^{n+1})\phi_3$ are the part due to source and sink flow. g^{n+1} can be written as

$$g^{n+1} = A_1(t^{n+1})(\delta_1(r,z) - \delta_3(r,z)) + A_2(t^{n+1})(\delta_2(r,z) - \delta_3(r,z)) \tag{4.53}$$

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

$$\Delta t \nabla^2 \phi_1^{n+1} = \nabla \cdot \vec{v}_* \tag{4.54}$$

$$\Delta t \nabla^2 \phi_2 = (\delta_1(r,z) - \delta_3(r,z)) \tag{4.55}$$

$$\Delta t \nabla^2 \phi_3 = (\delta_2(r,z) - \delta_3(r,z)) \tag{4.56}$$

ϕ_2 and ϕ_3 are independent of time, therefore, only need to be solved once at the beginning. ϕ_1^{n+1} has boundary condition $\nabla \phi_1^{n+1} \cdot n = 0$. After solving for ϕ_1^{n+1}, ϕ_2 and ϕ_3, we have the equation for p_{reg}^{n+1} to be

$$p_{reg}^{n+1} = \frac{3}{2}(\phi_1^{n+1} + A_1(t^{n+1})\phi_2 + A_2(t^{n+1})\phi_3) - \mu(\nabla \cdot \vec{v}_* - g^{n+1}) \tag{4.57}$$

In Eq. (4.57), $A_1(t^{n+1})$ is prescribed and $A_2(t^{n+1})$ is unknown from the current model we use. By combing Eq. (4.16,4.21,4.57), one can obtain the solution for $A_2(t^{n+1})$. The expression for $A_2(t^{n+1})$ is:

$$A_2(t^{n+1}) = \frac{p_2 - pp_2 - A_1(t^{n+1})(\overline{\phi}_{22} - \mu pq_1)}{\mu pq_2 + R_2 + \overline{\phi}_{32}} \tag{4.58}$$

$$pp_2 = \int_\Omega (p_s + \frac{3}{2}\phi_1^{n+1} - \mu\nabla \cdot u_*)(\delta_2 - \delta_3)\, dV \tag{4.59}$$

$$\overline{\phi}_{22} = \int_\Omega \frac{3}{2}\phi_2(\delta_2 - \delta_3)\, dV \tag{4.60}$$

$$\overline{\phi}_{32} = \int_\Omega \frac{3}{2}\phi_3(\delta_2 - \delta_3)\, dV \tag{4.61}$$

$$pq_1 = \int_\Omega (\delta_1 - \delta_3)(\delta_2 - \delta_3)\, dV \tag{4.62}$$

$$pq_2 = \int_\Omega (\delta_2 - \delta_3)(\delta_2 - \delta_3)\, dV \tag{4.63}$$

Once we have $A_2(t^{n+1})$, the regular pressure can be obtained from Eq. (4.57). Then we can obtain $\vec{v}_{reg}$ by

$$u_{reg}^{n+1} = u_* - \Delta t \frac{\partial \phi^{n+1}}{\partial r} \tag{4.64}$$

$$w_{reg}^{n+1} = w_* - \Delta t \frac{\partial \phi^{n+1}}{\partial z} \tag{4.65}$$

4.2.3 Solver in the method

To solve for the $\vec{v}_s$, $\vec{v}_*$ and ϕ in the velocity decomposition approach, we apply the fast sine or cosine transform on z-direction according to the boundary condition at z-direction boundary. Then the discretized system of equation can be reduced to a tri-diagonal system to solve. ϕ has the homogeneous Nuemann boundary condition on all four sides. After applying the fast cosine transform on z-direction, the tri-

diagonal system is:

$$(\frac{1}{dr^2}+\frac{1}{2drr_i})\widehat{\phi}_{i+1,k}+(-\frac{2}{dr^2}-4sin^2(\frac{\pi(k-1)}{2N_z})\frac{1}{dz^2})\widehat{\phi}_{i,k}+(\frac{1}{dr^2}-\frac{1}{2drr_i})\widehat{\phi}_{i-1,k}=\widehat{f}_{i,k} \tag{4.66}$$

Where dr, dz are the grid length on r and z direction, respectively. N_z is the number of the grid on z-direction and r_i denotes the r-coordinate at the ith grid. When k=1, the above system is rank deficient. By setting $\widehat{\phi}_{N_r,1}=0$, which is $\sum_{i=1}^{N_z}\phi_{N_r,i}=0$, we can obtain the unique solution for ϕ.

4.3 Numerical result

4.3.1 Example 1: Convergence for ellipsoid

To demonstrate the second order spatial accuracy of our method, we consider a spherical elastic surface deformed into an ellipsoid shape. (This stationary example does not include a source and sink.) At the plane $\theta = 0$, the cross section of the ellipsoid is described parametrically as $(0.8\cos(\phi), 0.6\sin(\phi))$. The equilibrium shape for this surface cross section is a circle with radius 0.6, centered at origin. In this example, the fluid domain is $[-1.5, 1.5] \times [-1.5, 1.5]$, and the fluid velocity field is computed for $N = 160$, 320, and 640, with uniform grid spacing $h = 1.5/N$ in each case at time $t = 0.3$. The immersed boundary is discretized using 88 markers. We use the solution computed on a high-resolution 1280×1280 grid as the reference solution to estimate errors. Table 4.1 displays the convergence results.

Table 4.1: Convergence results for p, u and w. Approximations exhibit approximately second-order convergence in space.

	p		u		w	
N	L_1 error/h^2	L_{inf} error/h^2	L_1 error/h^2	L_{inf} error/h^2	L_1 error/h^2	L_{inf} error/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

4.3.2 Example 2: Convergence for tube with source and sink

In this example, we do the convergence check for the elastic tube with source and sink. The source has the prescribed periodic strength $A_1(t) = 0.005(1.0 + \sin(\omega t))$ and the sink has the opposite strength as source, which is $A_2(t) = -A_1(t)$. The fluid domain is $[0.0, 0.9] \times [-3.6, 3.6]$, and the fluid velocity field is computed for $N_z = 320$, 640, and 1280, with uniform grid spacing $h = 3.6/N_z$ in each case at time t=0.3. Here N_z is the number of total grid on z-direction. The oscillating frequency is $\omega = 30$. $t = 0.3$ is approximately the time at the end of the first period. The equilibrium shape, source and sink location has been described in [28]. The immersed boundary is discretized using 320 markers. We use the solution computed on a high-resolution 2560×320 grid as the reference solution to estimate errors.

Table 4.2: Convergence results for p, u and w. Approximations exhibit second-order convergence in space.

	p		u		w	
N_z	L_1 error/h^2	L_{inf} error/h^2	L_1 error/h^2	L_{inf} error/h^2	L_1 error/h^2	L_{inf} error/h^2
320	19.32	214.3	1.941e-1	4.043	3.031e-1	6.643
640	24.72	253.2	9.740e-1	3.752	3.784e-1	5.990
1280	20.71	217.9	6.260e-2	5.532	2.615e-1	5.674

Besides spatial convergence test, we do the numerical test on time convergence in this example. In the time convergence study, the fluid pressure and velocity field is computed on the grid 640×80, with uniform grid spacing $h = 3.6/320$ in each case at time $t = 0.2$. The reference solution is the solution with time step $dt = 0.0005$. The time step size dt is refined from $dt = 0.004$ to $dt = 0.001$. The other condition is same as in the spatial convergence test.

Table 4.3: Convergence results for p, u and w. velocity exhibits second-order convergence in time and pressure exhibits between first and second order convergence in time.

	p		u		w	
dt	L_1 error/dt	L_{inf} error/dt	L_1 error/dt	L_{inf} error/dt	L_1 error/dt	L_{inf} error/dt
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

4.3.3 Simulation study

Different from Example. (4.3.2), we compute the sink magnetude by using Eq. (4.16). The fluctuating magnitude for the source is

$$A_1(t) = 0.005(1.0 + \sin(\omega t - 0.5\pi)) \tag{4.67}$$

Therefore, an external source or sink is needed outside the tube to balance the inflow or outflow. Its magnitude is in Eq. (4.12). The tube's boundary force, equilibrium position, position of source and sink has been described in [28]. The external source or sink is centered at $(r, z) = (0.55, 0.0)$ with domain $[r-0.1, r+0.1]\times[z-2.0, z+2.0]$.

Poiseuille characteristics

In this example, we choose inflow/outflow oscillation frequency $\omega = 20$. After the system reaches equilibrium, we plot the z direction velocity and shear stress along the cross section $z = 0.99$ (results are qualitatively similar at other cross sections). From the parabolic shape of the z-velocity (Fig. 4.1), and the v-shaped shear stress (Fig. 4.2), we can see that the flow is approximately Poiseuille flow, as could be expected for laminar flow through a roughly circular pipe. Besides plotting the velocity and shear stress, we also obtained the pressure drop Δp by using the different viscosity (Fig. 4.3). From the approximately linear relation between viscosity and pressure drop, we can also see the flow is approximately Poiseuille flow.

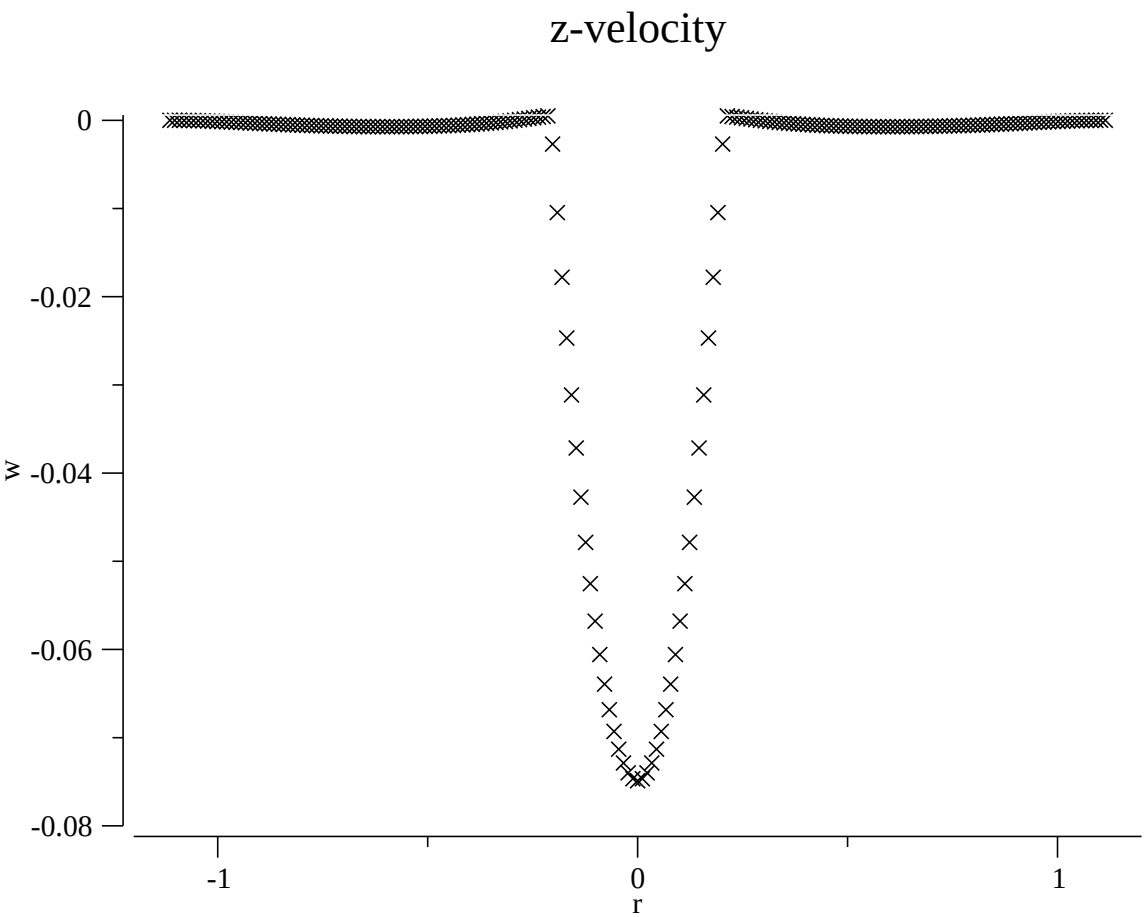

FIGURE 4.1: Velocity in the z direction at $z = 0.99$. Approximately parabolic profile is characteristic of Poiseuille flow.

Frequency-amplitude study

Figure (4.4) shows the maximum and minimum pressure values recorded over an equilibrium temporal cycle at point $(r, z) = (0, 1.8)$, for increasing pumping frequency. For pumping frequencies $5 \leqslant \omega \leqslant 80$, the amplitude of the pressure cycle decreases to the minimum between $10 \leqslant \omega \leqslant 20$ and then increases.

Figure (4.5) shows the maximum and minimum flow values obtained by integrating the vertical velocity field at $z = 0.99$ throughout an equilibrium temporal cycle. For all pumping frequencies, fluid flows in the downward direction. The flow amplitude exhibits similar trend as the pressure amplitude in Fig.4.4.

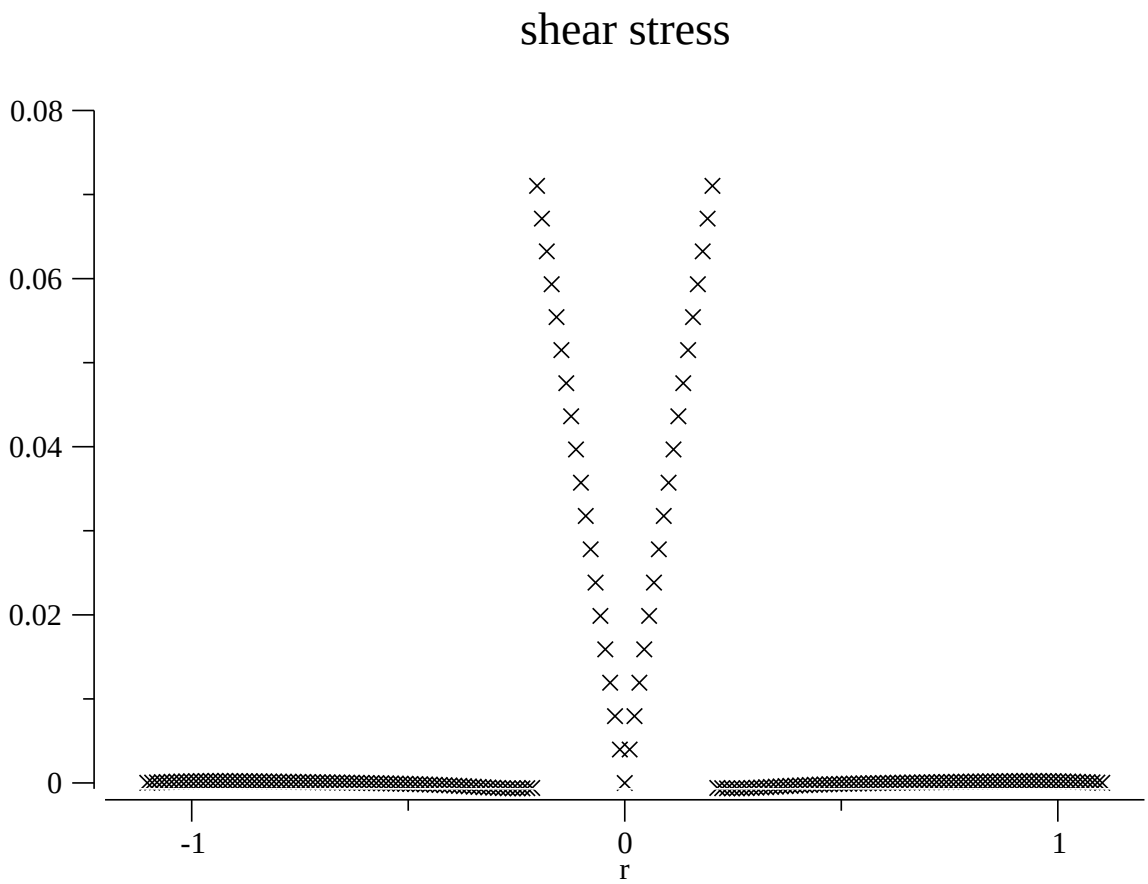

FIGURE 4.2: Shear stress at $z = 0.99$. V-shape is characteristic of Poiseuille flow.

Viscosity-amplitude study

Besides the frequency-amplitude study, we also explore how viscosity can affect the pressure and flow amplitude. Figure (4.6) shows the pressure amplitude recorded over an equilibrium temporal cycle at point $(r, z) = (0, 1.8)$, for increasing viscosity. For flow viscosity $0.01 \leqslant \mu \leqslant 0.15$, the amplitude of the pressure cycle decreases to the minimum between $0.04 \leqslant \mu \leqslant 0.06$ and then increases. Figure (4.7) shows the flow amplitude values obtained by integrating the vertical velocity field at $z = 0.99$ throughout an equilibrium temporal cycle. The flow amplitude exhibits similar trend as the pressure amplitude in Fig.4.6

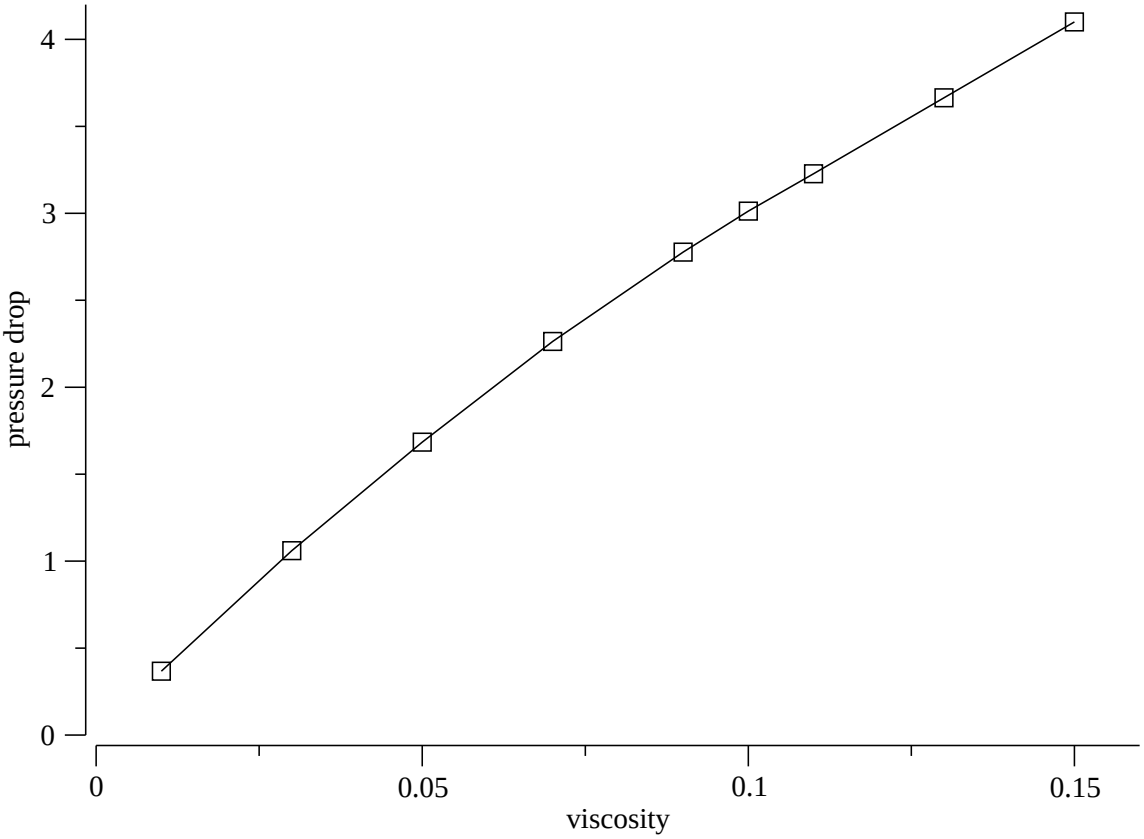

FIGURE 4.3: Pressure drop from z=1.8 to z=-1.8 from different μ. Approximately linear relation is characteristic of Poiseuille flow.

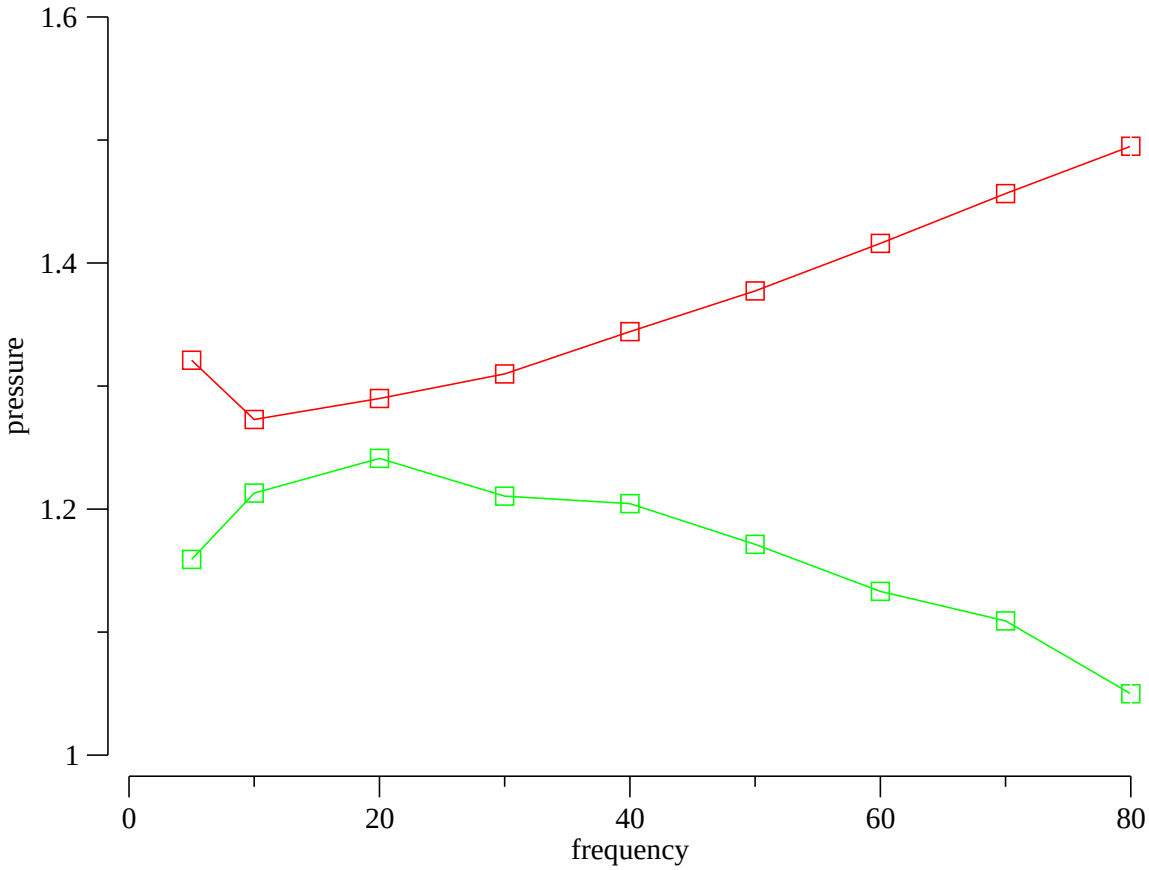

FIGURE 4.4: Minimum and maximum of pressure cycle at $(r, z) = (0, 1.8)$; pumping frequency $5 \leqslant \omega \leqslant 80$.

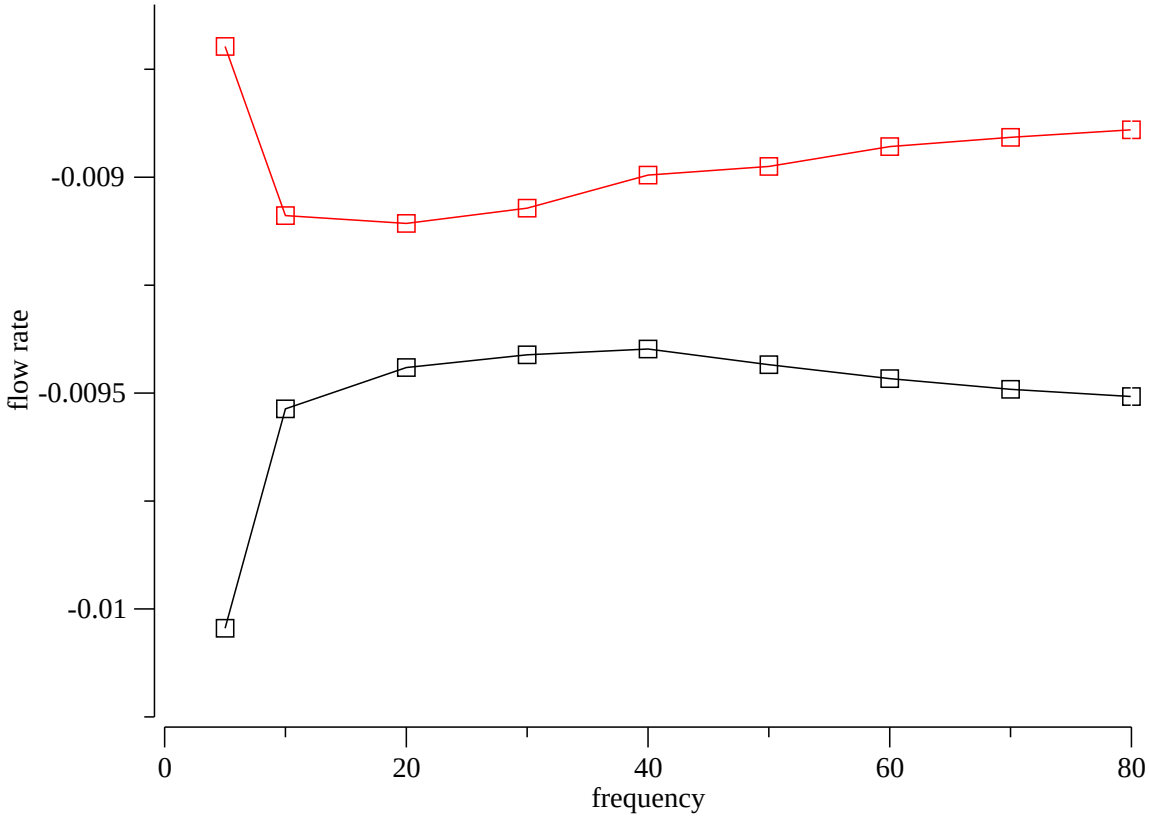

FIGURE 4.5: Minimum and maximum of flow across $z = 0.99$ for pumping frequency $5 \leqslant \omega \leqslant 80$.

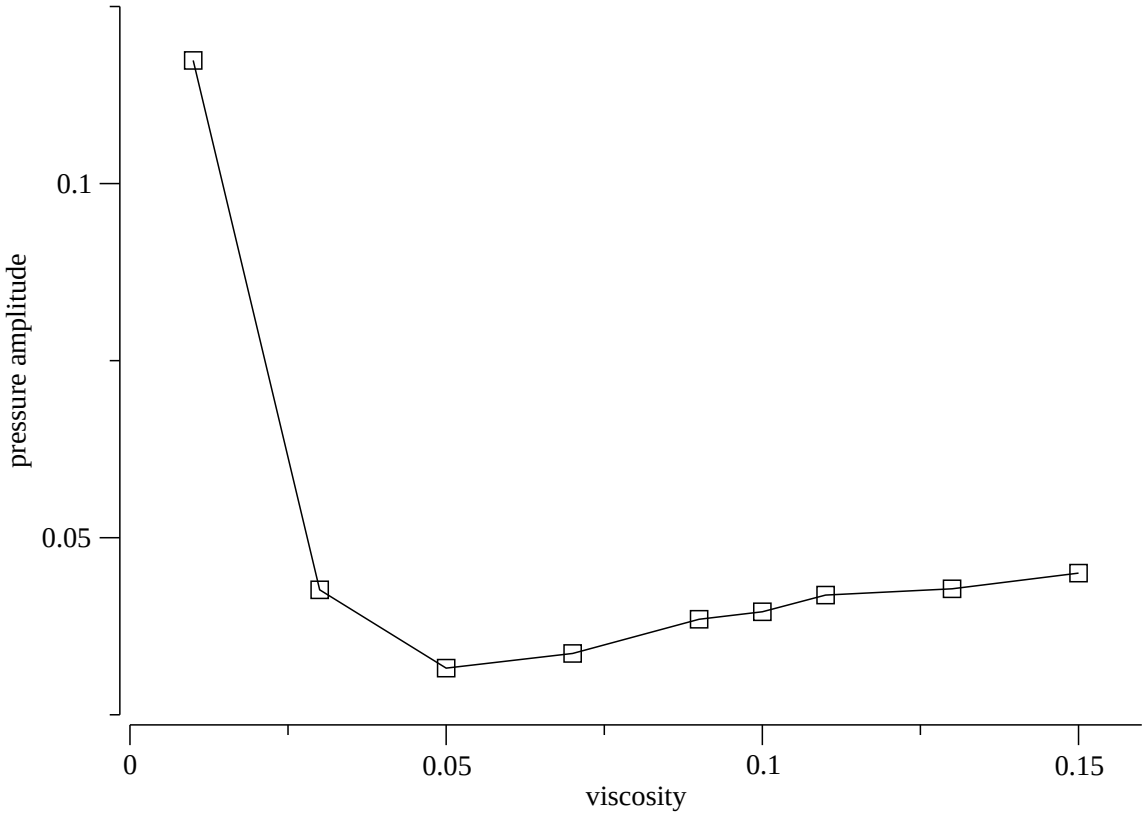

FIGURE 4.6: Pressure amplitude at $(r, z) = (0, 1.8)$; viscosity $0.01 \leqslant \mu \leqslant 0.15$.

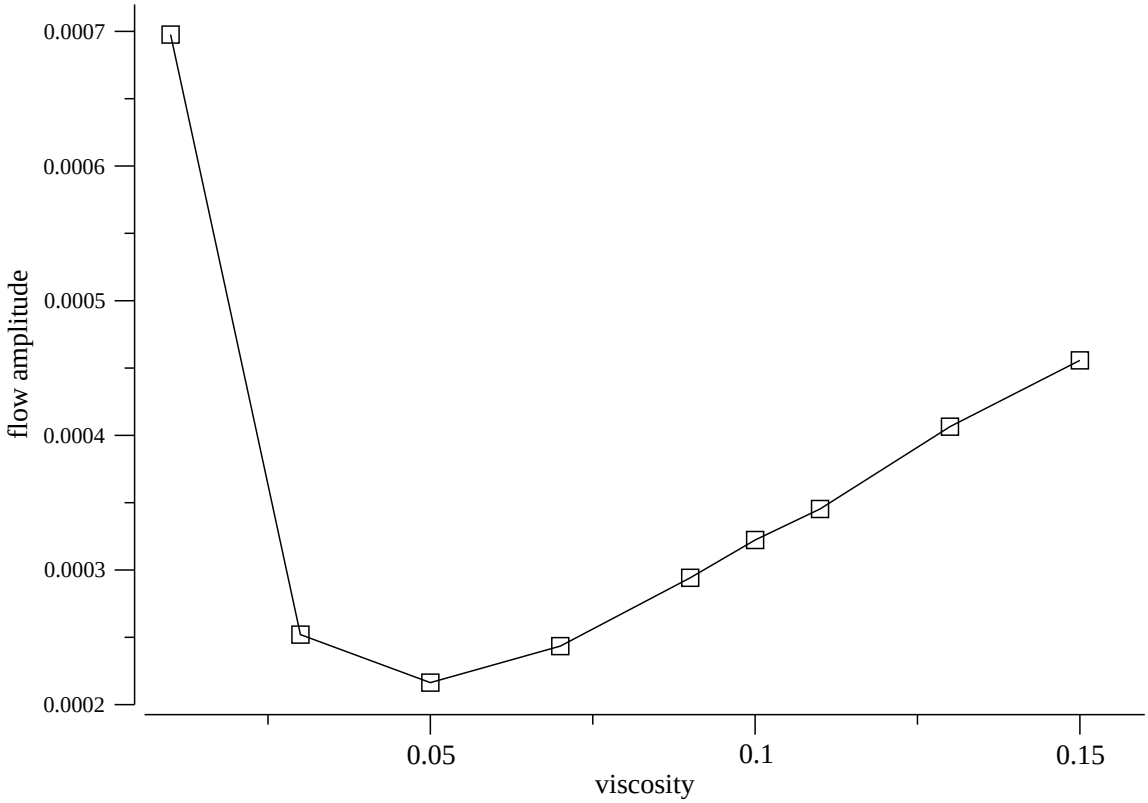

FIGURE 4.7: Amplitude of flow across $z = 0.99$ for viscosity $0.01 \leqslant \mu \leqslant 0.15$.

5

Conclusion

In Chapter 2, we introduced the method of corrected boundary integrals that yields second-order accurate approximations for singular and nearly singular integrals. In particular, the method computes the boundary integral solutions for the Stokes equations, with forces singularly supported along open curve segments. When the Stokes boundary integral solution is evaluated near the interface, the accuracy diminishes by using the standard quadrature rule. To attain the second order accuracy, we derived the correction for errors arising from the trapezoidal rule. The corrected boundary integral method can be combined with a mesh-based solver to yield a hybrid method that can compute Stokes solution everywhere on the computational domain.

To demonstrate the applicability of the methods, we apply the methods to a simple model of a single cilium to study the hydrodynamic interactions between the ciliary beat motion and the surrounding fluid. The model is formulated as an immersed boundary problem. Numerical results show that the method accurately computes fluid solution, and preserves jump discontinuities in the solution across the immersed boundaries. Although it is true that the motion of a boundary immersed in Stokes flow can be simulated by computing fluid velocity on the boundary only,

fluid velocity away from the boundary may still be of interest. For example, to understand how the motion of the cilium transport fluid, one might be interested in tracking tracer particles, which may be located near but not on the cilium. In that case, the method of corrected boundary integrals can prove useful.

The methods in the present study are developed for 2D flows. In 3D Stokes flows driven by forces supported on an open surface, pressure and velocity are given by surface integrals [33]. The singularities of the integral kernels are different in 2D and in 3D. Nonetheless, we believe that if one wants to accurately evaluate the surface integral solutions at a point, one may derive corrections using asymptotic analysis analogous to the 2D methods.

In Chapter 3, we developed numerical methods for simulating Stokes flows along a compliant, closed tube. We employed axisymmetric cylindrical coordinate system to simulate the fluid motion. Our hybrid method combined the boundary integral approach and the immersed interface method with jump conditions derived for the axisymmetric cylindrical coordinates. We used the method to simulate fluid flow through an elastic tube driven by prescribed oscillatory source and sink, and studied how the pressure and velocity oscillation amplitudes vary as functions of the oscillating frequency.

In Chapter 4, we developed numerical methods for simulating Navier-Stokes flows along a compliant, closed tube. This method used the velocity decomposition approach with the axisymmetric cylindrical coordinate system. The jump condition was derived in Chapter 3. In the simulation study, we employed the method to study how the pressure and velocity oscillation amplitudes vary with oscillating frequency and fluid viscosity. The model and numerical method can have potential application in the biological system, such as blood flow in arteries and arterioles. To simulate the blood flow, the axisymmetric wall can be used to model the blood vessel. Quantities of interest, such as pressure and shear stress, can be computed for

understanding the mechanism of cardiovascular disease. In addition, this model can be applied to simulate the mass transport in arteries and arterioles [1]. Coupled with the fluid velocity from the Navier-Stokes equation, the concentration of solute in the blood can be modeled by the advection diffusion equation [1]. The axisymmetric coordinate can reduce the degree of freedom, however, it may not be suitable for simulating the tube with complex geometry.

Bibliography

[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] G. K. Batchelor. *An Introduction to Fluid Dynamics*. Cambridge University Press, Cambridge, UK, 1967.

[3] J. T. Beale and M.-C. Lai. A method for computing nearly singular integrals. *SIAM J. Numer. Anal.*, 38:1902–25, 2001.

[4] J. T. Beale and A. T. Layton. On the accuracy of finite difference methods for elliptic problems with interfaces. *Comm. Appl. Math. Comput. Sci.*, 2006.

[5] J. T. Beale and A. T. Layton. A velocity decomposition approach for moving interfaces in viscous fluids. *J. Comput. Phys.,* in press, 2009.

[6] G. Biros, L. Ying, and D. Zorin. An embedded boundary integral solvers for the Stokes equations. *J. Comput. Phys.*, 193:317–348, 2004.

[7] 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.

[8] J. Blake. Fluid mechanics of cilliary propulsion. In L. J. Fauci and S. Gueron, editors, *Computational Modeling in Biological Fluid Dynamics*, NY, 1999. Springer-Verlag.

[9] J. P. Boris and D. L. Book. Flux-corrected transport. i:shasta, a fluid transport algorithm that works. *J. Comput. Phys.*, 11:38–69, 1973.

[10] D. L. Brown, R. Cortez, and M. L. Minion. Accurate projection methods for the incompressible Navier-Stokes equations. *J. Comput. Phys.*, 168:464–499, 2001.

[11] R. Cortez. The method of regularized stokeslets. *SIAM J. Sci. Comput.*, 23:1204–1225, 2001.

[12] C.W.Hirt and B.D.Nichols. Volume of fluid(vof) method for the dynamics of free boundaries. *J.Compute.Phys*, 39:201–225, 1981.

[13] J. Donea, S. Giuliani, and J. P. Halleux. An arbitrary lagrangian-eulerian finite element method for transient dynamic fluid-structure interactions. *Computer Methods in Applied Mechanics and Engineering*, 33:689–723, 1982.

[14] L. Fauci and A. McDonald. Sperm mobility in the presence of boundaries. *Bull. Math. Biol.*, 57:679–699, 1995.

[15] L. J. Fauci and A. L. Folgelson. Truncated Newton methods and the modeling of complex immersed elastic structures. *Comm. Pure Appl. Math.*, 66:787–818, 1993.

[16] G. R. Fulford and J. R. Blake. Muco-ciliary transport in the lung. *J Thero Biol*, 121:381–402, 1986.

[17] L. Greengard, M. C. Kropinski, and A. Mayo. Integral equatiosn methods for Stokes flow and isotropic elasticity in the plane. *J. Comput. Phys.*, 125:403–414, 1996.

[18] J. Helsing. Thin bridges in isotropic electrostatics. *J. Comput. Phys.*, 127:142–151, 1996.

[19] C. W. Hirt, A. A. Amsden, and J. L. Cook. An arbitrary lagrangian-eulerian computing method for all flow speeds. *J.Compute.Phys*, 14:227–253, 1974.

[20] 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.

[21] E. Jung and C. Peskin. Two dimensional simulations of valveless pumping using the immersed boundary method. *SIAM J.SCI. COMPUTE*, 23:19–45, 2001.

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

[23] P. Kjellgren and J. Hyvarinen. An arbitrary lagrangian-eulerian finite element method. *Computational Mechanics*, 21:81–90, 1998.

[24] P.M. Knupp, L. G. Margolin, and M. J. Shashkov. Reference jacobian optimization-based rezone strategies for arbitrary lagrangian-eulerian methods. *J. Comput. Phys*, 176:93–128, 2002.

[25] M. Lai and Z. Li. A remark on jump conditions for the three-dimensional navier-stokes equations involving an immersed moving membrane. *Applied Math. Letters.*, 14:149–154, 2001.

[26] 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.

[27] R. J. LeVeque and Z. Li. Immersed interface method for stokes flow with elastic boundaries or surface tension. *SIAM J. Numer. Anal.*, 18:709–735, 1997.

[28] Y. Li, S. Williams, and A. T. Layton. A hybrid immersed interface method for driven stokes flow in an elastic tube. *Numer. Math. Theor. Meth. Appl.*, submitted for publication.

[29] A. Mayo. Fast high order accurate solution of Laplace's equation on irregular regions. *SIAM J. Sci. Statist. Comput.*, 6:144–157, 1985.

[30] S. Osher and J. A. Sethian. Fronts propagating with curvature dependent speed: algorithms based on hamilton-jacobi formulation. *J. Comput. Phys.*, 79:12–49, 1988.

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

[32] C. S. Peskin and B. F. Printz. Improved volume conservation in the computation of flows with immersed elastic boundaries. *J. Comput. Phys.*, 105:33–46, 1993.

[33] C. Pozrikidis. *Boundary Integral and Singularity Methods for Linearized Viscous Flow*. Cambridge Univ. Press, Cambridge, 1992.

[34] M. E. Rosar. *A three dimensional model for fluid flow through a collapsible tube*. PhD thesis, New York University, 1994.

[35] M. J. Sanderson. Ciliary activity of cultured rabbit tracheal epithelium: beat pattern and metachrony. *J. Cell Sci.*, 47, 1981.

[36] J. Strain. Locally-corrected multidimensional quadrature rules for singular functions. *SIAM J. Sci. Comput.*, 6(4):992–1017, 1995.

[37] M. Surles. *Numerical approximation of layer potentials along curve segments*. PhD thesis, Duke University, 2008.

[38] P. N. Watton, X. Y. Luo, X. Wang, G. M. Bernacca, P. Molloy, and D. J. Wheatley. Dynamic modelling of prosthetic chorded mitral valves using the immersed boundary method. *J Biomech*, 40:613–626, 2007.

[39] A. M. Winslow. Equipotential zoning of two-dimensional meshes. *Technical Report UCRL-7312 Lawrence Livermore National Laboratory*, 963.

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

Biography

Yi Li is a Ph.D. student of Mathematics at Duke University in Durham, NC. He received the B.S. degree in Chemistry (2006) from Nanjing University in Nanjing, China.