Fast, High-order Algorithms for Simulating Vesicle Flows Through Periodic Geometries

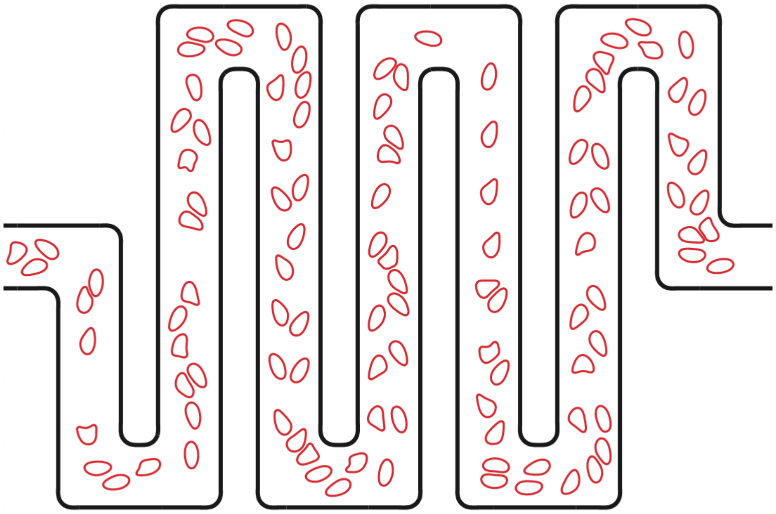
by

Gary Ray Marple

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Applied and Interdisciplinary Mathematics)
in the University of Michigan
2016

Doctoral Committee:

Assistant Professor Shravan K. Veerapaneni, Co-Chair
Associate Professor Lola Eniola-Adefeso, Co-Chair
Associate Professor Silas D. Alben
Professor Charles R. Doering
Professor Divakar Viswanath



ACKNOWLEDGMENTS

I would like to thank my advisor, Professor Shravan Veerapaneni, for his unwavering support during my studies at the University of Michigan. His motivation and persistent push to tackle ever more challenging problems is truly inspirational. He has been a reliable source of guidance and has helped me grow as a person (and a mathematician). I am grateful for collaborating with Professor Alex Barnett. Without the many thought provoking discussions and knowledge he shared, this thesis would not be possible. Many thanks to Dr. Eduardo Corona for the helpful discussions on fast solvers. I would like to express my gratitude for my co-advisor Professor Lola Eniola-Adefeso for her patience and my committee members Professors Silas Alben, Charles Doering, and Divakar Viswanath for agreeing to serve on my dissertation committee. I am grateful for the support and inspiration I received from Professor Bruce Lundberg and the other professors during my undergrad at Colorado State University - Pueblo. They helped me see my potential. Finally, I'd like to thank my roommate Andre Souza for the many thought provoking conversations.

TABLE OF CONTENTS

Acknowledgments . . . . **ii**

List of Figures . . . . **v**

List of Tables . . . . **x**

Chapter

1 Introduction **1**

1.1 Contributions 4
1.2 Limitations 5
1.3 Contents 6

2 Singly Periodic BVPs in Two Dimensions **8**

2.1 Introduction 8
2.2 Periodization scheme 8
2.2.1 Preliminaries: Stokes potentials and the non-periodic BVP 8
2.2.2 Dirichlet problem in a driven periodic pipe 11
2.2.3 Discretization of the linear system 17
2.3 Numerical results 17
2.3.1 Periodization scheme 20

3 Doubly Periodic BVPs in Two Dimensions **22**

3.1 Introduction 22
3.2 Dirichlet problem on a single inclusion 25
3.3 Schur complement system and its iterative solution 28
3.4 Computing the effective conductivity tensor 31

4 Triply Periodic BVPs in Three Dimension **39**

4.1 Introduction 39
4.2 Triply-periodic Stokes BVPs 39
4.2.1 Notation and preliminaries 41
4.2.2 Forming the extended linear system 42
4.2.3 Schur complement system and its iterative solution. 46
4.2.4 Numerical method 47

5 Vesicle Flows Through Singly Periodic Geometries **50**

5.1 Application to particulate flows 50

5.1.1 Solving for quasi-static fluid flow given the interfacial forces . . . 50
5.1.2 Time-stepping scheme 54
5.2 Long-time simulations 58
5.2.1 Arc length correction 58
5.2.2 Area correction 63
5.2.3 Fast reparameterization 63
5.3 Results 65

6 Application: Equilibrium Shapes of Planar Elastic Membranes 68

6.1 Introduction 68
6.2 Rod-theory formulation 70
6.3 Analytical Solutions 72
6.3.1 Clamped boundary conditions 73
6.3.2 Periodic boundary conditions 74
6.3.3 Simply supported boundary conditions 76
6.4 Asymmetric Shapes 77
6.5 Fluid-structure interaction problem 79
6.6 Numerical Scheme 81
6.6.1 Time-stepping scheme 82
6.7 Computational results 84
6.7.1 Error analysis 84
6.7.2 Equilibrium shapes 86
6.8 Conclusions 87
6.9 Appendix: Numerical solution of rod theory equations 88

7 Application: Gating of a mechanosensitive channel due to cellular flows 91

7.1 Introduction 91
7.2 Formulation 94
7.3 Results 97
7.4 Discussion 103

Bibliography 106

LIST OF FIGURES

2.2.1 Geometry for periodization scheme of Section 2.2. The grey shows the infinite periodic pipe, and the blue dots the quadrature nodes for the central domain and its near neighbors. The central domain Ω is bounded by $\Gamma = U \cup D$, and side walls L and R, and has the normal senses shown. The proxy points (red) lie on circle $\mathcal{C}$ of radius R_p. 12

2.3.1 Streamlines for pressure-driven flows through four geometries with no-slip boundary conditions. The background color indicates the magnitude of the velocity (red indicates high and blue indicates low). For the remainder of this chapter, we will refer to geometry (a) as the straight channel, (b) as the converging-diverging channel, (c) as the serpentine channel, and (d) as the channel with reservoirs. In all four geometries, we use $N = 4{,}000$ quadrature points on each wall, $K = 32$ points on each side, and $M = 128$ proxy points. The ring of proxy sources has radius $d = 2\pi$. 18

2.3.2 Logarithm of the relative errors for the velocity as we vary the number of points on the sides and on the ring of proxy sources for the converging-diverging geometry in Fig. 2.3.1(b). For each test point, the velocity was computed using $N = 600$ quadrature points per wall. The boundary integral equation was solved using GMRES. 19

2.3.3 $(*)$ straight channel, $(\circ)$ converging-diverging channel, $(-)$ serpentine channel, $(--)$ channel with reservoirs. (a) Relative errors as the number of side points and proxy sources are varied. For each geometry, we set the number of points on the walls to $N = 4{,}000$ and the number of proxy sources to $M = 4K$. The height of the inlet/outlet is denoted by $|L|$. Notice that the convergence appears to be independent of the complexity of the channel. (b) Relative errors as the number of points on the walls are varied. In each case, we used $K = 32$ points per side and $M = 128$ proxy sources. Spectral convergence was obtained for all four geometries. 21

2.3.4 Spatial plot of the logarithm of relative errors in velocity (a) and pressure (b). Errors were computed using the trapezoidal rule with 4,000 quadrature points per wall. The thin band near the walls, where low accuracy is obtained, may be resolved using a close evaluation scheme. 21

3.1.1 A pressure driven flow (left to right) with 1,000 irregularly shaped inclusions. The color indicates the velocity magnitude of the fluid (red indicates high, blue indicates low). The smallest and the largest inclusions differ in size by about a factor of 10. This simulation was done using two six-core 2.67 GHz Intel Xeon X5650 processors with 48 GB or RAM. 23

3.1.2 2D periodic problem in the case of a single inclusion Ω with boundary $\partial\Omega$. (a) Periodic BVP in $\mathbb{R}^2$, showing a possible unit cell $\mathcal{B}$ and its four walls L, R, D, U, and senses of wall normals. (b) Directly-summed "near" copies of $\partial\Omega$. (c) Circle of auxiliary sources (red dots) representing the field in U due to the infinite punctured lattice of "far" copies (dashed curves). In (b)–(c) blue arrows indicate the action (source to target) of the four matrix blocks A, B, C, and Q. 24

3.4.1 Efficient evaluation of fluxes (J_1, J_2) using far field interations alone. (a) Nine terms in L wall integral in J_1 from the 3×3 layer potential image sum in (3.4.7). (b) Re-interpretation as a sum over targets (nine copies of L) for the potential v. The red dotted line shows the closure of the boundary where flux conservation is applied. (c) Resulting weights of the flux integrals of v on nine wall segments; note all are distant from $\partial\Omega$. (d) Similar weights for J_2. 32

3.4.2 Logarithm of the errors for the velocity as we vary the number of points on the sides and on the ring of proxy sources for the worm shaped inclusion in Fig. 3.4.7. For each test point, the velocity was computing using 256 points on the inclusion. The boundary integral equation was solved using GMRES. . 34

3.4.3 The average time (in seconds) per GMRES iteration as the number of "star" shaped inclusions is varied. Each dot represents a data point. 35

3.4.4 Number of GMRES iterations to invert the boundary integral equation (GMRES resolution set to 10^{-16} with no preconditioner) as the number of circular inclusions is varied. The closeness parameter cl is defined as $cl =$ (Perimeter of Inclusion)/(Distance to Nearest Inclusion). The $clup$ is the maximum value of cl for all for all of the inclusions. Refer to Fig. 3.4.5 for an example of a distribution of circular islands with $clup = 10$. 35

3.4.5 Pressure driven flow (from left to right) with 1,000 circular shaped inclusions with $clup = 10$. The color indicates the magnitude of the fluid velocity (blue is slow, red is fast). 36

3.4.6 Pressure driven flow (from left to right) with 100 "star" shaped inclusions. Each inclusion had 1,000 points on its boundary. The fluid velocity is indicated by the background color (blue is slow, red is fast). 36

3.4.7 Pressure driven Stokes flow around a doubly periodic "worm" geometry. The color indicates the speed of the flow (blue is slow, red is fast). The boundary integral equation was solved using 256 points on each worm geometry. 37

3.4.8 Number of GMRES iterations as the $clup$ is varied for distributions of circular inclusions. The $clup$ is defined in Fig. 3.4.4. The GMRES tolerance was set to 10^{-16} and no preconditioners were used. 37

3.4.9 Errors as the number of side points and proxy points are varied. The number of proxy points was set to $M = 4K$, where K is the number of side points. Notice that the convergence rate appears to be independent of the number of islands. This is because interactions with the sides of the central cell are all far interactions which are smooth and hence, easy to resolve. 38

4.0.1 A plot of fluid streamlines in a triply periodic domain with 2 spheres. 40

4.2.1 Setup for the periodization scheme. The central unit cell has sides W_1 (front), W_{-1} (back), W_2 (right), W_{-2} (left), W_3 (up), and W_{-3} (down). The sphere of proxy points $\mathcal{C}$ surrounding the unit cell represents the contribution from the infinite lattice of "far" periodic copies. 43

4.2.2 Logarithm of the errors for the velocity as we vary the number of points on each side, which is given by n^2, and on the sphere of proxy sources for a single spherical inclusion. The number of proxy sources is given by $M = p(2p + 1)$ in accordance with (4.2.36). 48

5.1.1 (a) Snapshot from a simulation of 1,005 vesicles flowing through an arbitrary-shaped periodic channel. We used 64 discretization points per vesicle and 32,000 points each for the top and bottom walls. The vesicle-vesicle and vesicle-channel hydrodynamic interactions are computed via the Stokes FMM, and the fast direct solver in [26] is used to solve the channel BIEs. We used the close evaluation scheme of [1] for the vesicle-to-vesicle and vesicle-to-channel interactions (but not for channel-to-vesicle interactions due to its inapplicability). This simulation took 52 seconds per time step on a laptop with a 2.4 GHz dual-core Intel Core i5 processor and 8 GB of RAM. (b) Plot of the velocity magnitude (red indicates high and blue indicates low) corresponding to the disturbance field generated by the vesicles (obtained by subtracting the pressure-driven "empty pipe" flow from the total velocity field). 51

5.2.1 Errors without corrections $(--)$ and with both arc length and area corrections $(-)$. The relative error in arc length (a) and area (b) for a single vesicle flowing through the serpentine channel. After around $\sim 1,200$ time steps, the simulation breaks down due to numerical instabilities when no corrections are employed. In the simulation using both corrections, the vesicle successfully passed through the channel several times without incident. In both simulations, the time step was set to $\Delta t = 0.01$. 62

5.3.1 (a) A snapshot of 109 vesicles in the serpentine channel. A pressure difference is pushing the vesicles in the positive x-direction. (b) Average time per time step (in seconds) for the first 10 time steps as the number of vesicles is varied. Each dot represents a data point. Timings were performed using a laptop with a 2.4 GHz dual-core Intel Core i5 processor and 8 GB of RAM. 65

6.3.1 Plot of equilibrium shapes corresponding to the exact solutions (6.3.4) and (6.3.9) for different values of $m^2 = \{0, 0.1, 0.2, \ldots, 0.9\}$. 72

6.3.2 (Color online) Plots of equilibrium shapes corresponding to the exact solutions (6.3.16) and (6.3.18) in (I) and (II) respectively for various excess lengths (with shapes colored the same in (I) and (II) representing those with same value for m). Excess length is defined as the membrane arc length over the period 2π. Periodicity implies that multiple fold shapes obtained by concatenating each of these exact solutions is also an exact solution but with a higher excess length. In (III), we compare the bending energy, defined as $\int_{\gamma(s)} \kappa^2(s)\, ds$, of the shapes in (I) plotted over multiple periods and proportionally scaled back to $[0, 2\pi)$ (for example, a four-fold and a two-fold shapes are plotted in Figure 6.7.2). . . 76

6.4.1 Equilibrium shapes obtained by solving (6.4.1) through (6.4.8) that match the experimentally observed shapes reported in Fig. 1(A) of [2]. The solutions were obtained by modifying ϵ_{pre} while varying L to hold $x(L/2)$ constant. The first and second solutions from the top are odd type solutions as shown in Fig. 6.9.1. The last solution is an even type solution like those shown in Fig. 6.9.2. 77

6.4.2 Other classes of equilibrium shapes that can be obtained by solving (6.4.1) through (6.4.8). The solutions were obtained by numerically continuing ϵ_{pre} from the folded solution to the low amplitude solution while varying L to keep $x(L/2)$ constant. 79

6.5.1 (Color online) Snapshots of a bilipid membrane evolving to an equilibrium shape starting from an arbitrary initial shape. The streamlines are determined by evaluating the surrounding fluid velocity using the relation $\mathbf{v}(\mathbf{x}) = \int_\gamma G(\mathbf{x}, \mathbf{y})\, \mathbf{f}_{\text{mem}}$ for any $\mathbf{x}$ in the fluid domain. The color on the membrane indicates the magnitude of the tension σ. 81

6.7.1 (a) Membrane shapes in time as it relaxes to equilibrium in a quiescent flow. (b) Plots of the membrane bending energy and average tension as a function of time. 85

6.7.2 Evolution of a membrane with (a) four-fold shape to (b) two-fold shape to (c) a single-fold shape, in quiescent flow (see supplementary material for an animation). While the multiple-fold shapes have higher bending energy (as also pointed out in Figure 6.3.2), they correspond to stable equilibria. 87

6.9.1 Odd type equilibrium shapes for different values of ϵ_{pre} along with the internal moments M and the normal component of the internal force F_n obtained by solving (6.9.7). Plots were shifted vertically by 4 for spacing. 89

6.9.2 Even type equilibrium shapes for different values of ϵ_{pre} along with the internal moments M and the normal component of the internal force F_n obtained by solving (6.9.7). Plots were shifted vertically by 4 for spacing. 90

7.1.1 Multiscale characterization of a stretch-activated, membrane-bound channel in a flow. (A) Schematic representation of the protein-induced membrane deformation. Ions either are unable or allowed to cross the membrane depending, respectively, on whether the channel is closed (i) or open (ii) states. (B) (i) A 2D vesicle in a shear flow, with s as the arc length along the membrane. (ii) A vesicle entering and translating through a constriction due to a pressure-driven flow. 93

7.3.1 Energy landscape of a MS channel as a function of (dimensionless) channel radius r, for different values of uniform membrane tension: $\bar{\tau} = 14$ (blue line), $\bar{\tau} = 16.3$ (red line), and $\bar{\tau} = 18$ (green line), which correspond to a closed, critical, and open states respectively. The yellow-shaded regions ($r < r_{\text{closed}} = 3$ nm and $r > r_{\text{open}} = 5$ nm) are not accessible due to steric constraints. 98

7.3.2 Gating of a mechanosensitive channel on a vesicle subject to a shear flow at $\text{Ca} = 8$. (A) The state (open/closed) of the MS channel as a function of time in a period T, where insets show the steady-state shape of the vesicle (left) and the local membrane tension at the MS channel location as a function of time. (B) Energy landscape of the MS channel as a function of channel radius r at different times (or locations on the membrane). The blue dot, red square, green triangle, and black diamond correspond to $\dot{\gamma}t = 0, T/8, T/4, 3T/8$, respectively. 99

7.3.3 Percentage of time the MS channel spends in the open state over one period, P^T_{open}, as a function of the dimensionless shear rate, Ca. 100

7.3.4 Dynamics of a 2D vesicle through a channel with a constriction. (A) Deformation of a vesicle at different locations inside the channel. The green dot represents the location of the MS channel ($s = 0$). (B,C) Membrane tension profiles along the vesicle at the corresponding locations inside the channel shown in (A). Geometric parameters of the channel and constriction (scaled by R_0): $L/R_0 = 15.56$, $H/R_0 = 3.86$. The length and width of the constriction are, respectively, $L/2$ and $H/3$. The nondimensionalized pressure difference is $\frac{\Delta p R_0^3 H}{K_b L} = 8.17$. 102

7.3.5 (A): Visualization of the state of the MS channel as the vesicle flows inside a channel. Symbols (blue asterisks and red open circles) on the membrane represent 64 different locations of the MS channel on the membrane. The closed state is represented by blue asterisks and the open state is represented by red open circles. (B): Percentage of the number of locations where the MS channel is at the open state as the vesicle passes through different locations along the channel. 104

LIST OF TABLES

5.1 Convergence analysis for the arc length and area corrections. To perform the analysis, we evolved a vesicle using time step $\Delta t = 0.01/M$ until $t = 0.1$. We report the relative errors in the arc length L, area A, and position $\boldsymbol{x}$ with and without the corrections. The subscript "c" indicates that both the area and arc length corrections were used, and the subscript "nc" indicates that no corrections were used. The accepted values are labeled with subscript "acc". We now observe second-order asymptotic convergence for the arc length, which is in agreement with Theorem 5.2.2. 61

5.2 The CPU time distribution for the first 10 time steps of a simulation with 1,020 vesicles (64 points each) in the serpentine channel with $N = 29{,}580$ points per wall. Each time step took an average of 58 seconds on a laptop with a 2.4 GHz dual-core Intel Core i5 processor and 8 GB of RAM. 66

6.1 Convergence analysis of our numerical scheme measured on the basis of preserving the membrane inextensibility and fluid incompressibility constraints. Here, M is the number of spatial discretization points and the time-step size Δt is reduced by half as M is doubled. We report the relative errors in the length of the membrane L and area under it A measured at the end of the simulation in figure 6.7.1. We observe a first-order asymptotic convergence for the length and the area, which is consistent with our time-stepping scheme. . . 85

6.2 The analytical solutions in figure 6.3.2 (I) are verified by feeding them as initial shapes to our membrane-fluid solver and checking that the resultant velocity field is zero. We report the maximum magnitude values for the initial velocity field evaluated on the membrane in the second row. We simulate the evolution to a fixed time horizon and report the max-norm error in the difference of initial and final positions in the third row. Clearly, both the errors converge to zero verifying that these are indeed the equilibrium shapes. 86

7.1 Upper table: Summary of physical parameters of the bilayer membrane and the MS channel and their values used in this study. Lower table: Definitions of dimensionless parameters and their corresponding typical values. 96

CHAPTER 1

Introduction

Suspensions of rigid and/or deformable particles in viscous fluids flowing through complex geometries are ubiquitous in natural and engineering systems. Examples include drop, bubble, vesicle, swimmer, and red blood cell (RBC) suspensions. Understanding the spatial distribution of such particles is crucial in a wide range of applications including targeted drug delivery [3], enhanced oil recovery [4], and microfluidics for cell sorting and separation [5]. In several of these applications, the long-time behavior of the suspension is sought. For example: *What is the optimal size and shape of targeted drug carriers that maximizes their ability to reach the vascular walls escaping from flowing RBCs* [3, 6]? *What is the optimal design of a microfluidic device that differentially separates circulating tumor cells from blood cells* [7]? More generally, one is interested in estimating the rheological properties of a given particulate suspension in an applied flow, electric, or magnetic fields. A common mathematical construct that is employed in such a scenario is the periodicity of flow at the inlet and the outlet. Therefore, the natural computational problem that arises is to solve for the transient dynamics of a particulate flow through a complex periodic geometry driven either by pressure difference or other stimuli.

In this work, we provide a periodization algorithm that can be employed to model *vesicle suspensions* through complex geometries. Vesicles—often considered as mimics for biological cells, especially RBCs [8]—are comprised of bilipid membranes enclosing a viscous fluid and their diameter is typically less than 10 μm. The membrane mechanics is modeled by the Helfrich energy [9] combined with a local inextensibility constraint. At the length scale of the vesicles, the Reynolds number is extremely small. Therefore, the Stokes equations will be used to model the fluid interior and exterior to the vesicles. The suspension dynamics of this system is governed by the nonlinear membrane forces, the vesicle-vesicle and vesicle-channel non-local hydrodynamic interactions, and the applied flow boundary conditions.

Pioneered by Youngren and Acrivos [10, 11], BIE methods are widely used for particulate and other interfacial flows [12]. Their advantages over grid- and mesh-based methods are well-known: reduction in dimensionality, ease of achieving high-order accuracy, and availability of highly scalable fast algorithms. The classical approach for incorporating periodic boundary conditions within the BIE framework is to replace the free-space Green's function with one that satisfies the periodicity condition. This can be expressed as an infinite sum of source images. For instance, the double-layer potential defined on an open curve Γ which defines one period of a channel wall with lattice vector $\boldsymbol{d}$ can be written as

$$\boldsymbol{u}(\boldsymbol{x}) = \sum_{n\in\mathbb{Z}} \int_{\Gamma} D(\boldsymbol{x}, \boldsymbol{y} + n\boldsymbol{d})\, \boldsymbol{\tau}(\boldsymbol{y})\, ds_{\boldsymbol{y}}, \tag{1.0.1}$$

where $\boldsymbol{u}$ is the fluid velocity, D is the Stokes free-space double-layer kernel, and $\boldsymbol{\tau}$ is the density function defined on Γ. Classical algorithms, such as the *Ewald summation* [13, 14, 15] for accelerating the N-body calculation that arises from discretizing (1.0.1), use a partition of unity to split the discrete sum into rapidly converging sums for the *nearby* and *distant* interactions, handling the latter in the spectral domain. The local interactions are $\mathcal{O}(N)$ in number by construction, leaving distant interactions which can be evaluated accurately at the N targets by combining local interpolations onto a regular grid with the fast Fourier transform (FFT), a technique named particle-mesh Ewald [16]. Recently, an accurate variant called spectral Ewald has been developed for particulate flows [17, 18, 19] in periodic geometries.

Although such FFT-based methods are widely used, owing to their ease of implementation, they suffer from several drawbacks. Firstly, the FFT introduces a $\mathcal{O}(N \log N)$ complexity, and although the constants are rather small for FFT methods, the scalability of communication costs on multicore architectures is suboptimal (see [20] for a detailed discussion). Secondly, the lack of spatial adaptivity makes them somewhat impractical for problems with multi-scale physics. Finally, the "gridding" required is expensive, becoming even more so in three dimensions.

In this thesis, we provide a periodization scheme that avoids these drawbacks. The periodizing integral equation formulation is based on the ideas introduced in Barnett–Greengard [21] for quasi-periodic scattering problems. It uses direct free-space summation for the nearest-neighbor periodic images, whereas the flow field due to the distant images is captured using an auxiliary basis comprised of a small number of "proxy" sources. Periodic boundary conditions are imposed in an extended linear system (ELS) that determines both the densities on the geometry's walls (to enforce no-slip boundary conditions) and the proxy source strengths. Although one block of this ELS is rectangular and ill-conditioned, its pseudo-inverse is rapid to compute, allowing accuracy close to machine precision for singly periodic and 10-11 digits for doubly and triply periodic domains [22, 23]. The disturbance velocities and the hydrodynamic stresses due to the presence of vesicles enter the right-hand side of the ELS in such a way that the combined flow field is periodic from inlet to outlet and vanishes on the walls. The proposed integral formulation is versatile in handling the imposed flow boundary conditions: for applied pressure-drop across a channel, or imposed slip on channel walls (e.g., to model electroosmotic flows), simply modify the right-hand side of the ELS. The scheme can handle various dimensions of periodicity and easily extends to three dimensions.

The particulate flow solver is based on the work of Veerapaneni et al. [24]. It employs a semi-implicit time-stepping scheme to overcome the numerical stiffness associated with the integro-differential equations governing the vesicle evolution. A spectral (Fourier) basis is used to represent the vesicle and the boundaries of the confining geometry. The required spatial derivatives are computed via spectral differentiation and the singular integrals are also computed with spectral accuracy using the product quadrature rule given in Kress [25]. The FMM is used to accelerate the computation of the vesicle-vesicle hydrodynamic interactions. A simple correction term is introduced in the local inextensibility constraint applied at every time step. It eliminates error accumulation over long time-periods that usually leads to numerical instabilities.

1.1 Contributions

The periodization scheme has several important advantages.

- The scheme provides optimal $\mathcal{O}(N)$ scaling and is well suited for highly parallel architectures.

- Periodic boundary conditions are easy to enforce with no particle sums or meshes required.

- The method uses free-space Greens functions with a small auxiliary basis and imposes periodicity as an extra linear condition. As a result, we can exploit existing free-space solver libraries, quadratures, and fast algorithms (such as the FMM) to efficiently handle large numbers of vesicles in geometrically complex domains.

- Spectral convergence is observed as the number of proxy sources is increased. Singly periodic domains can obtain near machine precision while doubly and triply periodic domains give 10-11 digits of accuracy. In addition, the number of proxy sources needed to obtain a given accuracy is independent of the complexity of the geometry, even for domains with multi-scale features.

- The same approach can be used for problems in two or three dimensions.

Our periodization algorithm for vesicle flows through singly periodic domains was published in [26]. Our algorithm for boundary value problems in doubly periodic domains is in review for publication [27]. In addition, there are two more publications that are related to our periodization scheme [28, 29]. They will be reproduced as applications. The first describes the equilibrium shapes of planar elastic membranes, and it is the result of one of our initial attempts at periodization. The second is concerned with the opening and closing of mechanosensitive channels on a vesicle's membrane when exposed to shear flow. Our periodization algorithms were used in that paper to simulate a vesicle passing through a converging diverging channel.

1.2 Limitations

One of the main limitations with boundary integral methods comes from the evaluation of nearly-singular integrals that occurs when two objects are close to each other. In two dimensions, a recent algorithm for near-singular integration has been developed with spectral accuracy due to Barnett et al. [1]. This algorithm was used in this thesis to evaluate near vesicle-to-vesicle and vesicle-to-channel interactions. We have not yet applied any close evaluation schemes for the channel-to-vesicle interactions. This limitation prohibits us from performing simulations of very tightly-packed suspensions. One possible remedy is to switch the discretization of the channel from a global basis to local panel-based schemes, for which close evaluation schemes for Stokes potentials already exist [30]. In three dimensions, near singular integration is an even greater challenge. There is, however, a relatively new approach developed by Klockner et al., denoted QBX (quadrature by expansion), that promises to enable the fast evaluation of near singular layer potentials [31].

1.3 Contents

The next three chapters (Chapters 2-4) of this thesis focus on modeling Stokes flows, in the absence of vesicles, through singly, doubly, and triply periodic domains. Chapter 2 focuses on modeling flows through a periodic, empty channel with smooth walls of arbitrary shape. Singly periodic channels are important models for understanding blood flow through capillaries and arterials as well as microfluidic channels. For example, a periodic channel with straight walls could be used to predict macro properties of blood, such as the effective viscosity, as the density or reduced area of the vesicles is varied.

In Chapter 3, we extend the periodization scheme to doubly periodic domains with smooth randomly distributed, irregularly shaped inclusions (islands). Such a domain is useful (without vesicles) for modeling fluid flow through porous media. An example application would be for modeling water as it flows past irregularly shaped grains of sand. Another application (with vesicles) would be for modeling size based cell separation in a microfluidic device using deterministic lateral displacement for various sizes and shapes of pillars. For example, there is currently interest in understanding how circulating tumor cells (CTCs) can be more efficiently separated from red blood cells.

In Chapter 4, we extend the periodization scheme to model flow past randomly distributed inclusions with triply periodic boundary conditions. Such a setup is a natural extension of the scheme discussed in Chapter 3. In Chapter 5, we show how the periodization scheme, described in Chapter 2, can be modified to include vesicles. A similar approach can be used to model vesicle flows through doubly or triply periodic domains.

Chapters 6 and 7 focus on applications involving periodic boundary conditions. In Chapter 6, we solve for the equilibrium shapes of a periodic planar elastic membrane. We use a boundary integral approach (like we did for vesicles) to model the membrane dynamics. The work described in Chapter 6 was one of our initial attempts at periodization.

In Chapter 7, we model the opening/closing of mechanosensitive (MS) channels on the membrane of a vesicle/cell due to shear flow. Understanding the flow conditions that cause these MS channels to open are of great interested since they allow drugs and genetic material to enter the vesicle/cell. Unfortunately, the opening/close of MS channels is difficult to observe experimentally. Therefore, an in silico model, such as the one we provide in Chapter 7, may give additional insight.

CHAPTER 2

Singly Periodic BVPs in Two Dimensions

2.1 Introduction

We begin by describing the periodic solver for the steady Stokes equation in a periodic channel given velocity boundary conditions (Section 2.2). The accuracy and practicality of the scheme will be illustrated via numerical examples in Section 2.3. Chapter 5 will discuss how the periodization scheme can be combined with vesicle flow simulations.

2.2 Periodization scheme

2.2.1 Preliminaries: Stokes potentials and the non-periodic BVP

We first define the standard kernels and boundary integral operators used [32] (for our 2D case see [33, Sec. 2.2, 2.3]). Let $\mu > 0$ be the fluid viscosity, a scalar constant. Let $\boldsymbol{u}(\boldsymbol{x}) = (u_1(\boldsymbol{x}), u_2(\boldsymbol{x}))$ be the velocity field and $p(\boldsymbol{x})$ the scalar pressure field for $\boldsymbol{x} = (x_1, x_2) \in \mathbb{R}^2$. The pair $(\boldsymbol{u}, p)$ is a solution to the Stokes equations if

$$\begin{aligned} -\mu\Delta\boldsymbol{u} + \nabla p &= 0 && (2.2.1) \\ \nabla\cdot\boldsymbol{u} &= 0. && (2.2.2) \end{aligned}$$

These express force balance and incompressibility, respectively.

The Stokes single-layer kernel (stokeslet) from source point $\boldsymbol{y}$ to target point $\boldsymbol{x}$ has tensor components

$$S_{ij}(\boldsymbol{x},\boldsymbol{y}) = \frac{1}{4\pi\mu}\left(\delta_{ij}\log\frac{1}{r} + \frac{r_i r_j}{r^2}\right), \quad i,j = 1,2, \tag{2.2.3}$$

where $\boldsymbol{r} := \boldsymbol{x} - \boldsymbol{y}$, $r := \|\boldsymbol{r}\|$, and δ_{ij} is the Kronecker delta. Given a density (vector function) $\boldsymbol{\tau}$ on a source curve Γ, the single-layer representation for velocity is then $\boldsymbol{u} = \mathcal{S}_\Gamma\boldsymbol{\tau}$, i.e.,

$$\boldsymbol{u}(\boldsymbol{x}) = (\mathcal{S}_\Gamma\boldsymbol{\tau})(\boldsymbol{x}) := \int_\Gamma S(\boldsymbol{x},\boldsymbol{y})\boldsymbol{\tau}(\boldsymbol{y})ds_{\boldsymbol{y}}. \tag{2.2.4}$$

The associated pressure function is

$$p(\boldsymbol{x}) = (\mathcal{Q}_\Gamma\boldsymbol{\tau})(\boldsymbol{x}) := \int_\Gamma Q(\boldsymbol{x},\boldsymbol{y})\boldsymbol{\tau}(\boldsymbol{y})ds_{\boldsymbol{y}} \quad \text{where } Q_j(\boldsymbol{x},\boldsymbol{y}) = \frac{1}{2\pi}\frac{r_j}{r^2}\,, \qquad j = 1,2. \tag{2.2.5}$$

For the double-layer velocity representation $\boldsymbol{u} = \mathcal{D}_\Gamma\boldsymbol{\tau}$, we have, using $\boldsymbol{n}^{\boldsymbol{y}}$ to denote the surface normal at each point $\boldsymbol{y}$ on the source curve Γ,

$$\boldsymbol{u}(\boldsymbol{x}) = (\mathcal{D}_\Gamma\boldsymbol{\tau})(\boldsymbol{x}) := \int_\Gamma D(\boldsymbol{x},\boldsymbol{y})\boldsymbol{\tau}(\boldsymbol{y})ds_{\boldsymbol{y}}, \tag{2.2.6}$$

where

$$D_{ij}(\boldsymbol{x},\boldsymbol{y}) = \frac{1}{\pi}\frac{r_i r_j}{r^2}\frac{\boldsymbol{r}\cdot\boldsymbol{n}^{\boldsymbol{y}}}{r^2}\,, \qquad i,j = 1,2. \tag{2.2.7}$$

We write its associated pressure function as

$$p(\boldsymbol{x}) = (\mathcal{P}_\Gamma\boldsymbol{\tau})(\boldsymbol{x}) := \int_\Gamma P(\boldsymbol{x},\boldsymbol{y})\boldsymbol{\tau}(\boldsymbol{y})ds_{\boldsymbol{y}}, \tag{2.2.8}$$

where

$$P_j(\boldsymbol{x},\boldsymbol{y}) = \frac{\mu}{\pi}\left(-\frac{\boldsymbol{n}^{\boldsymbol{y}}_j}{r^2} + 2\boldsymbol{r}\cdot\boldsymbol{n}^{\boldsymbol{y}}\frac{r_j}{r^4}\right)\,, \qquad j = 1,2. \tag{2.2.9}$$

We use the notation $D_{\Gamma',\Gamma}$ to indicate the double-layer boundary integral operator from source curve Γ to target Γ', i.e., $D_{\Gamma',\Gamma}\boldsymbol{\tau} = (\mathcal{D}_\Gamma \boldsymbol{\tau})|_{\Gamma'}$. If the target and source curves are the same ($\Gamma' = \Gamma$) then $D_{\Gamma,\Gamma}$ is to be taken in the principal value sense and has a smooth kernel for smooth Γ. We have, for Γ a C^2-smooth curve and any $\boldsymbol{\tau} \in C(\Gamma)$, the jump relation

$$\lim_{h\to 0^+} (\mathcal{D}_\Gamma \boldsymbol{\tau})(\boldsymbol{x} - h\boldsymbol{n}^{\boldsymbol{x}}) = \big((-\tfrac{1}{2}I + D_{\Gamma,\Gamma})\boldsymbol{\tau}\big)(\boldsymbol{x}) \ , \qquad \boldsymbol{x} \in \Gamma \tag{2.2.10}$$

for the interior limit of velocity. Here, I is the 2×2 identity tensor. The non-periodic prototype BVP that we will need to solve is that the pair $(\boldsymbol{u}, p)$ satisfies the Stokes equations in a bounded domain Ω for given velocity (Dirichlet) data $\boldsymbol{u} = \boldsymbol{v}$ on its boundary $\partial\Omega$. To solve this problem, we insert the double-layer representation $\boldsymbol{u} = \mathcal{D}_{\partial\Omega}\boldsymbol{\tau}$ into (2.2.10) to get the 2nd-kind boundary integral equation (BIE) on $\partial\Omega$:

$$(-\tfrac{1}{2}I + D_{\partial\Omega,\partial\Omega})\boldsymbol{\tau} = \boldsymbol{v} \ . \tag{2.2.11}$$

This BVP, and the resulting BIE, has one consistency condition, $\int_{\partial\Omega} \boldsymbol{v} \cdot \boldsymbol{n} ds = 0$, and null-space of dimension one corresponding to adding a constant to p [33].

Since we will also need to impose *traction* (Neumann) matching conditions, we need the traction on a target curve due to the above representations. Given a function pair $(\boldsymbol{u}, p)$, the Cauchy stress tensor at any point has entries

$$\sigma_{ij}(\boldsymbol{u}, p) := -\delta_{ij} p + \mu(\partial_i u_j + \partial_j u_i) \ , \quad i, j = 1, 2. \tag{2.2.12}$$

The hydrodynamic traction of this pair, i.e., the force vector per unit length applied to the fluid at a surface point with outward unit normal $\boldsymbol{n}$, has components

$$T_i(\boldsymbol{u}, p) := \sigma_{ij}(\boldsymbol{u}, p) n_j = -p n_i + \mu(\partial_i u_j + \partial_j u_i) n_j \ , \quad i = 1, 2, \tag{2.2.13}$$

where summation over j is implied. Applying (2.2.13) to the pair $(\boldsymbol{u}(\boldsymbol{x}), p(\boldsymbol{x}))$ generated by the single-layer velocity (2.2.3) and pressure (2.2.5) kernel (with fixed source point $\boldsymbol{y}$), gives the single-layer traction kernel

$$K_{ik}(\boldsymbol{x},\boldsymbol{y}) = \sigma_{ij}(S_{jk}(\cdot,\boldsymbol{y}),Q_k(\cdot,\boldsymbol{y}))(\boldsymbol{x})\boldsymbol{n}_j^{\boldsymbol{x}} = -\frac{1}{\pi}\frac{r_ir_k}{r^2}\frac{\boldsymbol{r}\cdot\boldsymbol{n}^{\boldsymbol{x}}}{r^2}\ , \qquad i,k=1,2, \quad (2.2.14)$$

which we abbreviate by K. Likewise, applying (2.2.13) to the double-layer pair (2.2.6) and (2.2.8) gives, after a somewhat involved calculation (eg [34, (5.27)]), the double-layer traction kernel tensor

$$\begin{aligned} T_{ik}(\boldsymbol{x},\boldsymbol{y}) &= \sigma_{ij}(D_{jk}(\cdot,\boldsymbol{y}),P_k(\cdot,\boldsymbol{y}))(\boldsymbol{x})\boldsymbol{n}_j^{\boldsymbol{x}}\ , \qquad i,k=1,2 \qquad (2.2.15)\\ &= \frac{\mu}{\pi}\left[\left(\frac{\boldsymbol{n}^{\boldsymbol{y}}\cdot\boldsymbol{n}^{\boldsymbol{x}}}{r^2}-8d_{\boldsymbol{x}}d_{\boldsymbol{y}}\right)\frac{r_ir_k}{r^2}+d_{\boldsymbol{x}}d_{\boldsymbol{y}}\delta_{ik}+\frac{\boldsymbol{n}_i^{\boldsymbol{x}}\boldsymbol{n}_k^{\boldsymbol{y}}}{r^2}+d_{\boldsymbol{x}}\frac{r_k\boldsymbol{n}_i^{\boldsymbol{y}}}{r^2}+d_{\boldsymbol{y}}\frac{r_i\boldsymbol{n}_k^{\boldsymbol{x}}}{r^2}\right], \end{aligned}$$

where for notational convenience we defined the target and source "dipole functions"

$$d_{\boldsymbol{x}} = d_{\boldsymbol{x}}(\boldsymbol{x},\boldsymbol{y}) := (\boldsymbol{r}\cdot\boldsymbol{n}^{\boldsymbol{y}})/r^2\ , \qquad\qquad d_{\boldsymbol{y}} = d_{\boldsymbol{y}}(\boldsymbol{x},\boldsymbol{y}) := (\boldsymbol{r}\cdot\boldsymbol{n}^{\boldsymbol{x}})/r^2,$$

respectively. The use of the symbol T to mean the traction operator vs the double-layer traction kernel will be clear by context. The hypersingular boundary integral operator for the traction of the double-layer from source curve Γ to target Γ' we call $T_{\Gamma',\Gamma}$. To clarify,

$$(T_{\Gamma',\Gamma}\boldsymbol{\tau})(\boldsymbol{x}) = T(\mathcal{D}_\Gamma\boldsymbol{\tau},\mathcal{P}_\Gamma\boldsymbol{\tau})(\boldsymbol{x}) = \int_\Gamma T(\boldsymbol{x},\boldsymbol{y})\boldsymbol{\tau}(\boldsymbol{y})ds_{\boldsymbol{y}}\ , \qquad \boldsymbol{x}\in\Gamma',$$

where in the final expression, the kernel T has tensor components given in (2.2.16).

2.2.2 Dirichlet problem in a driven periodic pipe

We consider a single unit cell Ω confined by one period of the walls U above and D below. The full periodic pipe domain is then $\Omega_\Lambda := \{\boldsymbol{x}\in\mathbb{R}^2 : \boldsymbol{x}+n\boldsymbol{d}\in\Omega, n\in\mathbb{Z}\}$, where $\boldsymbol{d}=(d,0)$ is the lattice vector with period d. See Fig. 2.2.1.

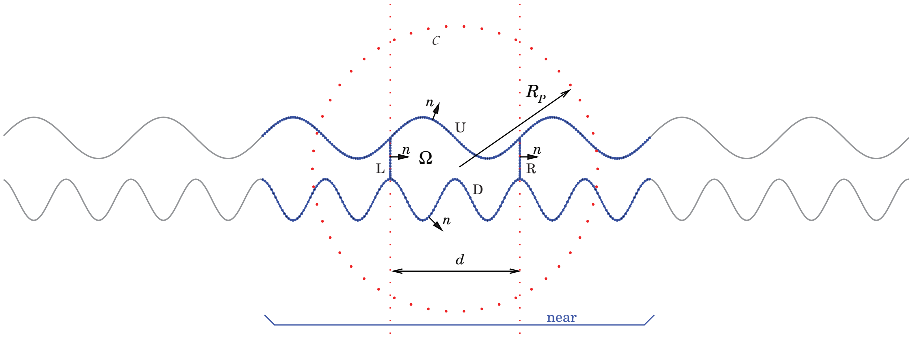


Figure 2.2.1: Geometry for periodization scheme of Section 2.2. The grey shows the infinite periodic pipe, and the blue dots the quadrature nodes for the central domain and its near neighbors. The central domain Ω is bounded by $\Gamma = U \cup D$, and side walls L and R, and has the normal senses shown. The proxy points (red) lie on circle $\mathcal{C}$ of radius R_p.

It is conceptually simplest to begin with the following strictly-periodic "empty pipe" problem. Given periodic velocity (Dirichlet) data $\boldsymbol{v}_U$ and $\boldsymbol{v}_D$ on the up and down walls, find a solution $(\boldsymbol{u}, p)$ in Ω_Λ that is periodic up to a constant pressure driving per period, i.e.,

$$\begin{aligned}
(\boldsymbol{u}, p) & \qquad \text{Stokes in } \Omega_\Lambda && (2.2.16)\\
\boldsymbol{u} &= \boldsymbol{v}_U \text{ on } U && (2.2.17)\\
\boldsymbol{u} &= \boldsymbol{v}_D \text{ on } D && (2.2.18)\\
\boldsymbol{u}(\boldsymbol{x}+\boldsymbol{d}) - \boldsymbol{u}(\boldsymbol{x}) &= 0, \quad \boldsymbol{x} \in \Omega_\Lambda && (2.2.19)\\
p(\boldsymbol{x}+\boldsymbol{d}) - p(\boldsymbol{x}) &= p_{\text{drive}}, \quad \boldsymbol{x} \in \Omega_\Lambda. && (2.2.20)
\end{aligned}$$

The consistency condition on the data is $\int_U \boldsymbol{v}_U \cdot \boldsymbol{n} ds + \int_D \boldsymbol{v}_D \cdot \boldsymbol{n} ds = 0$, and the nullity 1, as in the non-periodic case. When we include vesicles in Chapter 5, the data $\boldsymbol{v}_D$ and $\boldsymbol{v}_U$ will be (minus) the flow velocity induced by a periodized set of vesicles inside Ω_Λ.

A standard approach for solving this BVP in the strictly-periodic case $p_{\text{drive}} = 0$ would be to sum the double-layer kernel over all periodic copies in order to obtain the periodized version of the kernel:

$$D^P(\boldsymbol{x}, \boldsymbol{y}) = \sum_{n \in \mathbb{Z}} D(\boldsymbol{x}, \boldsymbol{y} + n\boldsymbol{d}). \qquad (2.2.21)$$

The representation is then, using $\Gamma = U \cup D$ to indicate the one period of the upper and lower walls,

$$\boldsymbol{u}(\boldsymbol{x}) = (\mathcal{D}^P_\Gamma \boldsymbol{\tau})(\boldsymbol{x}) = \int_U D^P(\boldsymbol{x},\boldsymbol{y})\boldsymbol{\tau}_U(\boldsymbol{y})\, ds_{\boldsymbol{y}} + \int_D D^P(\boldsymbol{x},\boldsymbol{y})\boldsymbol{\tau}_D(\boldsymbol{y})\, ds_{\boldsymbol{y}}. \tag{2.2.22}$$

By analogy with (2.2.11), the density $\boldsymbol{\tau} = [\boldsymbol{\tau}_U; \boldsymbol{\tau}_D]$ could then be determined by solving the 2nd-kind integral equation

$$\left(-\tfrac{1}{2}I + \mathcal{D}^P_{\Gamma,\Gamma}\right)\boldsymbol{\tau} = \boldsymbol{v} \tag{2.2.23}$$

with $\boldsymbol{v} = [\boldsymbol{v}_U; \boldsymbol{v}_D]$.

Remark 2.2.1. *If $p_{\textit{drive}} \neq 0$, this approach could also be used after substracting from $\boldsymbol{v}$ velocity data from the Poiseuille flow $\boldsymbol{u}(\boldsymbol{x}) = (\frac{1}{2}\alpha x_2^2, 0)$, $p(\boldsymbol{x}) = \alpha\mu x_1$, with $\alpha = p_{\textit{drive}}/(\mu d)$. The result is a strictly-periodic BVP with $p_{\textit{drive}} = 0$ and modified data $\boldsymbol{v}$. However, we will find the following approach much more convenient.*

Instead, we reformulate the BVP on the single unit cell Ω, introducing a left side wall L and right side wall $R = L + \boldsymbol{d}$ (Fig. 2.2.1). Note that, given a periodic pipe Ω_Λ, the choice of where to place the wall to subdivide the unit cell is arbitrary. We choose them to be vertical for convenience. Furthermore we relax the periodicity condition on the wall velocity data $\boldsymbol{v}_U$, $\boldsymbol{v}_D$, and impose between L and R periodicity conditions for velocity and traction with given arbitrary mismatch $\boldsymbol{g}_u$, $\boldsymbol{g}_T$, that we call the "discrepancies" [21]. Thus,

$$\begin{aligned}(\boldsymbol{u}, p) &\qquad \text{Stokes in } \Omega && (2.2.24)\\ \boldsymbol{u} &= \boldsymbol{v}_U \text{ on } U && (2.2.25)\\ \boldsymbol{u} &= \boldsymbol{v}_D \text{ on } D && (2.2.26)\\ \boldsymbol{u}_R - \boldsymbol{u}_L &= \boldsymbol{g}_u && (2.2.27)\\ T(\boldsymbol{u}, p)_R - T(\boldsymbol{u}, p)_L &= \boldsymbol{g}_T. && (2.2.28)\end{aligned}$$

By unique continuation from Cauchy data, if $\boldsymbol{v}_U$ and $\boldsymbol{v}_D$ are periodic, and we choose $\boldsymbol{g}_u \equiv \mathbf{0}$ and $\boldsymbol{g}_T = p_{\text{drive}}\boldsymbol{n}$, where $\boldsymbol{n}$ here indicates the normal $(1, 0)$ on the L and R walls, then the above BVP is equivalent to (2.2.16)–(2.2.20). The special case $\boldsymbol{v}_U \equiv \boldsymbol{v}_D \equiv \mathbf{0}$ creates pressure-driven flow in a periodic pipe free of vesicles. In the general case there is still a consistency condition on the data. The advantages of the above (non-periodic) BVP in the single unit cell are that the data may be induced by a sum over vesicles which includes only the nearest images, and that pressure driving is incorporated naturally.

To solve (2.2.24)–(2.2.28), we use a kernel containing only the *near-field* images, plus a small auxiliary basis for smooth Stokes solutions in Ω to account for the effect of the infinite number of *far-field* images. For the latter, we use the "method of fundamental solutions" (MFS) basis [35, 36] of stokeslets with sources lying on a circular contour $\mathcal{C}$ enclosing Ω. (These are also known as "proxy points" [37].) To be precise, the velocity representation is

$$\boldsymbol{u} \;=\; \mathcal{D}_\Gamma^{\text{near}}\boldsymbol{\tau} + \sum_{m=1}^{M} \boldsymbol{\xi}_m \phi_m, \tag{2.2.29}$$

where

$$(\mathcal{D}_\Gamma^{\text{near}}\boldsymbol{\tau})(\boldsymbol{x}) := \sum_{|n|\le 1} \int_U D(\boldsymbol{x}, \boldsymbol{y} + n\boldsymbol{d})\boldsymbol{\tau}_U(\boldsymbol{y})\, ds_{\boldsymbol{y}} + \sum_{|n|\le 1} \int_D D(\boldsymbol{x}, \boldsymbol{y} + n\boldsymbol{d})\boldsymbol{\tau}_D(\boldsymbol{y})\, ds_{\boldsymbol{y}} \tag{2.2.30}$$

is a sum over free-space kernels living on the walls in the central unit cell and its two near neighbors. The second term contains basis functions ϕ_m that satisfy the Stokes equation in the physical domain living in the central unit cell. The basis $\{\phi_m\}$ needs to accurately represent any field due to the "far" periodic copies (i.e. those indexed $\dots, -3, -2, 2, 3, \dots$). The source points $\{\boldsymbol{y}_m\}_{m=1}^{M}$ are equispaced on a circle of sufficiently large radius R_P centered on the central unit cell, and

$$\phi_m(\boldsymbol{x}) = S(\boldsymbol{x}, \boldsymbol{y}_m), \qquad m = 1, \dots, M \tag{2.2.31}$$

is the corresponding set of stokeslet velocity fields. Each coefficient $\boldsymbol{\xi}_m$ lives in $\mathbb{R}^2$, resulting in $2M$ unknowns. This may be viewed as approximating a single-layer density lying on the circle, which is able to represent in its interior any field due to sources lying outside. Since the sources are distant from Ω, the convergence is exponential with a rapid rate; we only need $M = \mathcal{O}(1)$ (typically less than 10^2) *independent of the complexity of the channel walls or the number of quadrature points needed to accurately represent them.*

The pressure representation corresponding to (2.2.29) is (summing (2.2.8) in the same fashion),

$$p = \mathcal{P}_\Gamma^{\text{near}}\boldsymbol{\tau} + \sum_{m=1}^{M} \boldsymbol{\xi}_m \varphi_m, \qquad \text{where } \varphi_m(\boldsymbol{x}) = Q(\boldsymbol{x}, \boldsymbol{y}_m), \quad m = 1, \ldots, M. \tag{2.2.32}$$

For any density $\boldsymbol{\tau}$ and coefficients $\{\boldsymbol{\xi}_m\}$, $(\boldsymbol{u}, p)$ solves the Stokes equations in Ω.

Remark 2.2.2. *There are constraints on the radius R_P: larger R_P allows for more rapid error convergence with respect to M, but if R_P is larger than $3d/2$, then the circle encloses some image sources and the size of the coefficients $\boldsymbol{\xi}_m$ grow exponentially large, resulting in catastrophic cancellation. Hence, we fix $R_P = d$ in this study.*

Constructing a linear system is now simply a matter of inserting the representation (2.2.29) into each of the conditions (2.2.25)–(2.2.28), which we now do. Imposing the velocity data on U and D using the jump relation (2.2.10) (which only affects the $n = 0$ term in (2.2.30)) gives two coupled boundary integral-algebraic equations,

$$(-\tfrac{1}{2}I + D_{U,U}^{\text{near}})\boldsymbol{\tau}_U + D_{U,D}^{\text{near}}\boldsymbol{\tau}_D + \sum_{m=1}^{M} \phi_m|_U \boldsymbol{\xi}_m \quad = \quad \boldsymbol{v}_U \qquad \text{on } U \tag{2.2.33}$$

$$D_{D,U}^{\text{near}}\boldsymbol{\tau}_U + (-\tfrac{1}{2}I + D_{D,D}^{\text{near}})\boldsymbol{\tau}_D + \sum_{m=1}^{M} \phi_m|_D \boldsymbol{\xi}_m \quad = \quad \boldsymbol{v}_D \qquad \text{on } D\,, \tag{2.2.34}$$

which we may summarize as

$$A\boldsymbol{\tau} + B\boldsymbol{\xi} = \boldsymbol{v}\,.$$

Imposing periodicity in matching velocity and traction data (2.2.27)–(2.2.28) gives, after noticing cancellations of all of the touching wall-wall interactions,

$$\begin{aligned}(D_{R,U-\boldsymbol{d}} - D_{L,U+\boldsymbol{d}})\boldsymbol{\tau}_U + (D_{R,D-\boldsymbol{d}} - D_{L,D+\boldsymbol{d}})\boldsymbol{\tau}_D & \\ + \sum_{m=1}^{M} (\phi_m|_R - \phi_m|_L)\boldsymbol{\xi}_m &= \boldsymbol{g}_u \qquad (2.2.35)\\ (T_{R,U-\boldsymbol{d}} - T_{L,U+\boldsymbol{d}})\boldsymbol{\tau}_U + (T_{R,D-\boldsymbol{d}} - T_{L,D+\boldsymbol{d}})\boldsymbol{\tau}_D & \\ + \sum_{m=1}^{M} \big(T(\phi_m, \varphi_m)|_R - T(\phi_m, \varphi_m)|_L\big)\boldsymbol{\xi}_m &= \boldsymbol{g}_T \qquad (2.2.36)\end{aligned}$$

which we summarize as

$$C\boldsymbol{\tau} + Q\boldsymbol{\xi} = \boldsymbol{g}\ .$$

The four coupled boundary integral-algebraic equations (2.2.33)–(2.2.36) may be stacked in pairs and written in a block form

$$\begin{bmatrix} A & B \\ C & Q \end{bmatrix} \begin{bmatrix} \boldsymbol{\tau} \\ \boldsymbol{\xi} \end{bmatrix} = \begin{bmatrix} \boldsymbol{v} \\ \boldsymbol{g} \end{bmatrix}. \qquad (2.2.37)$$

To recap, the roles of the block matrices are as follows: A is a 2nd-kind operator mapping wall densities to (U, D) wall velocities, B maps auxiliary coefficients to wall velocities, C maps wall densities to their discrepancies in the periodicity conditions, and Q maps auxiliary coefficients to their discrepancies. The functions $\boldsymbol{v}$ and $\boldsymbol{g}$ contain the Dirichlet and discrepancy data.

The solution to the system is then

$$\boldsymbol{\tau} = \big(A - B(Q^{+}C)\big)^{-1}\big(\boldsymbol{v} - B(Q^{+}\boldsymbol{g})\big) \qquad (2.2.38)$$

$$\boldsymbol{\xi} = Q^{+}\boldsymbol{g} - (Q^{+}C)\boldsymbol{\tau} \qquad (2.2.39)$$

where we used the Schur complement to reduce the dimensionality of the solve. One could also use the Fast Direct Solver discussed in [26].

2.2.3 Discretization of the linear system

The coupled integral-algebraic equations (2.2.37) need to be discretized; this is performed in a standard fashion using N quadrature nodes $\{\boldsymbol{x}_{U,j}\}_{j=1}^{N}$ on U, N nodes $\{\boldsymbol{x}_{D,j}\}_{j=1}^{N}$ on D, and K nodes $\{\boldsymbol{x}_{L,j}\}_{j=1}^{K}$ on L (the nodes on R being those on L displaced by $\boldsymbol{d}$). The nodes on U and D are generated using the periodic trapezoid rule applied to a smooth parametrization of the curves, while the L and R wall "collocation" nodes are chosen to be Gauss–Legendre in the vertical coordinate (no weights are needed for these nodes) [38, Ch. 9].

The Nyström method [39, Sec. 12.2] is used to discretize A. For instance, given the quadrature weights $\{w_{U,j}\}_{j=1}^{N}$ on U, the matrix discretization of the $1,1$ tensor block of $D_{U,U}$ has elements

$$\mathsf{D}_{ij} = \begin{cases} D_{11}(\boldsymbol{x}_{U,i}, \boldsymbol{x}_{U,j}) w_{U,j}, & i \neq j \\ -\frac{\kappa(\boldsymbol{x}_{U,j})}{2\pi}(\boldsymbol{t}_1(\boldsymbol{x}_{U,j}))^2 w_{U,j}, & i = j \end{cases}, \tag{2.2.40}$$

which uses the diagonal limit $\lim_{\boldsymbol{y}\to\boldsymbol{x}} D_{ij}(\boldsymbol{x},\boldsymbol{y}) = -\frac{\kappa(\boldsymbol{x})}{2\pi}\boldsymbol{t}_i(\boldsymbol{x})\boldsymbol{t}_j(\boldsymbol{x})$, where κ is the curvature of the boundary, and $\boldsymbol{t}$ the unit tangent vector. Other blocks are filled similarly. The result is to replace (2.2.37) by a discrete linear system of identical structure, which will be solved with a fast direct solver described in [26]. For more details on a similar periodic discretization scheme, see [22].

2.3 Numerical results

In this section, we test the performance of our algorithm on simulating flows through four geometries ranging in complexity from a flat channel to a more complicated space-filling geometry. We apply a pressure-drop on each side which drives the flow in the positive

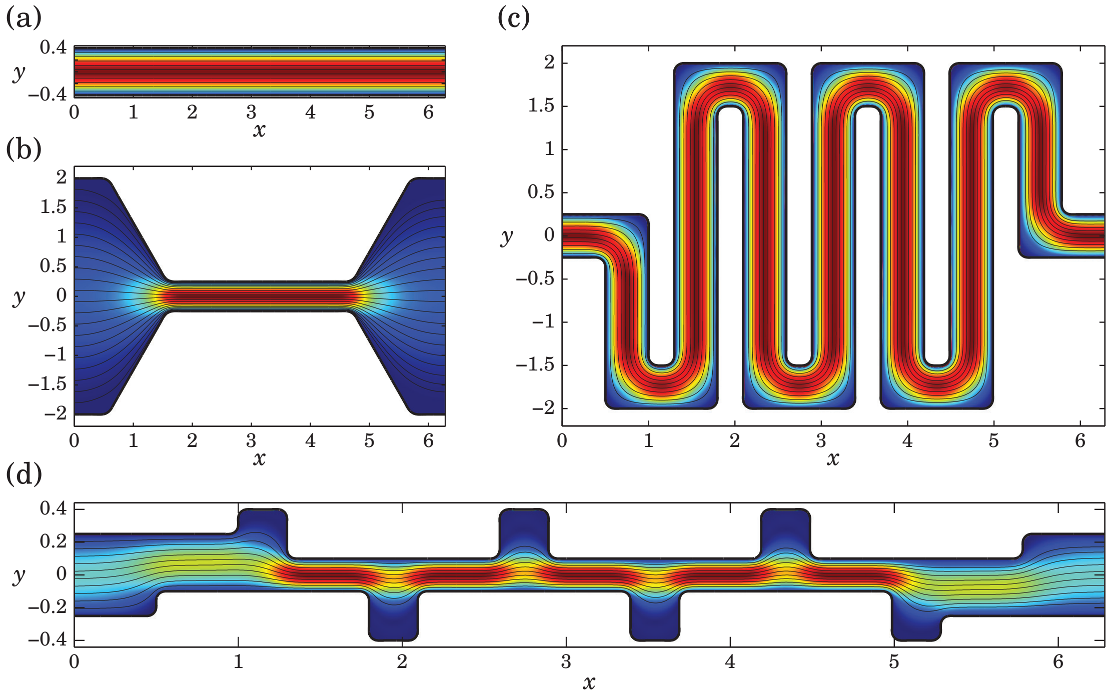


Figure 2.3.1: Streamlines for pressure-driven flows through four geometries with no-slip boundary conditions. The background color indicates the magnitude of the velocity (red indicates high and blue indicates low). For the remainder of this chapter, we will refer to geometry (a) as the straight channel, (b) as the converging-diverging channel, (c) as the serpentine channel, and (d) as the channel with reservoirs. In all four geometries, we use $N = 4{,}000$ quadrature points on each wall, $K = 32$ points on each side, and $M = 128$ proxy points. The ring of proxy sources has radius $d = 2\pi$.

x-direction while simultaneously enforcing no-slip boundary conditions on the walls. The streamlines are plotted in Fig. 2.3.1 along with the magnitude of the velocity which is indicated by the color in the background. We will refer to geometry (a) as the straight channel, (b) as the converging-diverging channel, (c) as the serpentine channel, and (d) as the channel with reservoirs. Note that all the wall geometries are $\mathcal{C}^\infty$ curves, constructed using partitions of unity to avoid sharp corners. We used the trapezoidal rule, with $N = 4{,}000$ nodes on each wall (unless otherwise stated), to evaluate the layer potentials at interior targets for this test.[1]

[1] For some applications, the trapezoidal rule may not be an ideal choice since it cannot yield uniform convergence for points that are very close to the walls. For example, notice the thin band near the wall in Fig. 2.3.4. Accurate close-evaluation schemes must be used instead. In our setting, the no-slip condition at the walls leads to low velocities near the walls and in general helps alleviate numerical instabilities naturally

In the following subsection, we analyze the convergence of the periodization scheme as we vary the number of proxy points (denoted by M), discretization points on L and R (denoted by K), and quadrature points on the walls (denoted by N). For these tests, we set the boundary conditions and pressure-drop such that the horizontal Poiseuille flow $\boldsymbol{u}_e(\boldsymbol{x}) = (\alpha x_2^2, 0)$, $p_e(\boldsymbol{x}) = \alpha\mu x_1$, with $\alpha = p_{\text{drive}}/(2\mu d)$ satisfies the BVP. For all tests, we use $\alpha = 0.2$, $\mu = 0.7$, and $d = 2\pi$. The relative errors are defined as

$$\text{relative error in velocity} = \frac{\|\boldsymbol{u} - \boldsymbol{u}_e\|_2}{\max\limits_{\boldsymbol{x}\in\Omega}\left(\|\boldsymbol{u}_e\|_2\right)} \tag{2.3.1}$$

$$\text{relative error in pressure} = \left|\frac{p - p_e}{p_{\text{drive}}}\right|, \tag{2.3.2}$$

where $(\boldsymbol{u}, p)$ is the numerical solution obtained by the algorithm. We report the maximum relative error at the same three points $(0.8, 0)$, $(3.2, 0)$, and $(4.8, 0)$ for all four geometries (not too close to the walls).

for vesicle flow simulations.

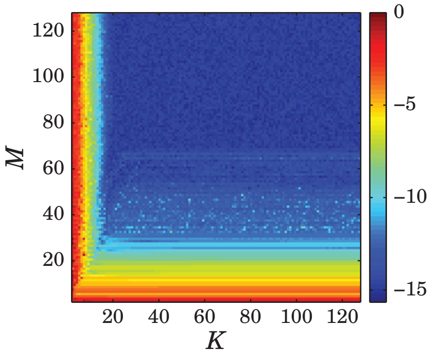


Figure 2.3.2: Logarithm of the relative errors for the velocity as we vary the number of points on the sides and on the ring of proxy sources for the converging-diverging geometry in Fig. 2.3.1(b). For each test point, the velocity was computed using $N = 600$ quadrature points per wall. The boundary integral equation was solved using GMRES.

2.3.1 Periodization scheme

Here, we analyze the convergence properties of the periodization scheme. We begin by focusing on the sides L and R and the proxy sources $\mathcal{C}$. We compute the relative errors for the velocity in the converging-diverging channel as the number of side points K and proxy sources M are varied. The number of quadrature points along the walls is fixed to $N = 600$, which, based on previous experimentation, is enough to resolve the flow to near machine precision. As can be seen in Fig. 2.3.2, rapid convergence is achieved and high accuracy is obtained using a relatively small K and M. This is mainly because of the analytical cancellation of close interactions of L and R with the walls, as shown in equations (2.2.35) and (2.2.36). Since flow far from a disturbance is generally smooth with rapidly decaying Fourier modes, the distant interactions relating to L, R, and $\mathcal{C}$ are easily resolved.

Next, we investigate how the complexity of the geometry affects the convergence rate of the side points and the proxy sources. This is done by setting $M = 4K$ and measuring the relative errors pertaining to the four test geometries as K is varied. Fig. 2.3.3(a) shows the results. Notice that once we account for the height of the inlet/outlet, the convergence of all four geometries is almost exactly the same, independent of the complexity of the channel.

Finally, we focus on the walls U and D. Fig. 2.3.3(b) shows the relative errors for all four test geometries as the number of quadrature points N is varied. In all cases, we observe spectral convergence, as expected for smooth curves. The relative errors for both the velocity and pressure inside the serpentine channel are plotted in Fig. 2.3.4. The small band of larger errors near the wall is the result of using the global trapezoidal rule as opposed to a close evaluation scheme for the walls. It will turn out that in some cases this error band may cause problems for vesicle simulations if vesicles drift too close to the walls, as we will discuss in Chapter 5.

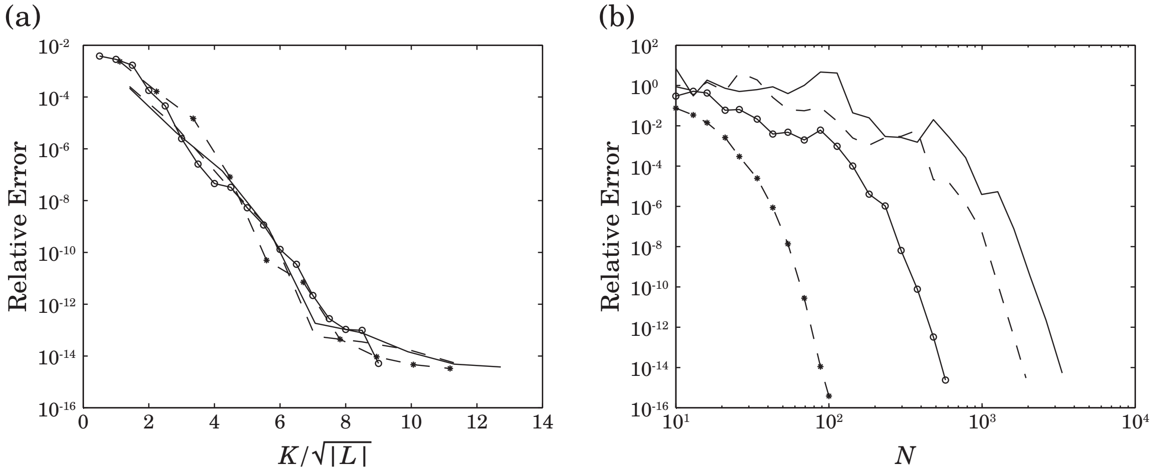


Figure 2.3.3: $(*)$ straight channel, $(\circ)$ converging-diverging channel, $(-)$ serpentine channel, $(--)$ channel with reservoirs. (a) Relative errors as the number of side points and proxy sources are varied. For each geometry, we set the number of points on the walls to $N = 4{,}000$ and the number of proxy sources to $M = 4K$. The height of the inlet/outlet is denoted by $|L|$. Notice that the convergence appears to be independent of the complexity of the channel. (b) Relative errors as the number of points on the walls are varied. In each case, we used $K = 32$ points per side and $M = 128$ proxy sources. Spectral convergence was obtained for all four geometries.

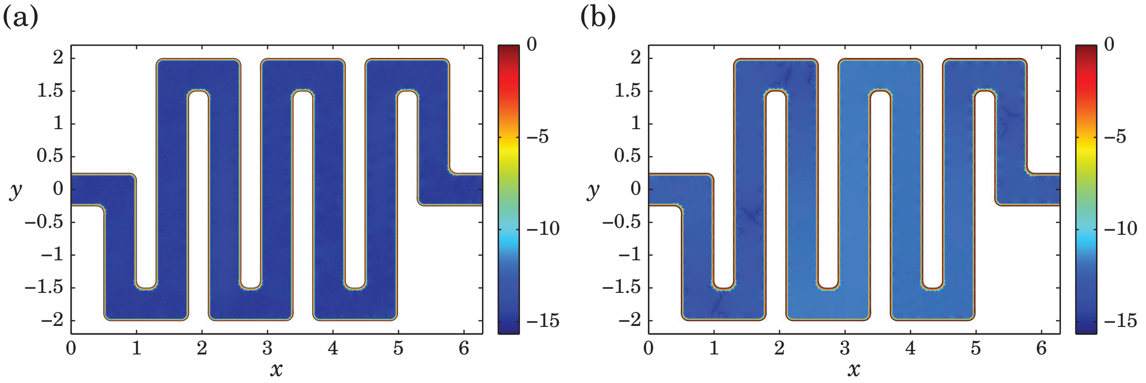


Figure 2.3.4: Spatial plot of the logarithm of relative errors in velocity (a) and pressure (b). Errors were computed using the trapezoidal rule with 4,000 quadrature points per wall. The thin band near the walls, where low accuracy is obtained, may be resolved using a close evaluation scheme.

CHAPTER 3

Doubly Periodic BVPs in Two Dimensions

3.1 Introduction

Doubly periodic boundary value problems (BVPs) are very common in engineering and science, either for modeling the response of periodic composite or multi-phase materials, or for periodized (representative volume element, or "super-cell") simulations of non-periodic random materials and fluid flows. The problem of computing, for a given microscopic composite, the effective macroscopic uniform medium properties under external driving is called *homogenization* and has a vast body of literature. This chapter will focus on extending the periodization scheme for these types of problems. That is, a Stokes flow through a doubly periodic domain containing a large number of smooth irregularly shaped inclusions. Fig. 3.1.2(a) sketches such a problem in the case of a single inclusion Ω per unit cell.

As before, we choose a unit cell $\mathcal{B}$, which may intersect $\partial\Omega$, and impose periodicity via auxiliary conditions on solution differences between pairs of walls (e.g. the L–R wall pair; see Fig. 3.1.2(a)). We sum our potential representation over only the 3×3 nearest neighbor images of $\partial\Omega$ (see Fig. 3.1.2(b)) but add a small auxiliary basis of smooth PDE solutions in U, with coefficient vector $\boldsymbol{\xi}$, that efficiently represents the distant contribution of the rest of the infinite image lattice. The result is the familiar 2×2 block "extended linear system"

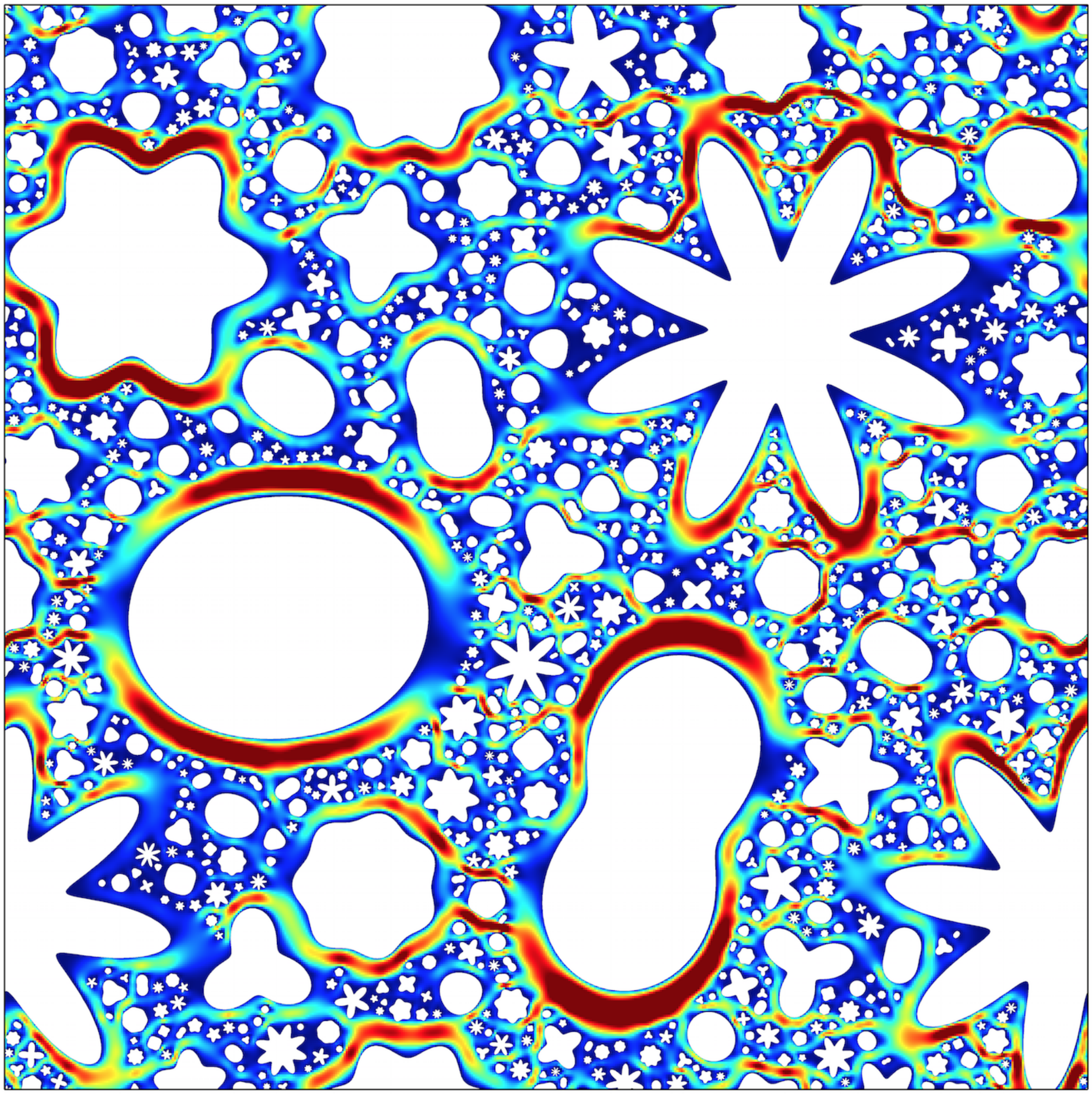


Figure 3.1.1: A pressure driven flow (left to right) with 1,000 irregularly shaped inclusions. The color indicates the velocity magnitude of the fluid (red indicates high, blue indicates low). The smallest and the largest inclusions differ in size by about a factor of 10. This simulation was done using two six-core 2.67 GHz Intel Xeon X5650 processors with 48 GB or RAM.

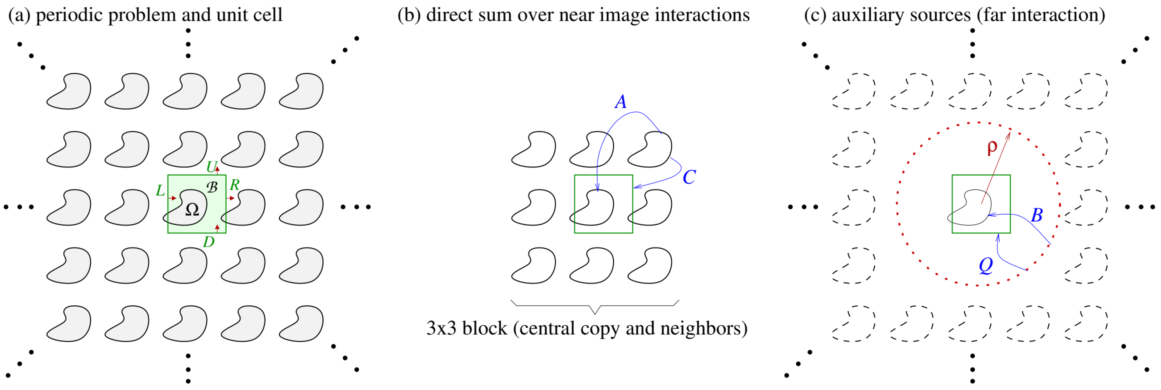

(a) periodic problem and unit cell
(b) direct sum over near image interactions
(c) auxiliary sources (far interaction)
U
L
ℬ
R
Ω
D
A
C
3x3 block (central copy and neighbors)
ρ
B
Q

Figure 3.1.2: 2D periodic problem in the case of a single inclusion Ω with boundary $\partial\Omega$. (a) Periodic BVP in $\mathbb{R}^2$, showing a possible unit cell $\mathcal{B}$ and its four walls L, R, D, U, and senses of wall normals. (b) Directly-summed "near" copies of $\partial\Omega$. (c) Circle of auxiliary sources (red dots) representing the field in U due to the infinite punctured lattice of "far" copies (dashed curves). In (b)–(c) blue arrows indicate the action (source to target) of the four matrix blocks A, B, C, and Q.

(ELS),

$$\begin{bmatrix} A & B \\ C & Q \end{bmatrix} \begin{bmatrix} \boldsymbol{\tau} \\ \boldsymbol{\xi} \end{bmatrix} = \begin{bmatrix} \boldsymbol{f} \\ \boldsymbol{g} \end{bmatrix}, \tag{3.1.1}$$

where the first block row applies the Dirichlet boundary condition on $\partial\Omega$, and the second block row imposes the periodicity conditions between pairs of walls. A non-zero $\boldsymbol{g}$ accounts for the applied macroscopic pressure gradient in the form of a jump in value across a single unit cell. Fig. 3.1.2(b)–(c) shows the interactions described by the four operators A, B, C, and Q. Again, the linear system is generally *rectangular and highly ill-conditioned*, but when solved in a backward-stable least-squares sense can result in accuracies close to machine precision.

There are three main routes to solving (3.1.1). (a) The simplest is to use a dense direct least-squares solve (e.g., via QR), but the $O(N^3)$ cost becomes impractical for large problems with $N \gg 10^4$. To create fast algorithms one exploits the fact that the numbers of auxiliary rows and columns in (3.1.1) are each $O(1)$ (of order 10^2), as follows. (b) One may attempt to eliminate $\boldsymbol{\xi}$ to get the Schur complement square linear system involving

A_{per}, the periodized version of A, namely

$$A_{\text{per}}\boldsymbol{\tau} \;:=\; (A - BQ^{+}C)\boldsymbol{\tau} \;=\; \boldsymbol{f} - BQ^{+}\boldsymbol{g}\,, \tag{3.1.2}$$

where Q^{+} is a pseudoinverse of Q (see Section 3.3). Assuming A_{per} exists, then (3.1.2) is well-conditioned and can be solved iteratively using fast multipole acceleration to apply $A\boldsymbol{\tau}$, while applying the second term in its low-rank factored form $-B((Q^{+}C)\boldsymbol{\tau})$. (c) One may instead eliminate $\boldsymbol{\tau}$ by forming and applying a compressed representation of A^{-1} via a *fast direct solver* [37]; this has proven useful for the case of fixed periodic geometry with multiple boundary data [40, 41]. Both methods (b) and (c) may achieve $\mathcal{O}(N)$ complexity, and hence are appropriate for large problems. In this chapter, we focus on approach (b).

One of our contributions is handling a subtlety that was absent in the singly periodic case but can cause difficulties when using iterative solvers for doubly periodic problems. That is, depending on the choice of integral representation, A_{per} may cease to exist; more precisely, for existence, its domain must be restricted to a linear subspace of densities $\boldsymbol{\tau}$. As an example, there is no solution to the periodic Stokes problem when the unit cell contains a nonzero net density, because the sum, over the infinite lattice, of the logarithmic Greens function clearly diverges. Each PDE and integral representation thus brings *consistency conditions* on $\boldsymbol{\tau}$; we show how these can be accommodated via a rank-3 modification to Q in (3.1.1), which results in a well-conditioned system (3.1.2) that can then be solved iteratively via, e.g., GMRES. The rank-3 perturbations which remove the left null-space of Q are inspired by the randomized methods of Sifuentes et al. [42].

3.2 Dirichlet problem on a single inclusion

We now present the doubly periodic Stokes problem in the exterior of a periodic lattice of smooth inclusions. For simplicity, we assume a single inclusion Ω per unit cell with boundary Γ. Let $\boldsymbol{e}_1$ and $\boldsymbol{e}_2$ be vectors defining the lattice in $\mathbb{R}^2$. For simplicity, we work with the unit square case so that $\boldsymbol{e}_1 = (1, 0)$ and $\boldsymbol{e}_2 = (0, 1)$. Let $\Omega_\Lambda := \{\boldsymbol{x} \in \mathbb{R}^2 \mid \boldsymbol{x} + m\boldsymbol{e}_1 + n\boldsymbol{e}_2 \in \Omega \text{ for some } m, n \in \mathbb{Z}\}$ represent the infinite lattice of inclusions. The velocity, pressure pair $(\boldsymbol{u}, p)$ solve the BVP

$$(\boldsymbol{u}, p) \qquad \text{Stokes in } \mathbb{R}^2 \setminus \overline{\Omega_\Lambda} \tag{3.2.1}$$

$$\boldsymbol{u} = \boldsymbol{0}, \quad \boldsymbol{x} \in \partial\Omega_\Lambda \tag{3.2.2}$$

$$\boldsymbol{u}(\boldsymbol{x} + \boldsymbol{e}_1) - \boldsymbol{u}(\boldsymbol{x}) = \boldsymbol{0}, \quad \boldsymbol{x} \in \mathbb{R}^2 \setminus \overline{\Omega_\Lambda} \tag{3.2.3}$$

$$\boldsymbol{u}(\boldsymbol{x} + \boldsymbol{e}_2) - \boldsymbol{u}(\boldsymbol{x}) = \boldsymbol{0}, \quad \boldsymbol{x} \in \mathbb{R}^2 \setminus \overline{\Omega_\Lambda} \tag{3.2.4}$$

$$p(\boldsymbol{x} + \boldsymbol{e}_1) - p(\boldsymbol{x}) = p_1, \quad \boldsymbol{x} \in \mathbb{R}^2 \setminus \overline{\Omega_\Lambda} \tag{3.2.5}$$

$$p(\boldsymbol{x} + \boldsymbol{e}_2) - p(\boldsymbol{x}) = p_2, \quad \boldsymbol{x} \in \mathbb{R}^2 \setminus \overline{\Omega_\Lambda} \tag{3.2.6}$$

where $\boldsymbol{p} = (p_1, p_2)$ specifies the macroscopic pressure drop. We reformulate the BVP on the single unit cell by introducing a left wall L, a right wall $R = L + \boldsymbol{e}_1$, down wall D, and an up wall $U = D + \boldsymbol{e}_2$. Imposing periodicity conditions gives us

$$(\boldsymbol{u}, p) \qquad \text{Stokes in } \mathbb{R}^2 \setminus \overline{\Omega_\Lambda} \tag{3.2.7}$$

$$\boldsymbol{u} = \boldsymbol{0}, \quad \boldsymbol{x} \in \partial\Omega_\Lambda \tag{3.2.8}$$

$$\boldsymbol{u}_R - \boldsymbol{u}_L = \boldsymbol{0} \tag{3.2.9}$$

$$\boldsymbol{u}_U - \boldsymbol{u}_D = \boldsymbol{0} \tag{3.2.10}$$

$$T(\boldsymbol{u}, p)_R - T(\boldsymbol{u}, p)_L = p_1\boldsymbol{e}_1 \tag{3.2.11}$$

$$T(\boldsymbol{u}, p)_U - T(\boldsymbol{u}, p)_D = p_2\boldsymbol{e}_2. \tag{3.2.12}$$

By unique continuation from Cauchy data, the above BVP is equivalent to (3.2.1)-(3.2.6). We represent the solution to (3.2.7)-(3.2.12) as

$$\boldsymbol{u} = (\mathcal{S} + \mathcal{D})^{\text{near}}_{\Gamma}\boldsymbol{\tau} + \sum_{m=1}^{M} \phi_m \boldsymbol{\xi}_m \tag{3.2.13}$$

where

$$\left((\mathcal{S}+\mathcal{D})_{\Gamma}^{\text{near}}\boldsymbol{\tau}\right)(\boldsymbol{x}) = \sum_{\boldsymbol{d}\in\mathbb{Z}^2\cap[-1,1]^2}\int_{\Gamma}\left(S(\boldsymbol{x},\boldsymbol{y}+\boldsymbol{d})+D(\boldsymbol{x},\boldsymbol{y}+\boldsymbol{d})\right)\boldsymbol{\tau}(\boldsymbol{x})ds_{\boldsymbol{y}} \tag{3.2.14}$$

and

$$\phi_m(\boldsymbol{x}) = S(\boldsymbol{x},\boldsymbol{y}_m), \quad m = 1,\ldots,M. \tag{3.2.15}$$

The vector $\boldsymbol{y}_m$ represents the mth proxy point. We decided to use the sum of a single-layer and a double-layer, $S+D$, to represent the near interactions. For modeling flow in the exterior of Ω, the $S+D$ representation is preferred over just a single-layer since the self interaction matrix has full rank. This leads to a system with improved conditioning.

The pressure representation corresponding to (3.2.13) is then

$$p = \mathcal{P}_{\Gamma}^{\text{near}}\boldsymbol{\tau} + \sum_{m=1}^{M}\varphi_m\boldsymbol{\xi}_m, \tag{3.2.16}$$

where

$$\left(\mathcal{P}_{\Gamma}^{\text{near}}\boldsymbol{\tau}\right)(\boldsymbol{x}) = \sum_{\boldsymbol{d}\in\mathbb{Z}^2\cap[-1,1]^2}\int_{\Gamma}\left(Q(\boldsymbol{x},\boldsymbol{y}+\boldsymbol{d})+P(\boldsymbol{x},\boldsymbol{y}+\boldsymbol{d})\right)\boldsymbol{\tau}(\boldsymbol{x})ds_{\boldsymbol{y}} \tag{3.2.17}$$

and $\varphi_m(\boldsymbol{x}) = Q(\boldsymbol{x},\boldsymbol{y}_m)$.

Constructing the linear system now is simply a matter of plugging (3.2.13) into (3.2.7)-(3.2.12). Enforcing Dirichlet boundary conditions gives

$$\left(-\frac{1}{2}I + (S+D)_{\Gamma,\Gamma}^{\text{near}}\right)\boldsymbol{\tau} + \sum_{m=1}^{M}\phi_m|_{\Gamma}\boldsymbol{\xi}_m = \boldsymbol{f} \tag{3.2.18}$$

which we summarize as

$$A\boldsymbol{\tau} + B\boldsymbol{\xi} = \boldsymbol{f}. \tag{3.2.19}$$

Imposing periodicity by matching velocity and traction data in (3.2.9)-(3.2.12) gives

$$\sum_{\boldsymbol{d}\in\mathcal{W}_1}\left((S+D)_{R,\Gamma-\boldsymbol{e}_1+\boldsymbol{d}}-(S+D)_{L,\Gamma+\boldsymbol{e}_1+\boldsymbol{d}}\right)\boldsymbol{\tau}+\sum_{j=1}^{M}\left(\phi_j|_R-\phi_j|_L\right)\boldsymbol{\xi}_j=\mathbf{0} \tag{3.2.20}$$

$$\begin{aligned}\sum_{\boldsymbol{d}\in\mathcal{W}_1}&\left((K+T)_{R,\Gamma-\boldsymbol{e}_1+\boldsymbol{d}}-(K+T)_{L,\Gamma+\boldsymbol{e}_1+\boldsymbol{d}}\right)\boldsymbol{\tau}\\&+\sum_{j=1}^{M}\left(T(\phi_j,\varphi_j)|_R-T(\phi_j,\varphi_j)|_L\right)\boldsymbol{\xi}_j=p_1\boldsymbol{e}_1\end{aligned} \tag{3.2.21}$$

$$\sum_{\boldsymbol{d}\in\mathcal{W}_2}\left((S+D)_{U,\Gamma-\boldsymbol{e}_2+\boldsymbol{d}}-(S+D)_{D,\Gamma+\boldsymbol{e}_2+\boldsymbol{d}}\right)\boldsymbol{\tau}+\sum_{j=1}^{M}\left(\phi_j|_U-\phi_j|_D\right)\boldsymbol{\xi}_j=\mathbf{0} \tag{3.2.22}$$

$$\begin{aligned}\sum_{\boldsymbol{d}\in\mathcal{W}_2}&\left((K+T)_{U,\Gamma-\boldsymbol{e}_2+\boldsymbol{d}}-(K+T)_{D,\Gamma+\boldsymbol{e}_2+\boldsymbol{d}}\right)\boldsymbol{\tau}\\&+\sum_{j=1}^{M}\left(T(\phi_j,\varphi_j)|_U-T(\phi_j,\varphi_j)|_D\right)\boldsymbol{\xi}_j=p_2\boldsymbol{e}_2\end{aligned} \tag{3.2.23}$$

where

$$\mathcal{W}_i=\left\{\boldsymbol{d}\in\mathbb{Z}^2\cap[-1,1]^2 \mid \boldsymbol{d}\cdot\boldsymbol{e}_i=0\right\}. \tag{3.2.24}$$

We summarize these conditions as

$$C\boldsymbol{\tau}+Q\boldsymbol{\xi}=\boldsymbol{g}. \tag{3.2.25}$$

This leads to the extended linear system

$$\begin{bmatrix} A & B \\ C & Q \end{bmatrix}\begin{bmatrix} \boldsymbol{\tau} \\ \boldsymbol{\xi} \end{bmatrix}=\begin{bmatrix} \boldsymbol{f} \\ \boldsymbol{g} \end{bmatrix}, \tag{3.2.26}$$

3.3 Schur complement system and its iterative solution

In this section, we show how to modify Q in order to perform the Schur complement on the system in (3.2.26). Simply numerically computing $Q^{+}C$, as was done in the singly periodic case, is not possible since range of C is not spanned by the columns of Q. This is because there are 3 consistency conditions on $\boldsymbol{\tau}$ that must be handled before we can apply Q^+. In the representation for $\boldsymbol{u}$ in (3.2.13), we chose to use a single and a double-layer potential on Γ and its neighboring copies. However, we have to be careful to make sure that $\boldsymbol{\tau}$ satisfies the consistency conditions. For a single-layer, we must have a density with mean zero. That is,

$$\int_\Gamma \boldsymbol{\tau}\, ds = \mathbf{0}. \tag{3.3.1}$$

Otherwise, the infinite sum of periodic copies would diverge. This gives us two conditions. The third comes from the double-layer potential. For that, we require

$$\int_\Gamma \boldsymbol{\tau} \cdot \boldsymbol{n}\, ds = 0, \tag{3.3.2}$$

where $\boldsymbol{n}$ is the outward pointing normal to Γ. The reason for this is that a density that does not satisfy condition (3.3.2) could give a velocity profile that violates conservation of fluid mass. In other words, the double-layer is a source of divergence without the extra condition. This is not periodizable, since our velocity conditions on the side walls require there to be no net flow into or out of the central cell.

To enforce these conditions, we apply a rank-3 modification to Q in order to expand the range of Q in such a way that a solution to the modified ELS will allow us to easily construct a solution to the original ELS. Also, the modified ELS will no longer require the consistency conditions on $\boldsymbol{\tau}$.

Recall that Q maps the coefficient vectors $\boldsymbol{\xi}$ on the ring of proxy points to the velocity and traction differences on the sides of the central cell. The space of velocity and traction differences that can be generated by Q, unfortunately, does not include all of the velocity and traction differences that could be generated by densities on Γ that violate any of the three consistency conditions. In other words, we need to extend the range of Q in order to include these velocity and traction differences. An obvious basis for these "unacceptable" densities would be $\{\boldsymbol{\tau}_1(\boldsymbol{x}) := \boldsymbol{e}_1, \boldsymbol{\tau}_2(\boldsymbol{x}) := \boldsymbol{e}_2, \boldsymbol{\tau}_3(\boldsymbol{x}) := \boldsymbol{n}(\boldsymbol{x})\}$, where $\boldsymbol{x} \in \Gamma$. Let H be a matrix with these densities (scaled appropriately) as columns. Let

$$V := CH, \quad W := BH, \quad Q_V := Q + VR^T, \quad B_W := B + WR^T, \tag{3.3.3}$$

where R is a random matrix with $2M$ rows (B and Q have $2M$ columns) and 3 columns.

Proposition 3.3.1. *Suppose $\boldsymbol{z} \in$ Col$(Q) \oplus$ Col(V), where V is defined in (3.3.3), and that Q has at least 3 more columns than rows. Then, there exists an $\boldsymbol{x} \in \mathbb{R}^{2M}$ such that $\boldsymbol{z} = (Q + VR^T)\boldsymbol{x}$ with probability 1.*

Proof. Since Q has at least 3 more columns than rows, we can guarantee the nullity of Q is at least 3. Suppose $\boldsymbol{z} \in \text{Col}(Q) \oplus \text{Col}(V)$. That is, there exists a $\boldsymbol{q} \in \mathbb{R}^{2M}$ and a $\boldsymbol{v} \in \mathbb{R}^3$ such that $\boldsymbol{z} = Q\boldsymbol{q} + V\boldsymbol{v}$. Since $\text{Null}(Q) \perp \text{Row}(Q)$ and $\text{Null}(Q) \oplus \text{Row}(Q) = \mathbb{R}^{2M}$, we can decompose $\boldsymbol{q}$ into a component that is within the row space and another that is within the null space of Q. That is, $\boldsymbol{q} = \boldsymbol{q}_{\text{row}} + \boldsymbol{q}_{\text{null}}$, where $\boldsymbol{q}_{\text{row}} \in \text{Row}(Q)$ and $\boldsymbol{q}_{\text{null}} \in \text{Null}(Q)$. Since $Q\boldsymbol{q}_{\text{null}} = \mathbf{0}$, we have that $\boldsymbol{z} = Q\boldsymbol{q}_{\text{row}} + V\boldsymbol{v}$. Let $\boldsymbol{x} = \boldsymbol{q}_{\text{row}} + \boldsymbol{w}_{\text{null}}$ where $\boldsymbol{w}_{\text{null}} \in \text{Null}(Q)$ is some vector to be determined. Since R is a random matrix, there is a probability of 1 that its columns have nonzero components within $\text{Null}(Q)$. That is, we can decompose R into $R = R_{\text{row}} + R_{\text{null}}$ where each column of R_{null} is nonzero. It can also be shown that the columns of R_{null} are linearly independent with probability 1. We now have

$$\begin{align}
\boldsymbol{z} &= Q\boldsymbol{q}_{\text{row}} + V\boldsymbol{v} \tag{3.3.4}\\
&= (Q + VR^T)\boldsymbol{x} \tag{3.3.5}\\
&= \left(Q + V(R^T_{\text{row}} + R^T_{\text{null}})\right)(\boldsymbol{q}_{\text{row}} + \boldsymbol{w}_{\text{null}}) \tag{3.3.6}\\
&= Q\boldsymbol{q}_{\text{row}} + VR^T_{\text{row}}\boldsymbol{q}_{\text{row}} + VR^T_{\text{null}}\boldsymbol{w}_{\text{null}}. \tag{3.3.7}
\end{align}$$

The system

$$R^T_{\text{null}}\boldsymbol{w}_{\text{null}} = \boldsymbol{v} - R^T_{\text{row}}\boldsymbol{q}_{\text{row}} \tag{3.3.8}$$

is guaranteed to have a solution in $\text{Col}(R_{\text{null}}) \subseteq \text{Null}(Q)$ since R_{null} has linearly independent columns. □

To obtain a solution to (3.2.26), perform the Schur complement on

$$\begin{bmatrix} A & B_W \\ C & Q_V \end{bmatrix} \begin{bmatrix} \boldsymbol{\zeta} \\ \boldsymbol{\xi} \end{bmatrix} = \begin{bmatrix} \boldsymbol{f} \\ \boldsymbol{g} \end{bmatrix}. \tag{3.3.9}$$

That is, we can obtain $\boldsymbol{\zeta}$ by solving

$$(A - B_W(Q_V^+ C))\boldsymbol{\zeta} = \boldsymbol{f} - B_W(Q_V^+ \boldsymbol{g}) \tag{3.3.10}$$

using GMRES. (We suggest using a QR factorization of Q_V to apply Q_V^+.) The proxy coefficients $\boldsymbol{\xi}$ are then

$$\boldsymbol{\xi} = (Q_V^+ \boldsymbol{g}) - (Q_V^+ C)\boldsymbol{\zeta}. \tag{3.3.11}$$

We can now construct the density $\boldsymbol{\tau}$ for (3.2.26):

$$\boldsymbol{\tau} = \boldsymbol{\zeta} + H(R^T \boldsymbol{\xi}). \tag{3.3.12}$$

Remark 3.3.2. *Numerical experiments show that this approach for iteratively solving the boundary integral equation tends to lose about 3 decimal places in the final answer. Developing an approach that avoids this is a project for future work.*

3.4 Computing the effective conductivity tensor

The periodization scheme presented in this chapter can be easily modified to solve Laplace's equations. In this section we will describe an efficient approach for computing the effective conductivity tensor in the case of Laplace. An analogous approach can be used in the case of Stokes for computing the effective viscosity due to a distribution of inclusions.

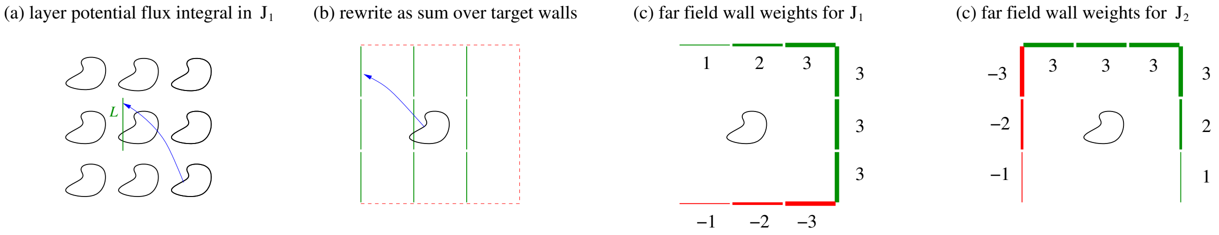

(a) layer potential flux integral in J1
L
(b) rewrite as sum over target walls
(c) far field wall weights for J1
1
2
3
3
3
3
−1
−2
−3
(c) far field wall weights for J2
−3
3
3
3
3
−2
2
−1
1

Figure 3.4.1: Efficient evaluation of fluxes (J_1, J_2) using far field interations alone. (a) Nine terms in L wall integral in J_1 from the 3×3 layer potential image sum in (3.4.7). (b) Re-interpretation as a sum over targets (nine copies of L) for the potential v. The red dotted line shows the closure of the boundary where flux conservation is applied. (c) Resulting weights of the flux integrals of v on nine wall segments; note all are distant from $\partial\Omega$. (d) Similar weights for J_2.

The analogous Neumann BVP for Laplace can be written as

$$\Delta u = 0 \quad \text{in } \mathbb{R}^2 \setminus \overline{\Omega_\Lambda} \tag{3.4.1}$$

$$u_n = 0 \quad \text{on } \partial\Omega \tag{3.4.2}$$

$$u_R - u_L = \varrho_1 \tag{3.4.3}$$

$$u_U - u_D = \varrho_2 \tag{3.4.4}$$

$$u_{nR} - u_{nL} = 0 \tag{3.4.5}$$

$$u_{nU} - u_{nD} = 0 \tag{3.4.6}$$

where we use the notation u_{nL} to represent the normal derivative using the normal on L. (Note this points to the right, as shown in the Fig. 3.1.2.) The representation for u is given by

$$u = \mathcal{S}^{\text{near}}_{\partial\Omega}\tau + \sum_{j=1}^{M} \xi_j \phi_j, \quad \mathcal{S}^{\text{near}}_{\partial\Omega}\tau := \sum_{m,n\in\{-1,0,1\}} \int_{\partial\Omega} G(\boldsymbol{x}, \boldsymbol{y} + m\boldsymbol{e}_1 + n\boldsymbol{e}_2)\tau(\boldsymbol{y}) ds_{\boldsymbol{y}}, \tag{3.4.7}$$

where all functions are defined analogously to the Stokes case. The macroscopic effective conductivity $\kappa \in \mathbb{R}^{2\times 2}$ expresses how the mean flux depends on the driving. Let $\boldsymbol{J} :=$

(J_1, J_2) be the mean flux, with components

$$J_1 := \frac{1}{H}\int_L u_n ds \ , \qquad \text{and} \ \ J_2 := \frac{1}{H}\int_D u_n ds \ , \tag{3.4.8}$$

then the conductivity tensor is defined by

$$\boldsymbol{J} = \kappa \boldsymbol{p} \ . \tag{3.4.9}$$

As is well known, to extract the four elements of κ, it is sufficient to solve two BVPs, one with $p_1 = 1, p_2 = 0$ (from which one may read off $\kappa_{11} = J_1$ and $\kappa_{21} = J_2$), and the other with $p_1 = 0, p_2 = 1$ (and read off $\kappa_{12} = J_1$ and $\kappa_{22} = J_2$). Note that κ is symmetric, hence $\kappa_{12} - \kappa_{21}$ provides an independent gauge of the accuracy of the numerical solution.

For large-scale problems, approximating the integrals (3.4.8) directly by quadrature on the walls L and D is inconvenient, because when inclusions intersect the walls this forces the integral to be broken into intervals. To avoid this, one could deform the integration paths to avoid inclusions, but finding such a smooth path is complicated, and needs many quadrature nodes due to the nearby inclusions. Instead, we propose the following method which pushes all interactions to the far field and thus requires only $m \approx 20$ target nodes per wall.

Proposition 3.4.1. *Let u be represented by* (3.4.7) *and solve the BVP* (3.4.1)–(3.4.6) *in $\mathcal{B}$. Define $v = \mathcal{S}_{\partial\Omega}\tau$. Then the horizontal flux in* (3.4.8) *can be written*

$$J_1 = \frac{1}{H}\Bigg[\sum_{m\in\{-1,0,1\}} (m{+}2)\left(\int_{U+m\boldsymbol{e}_1+\boldsymbol{e}_2} v_n ds - \int_{D+m\boldsymbol{e}_1-\boldsymbol{e}_2} v_n ds\right) + 3\sum_{n\in\{-1,0,1\}}\int_{R+\boldsymbol{e}_1+n\boldsymbol{e}_2} v_n ds + \int_L \sum_{j=1}^{M} \frac{\partial \phi_j}{\partial n}\xi_j ds\Bigg] \ . \tag{3.4.10}$$

The flux integrals involving v are on the *distant* walls of the 3×3 "super-cell", with locations and weights shown in Fig. 3.4.1(c). The final term involves a smooth integrand on the original wall L. A similar far-field formula for J_2 is achieved by reflection through

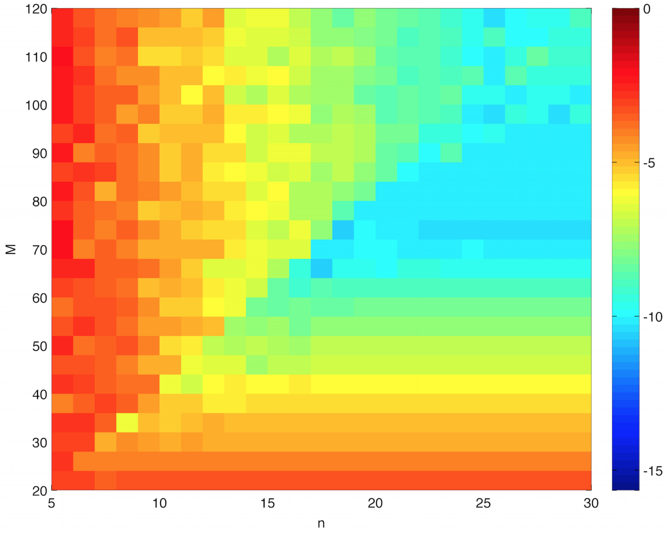


Figure 3.4.2: Logarithm of the errors for the velocity as we vary the number of points on the sides and on the ring of proxy sources for the worm shaped inclusion in Fig. 3.4.7. For each test point, the velocity was computing using 256 points on the inclusion. The boundary integral equation was solved using GMRES.

the line $x_1 = x_2$, with weights shown in Fig. 3.4.1(d).

Proof. Substituting (3.4.7) into J_1 in (3.4.8) involves a 3×3 sum over density sources (Fig. 3.4.1(a)) which by translational invariance we reinterpret as a sum over displaced target copies as in panel (b). These nine copies of L form three continuous vertical walls. Since v is harmonic in $\mathbb{R}^2 \backslash \overline{\Omega}$, then $\int_\Gamma v_n = 0$ for any closed curve Γ that does not enclose nor touch $\partial\Omega$. However, since $u_n = 0$ on $\partial\Omega$ and $u - v$ is harmonic in a neighborhood of Ω, this also holds if Γ encloses $\partial\Omega$. Thus the flux through each length-3 vertical wall is equal to the flux through its closure to the right along the dotted contour shown in panel (b). Summing these three contour closures, with appropriate normal senses, gives the weights in panel (c), i.e. (3.4.10). One may check that the result is unaffected by intersections of $\partial\Omega$ with the original unit cell walls. □

We have tested that this formula matches the naive quadrature of (3.4.8) when $\partial\Omega$ is far from L. For the solution in panel (a) we find $|\kappa_{12} - \kappa_{21}| = 6 \times 10^{-14}$, indicating accuracy of the computed tensor.

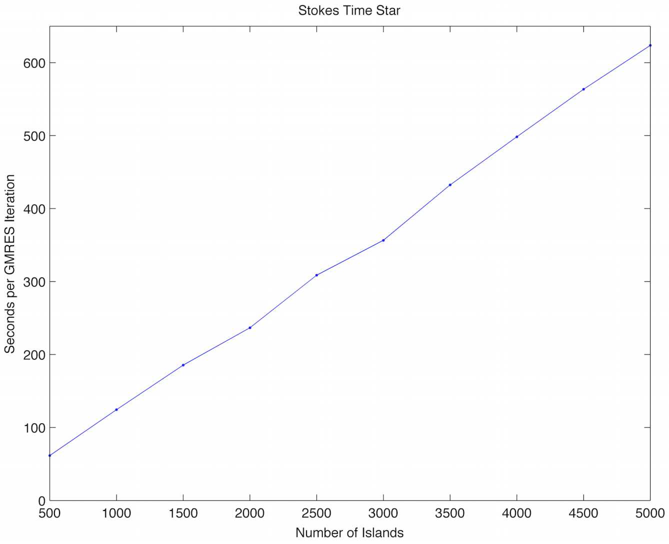


Figure 3.4.3: The average time (in seconds) per GMRES iteration as the number of "star" shaped inclusions is varied. Each dot represents a data point.

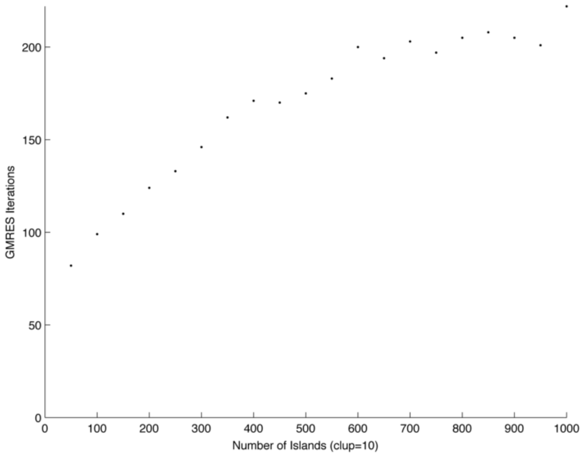


Figure 3.4.4: Number of GMRES iterations to invert the boundary integral equation (GMRES resolution set to 10^{-16} with no preconditioner) as the number of circular inclusions is varied. The closeness parameter cl is defined as cl = (Perimeter of Inclusion)/(Distance to Nearest Inclusion). The $clup$ is the maximum value of cl for all for all of the inclusions. Refer to Fig. 3.4.5 for an example of a distribution of circular islands with $clup = 10$.

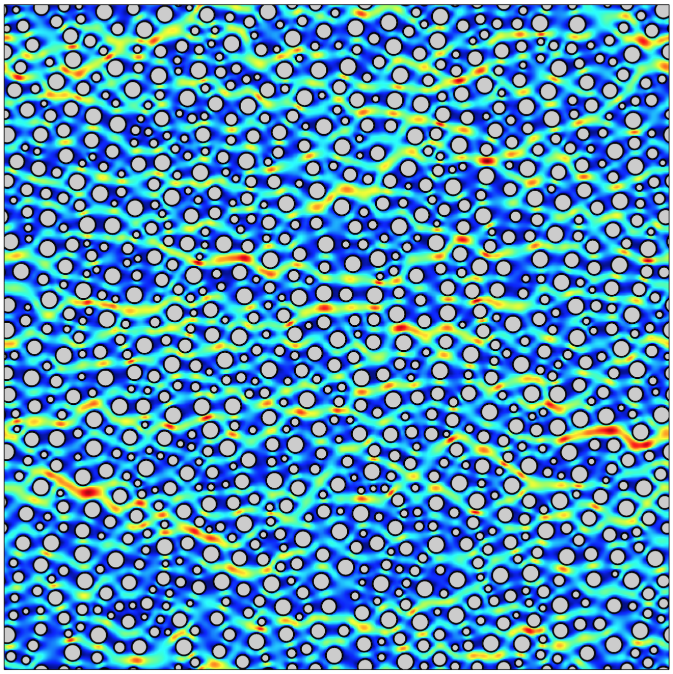


Figure 3.4.5: Pressure driven flow (from left to right) with 1,000 circular shaped inclusions with $clup = 10$. The color indicates the magnitude of the fluid velocity (blue is slow, red is fast).

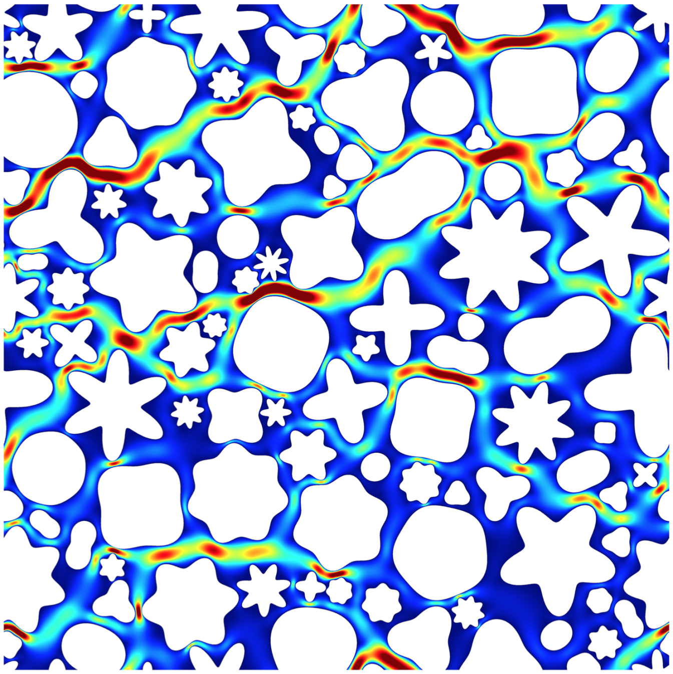


Figure 3.4.6: Pressure driven flow (from left to right) with 100 "star" shaped inclusions. Each inclusion had 1,000 points on its boundary. The fluid velocity is indicated by the background color (blue is slow, red is fast).

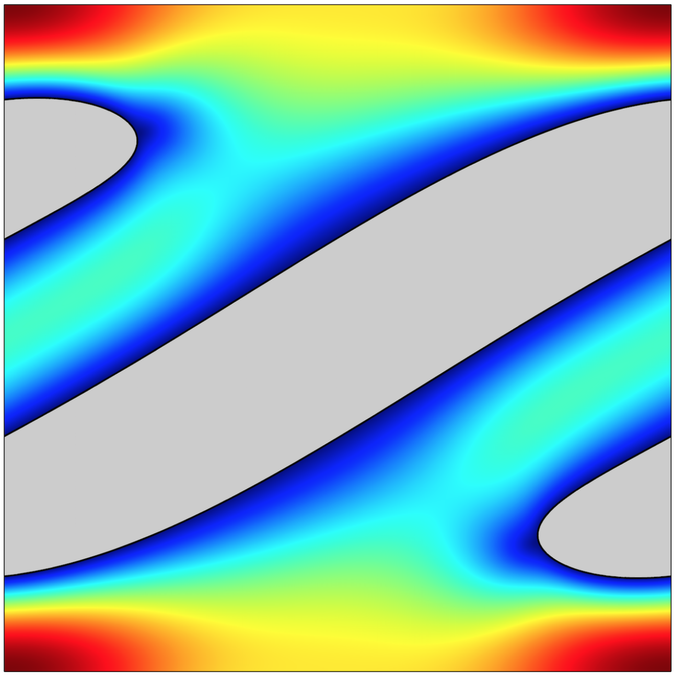


Figure 3.4.7: Pressure driven Stokes flow around a doubly periodic “worm” geometry. The color indicates the speed of the flow (blue is slow, red is fast). The boundary integral equation was solved using 256 points on each worm geometry.

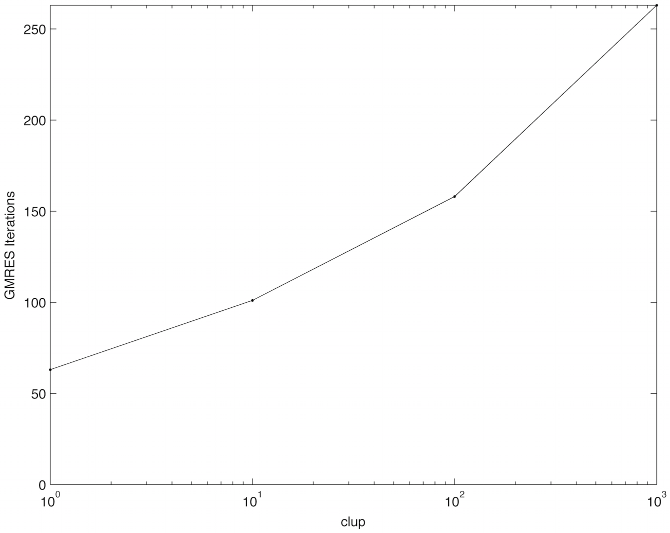


Figure 3.4.8: Number of GMRES iterations as the $clup$ is varied for distributions of circular inclusions. The $clup$ is defined in Fig. 3.4.4. The GMRES tolerance was set to 10^{-16} and no preconditioners were used.

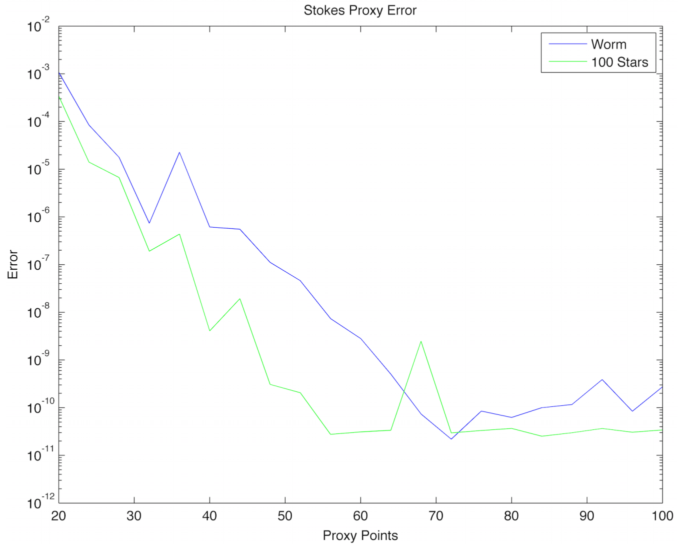


Figure 3.4.9: Errors as the number of side points and proxy points are varied. The number of proxy points was set to $M = 4K$, where K is the number of side points. Notice that the convergence rate appears to be independent of the number of islands. This is because interactions with the sides of the central cell are all far interactions which are smooth and hence, easy to resolve.

CHAPTER 4

Triply Periodic BVPs in Three Dimension

4.1 Introduction

In this chapter, we extend the periodization scheme to triply periodic flows in 3D. We begin by defining the 3D Stokes kernels necessary to impose periodic boundary conditions. We then show that performing the Schur complement can be done using a similar approach to the one used in Chapter 3.

4.2 Triply-periodic Stokes BVPs

We now present the triply periodic Stokes problem in the exterior of the periodic lattice of smooth inclusions. For simplicity, consider a single rigid body Ω in the central unit cell with boundary Γ. Let $\Omega_\Lambda := \{\boldsymbol{x} \in \mathbb{R}^3 : \boldsymbol{x} + l\boldsymbol{e}_1 + m\boldsymbol{e}_2 + n\boldsymbol{e}_3 \in E$ for some $l, m, n \in \mathbb{Z}\}$ represent the infinite lattice of rigid bodies. Let $\boldsymbol{e}_1$, $\boldsymbol{e}_2$, and $\boldsymbol{e}_3$ be vectors defining the lattice in $\mathbb{R}^3$. For simplicity, we work with the unit square case so that $\boldsymbol{e}_1 = (1, 0, 0)$, $\boldsymbol{e}_2 = (0, 1, 0)$, and $\boldsymbol{e}_3 = (0, 0, 1)$. The fluid velocity $\boldsymbol{u}(\boldsymbol{x}) \in \mathbb{R}^3$ and corresponding

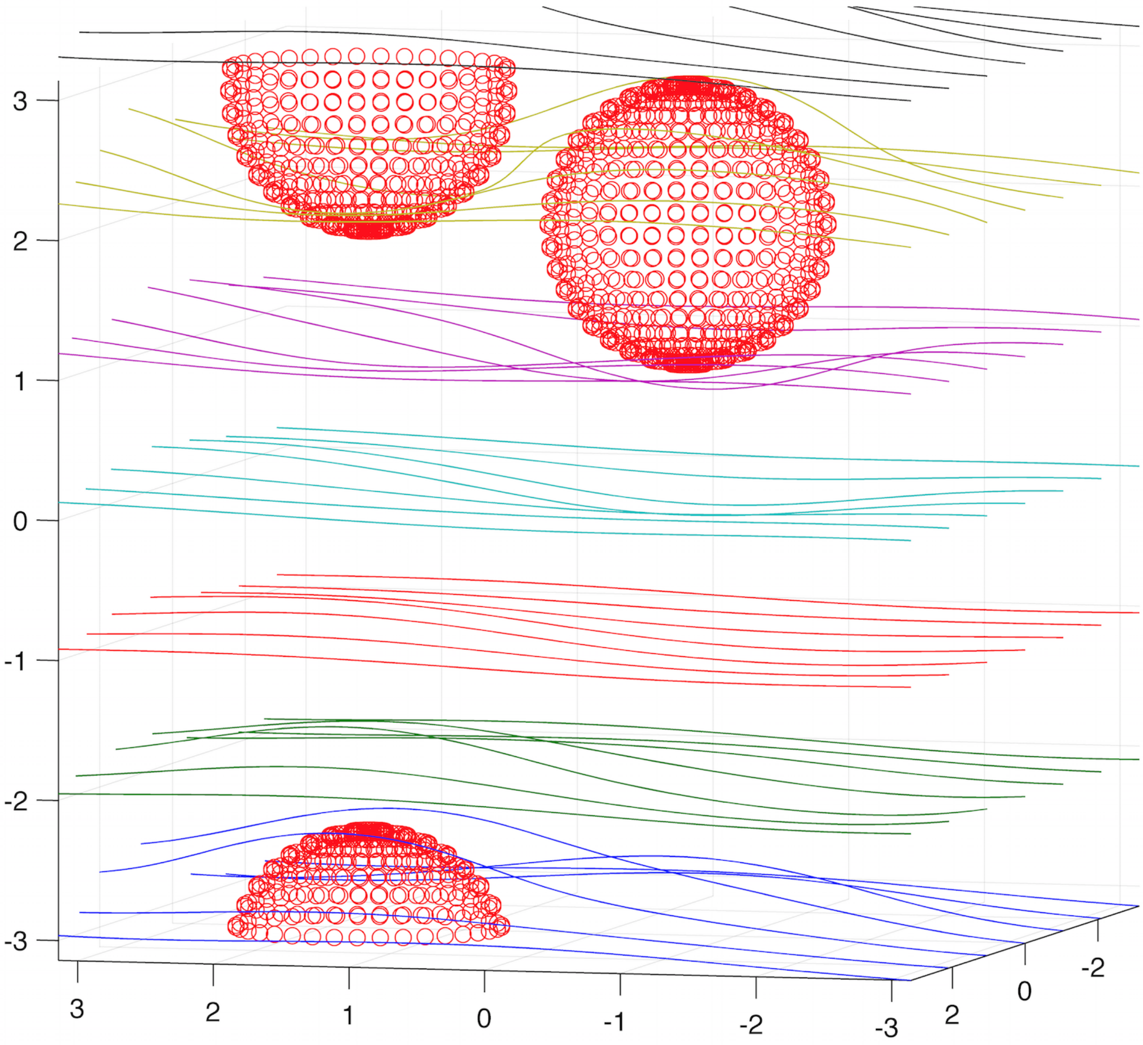


Figure 4.0.1: A plot of fluid streamlines in a triply periodic domain with 2 spheres.

pressure $p(\boldsymbol{x}) \in \mathbb{R}$ solve the BVP

$$-\Delta \boldsymbol{u} + \nabla p = \boldsymbol{0}, \quad \boldsymbol{x} \in \mathbb{R}^3 \setminus \overline{\Omega_\Lambda} \tag{4.2.1}$$

$$\nabla \cdot \boldsymbol{u} = 0, \quad \boldsymbol{x} \in \mathbb{R}^3 \setminus \overline{\Omega_\Lambda} \tag{4.2.2}$$

$$\boldsymbol{u}(\boldsymbol{x}) = \boldsymbol{0}, \quad \boldsymbol{x} \in \partial\Omega_\Lambda \tag{4.2.3}$$

$$\boldsymbol{u}(\boldsymbol{x} + \boldsymbol{e}_i) - \boldsymbol{u}(\boldsymbol{x}) = \boldsymbol{0}, \quad \forall \boldsymbol{x} \in \mathbb{R}^3 \setminus \overline{\Omega_\Lambda}, \quad \text{for} \quad i = 1, 2, 3 \tag{4.2.4}$$

$$p(\boldsymbol{x} + \boldsymbol{e}_i) - p(\boldsymbol{x}) = \varrho_i, \quad \forall \boldsymbol{x} \in \mathbb{R}^3 \setminus \overline{\Omega_\Lambda}, \quad \text{for} \quad i = 1, 2, 3 \tag{4.2.5}$$

where $\boldsymbol{\varrho} = (\varrho_1, \varrho_2, \varrho_3)$ specifies the macroscopic pressure drop. For simplicity, we set the fluid viscosity $\mu = 1$.

4.2.1 Notation and preliminaries

The Stokes single-layer kernel (Stokeslet) from source point $\boldsymbol{y}$ to target point $\boldsymbol{x}$ in $\mathbb{R}^3$ is given by

$$S_{i,j}(\boldsymbol{x}, \boldsymbol{y}) = \frac{1}{8\pi}\left(\frac{\delta_{i,j}}{|\boldsymbol{x} - \boldsymbol{y}|} + \frac{(x_i - y_i)(x_j - y_j)}{|\boldsymbol{x} - \boldsymbol{y}|^3}\right). \tag{4.2.6}$$

The single-layer potential operator $\mathcal{S}_\Gamma$ is given by

$$(\mathcal{S}_\Gamma \boldsymbol{\tau})(\boldsymbol{x})_i = \int_\Gamma S_{i,j}(\boldsymbol{x}, \boldsymbol{y}) \tau_j(\boldsymbol{y}) dS_y, \tag{4.2.7}$$

where $\boldsymbol{\tau} \in \mathbb{R}^3$ is a density on Γ. The associated pressure function is

$$(\mathcal{P}_\Gamma \boldsymbol{\tau})(\boldsymbol{x}) = \int_\Gamma P_j(\boldsymbol{x}, \boldsymbol{y}) \tau_j(\boldsymbol{y}) dS_y, \tag{4.2.8}$$

where

$$P_j(\boldsymbol{x}, \boldsymbol{y}) = \frac{1}{4\pi} \frac{x_j - y_j}{|\boldsymbol{x} - \boldsymbol{y}|^3}. \tag{4.2.9}$$

The traction kernel is given by

$$T_{i,j,k}(\boldsymbol{x},\boldsymbol{y}) = -\frac{3}{4\pi}\frac{(x_i-y_i)(x_j-y_j)(x_k-y_k)}{|\boldsymbol{x}-\boldsymbol{y}|^5}. \tag{4.2.10}$$

The symbol T will be used to represent the single-layer traction kernel and the traction operator. The difference will be clear by context. The boundary integral operator for the traction of the single-layer $T_{\Gamma',\Gamma}$ from source Γ to target surface Γ' is given by

$$(T_{\Gamma',\Gamma}\boldsymbol{\tau})(\boldsymbol{x})_i = T(\mathcal{S}_\Gamma\boldsymbol{\tau},\mathcal{P}_\Gamma\boldsymbol{\tau})(\boldsymbol{x})_i = n_k(\boldsymbol{x})\int_\Gamma T_{i,j,k}(\boldsymbol{x},\boldsymbol{y})\tau_j(\boldsymbol{y})dS_y, \tag{4.2.11}$$

where $\boldsymbol{n}(\boldsymbol{x})$ is the unit outward normal vector at $\boldsymbol{x}\in\Gamma'$.

4.2.2 Forming the extended linear system

Analogous to the 2D case, we separate the contributions from the infinite lattice of rigid bodies to the central unit cell into “near” and “far” contributions. The near contributions will include the central unit cell and 26 of its neighboring blocks. Contributions from all remaining blocks will be considered far interactions and will be accounted for by a sphere of proxy sources (refer to Fig. 4.2.1). That is, the velocity representation within the central unit cell is given by

$$\boldsymbol{u}(\boldsymbol{x}) = (\mathcal{S}_\Gamma^{\text{near}}\boldsymbol{\tau})(\boldsymbol{x}) + \sum_{j=1}^{M}\phi_j(\boldsymbol{x})\boldsymbol{\xi}_j, \tag{4.2.12}$$

where

$$(\mathcal{S}_\Gamma^{\text{near}}\boldsymbol{\tau})(\boldsymbol{x})_i = \sum_{\boldsymbol{d}\in\mathbb{Z}^3\cap[-1,1]^3}\int_\Gamma S_{i,j}(\boldsymbol{x},\boldsymbol{y}+\boldsymbol{d})\tau_j(\boldsymbol{y})dS_y \tag{4.2.13}$$

is a sum over the $3\times3\times3$ near neighbors. The second term contains a linear combination of basis functions ϕ_j, where

$$\phi_j(\boldsymbol{x}) = S(\boldsymbol{x},\boldsymbol{y}_j), \qquad j=1,\dots,M \tag{4.2.14}$$

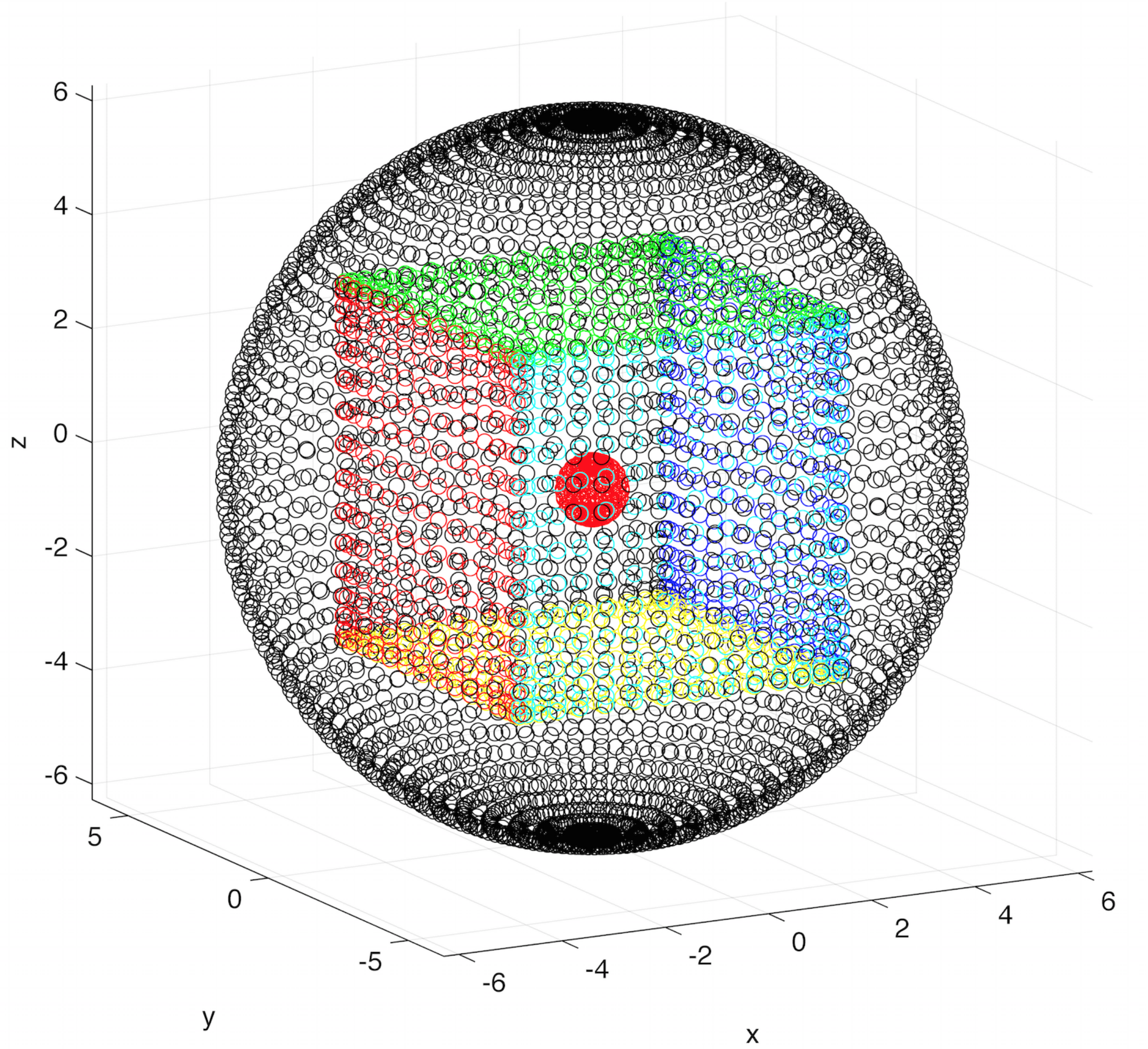


Figure 4.2.1: Setup for the periodization scheme. The central unit cell has sides W_1 (front), W_{-1} (back), W_2 (right), W_{-2} (left), W_3 (up), and W_{-3} (down). The sphere of proxy points $\mathcal{C}$ surrounding the unit cell represents the contribution from the infinite lattice of "far" periodic copies.

and $\{\boldsymbol{y}_j\}_{j=1}^M$ is the set of source points on the sphere $\mathcal{C}$ (refer to Fig. 4.2.1). The pressure representation is then

$$p(\boldsymbol{x}) = \mathcal{P}_\Gamma^{\text{near}}\boldsymbol{\tau} + \sum_{j=1}^M \varphi_j(\boldsymbol{x})\boldsymbol{\xi}_j, \tag{4.2.15}$$

where

$$(\mathcal{P}_\Gamma^{\text{near}}\boldsymbol{\tau})(\boldsymbol{x})_i = \sum_{\boldsymbol{d}\in\mathbb{Z}^3\cap[-1,1]^3} \int_\Gamma P_{i,j}(\boldsymbol{x}, \boldsymbol{y}+\boldsymbol{d})\tau_j(\boldsymbol{y})dS_y \tag{4.2.16}$$

and $\varphi_j(\boldsymbol{x}) = P(\boldsymbol{x}, \boldsymbol{y}_j)$ for $j = 1, \dots, M$.

To form the extended linear system, we require that a solution satisfies Dirichlet boundary conditions on Γ along with appropriate conditions for periodization on the sides of the central unit cell. To be specific, we require that

$$\boldsymbol{u}|_\Gamma = \boldsymbol{f} \tag{4.2.17}$$

$$\boldsymbol{u}|_{W_i} - \boldsymbol{u}|_{W_{-i}} = \boldsymbol{0}, \qquad \text{for} \quad i = 1, 2, 3 \tag{4.2.18}$$

$$T(\boldsymbol{u}, p)|_{W_i} - T(\boldsymbol{u}, p)|_{W_{-i}} = \varrho_i \boldsymbol{e}_i, \qquad \text{for} \quad i = 1, 2, 3 \tag{4.2.19}$$

Clearly, any solution to (4.2.1)-(4.2.5) will satisfy (4.2.17)-(4.2.19). Because of the unique continuation of Cauchy data, a solution to the Stokes equations that also satisfies (4.2.17)-(4.2.19) will solve (4.2.1)-(4.2.5).

In terms of near and far contributions, the Dirichlet boundary conditions imply that

$$S_{\Gamma,\Gamma}^{\text{near}}\boldsymbol{\tau} + \sum_{j=1}^M \phi_j|_\Gamma\, \boldsymbol{\xi}_j = \boldsymbol{f}, \tag{4.2.20}$$

which summarizes as

$$A\boldsymbol{\tau} + B\boldsymbol{\xi} = \boldsymbol{f}. \tag{4.2.21}$$

When enforcing the conditions on the sides of the central unit cell, many terms cancel, as they did in the 2D case, leaving

$$\sum_{\boldsymbol{d}\in\mathcal{W}_i}\left(S_{W_i,\Gamma-\boldsymbol{e}_i+\boldsymbol{d}}-S_{W_{-i},\Gamma+\boldsymbol{e}_i+\boldsymbol{d}}\right)\boldsymbol{\tau}+\sum_{j=1}^{M}\left(\phi_j|_{W_i}-\phi_j|_{W_{-i}}\right)\boldsymbol{\xi}_j=\boldsymbol{0} \tag{4.2.22}$$

$$\sum_{\boldsymbol{d}\in\mathcal{W}_i}\left(T_{W_i,\Gamma-\boldsymbol{e}_i+\boldsymbol{d}}-T_{W_{-i},\Gamma+\boldsymbol{e}_i+\boldsymbol{d}}\right)\boldsymbol{\tau}+\sum_{j=1}^{M}\left(T(\phi_j,\varphi_j)|_{W_i}-T(\phi_j,\varphi_j)|_{W_{-i}}\right)\boldsymbol{\xi}_j=\varrho_i\boldsymbol{e}_i \tag{4.2.23}$$

where

$$\mathcal{W}_i=\left\{\boldsymbol{d}\in\mathbb{Z}^3\cap[-1,1]^3\mid\boldsymbol{d}\cdot\boldsymbol{e}_i=0\right\},\quad\text{for}\quad i=1,2,3. \tag{4.2.24}$$

We summarize these conditions as

$$C\boldsymbol{\tau}+Q\boldsymbol{\xi}=\boldsymbol{g}. \tag{4.2.25}$$

This leads us to the extended linear system

$$\begin{bmatrix} A & B \\ C & Q \end{bmatrix}\begin{bmatrix} \boldsymbol{\tau} \\ \boldsymbol{\xi} \end{bmatrix}=\begin{bmatrix} \boldsymbol{f} \\ \boldsymbol{g} \end{bmatrix}. \tag{4.2.26}$$

We could obtain $\boldsymbol{\tau}$ and $\boldsymbol{\xi}$ directly using a least-squares solve (e.g. using QR) but at a cost of $O(N^3)$. To accelerate the solve, it's preferable to perform the Schur complement on the system so that $\boldsymbol{\tau}$ can be obtained by solving

$$(A-B(Q^+C))\boldsymbol{\tau}=-B(Q^+\boldsymbol{g}). \tag{4.2.27}$$

using an iterative method, such as GMRES. Unfortunately, the column space of C is not contained within the range of Q, as was the case in Chapter 3. In the next section, we show how to modify Q in order to perform the Schur complement.

4.2.3 Schur complement system and its iterative solution.

In this section, we show how to modify Q in order to perform the Schur complement on the system in (4.2.26). The approach is analogous to the doubly periodic case. Since we're only using a single-layer potential on our inclusions, we have

$$\int_\Gamma \boldsymbol{\tau}\, ds = \mathbf{0} \tag{4.2.28}$$

as our only consistency condition. We would like to increase the column space of Q to handle densities that violate (4.2.28). An obvious basis of "unacceptable" densities would be $\{\boldsymbol{\tau}_1(\boldsymbol{x}) := \boldsymbol{e}_1, \boldsymbol{\tau}_1(\boldsymbol{x}) := \boldsymbol{e}_2, \boldsymbol{\tau}_1(\boldsymbol{x}) := \boldsymbol{e}_3\}$, where $\boldsymbol{x} \in \Gamma$. Let H be a matrix with these densities (scaled appropriately) as columns. Let

$$V := CH, \quad W := BH, \quad Q_V := Q + VR^T, \quad B_W := B + WR^T, \tag{4.2.29}$$

where R is a random matrix with $3M$ rows (B and Q have $2M$ columns) and 3 columns.

To obtain a solution to (4.2.26), perform the Schur complement on

$$\begin{bmatrix} A & B_W \\ C & Q_V \end{bmatrix} \begin{bmatrix} \boldsymbol{\zeta} \\ \boldsymbol{\xi} \end{bmatrix} = \begin{bmatrix} \boldsymbol{f} \\ \boldsymbol{g} \end{bmatrix}. \tag{4.2.30}$$

That is, we can obtain $\boldsymbol{\zeta}$ by solving

$$(A - B_W(Q_V^+ C))\boldsymbol{\zeta} = \boldsymbol{f} - B_W(Q_V^+ \boldsymbol{g}) \tag{4.2.31}$$

using GMRES. (We suggest using a QR factorization of Q_V to apply Q_V^+.) The proxy coefficients $\boldsymbol{\xi}$ are then

$$\boldsymbol{\xi} = (Q_V^+ \boldsymbol{g}) - (Q_V^+ C)\boldsymbol{\zeta}. \tag{4.2.32}$$

We can now construct the density $\boldsymbol{\tau}$ for (4.2.26):

$$\boldsymbol{\tau} = \boldsymbol{\zeta} + H(R^T \boldsymbol{\xi}). \tag{4.2.33}$$

Remark 4.2.1. *Numerical experiments show that this approach for iteratively solving the boundary integral equation tends to lose about 3 decimal places in the final answer. Developing an approach that avoids this is a project for future work.*

4.2.4 Numerical method

In this section, we describe our approach for discretizing the integral operators. To represent the surface of the 3D bodies and the densities defined on them, we use spherical harmonics. That is,

$$\boldsymbol{x}(\theta, \phi) = \sum_{n=0}^{p} \sum_{m=-n}^{n} \boldsymbol{x}_n^m Y_n^m(\theta, \phi), \quad \theta \in [0, \pi], \quad \phi \in [0, 2\pi], \tag{4.2.34}$$

where θ is the polar angle, ϕ is the azimuthal angle, p is the order of the expansion, and each $\boldsymbol{x}_n^m \in \mathbb{C}^3$ is a spherical harmonic coefficient. The spherical harmonic of degree n and order m can be expressed in terms of the associated Legendre functions P_n^m as

$$Y_n^m(\theta, \phi) = \sqrt{\frac{2n+1}{4\pi}} \sqrt{\frac{(n-|m|)!}{(n+|m|)!}} P_n^{|m|}(\cos\theta) e^{im\phi}. \tag{4.2.35}$$

For numerical integration, we discretized the polar angle θ using Gauss-Legendre quadrature and the azimuthal angle ϕ using the trapezoidal rule. That is,

$$\left\{\theta_j = \cos^{-1}(t_j),\ j = 0, \ldots, p\right\}, \quad \text{and} \quad \left\{\phi_k = \frac{2\pi k}{2p+2},\ k = 0, \ldots, 2p+1\right\}, \tag{4.2.36}$$

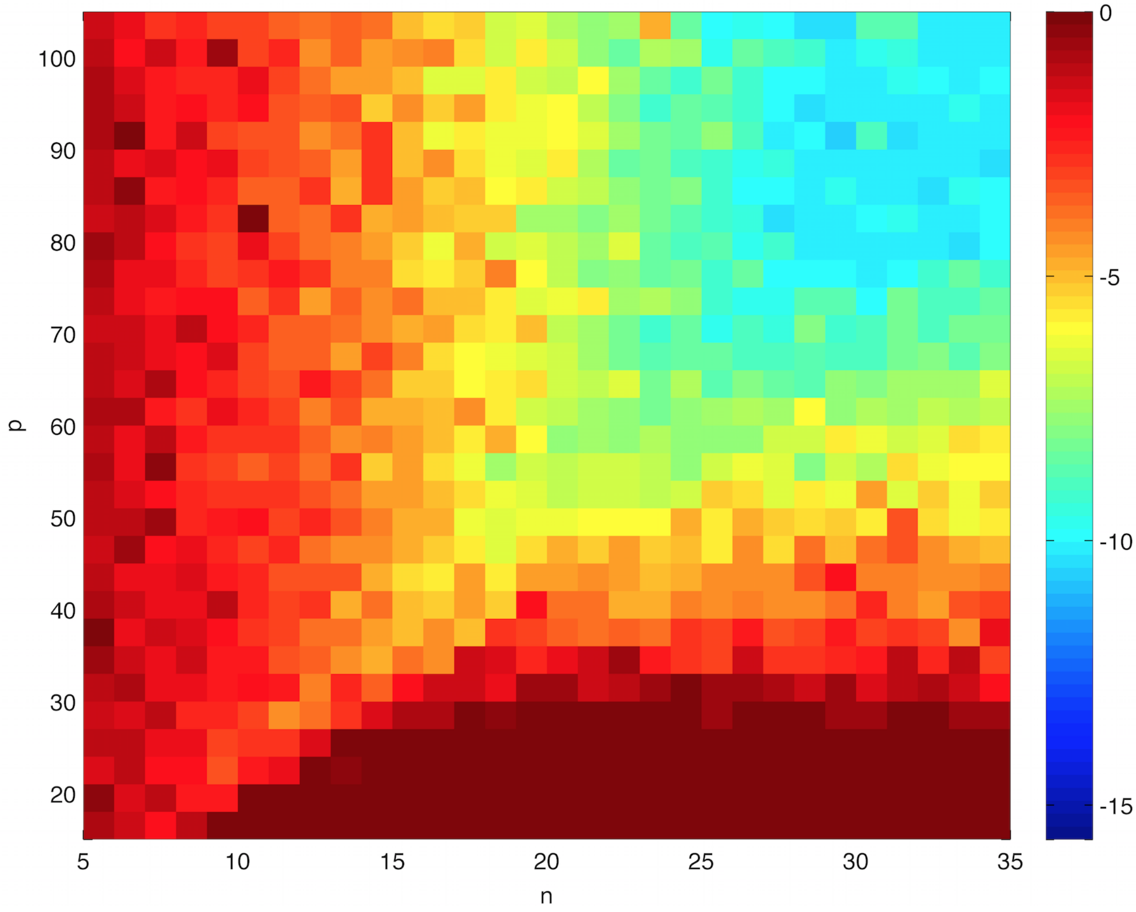


Figure 4.2.2: Logarithm of the errors for the velocity as we vary the number of points on each side, which is given by n^2, and on the sphere of proxy sources for a single spherical inclusion. The number of proxy sources is given by $M = p(2p + 1)$ in accordance with (4.2.36).

where the t_j's are the nodes of the $(p+1)$-point Gauss-Legendre quadrature on $[-1,1]$. This leads us to the following quadrature rule:

$$\int_{\Gamma_i} \boldsymbol{\tau}(\boldsymbol{y}) dS_y = \int_0^{2\pi} \int_0^{\pi} \boldsymbol{\tau}(\boldsymbol{y}(\theta,\phi)) W(\theta,\phi) d\theta d\phi \tag{4.2.37}$$

$$= \sum_{j=0}^{p} \sum_{k=0}^{2p+1} \frac{2\pi\lambda_j}{(2p+2)\sin\theta_j} \boldsymbol{\tau}(\boldsymbol{y}(\theta_j,\phi_k)) W(\theta_j,\phi_k). \tag{4.2.38}$$

The λ_j's are the Gauss-Legendre quadrature weights, and the area element W is given by

$$W = \sqrt{(\boldsymbol{x}_\theta \cdot \boldsymbol{x}_\theta)(\boldsymbol{x}_\phi \cdot \boldsymbol{x}_\phi) - (\boldsymbol{x}_\theta \cdot \boldsymbol{x}_\phi)^2} \tag{4.2.39}$$

To compute derivatives of the coordinate functions $\boldsymbol{x}$, we simply differential each of the basis function. That is,

$$\boldsymbol{x}_\theta(\theta_j,\phi_k) = \sum_{n=0}^{p} \sum_{m=-n}^{n} \boldsymbol{x}_n^m (Y_n^m(\theta_j,\phi_k))_\theta \tag{4.2.40}$$

$$\boldsymbol{x}_\phi(\theta_j,\phi_k) = \sum_{n=0}^{p} \sum_{m=-n}^{n} \boldsymbol{x}_n^m (Y_n^m(\theta_j,\phi_k))_\phi \tag{4.2.41}$$

This quadrature rule is used to compute all "far" off-boundary interactions. However, for interactions that are close to the boundary, the layer potentials become nearly-singular and a close evaluation scheme is needed.

Lastly, the fast spherical grid rotation algorithm introduced in [43] is used to compute the weakly-singular integral operators, such as $\mathcal{K}_{i,i}[\cdot]$. We refer to Theorem 1 of [43] for the quadrature rule, which is based on the idea that if the spherical grid is rotated so that the target point becomes the north (or south) pole, the integrand transforms into a smooth function. This scheme is spectrally accurate and has a compulational complexity of $\mathcal{O}(p^4 \log p)$.

CHAPTER 5

Vesicle Flows Through Singly Periodic Geometries

5.1 Application to particulate flows

In this chapter, we extend the scheme of Veerapaneni et al [24] to periodic flow of vesicle suspensions in rigid pipe-like geometries. We begin with just a single (periodized) vesicle with its boundary γ lying in the periodic unit cell Ω described in Chapter 2. We solve the steady-state flow problem given forces on the vesicle, and then insert this solver into the time-stepping scheme.

5.1.1 Solving for quasi-static fluid flow given the interfacial forces

For simplicity, we consider the case without viscosity contrast. Let $\boldsymbol{x}(s)$ parametrize the vesicle membrane γ according to arc length s. The membrane generates forces on the fluid due to bending $\boldsymbol{f}_b = -\kappa_B \boldsymbol{x}_{ssss}$ and tension $\boldsymbol{f}_\sigma = (\sigma \boldsymbol{x}_s)_s$, where κ_B is the bending modulus and σ is the tension. The total force at each point on γ is then $\boldsymbol{f} = \boldsymbol{f}_b + \boldsymbol{f}_\sigma$. Stress balance and no-slip conditions at the membrane-fluid interface imply the jump conditions $[\![T(\boldsymbol{u}, p)]\!]_\gamma = \boldsymbol{f}$ and $[\![\boldsymbol{u}]\!]_\gamma = 0$ respectively, where $[\![\cdot]\!]_\gamma$ denotes the jump across γ. In the case of an isolated vesicle in free-space and assuming $\boldsymbol{f}$ is known, the representation for the fluid velocity $\boldsymbol{u} = \mathcal{S}_\gamma \boldsymbol{f}$ and the pressure $p = \mathcal{Q}_\gamma \boldsymbol{f}$ satisfies these jump conditions as

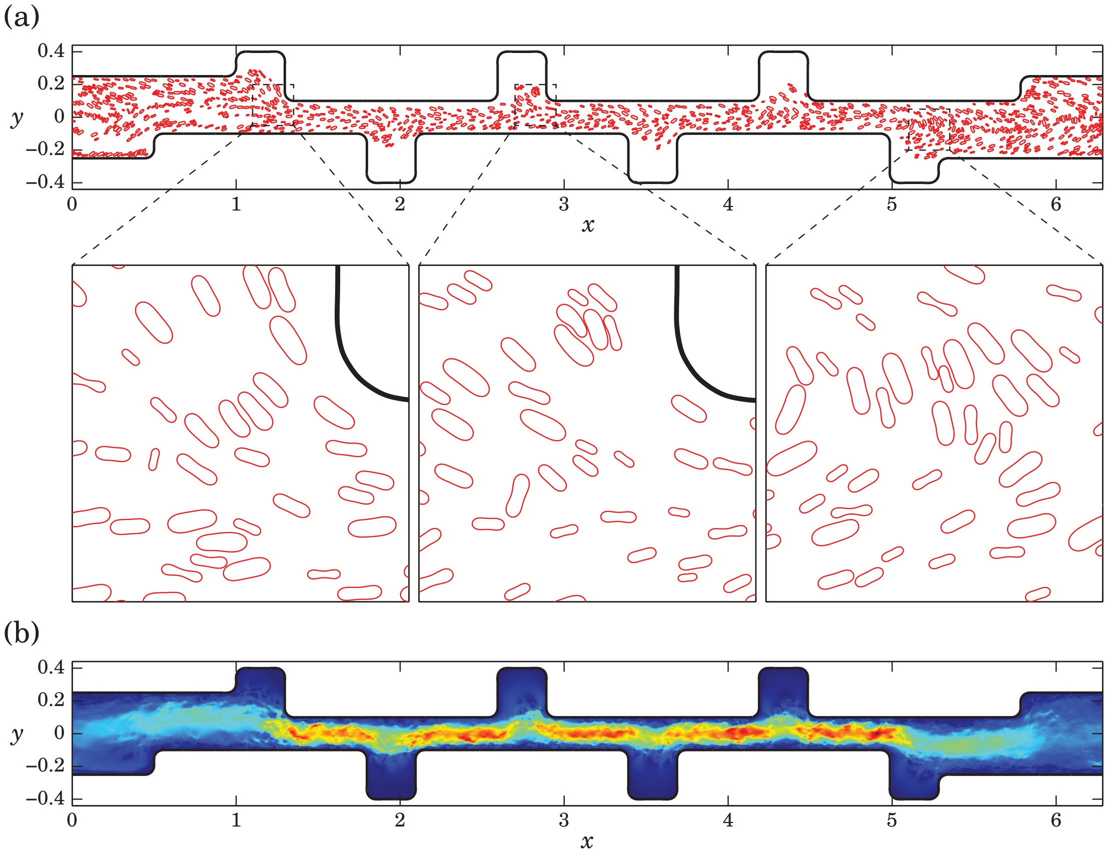


Figure 5.1.1: (a) Snapshot from a simulation of 1,005 vesicles flowing through an arbitrary-shaped periodic channel. We used 64 discretization points per vesicle and 32,000 points each for the top and bottom walls. The vesicle-vesicle and vesicle-channel hydrodynamic interactions are computed via the Stokes FMM, and the fast direct solver in [26] is used to solve the channel BIEs. We used the close evaluation scheme of [1] for the vesicle-to-vesicle and vesicle-to-channel interactions (but not for channel-to-vesicle interactions due to its inapplicability). This simulation took 52 seconds per time step on a laptop with a 2.4 GHz dual-core Intel Core i5 processor and 8 GB of RAM. (b) Plot of the velocity magnitude (red indicates high and blue indicates low) corresponding to the disturbance field generated by the vesicles (obtained by subtracting the pressure-driven "empty pipe" flow from the total velocity field).

well as the Stokes equations in the bulk.

Our goal in this section is, given only the forces $\boldsymbol{f}$ on a periodized vesicle $\gamma+n\boldsymbol{d}$, $n \in \mathbb{Z}$, to solve for the resulting fluid flow $\boldsymbol{u}$ in the periodic channel Ω_Λ which satisfies the no-slip boundary conditions on the channel walls, and which is periodic up to a given pressure drop p_{drive} across a single unit cell. The basic idea is to write $\boldsymbol{u}$ as a sum of the "imposed" flow that the vesicle would generate in an unconstrained fluid, plus a "response" flow due to the confining geometry Ω. (This is the same concept as the incident and scattered wave in scattering theory [44].) A standard approach in the case $p_{\text{drive}} = 0$ might be to use a periodic imposed flow $\sum_{n\in\mathbb{Z}} \mathcal{S}_{\gamma+n\boldsymbol{d}}\boldsymbol{f}$ and for the response to solve the periodic BVP (2.2.16)–(2.2.20) with velocity data given by the negative of the imposed flow measured on U and D. The sum of imposed and response flows then would meet our goal. However, this approach has the disadvantage of relying on periodic Greens functions.

Instead, we use the following representation for the physical flow velocity:

$$\boldsymbol{u} = \mathcal{S}_\gamma^{\text{near}}\boldsymbol{f} + \boldsymbol{u}_{\text{resp}}, \qquad \text{where} \qquad \mathcal{S}_\gamma^{\text{near}}\boldsymbol{f} := \sum_{|n|\le 1} \mathcal{S}_{\gamma+n\boldsymbol{d}}\boldsymbol{f}\ . \tag{5.1.1}$$

The imposed flow (the first term) involves only the vesicle and its immediate neighbor images, as with (2.2.30). We define the associated imposed pressure similarly: $\mathcal{Q}_\gamma^{\text{near}}\boldsymbol{f} := \sum_{|n|\le 1} \mathcal{Q}_{\gamma+n\boldsymbol{d}}\boldsymbol{f}$.

The response $\boldsymbol{u}_{\text{resp}} = \boldsymbol{u}_{\text{resp}}[\boldsymbol{f}, p_{\text{drive}}]$ is then the solution to the single-unit-cell BVP (2.2.24)–(2.2.28) with the following data involving traces of the imposed flow on the walls:

$$\begin{aligned} \boldsymbol{v}_U &= -\mathcal{S}_\gamma^{\text{near}}\boldsymbol{f}|_U && (5.1.2)\\ \boldsymbol{v}_D &= -\mathcal{S}_\gamma^{\text{near}}\boldsymbol{f}|_D && (5.1.3)\\ \boldsymbol{g}_u &= -\mathcal{S}_{\gamma-\boldsymbol{d}}\boldsymbol{f}|_R + \mathcal{S}_{\gamma+\boldsymbol{d}}\boldsymbol{f}|_L && (5.1.4)\\ \boldsymbol{g}_T &= -T(\mathcal{S}_{\gamma-\boldsymbol{d}}\boldsymbol{f}, \mathcal{Q}_{\gamma-\boldsymbol{d}}\boldsymbol{f})|_R + T(\mathcal{S}_{\gamma+\boldsymbol{d}}\boldsymbol{f}, \mathcal{Q}_{\gamma+\boldsymbol{d}}\boldsymbol{f})|_L + p_{\text{drive}}\boldsymbol{n} && (5.1.5) \end{aligned}$$

It is simple to check that (5.1.1) then satisfies no-slip velocity data on U and D, is periodic, and the pressure representation has the required pressure drop (2.2.20). Note that, as in the C block of the previous section, there is cancellation in the discrepancies $\boldsymbol{g}_u$ and $\boldsymbol{g}_T$ so that even when vesicles come close to, or intersect, L or R, there are no near-field terms. Effectively, the L and R walls are "invisible" to the vesicles.

To summarize, the algorithm for solving the static periodic pipe flow problem given vesicle forces $\boldsymbol{f}$ and the driving p_{drive} has three main steps:

i) Evaluate the right-hand side data (5.1.2)–(5.1.5), ie

$$\begin{bmatrix} \boldsymbol{v} \\ \boldsymbol{g} \end{bmatrix} = \begin{bmatrix} -S^{\text{near}}_{U,\gamma} \\ -S^{\text{near}}_{D,\gamma} \\ -S_{R,\gamma-\boldsymbol{d}} + S_{L,\gamma+\boldsymbol{d}} \\ -K_{R,\gamma-\boldsymbol{d}} + K_{L,\gamma+\boldsymbol{d}} \end{bmatrix} \boldsymbol{f} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \boldsymbol{n} \end{bmatrix} p_{\text{drive}} \; ;$$

this will be done with the FMM, except for when the vesicle is close to the wall, in which case a recent spectral close evaluation scheme for the single-layer potential is used [1].

ii) Solve the rectangular linear system (2.2.37) for the density $\boldsymbol{\tau}$ and coefficient vector $\boldsymbol{\xi}$; this is done with the fast direct solver described in [26].

iii) Evaluate $\boldsymbol{u}_{\text{resp}}$ (being the solution to (2.2.24)–(2.2.28)) using the representation (2.2.29); this will again be done via the FMM to get $\boldsymbol{u}_{\text{resp}}|_\gamma$. It is clear that $\boldsymbol{u}_{\text{resp}}$ is linear both in $\boldsymbol{f}$ and p_{drive}.

This three-step procedure to solve for $\boldsymbol{u}_{\text{resp}}[\boldsymbol{f}, p_{\text{drive}}]|_\gamma$ will become one piece of the following evolution scheme.

5.1.2 Time-stepping scheme

So far we have only described a quasi-static solution for $\boldsymbol{u}$ driven by $\boldsymbol{f}$ and p_{drive}. To close the system, we enforce no-slip conditions on the vesicle, $\dot{\boldsymbol{x}} = \boldsymbol{u}|_\gamma$, where $\dot{} = \partial/\partial t$. Substituting (5.1.1) gives the first equation in the integro-differential system of evolution equations for the vesicle dynamics, namely

$$\begin{aligned} \dot{\boldsymbol{x}} &= S^{\text{near}}_{\gamma,\gamma}\boldsymbol{f} + \boldsymbol{u}_{\text{resp}}[\boldsymbol{f}, p_{\text{drive}}]|_\gamma && (5.1.6) \\ 0 &= \boldsymbol{x}_s \cdot \dot{\boldsymbol{x}}_s && (5.1.7) \end{aligned}$$

where $\boldsymbol{f} = -\kappa_B \boldsymbol{x}_{ssss} + (\sigma \boldsymbol{x}_s)_s$. Unlike other particulate systems (e.g., drops), the interfacial tension $\sigma(s,t)$ is not known *a priori* and needs to be determined as part of the solution. It serves as a Lagrange multiplier to enforce the local inextensibility constraint—the second equation (5.1.7) in this system—that the surface divergence of the membrane velocity is zero. This system is driven by p_{drive}, which could vary in time (we take it as constant in our experiments).

The governing equations (5.1.6)–(5.1.7) are numerically stiff owing to the presence of high-order spatial derivatives in the bending force. As shown in [24], explicit time-stepping schemes, such as the forward Euler method, suffer from a third-order constraint on the time step size, rendering them prohibitively expensive for simulating vesicle suspensions. Therefore, we use the semi-implicit scheme formulated in [24] with a few modifications to improve the overall numerical accuracy and stability. Given a time step size Δt and the membrane position and tension at the kth time step, $(\boldsymbol{x}^k, \sigma^k)$, we evolve to $(\boldsymbol{x}^{k+1}, \sigma^{k+1})$ by using a first-order semi-implicit time-stepping scheme on (5.1.6)-(5.1.7). For implementational convenience, however, we treat the discretized membrane velocity, denoted with a slight abuse of notation by $\mathbf{u} = (\boldsymbol{x}^{k+1} - \boldsymbol{x}^k)/\Delta t$, as the unknown instead of $\boldsymbol{x}^{k+1}$. The scheme, then, is given by

$$
\begin{aligned}
\mathbf{u} - \; S^{\text{near}}_{\gamma,\gamma}[-\Delta t \kappa_B \mathbf{u}_{ssss} + (\sigma^{k+1} \boldsymbol{x}^k_s)_s] =& S^{\text{near}}_{\gamma,\gamma}[-\kappa_B \boldsymbol{x}^k_{ssss}] \\
& + \boldsymbol{u}_{\text{resp}}[-\kappa_B \boldsymbol{x}^k_{ssss} + (\sigma^k \boldsymbol{x}^k_s)_s, p_{\text{drive}}]|_\gamma \\
\boldsymbol{x}^k_s \cdot \mathbf{u}_s =& 0.
\end{aligned} \tag{5.1.8}
$$

Since the bending force is a nonlinear function of the membrane position, the standard principle of semi-implicit schemes—to treat the terms with highest-order spatial derivatives implicitly [45]—has been applied to the particular linearization. The tension is treated implicitly and the vesicle-channel interaction, explicitly[1]. The single-layer operator $S^{\text{near}}_{\gamma,\gamma}[\cdot]$ as well as the differential operator $(\cdot)_s$ are constructed using $\boldsymbol{x}^k$. In summary, we solve the following linear system for the unknowns $(\mathbf{u}, \sigma^{k+1})$:

$$
\begin{bmatrix} I + \Delta t \kappa_B S^{\text{near}}_{\gamma,\gamma} \partial_{ssss} & -S^{\text{near}}_{\gamma,\gamma} \partial_s (\boldsymbol{x}^k_s \cdot) \\ \boldsymbol{x}^k_s \cdot \partial_s & 0 \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \sigma^{k+1} \end{bmatrix} = \begin{bmatrix} -\kappa_B S^{\text{near}}_{\gamma,\gamma} \boldsymbol{x}^k_{ssss} + \boldsymbol{u}_{\text{resp}}|_\gamma \\ 0 \end{bmatrix} \tag{5.1.9}
$$

with given $(\boldsymbol{x}^k, \sigma^k)$ and then update the membrane positions as $\boldsymbol{x}^{k+1} = \boldsymbol{x}^k + \Delta t \mathbf{u}$. The operators are discretized using a spectrally-accurate Nyström method (with periodic Kress corrections for the log singularity [39, Sec. 12.3]) for the single-layer operator, and a standard periodic spectral scheme for the differentiation operators. The resulting discrete linear system is solved via GMRES, with all distant interactions applied using the Stokes FMM (e.g., see Appendix D of [24]).

On the initial time step, we generally set $\sigma_0 = 0$. One could obtain an improved initial guess for σ_0 by setting $\sigma_0 = 0$ and solving (5.1.8) and (5.1.8) for σ with $\Delta t = 0$. This could be thought of as the first iteration of a Picard iteration for σ. To enable long-time simulations, we incorporate three supplementary steps in our time-stepping scheme. First,

[1] When the vesicle is located very close to the channel, say $\mathcal{O}(h)$ away where h is the lowest distance between spatial grid points on the vesicle, a semi-implicit treatment of $\boldsymbol{u}_{\text{resp}}$ would be more efficient since the interaction force also induces numerical stiffness in this scenario. Such a scheme, however, requires non-trivial modifications to the fast direct solver [26]. Therefore, we postponed this exercise to future work.

we modify the constraint equation (5.1.8) as[2]

$$\boldsymbol{x}_s^k \cdot \mathbf{u}_s = \frac{L_0 - L_k}{\Delta t L_k}, \tag{5.1.10}$$

where L_k represents the perimeter of the vesicle at the kth time step. We will refer to this as the "arc length correction" (ALC). We prove in Section 5.2.1 that without the ALC, the perimeter of a vesicle will increase monotonically with the number of time steps (total error still scales as $\mathcal{O}(\Delta t)$). This would mean that the vesicle's reduced area can become very low when a large number of time steps are taken. Consequently, the simulated dynamics may correspond to a totally different system than what was originally intended (e.g., a tank-treading vesicle in shear flow might tumble if the reduced area is lowered enough). Executing the ALC at every time step, on the other hand, guarantees a $\mathcal{O}(\Delta t^2)$ convergence rate, but, more importantly, the error is independent of the number of time steps for a fixed Δt (see Theorem 5.2.2). Second, we correct the error incurred in the enclosed area of the vesicle after every time step by solving a quadratic equation in one variable (see Section 5.2.2). Finally, we reparameterize γ at every time step so that spatial discretization points are located *approximately* equal arc lengths apart (see Section 5.2.3).

Suspension flow. Although we have presented the case with a single (periodized) vesicle γ, the above scheme carries over naturally to multiple vesicles. Since such an extension has been described previously in other contexts (e.g., see [24] for free-space and [47] for constrained geometry problems) and does not modify in any way our periodization scheme, we only highlight the main steps here. First, the single-layer potential in the representation (5.1.1) is replaced with a sum of such potentials over all of the vesicles. The equations for the imposed flow data on the walls (5.1.2)–(5.1.5) are then modified accordingly. In discretizing the evolution equation for each individual vesicle, the bending force in the self-interaction term is treated semi-implicitly, similarly to (5.1.8)–(5.1.8). However, the

[2] The main idea here is similar in spirit to the correction formula applied in [46], but in our scheme, we do not introduce any penalty parameters and also rigorously prove its convergence rate (Section 5.2.1).

Algorithm 1 Main (single vesicle)

```
Step 1:   Compress $A^{-1}$ using the fast direct solver [26]
for $k = 0 : N_{\Delta t} - 1$ do
    Step 2:   Compute bending and tension forces ($\boldsymbol{f}_\sigma = \mathbf{0}$ if $k = 0$)
              $\boldsymbol{f}_b = -\kappa_B \boldsymbol{x}^k_{ssss}, \quad \boldsymbol{f}_\sigma = \left(\sigma^k \boldsymbol{x}^k_s\right)_s, \quad \boldsymbol{f} = \boldsymbol{f}_b + \boldsymbol{f}_\sigma \qquad (5.1.11)$
    Step 3:   Compute vesicle-wall and vesicle-side interactions
              (step (i) from Section 5.1.1)
              $\boldsymbol{v}_U = -\mathcal{S}^{\text{near}}_\gamma \boldsymbol{f}\big|_U \qquad (5.1.12)$
              $\boldsymbol{v}_D = -\mathcal{S}^{\text{near}}_\gamma \boldsymbol{f}\big|_D \qquad (5.1.13)$
              $\boldsymbol{g}_u = -\mathcal{S}_{\gamma-\boldsymbol{d}} \boldsymbol{f}|_R + \mathcal{S}_{\gamma+\boldsymbol{d}} \boldsymbol{f}|_L \qquad (5.1.14)$
              $\boldsymbol{g}_T = -T\left(\mathcal{S}_{\gamma-\boldsymbol{d}} \boldsymbol{f}, \mathcal{Q}_{\gamma-\boldsymbol{d}} \boldsymbol{f}\right)|_R + T\left(\mathcal{S}_{\gamma+\boldsymbol{d}} \boldsymbol{f}, \mathcal{Q}_{\gamma+\boldsymbol{d}} \boldsymbol{f}\right)|_L + p_{\text{drive}} \boldsymbol{n} \qquad (5.1.15)$
    Step 4:   Solve for $\boldsymbol{\tau}$, $\boldsymbol{\xi}$ using the fast direct solver (step (ii) from Section 5.1.1)
    Step 5:   Compute wall-vesicle and proxy-vesicle interactions
              (step (iii) from Section 5.1.1)
              $\boldsymbol{u}_{\text{resp}} = \mathcal{D}^{\text{near}}_\Gamma \boldsymbol{\tau}|_\gamma + \sum_{m=1}^{M} \boldsymbol{c}_m \phi_m|_\gamma \qquad (5.1.16)$
    Step 6:   Solve (5.1.9) for $\boldsymbol{u}$ and $\sigma^{k+1}$ using GMRES with the constraint equation
              modified as (5.1.10).
    Step 7:   Set $\boldsymbol{x}^{k+1} = \boldsymbol{x}^k + \Delta t\, \boldsymbol{u}$
    Step 8:   Apply area correction to $\boldsymbol{x}^{k+1}$ (Section 5.2.2)
    Step 9:   Apply the reparameterization scheme on $\boldsymbol{x}^{k+1}$
              (Algorithm 2 of Section 5.2.3)
end for
```

bending forces in the vesicle-vesicle interaction terms can either be treated semi-implicitly, resulting in a dense linear system, or explicitly, resulting in a block tri-diagonal system. While the latter scheme has a marginally smaller computational cost, the former scheme has better stability properties in general since vesicle-vesicle interactions can induce stiffness into the evolution equations when they are located close to each other. In our implementation, we treat all vesicle interactions semi-implicitly. We use a recent single-layer close evaluation scheme [1] to compute the nearby vesicle interactions, whereas for distant interactions, we use the FMM.

5.2 Long-time simulations

5.2.1 Arc length correction

In this section, we present a derivation of the arc length correction formula. The arc length correction is useful for long-time vesicle simulations where the accumulation of errors in the arc length may become significant, leading to elongated vesicles. By placing a small correction term on the right-hand side of the inextensiblity condition, we prevent this accumulation while preserving the membrane's original length with second-order asymptotic convergence.

We begin by understanding how errors accumulate. Let $\boldsymbol{x}^k(\alpha)$, where $\alpha \in [0, 2\pi)$, be a parameterization of the membrane γ_k. The cumulative error on the kth time step, denoted by $\mathcal{E}_k$, is given by

$$\mathcal{E}_k = \int_{\gamma_k} ds - \int_{\gamma_0} ds. \tag{5.2.1}$$

To find $\mathcal{E}_{k+1}$, we will use the identity

$$\begin{aligned}
\|\boldsymbol{x}_\alpha^{k+1}\|_2 &= \|\boldsymbol{x}_\alpha^k + \Delta t \boldsymbol{u}_\alpha\|_2 &(5.2.2)\\
&= \left(\left(x_\alpha^k + \Delta t u_\alpha\right)^2 + \left(y_\alpha^k + \Delta t v_\alpha\right)^2\right)^{\frac{1}{2}} &(5.2.3)\\
&= \left(\left((x_\alpha^k)^2 + (y_\alpha^k)^2\right) + 2\Delta t \left(u_\alpha x_\alpha^k + v_\alpha y_\alpha^k\right) + \Delta t^2 \left((u_\alpha)^2 + (v_\alpha)^2\right)\right)^{\frac{1}{2}} &(5.2.4)\\
&= \left(1 + 2\Delta t(\boldsymbol{x}_s^k \cdot \boldsymbol{u}_s) + \Delta t^2 \|\boldsymbol{u}_s\|_2^2\right)^{\frac{1}{2}} \|\boldsymbol{x}_\alpha^k\|_2, &(5.2.5)
\end{aligned}$$

where $\boldsymbol{x}^k = (x^k, y^k)$ and $\boldsymbol{u} = (u, v)$. The error is then

$$\mathcal{E}_{k+1} = \int_{\gamma^k} \left(1 + 2\Delta t(\boldsymbol{x}_s^k \cdot \boldsymbol{u}_s) + \Delta t^2 \|\boldsymbol{u}_s\|_2^2\right)^{\frac{1}{2}} ds - \int_{\gamma_0} ds. \tag{5.2.6}$$

Observe that setting $\boldsymbol{x}_s^k \cdot \boldsymbol{u}_s = 0$ will cause the arc length to increase monotonically since

$\|\boldsymbol{u}_s\|_2^2 \geq 0$. We now perform a Taylor expansion around $\Delta t = 0$ to get

$$\mathcal{E}_{k+1} = \mathcal{E}_k + \Delta t \int_{\gamma^k} \boldsymbol{x}_s^k \cdot \boldsymbol{u}_s \, ds + \frac{1}{2}\Delta t^2 \int_{\gamma^k} \|\boldsymbol{u}_s\|_2^2 \, ds - \frac{1}{2}\Delta t^2 \int_{\gamma^k} (\boldsymbol{x}_s^k \cdot \boldsymbol{u}_s)^2 \, ds + \mathcal{O}(\Delta t^3). \tag{5.2.7}$$

If we set $\boldsymbol{x}_s^k \cdot \boldsymbol{u}_s = 0$, we see that with each time step, the arc length incurs an error on the order of Δt^2. If n is the total number of time steps, we would then expect the cumulative error to scale like $\mathcal{E}_n = \mathcal{O}(n\Delta t^2)$. This poses a problem for long-time simulations, where n is very large. To address this, we need to prevent the $\mathcal{E}_k$ term in (5.2.7) from causing any significant accumulation. A simple remedy is to let

$$\boldsymbol{x}_s^k \cdot \boldsymbol{u}_s = -\frac{\mathcal{E}_k}{\Delta t \int_{\gamma^k} ds} = \frac{\int_{\gamma^0} ds - \int_{\gamma^k} ds}{\Delta t \int_{\gamma^k} ds}. \tag{5.2.8}$$

Each time step, we are still incurring an error on the order of Δt^2. However, we propose that the cumulative error now scales as $\mathcal{E}_n = \mathcal{O}(\Delta t^2)$ and that there exists an upper bound for the relative error that scales as Δt^2, independent of n. We only require that $\|\boldsymbol{u}_s\|_2$ be bounded.

To begin, let $L_k = \int_{\gamma^k} ds$. Using (5.2.1), (5.2.6), and (5.2.8), we find that

$$L_{k+1} = \int_{\gamma^k} \left(1 + 2\frac{L_0 - L_k}{L_k} + \Delta t^2 \|\boldsymbol{u}_s\|_2^2\right)^{\frac{1}{2}} ds. \tag{5.2.9}$$

Lemma 5.2.1. *Suppose there exists a constant C such that $\|\boldsymbol{u}_s\|_2 \leq C$ for all time. Then,*

$$1 - \frac{\Delta t^2 C^2}{2 - \Delta t^2 C^2} \leq \frac{L_k}{L_0} \leq 1 + \frac{\Delta t^2 C^2}{2 - \Delta t^2 C^2} \tag{5.2.10}$$

for all $k \geq 0$ when Δt is sufficiently small.

Proof. Clearly, (5.2.10) holds for the base case $L_k = L_0$. Assume the induction hypothesis.

We will use the fact that

$$\frac{L_k}{L_0}\left(1+2\frac{L_0-L_k}{L_k}\right)^{\frac{1}{2}} \leq \frac{L_{k+1}}{L_0} \leq \frac{L_k}{L_0}\left(1+2\frac{L_0-L_k}{L_k}+\Delta t^2C^2\right)^{\frac{1}{2}} \tag{5.2.11}$$

to place the desired bounds on L_{k+1}/L_0.

We begin by showing

$$1-\frac{\Delta t^2C^2}{2-\Delta t^2C^2} \leq \frac{L_k}{L_0}\left(1+2\frac{L_0-L_k}{L_k}\right)^{\frac{1}{2}}. \tag{5.2.12}$$

For simplicity, let $z = L_k/L_0$ and $r = \Delta t^2C^2$. We need to show that

$$1-\frac{r}{2-r} \leq z\left(\frac{2}{z}-1\right)^{\frac{1}{2}}, \quad \text{for} \quad z \in I_r = \left[1-\frac{r}{2-r}, 1+\frac{r}{2-r}\right]. \tag{5.2.13}$$

The right-hand side of the inequality in (5.2.13) is the top half of a circle centered about the point $(1, 0)$ with radius 1. The minimum occurs on both endpoints of I_r and is given by

$$\min_{z\in I_r} z\left(\frac{2}{z}-1\right)^{\frac{1}{2}} = \frac{2}{2-r}\left(1-r\right)^{\frac{1}{2}}. \tag{5.2.14}$$

We find that (5.2.13) holds as long as $r \leq 1$. Therefore, (5.2.12) holds as long as $\Delta t \leq 1/C$.

All that remains is to show that

$$\frac{L_k}{L_0}\left(1+2\frac{L_0-L_k}{L_k}+\Delta t^2C^2\right)^{\frac{1}{2}} \leq 1+\frac{\Delta t^2C^2}{2-\Delta t^2C^2}, \tag{5.2.15}$$

which is equivalent to

$$z\left(\frac{2}{z}-1+r\right)^{\frac{1}{2}} \leq 1+\frac{r}{2-r}, \quad \text{for} \quad z \in I_r. \tag{5.2.16}$$

The left-hand side of the inequality in (5.2.16) is the top half of an ellipse centered about

Table 5.1: Convergence analysis for the arc length and area corrections. To perform the analysis, we evolved a vesicle using time step $\Delta t = 0.01/M$ until $t = 0.1$. We report the relative errors in the arc length L, area A, and position $\boldsymbol{x}$ with and without the corrections. The subscript "c" indicates that both the area and arc length corrections were used, and the subscript "nc" indicates that no corrections were used. The accepted values are labeled with subscript "acc". We now observe second-order asymptotic convergence for the arc length, which is in agreement with Theorem 5.2.2.

M	$\left\lvert\frac{L_c - L_{acc}}{L_{acc}}\right\rvert$	$\left\lvert\frac{A_c - A_{acc}}{A_{acc}}\right\rvert$	$\left\lVert\frac{\boldsymbol{x}_c - \boldsymbol{x}_{acc}}{L_{acc}}\right\rVert_\infty$	$\left\lvert\frac{L_{nc} - L_{acc}}{L_{acc}}\right\rvert$	$\left\lvert\frac{A_{nc} - A_{acc}}{A_{acc}}\right\rvert$	$\left\lVert\frac{\boldsymbol{x}_{nc} - \boldsymbol{x}_{acc}}{L_{acc}}\right\rVert_\infty$
1	8.65×10^{-5}	1.21×10^{-10}	3.02×10^{-4}	1.15×10^{-3}	3.12×10^{-4}	4.78×10^{-4}
2	2.09×10^{-5}	7.16×10^{-12}	1.55×10^{-4}	5.77×10^{-4}	1.57×10^{-4}	2.38×10^{-4}
4	5.15×10^{-6}	4.35×10^{-13}	7.80×10^{-5}	2.90×10^{-4}	7.91×10^{-5}	1.17×10^{-4}
8	1.28×10^{-6}	2.72×10^{-14}	3.90×10^{-5}	1.45×10^{-4}	3.97×10^{-5}	5.68×10^{-5}
16	3.18×10^{-7}	3.09×10^{-15}	1.93×10^{-5}	7.26×10^{-5}	1.99×10^{-5}	2.65×10^{-5}

$(\frac{1}{1-r}, 0)$. The maximum occurs on the right endpoint of I_r and is given by

$$\max_{z \in I_r} z \left(\frac{2}{z} - 1 + r \right)^{\frac{1}{2}} = 1 + \frac{r}{2-r}. \tag{5.2.17}$$

Therefore, (5.2.16) holds, which completes the proof of Lemma 5.2.1. □

Theorem 5.2.2. *Suppose there exists a constant C such that $\|\boldsymbol{u}_s\|_2 \leq C$ for all time. Then, the condition*

$$\boldsymbol{x}_s^k \cdot \boldsymbol{u}_s = \frac{L_0 - L_k}{\Delta t L_k}, \tag{5.2.18}$$

preserves arc length with relative error

$$\left| \frac{L_{k+1} - L_0}{L_0} \right| \leq \frac{\Delta t^2 C^2}{2 - \Delta t^2 C^2} = \frac{\Delta t^2 C^2}{2} + \mathcal{O}(\Delta t^4) \tag{5.2.19}$$

for all $k \geq 0$ when $\Delta t \leq 1/C$.

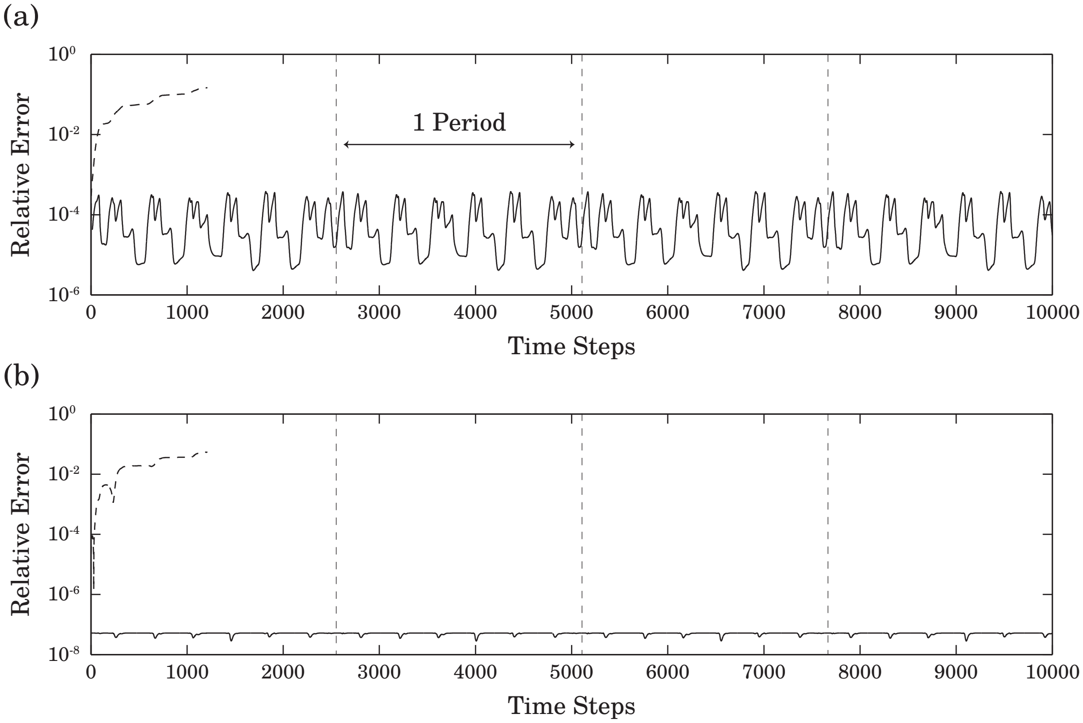


Figure 5.2.1: Errors without corrections $(--)$ and with both arc length and area corrections $(-)$. The relative error in arc length (a) and area (b) for a single vesicle flowing through the serpentine channel. After around $\sim 1,200$ time steps, the simulation breaks down due to numerical instabilities when no corrections are employed. In the simulation using both corrections, the vesicle successfully passed through the channel several times without incident. In both simulations, the time step was set to $\Delta t = 0.01$.

5.2.2 Area correction

Although the fluid incompressibility condition is satisfied exactly by the single-layer kernel, errors in the area of vesicles (owing to discretization) accumulate over time and have a compounding effect in the case of long-time simulations. In this section, we discuss a simple and efficient method to correct the area errors whenever the vesicle shapes are updated i.e., at every time step. Let $\boldsymbol{x}^k(\alpha) = (x^k(\alpha), y^k(\alpha))$ with $\alpha \in [0, 2\pi)$ represent the position of a vesicle's membrane on the kth time step. The initial area is given by

$$A_0 = \int_0^{2\pi} x^0 y^0_\alpha \, d\alpha. \tag{5.2.20}$$

To correct $\boldsymbol{x}^k$, we simply add a normal vector $\boldsymbol{n} = (y^k_\alpha, -x^k_\alpha)$ scaled by a small unknown constant c, which is computed by requiring that the area enclosed by $\boldsymbol{x}^k + c\boldsymbol{n}$ equals A_0. That is,

$$\int_0^{2\pi} \left(x^k + c y^k_\alpha\right) \left(y^k - c x^k_\alpha\right)_\alpha \, d\alpha = A_0. \tag{5.2.21}$$

Expanding the integrand gives

$$\int_0^{2\pi} x^k y^k_\alpha \, d\alpha - c \int_0^{2\pi} x x^k_{\alpha\alpha} \, d\alpha + c \int_0^{2\pi} \left(y^k_\alpha\right)^2 \, d\alpha - c^2 \int_0^{2\pi} y^k_\alpha x^k_{\alpha\alpha} \, d\alpha = A_0. \tag{5.2.22}$$

We take c to be the closest root to zero of this quadratic equation.

5.2.3 Fast reparameterization

For long-time simulations, it's important that we keep discretization points well separated. The classical approach for reparameterizing evolving geometries in 2D is to introduce an auxiliary tangential velocity in the kinematic condition that helps maintain parameterization quality under some metric (e.g., equispaced in arc length) [48]. This approach is very effective and widely used. In our setting, however, in addition to evolving the shape, we need to keep track of the membrane tension from the previous time step. While this can be accomplished by advecting the scalar field (tension) with the auxiliary velocity field, it effects the overall accuracy and stability of our time stepping scheme. Moreover, our requirements are different: we do not need to maintain exact equi-arc-length parameterization but rather a scheme that overcomes the error amplification due to large shape deformations. For this reason, we take a different approach that avoids an auxiliary advection equation solve at each time step. The scheme only uses Fourier interpolation and redistributes the discretization points on the vesicle membrane so that they are *nearly* equispaced.

Let $\boldsymbol{x}(\alpha)$, where $\alpha \in [0, 2\pi)$, be a parameterization for a vesicle's membrane γ. We begin by expressing $ds/d\alpha$ as a Fourier series. That is,

$$\frac{ds}{d\alpha} = \|\boldsymbol{x}_\alpha\|_2 = \sum_{k=-M/2+1}^{M/2} C_k e^{ik\alpha}, \tag{5.2.23}$$

where M is the number of discretization points. Integrating both sides gives

$$s(\alpha) = \int_0^\alpha \|\boldsymbol{x}_\beta\|_2 \, d\beta = K + C_0\alpha + \sum_{k\neq 0} \frac{C_k}{ik} e^{ik\alpha}, \tag{5.2.24}$$

where

$$K = -\sum_{k\neq 0} \frac{C_k}{ik}. \tag{5.2.25}$$

Our goal is to find a set of discrete points $\{\alpha_k^*\}_{k=0}^{M-1}$ in the parametric domain that corresponds to $\{s_j = s(2\pi j/M), j = 0, \ldots, M-1\}$. We simply use a piecewise linear interpolant of $s(\alpha)$ to determine these points; see Algorithm 2 for more details. The result is that we obtain points on the curve that are nearly equispaced (up to the accuracy of linear interpolation). The coordinate positions $\boldsymbol{x}$ are then evaluated at α^* via Fourier interpolation. Note that the linear interpolation does not interfere with the overall spectral accuracy of the method (in space) because it is used only to find a new set of points α^*.

Algorithm 2 Compute α^*

```
$\alpha_0^* = 0, j = 0$
for $k = 1 : M - 1$ do
    while $\frac{s(2\pi)}{M}k \notin [s_j, s_{j+1})$ do
        $j = j + 1$
    end while
    $\alpha_k^* = \frac{2\pi}{M}\left(j + \frac{\frac{s(2\pi)}{M}k - s_j}{s_{j+1} - s_j}\right)$
end for
```

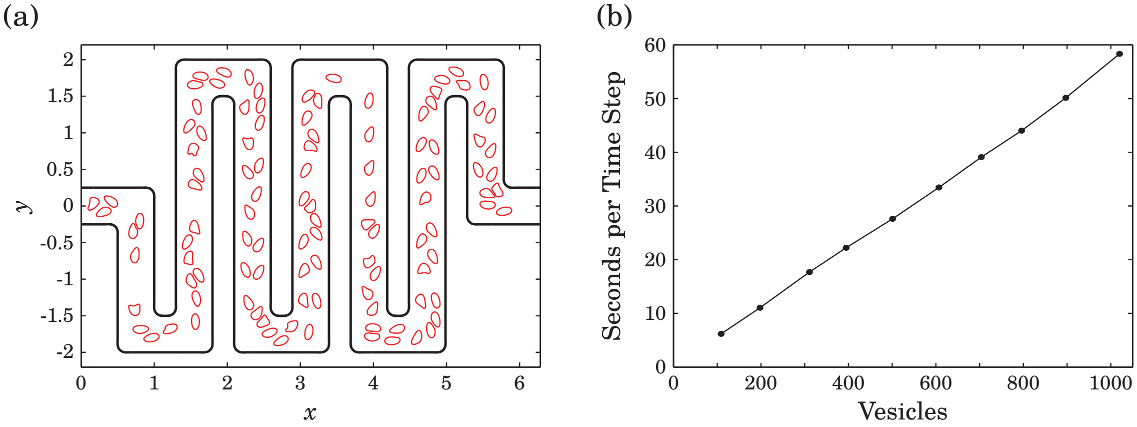


Figure 5.3.1: (a) A snapshot of 109 vesicles in the serpentine channel. A pressure difference is pushing the vesicles in the positive x-direction. (b) Average time per time step (in seconds) for the first 10 time steps as the number of vesicles is varied. Each dot represents a data point. Timings were performed using a laptop with a 2.4 GHz dual-core Intel Core i5 processor and 8 GB of RAM.

5.3 Results

We now give numerical results for the vesicle flow simulation described in Section 5.1. A snapshot of 109 vesicles flowing through the serpentine channel is shown in Fig. 5.3.1(a), where a pressure difference between the left and right sides drives the flow in the positive x-direction. When modeling such flows, it is vitally important that vesicles avoid collisions with other vesicles and with the walls. We have found that using 64 points per vesicle along with a close evaluation scheme is usually enough to prevent vesicle-vesicle collisions, as long as vesicle shapes are not elongated and the time step is not too big. For the simulations in Figs. 5.1.1(a) and 5.3.1(a), the time step was set to $\Delta t = 0.005$. When modeling pressure driven flows with no-slip boundary conditions, the vesicles often keep a safe distance from the walls and a close evaluation scheme is not always required. However, there are plenty of situations where this is not the case. In general, vesicle-wall collisions tend to occur more often when the channel geometry has bends or tight spaces, the number of vesicles is large, or the bending modulus κ_B is high.

We now focus on the performance of the algorithm. For the Stokes single- and double-layer FMM we used the potential and first derivatives output by the Fortran/OpenMP Laplace FMM implentation of Gimbutas–Greengard [49], combined with standard formulae [1, Eq. (2.8)–(2.9)]. Fig. 5.3.1(b) shows the scaling as the number of vesicles in the serpentine geometry is increased. When performing the timings, the vesicle dimensions were scaled to maintain the same relative spacing. Each vesicle had 64 discretization points and the number of quadrature points on the walls was set to 29 times the number of vesicles. The algorithm maintained linear scaling to over 1,000 vesicles on a laptop with a 2.4 GHz dual-core Intel Core i5 processor and 8 GB of RAM. In Table 5.2, we give the CPU time distribution for the simulation with 1,020 vesicles. Approximately 82% of the computational time was spent computing vesicle-vesicle, vesicle-wall, or wall-vesicle

Table 5.2: The CPU time distribution for the first 10 time steps of a simulation with 1,020 vesicles (64 points each) in the serpentine channel with $N = 29{,}580$ points per wall. Each time step took an average of 58 seconds on a laptop with a 2.4 GHz dual-core Intel Core i5 processor and 8 GB of RAM.

Operation	Percentage
vesicle to vesicle interactions	63.49
vesicle to wall interactions	14.77
preconditioner	7.30
wall to vesicle interactions	3.80
bending and tension forces	3.17
inextensibility operator	1.41
solve for $\boldsymbol{\tau}$, $\boldsymbol{\xi}$ using A^{-1}	1.40
vesicle to side interactions	1.31
proxy point to vesicle interactions	1.23
vesicle area corrections	1.16
other	0.96

interactions. The majority of this expense was handled by the FMM. Approximately 7% of the time was used to construct a preconditioner which was a sparse block diagonal matrix consisting of vesicle-vesicle self interactions. Notice that while almost one half of the points are associated with the fixed geometry, the solve associated with these points takes less than 2% of the total time for the time step, thanks to the fast direct solver.

CHAPTER 6

Application: Equilibrium Shapes of Planar Elastic Membranes

This material was first published in [28].

6.1 Introduction

Understanding the wrinkling and folding patterns of elastic sheets over a soft substrate has received much attention recently owing to a large number of practical applications [50]. Recent experiments of [2] appeared to show that a wrinkle to fold transition of the sheet occurs whenever it is compressed beyond a certain threshold. The fold patterns are highly localized which have also been reproduced numerically in [2]. Elegant analytic solutions for the fold shapes have been derived in [51] (and extended recently to sheets of finite length in [52]) by showing that the Euler's elastica equation governing the elastic sheet belongs to the stationary-sine-Gordon-modified-KdV hierarchy, which is integrable. Two classes of solutions, corresponding to a symmetric and an antisymmetric fold, have been reported. However, these solutions are restricted to the primary single-fold shapes. More recently [53, 54] have also used a rod theory formulation to study the shapes of floating elasticas. In [54], the authors first linearize the rod equations and get buckling thresholds as well as symmetric and antisymmetric mode shapes. Then they perform a non-linear post-buckling analysis to get the shapes of the elastica with large deflections of the rod. Finally, they check for the stability of these post-buckled shapes. [54] resolve the forces in the rod in a lab-fixed coordinate system and solve the resulting non-linear ODEs numerically to get the post-buckled shapes.

Here, we present new analytical solutions for the equilibrium shapes. Furthermore, we obtain more general multiple-fold shapes numerically by solving a set of first-order ordinary differential equations (ODEs) derived using large deflection theory of rods. Unlike the far-field boundary conditions applied in [51] (for the sake of deriving exact solutions), we apply clamped boundary conditions. This is consistent with the experimental set-up of [2] wherein the lateral edges of a thin polymer sheet resting on the surface of water are clamped while the sheet itself is compressed laterally. By simply varying the applied strain, which enters as a boundary condition in our formulation, we get shapes with varying complexity from smooth to highly localized to nearly touching configurations. In contrast to [54], we resolve the force in the rod along coordinates that are tangential and normal to the rod and obtain analytical solutions to the resulting non-linear ODEs.

Using the same formalism, we consider another important problem, that of determining the periodic equilibrium shapes of lipid membranes immersed in a viscous fluid. Periodic equilibria of lipid bilayers in the form of minimal surfaces (with cubic symmetry) have been known for a long time [55]. Periodic cylindrical equilibria are also known [56]. Indeed, such periodic shapes are observed in cellular organelles such as mitochondria, chloroplasts, and endoplasmic reticula [57, 58]. Periodic shapes with cylindrical symmetry have been realized in suspended lipid bilayers [59]. Moreover, periodic membrane mechanics is often used as a paradigm for understanding the complex multi-physics of soft particles such as vesicles and red blood cells. For example, [60] studied the interplay of membrane compositional dynamics with solvent hydrodynamics while [61] considered the vesicle electrohydrodynamics.

Building on our previous work on closed vesicles [62], we derive exact solutions in terms of elliptic functions to the ODEs of rod theory formulation with periodic boundary conditions. Excess length characterizes the obtained equilibrium shapes. We show that single-fold shapes have lower elastic energy compared to multiple-fold shapes for a fixed excess length. While useful in their own right, the exact solutions, perhaps more importantly, can be used for validating numerical schemes that are applicable in the broader context of simulating biomembranes in viscous fluids. We formulate and implement a boundary integral method to simulate the dynamics of a periodic lipid bilayer membrane suspended in a viscous fluid. The membrane dynamics are characterized by a competition between elastic energy, local inextensibility, and non-local hydrodynamic forces. We develop a high-order method to solve the integro-differential equations for membrane evolution to equilibrium by the use of Fourier representations and a high-order singular integration scheme. We show that for any arbitrary initial profile, the membrane always relaxes to one of the exact solutions we derived using the rod theory formulation.

6.2 Rod-theory formulation

In this section, we provide a rod-theory based approach for computing equilibrium configurations of planar membranes. Since we focus primarily on cylindrical membrane shapes, it is sufficient to think of the membrane as an inextensible planar rod [63]. Indeed, recent experiments on membranes or thin films at the interface between two liquids have demonstrated a variety of shapes that are similar to those obtained in our computations. All of these membrane shapes are cylindrical so that curvature is a scalar variable, just as it is for a planar rod. At any cross-section of this rod located at arc length s, there are two components of the internal force $F_x(s)$ and $F_y(s)$ and one internal moment $M(s)$. The tangent to the rod makes an angle $\theta(s)$ with the laboratory x-axis. Therefore,

$$\cos\theta = \frac{dx}{ds} \quad \text{and} \quad \sin\theta = \frac{dy}{ds}. \tag{6.2.1}$$

The equilibrium equations for the rod are well-known and are given by [64]

$$\begin{aligned}\frac{dF_x}{ds} + b_x &= 0, &(6.2.2)\\ \frac{dF_y}{ds} + b_y &= 0, &(6.2.3)\\ \frac{dM}{ds} - F_x \sin\theta + F_y \cos\theta + m &= 0, &(6.2.4)\end{aligned}$$

where $b_x(s)$ and $b_y(s)$ are body forces per unit length and $m(s)$ is a body moment per unit length. In experiments on thin films at liquid interfaces, the body forces arise from hydrostatic pressure exerted by a liquid column on the film. Such a pressure acts normal to the surface $\mathbf{n}$ of the film and takes the form $\mathbf{b}(s) = b(s)\mathbf{n}$, where $b(s)$ is a scalar function. Since there are no body moments in the films studied in these experiments, we will set $m = 0$ everywhere. Given this form of the body force, it is natural to resolve the internal force $\mathbf{F}$ in the rod along the tangent $\mathbf{t}$ and normal $\mathbf{n}$. Then, $\mathbf{F} = F_t\mathbf{t} + F_n\mathbf{n}$ and the equilibrium equations can be written as

$$\begin{aligned}\frac{dF_t}{ds} + F_n\kappa &= 0, &(6.2.5)\\ \frac{dF_n}{ds} - F_t\kappa + b &= 0, &(6.2.6)\\ \frac{dM}{ds} + F_n &= 0, &(6.2.7)\end{aligned}$$

where $\kappa = \frac{d\theta}{ds}$ is the curvature and the rod is assumed to be inextensible. The tangent and normal to the rod are related through the standard Frenet-Serret formulas on the plane, given by

$$\frac{d\mathbf{t}}{ds} = -\kappa\mathbf{n} \quad \text{and} \quad \frac{d\mathbf{n}}{ds} = \kappa\mathbf{t}. \qquad (6.2.8)$$

We also assume that the moment M and curvature κ are related through $M = K_b\kappa$ where K_b is a bending modulus of the membrane. Then, (6.2.5) and (6.2.7) can be combined to

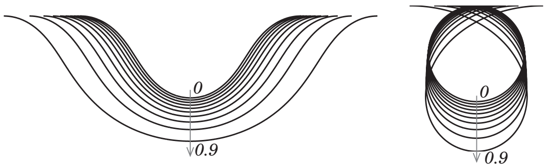


Figure 6.3.1: Plot of equilibrium shapes corresponding to the exact solutions (6.3.4) and (6.3.9) for different values of $m^2 = \{0, 0.1, 0.2, \ldots, 0.9\}$.

yield

$$F_t = \frac{K_b}{2}\kappa^2 + C, \tag{6.2.9}$$

where C is a constant that must be determined from the boundary conditions. When we plug (6.2.9) into (6.2.6), we get

$$K_b \frac{d^2\kappa}{ds^2} + \frac{K_b}{2}\kappa^3 + C\kappa - b = 0. \tag{6.2.10}$$

If we recognize $-C/K_b$ as the tension σ and $-b/K_b$ as the pressure difference p between the bottom and top of the film, then this is the same equation that was solved previously for closed vesicles [62], assuming a constant pressure difference.

6.3 Analytical Solutions

In this section, we derive analytic solutions to (6.2.10) for three kinds of boundary conditions—clamped, periodic and simply supported—in terms of the Jacobi elliptic functions which are defined using the integral $u = \int_0^\phi 1/\sqrt{1 - m\sin^2\theta}\, d\theta$ as

$$\begin{aligned} \mathrm{sn}(u|m) &= \sin(\phi), && (6.3.1)\\ \mathrm{cn}(u|m) &= \cos(\phi), && (6.3.2)\\ \mathrm{dn}(u|m) &= \sqrt{1 - m\sin^2\phi}. && (6.3.3) \end{aligned}$$

6.3.1 Clamped boundary conditions

For clamped boundary conditions, we solve (6.2.10) with a non-zero pressure difference between the two sides of the film. In order to illustrate this method and to relate to the experimental results of [2], we assume that $b = -ky = -\rho g y$, where ρ is the density difference between the liquids on the two sides of the film, g is the acceleration due to gravity, and $k = \rho g$ is the stiffness of the "substrate". Under this assumption, an analytical solution to (6.2.10) is given by

$$\kappa(s) = \frac{2\sqrt{1-m^2}}{\mathrm{dn}(s|m)}, \tag{6.3.4}$$

for which

$$\frac{C}{K_b} = m^2 - 2 - \frac{\rho g}{K_b m^2}. \tag{6.3.5}$$

This gives

$$\begin{aligned} \theta(s) &= 2\sqrt{1-m^2}\frac{\mathrm{sn}(s|m)\mathrm{cn}(s|m)}{\mathrm{dn}^2(s|m)}, && (6.3.6) \\ x(s) &= \int \left(2\frac{\mathrm{cn}^2(s|m)}{\mathrm{dn}^2(s|m)} - 1 \right) ds, && (6.3.7) \\ y(s) &= 2\frac{\sqrt{1-m^2}}{m^2}\frac{1}{\mathrm{dn}(s|m)}. && (6.3.8) \end{aligned}$$

Clearly, $\theta(s = \pm K(m)) = 0$ where $K(m)$ is the complete elliptic integral of the first kind. If we take $L = 2K(m)$ then this solution represents clamped-clamped boundary conditions. The parameter m can be determined by enforcing $x(\frac{L}{2}) - x(-\frac{L}{2}) = \frac{L}{1+\epsilon_{pre}}$ where ϵ_{pre} is a prescribed pre-strain. Alternatively, ϵ_{pre} is determined by the compression applied on the film which specifies the end-to-end distance $x(\frac{L}{2}) - x(-\frac{L}{2})$. We can always re-scale the length L by a length scale λ such that $\frac{L}{\lambda} = K(m)$. The constant C must then be modified accordingly. Another solution to (6.2.10) is

$$\kappa(s) = 2\mathrm{dn}(s|m), \tag{6.3.9}$$

for which

$$\frac{C}{K_b} = m^2 - 2 + \frac{\rho g}{K_b m^2}. \tag{6.3.10}$$

This gives

$$\begin{aligned} \theta(s) &= 2\sin^{-1}(\mathrm{sn}(s|m)), && (6.3.11)\\ x(s) &= \frac{2}{m^2}E(s|m) + (1 - \frac{2}{m^2})s, && (6.3.12)\\ y(s) &= -\frac{2}{m^2}\mathrm{dn}(s|k), && (6.3.13) \end{aligned}$$

where $E(s|m)$ is the incomplete elliptic integral of the second kind with modulus m. Again, if we take $L = 2K(m)$, then since $\theta(s = \pm K(m)) = \pm\pi$, these solutions correspond to clamped boundary conditions but result in self-intersecting shapes. The equilibrium shapes corresponding to (6.3.4) and (6.3.9) are plotted in Figure 6.3.1 for various values of m.

6.3.2 Periodic boundary conditions

In the special case of $k = 0$ (which implies $b = 0$) it is possible to recast (6.2.10) as

$$\frac{d}{ds}\left(\left(\frac{1}{2}\frac{d\kappa}{ds}\right)^2 + \frac{\kappa^4}{8} + \frac{C\kappa^2}{2}\right) = 0, \tag{6.3.14}$$

which implies

$$\left(\frac{1}{2}\frac{d\kappa}{ds}\right)^2 + \frac{\kappa^4}{8} + \frac{C\kappa^2}{2} = B, \tag{6.3.15}$$

where B is a constant. This results in closed orbits on the $\kappa - \frac{d\kappa}{ds}$ plane which suggests periodic solutions for $\kappa(s)$. One class of periodic solution in terms of elliptic functions is given by

$$\kappa(s) = 2m\text{cn}(s|m), \tag{6.3.16}$$

which results in

$$\begin{aligned} x(s) &= 2E(s|m) - s, \\ y(s) &= 2m\text{cn}(s|m), \\ \theta(s) &= 2\cos^{-1}(\text{dn}(s|m)), \end{aligned} \tag{6.3.17}$$

where $E(s|m)$ is the incomplete elliptic integral of the second kind with modulus m [65]. Figure 6.3.2 (I) shows the equilibrium shapes for various values of m.

Another class of shapes that solve (6.2.10) with $b = 0$ is given by

$$\kappa(s) = 2m\sqrt{1-m^2}\frac{\text{sn}(s|m)}{\text{dn}(s|m)}. \tag{6.3.18}$$

To realize periodic boundary conditions, we set $L = 4qK(m)$ where $K(m)$ is the complete elliptic integral of the first kind and q is a positive integer. By integration we get

$$\theta(s) = 2\sin^{-1}\left(-m\frac{\text{cn}(s|m)}{\text{dn}(m|k)}\right), \tag{6.3.19}$$

$$\cos(\theta) = 1 - 2m^2\frac{\text{cn}^2(s|m)}{\text{dn}^2(s|m)}, \tag{6.3.20}$$

$$\sin(\theta) = -2m\sqrt{1-k^2}\frac{\text{cn}(s|m)}{\text{dn}^2(s|m)}. \tag{6.3.21}$$

We obtain $x(s)$ and $y(s)$ by integrating numerically. The equilibrium shapes thus obtained are plotted in Figure 6.3.2 (II) for various values of m and $q = 1$.

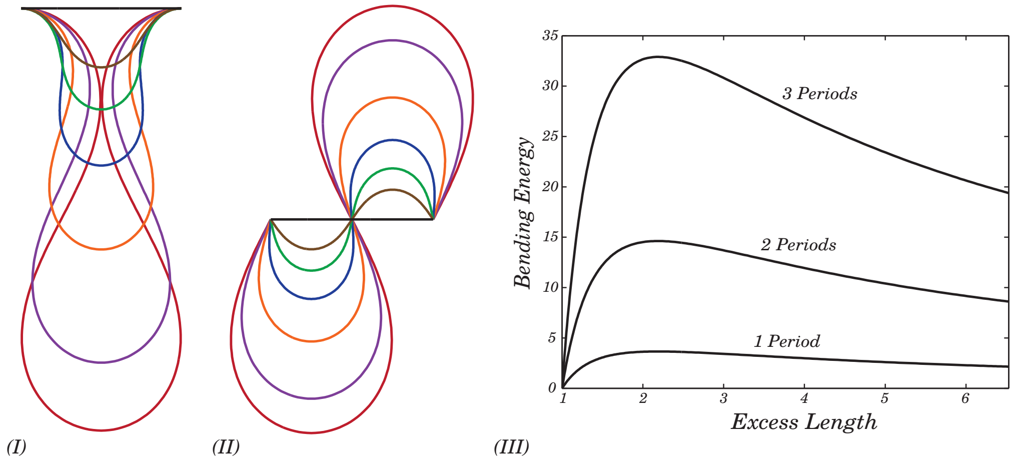


Figure 6.3.2: (Color online) Plots of equilibrium shapes corresponding to the exact solutions (6.3.16) and (6.3.18) in (I) and (II) respectively for various excess lengths (with shapes colored the same in (I) and (II) representing those with same value for m). Excess length is defined as the membrane arc length over the period 2π. Periodicity implies that multiple fold shapes obtained by concatenating each of these exact solutions is also an exact solution but with a higher excess length. In (III), we compare the bending energy, defined as $\int_{\gamma(s)} \kappa^2(s)\, ds$, of the shapes in (I) plotted over multiple periods and proportionally scaled back to $[0, 2\pi)$ (for example, a four-fold and a two-fold shapes are plotted in Figure 6.7.2).

6.3.3 Simply supported boundary conditions

Interestingly, (6.3.16) and (6.3.17) will also be a solution to (6.2.10) with $k = \rho g$ if we select the arbitrary constant C as

$$\frac{C}{K_b} = 1 - 2m^2 - \frac{\rho g}{K_b}. \tag{6.3.22}$$

Here, as before, m must be determined from the boundary conditions. The appropriate boundary condition is

$$x\left(\frac{L}{2}\right) - x\left(-\frac{L}{2}\right) = 4E\left(\frac{L}{2}\middle| m\right) - L = \frac{L}{1+\epsilon_{pre}}, \tag{6.3.23}$$

where ϵ_{pre} is a pre-strain or is determined from the compression applied on the film. For given L this is an equation for m which can be solved numerically. We also see that $y(s = K(m)) = 0$ and $\theta(s = K(m)) \neq 0$ where $K(m)$ is the complete elliptic integral of

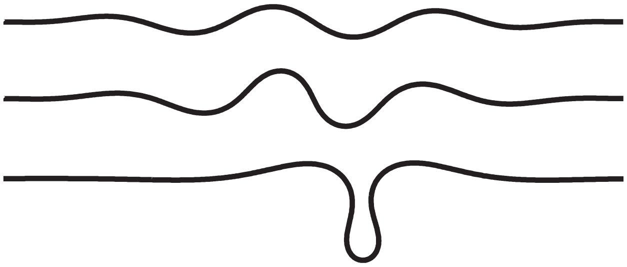


Figure 6.4.1: Equilibrium shapes obtained by solving (6.4.1) through (6.4.8) that match the experimentally observed shapes reported in Fig. 1(A) of [2]. The solutions were obtained by modifying ϵ_{pre} while varying L to hold $x(L/2)$ constant. The first and second solutions from the top are odd type solutions as shown in Fig. 6.9.1. The last solution is an even type solution like those shown in Fig. 6.9.2.

the first kind. This means that (6.3.16) and (6.3.17) are solutions of (6.2.10) with simply supported boundary conditions with $L = 2K(m)$ which correspond to zero deflection and zero curvature at the two ends. Note that $\theta(s)$ in (6.3.17) can be written as

$$\theta(s) = 4\tan^{-1}\left(\frac{m\mathrm{sn}(s|m)}{1+\mathrm{dn}(s|m)}\right). \tag{6.3.24}$$

This is reminiscent of the solution in [51] where the symmetric solution for clamped-clamped boundary conditions was

$$\theta(s) = 4\tan^{-1}\left(\frac{\kappa\sin(ks)}{k\cosh(\kappa s)}\right), \tag{6.3.25}$$

with $k^2 + \kappa^2 = 1$. Although these solutions correspond to different boundary conditions, they look similar if we take $\kappa << 1$ and $m^2 = 2\kappa + \kappa^3$ (which would mean a small ϵ_{pre}).

6.4 Asymmetric Shapes

All of the analytical solutions derived in the previous section give rise to symmetric equilibrium shapes. In experiments, however, many asymmetric shapes are commonly observed (e.g., in [2]). In this section, we develop a numerical procedure to obtain such shapes. First, we recast our problem as a system of first order ODEs (using equations (6.2.5) through (6.2.9)):

$$\begin{aligned}
\frac{dx}{ds} &= \cos\theta, && (6.4.1)\\
\frac{dy}{ds} &= \sin\theta, && (6.4.2)\\
\frac{d\theta}{ds} &= \frac{M}{K_b}, && (6.4.3)\\
\frac{dM}{ds} &= -F_n, && (6.4.4)\\
\frac{dF_n}{ds} &= ky + \frac{K_b}{2}\frac{M^3}{K_b^3} + C\frac{M}{K_b}. && (6.4.5)
\end{aligned}$$

In the experiments of [2] the end-to-end distance $x(L/2) - x(-L/2)$ is reduced in order to buckle the film. We account for this by using a strain ϵ_{pre} through our boundary conditions:

$$\begin{aligned}
x(L/2) - x(-L/2) &= \frac{L}{1+\epsilon_{pre}}, && (6.4.6)\\
y(-L/2) = 0, \quad y(L/2) &= 0, && (6.4.7)\\
\theta(-L/2) = 0, \quad \theta(L/2) &= 0. && (6.4.8)
\end{aligned}$$

We solve the system consisting of (6.4.1) through (6.4.8) numerically. Details of the solver are presented in the appendix. Select solutions are plotted in Figures 6.4.1 to 6.4.2 with the following motivation. First, in Figure 6.4.1, we show solutions that closely match the experimentally obtained shapes reported in Figure 1(A) of [2]. Second, odd and even type solutions for varying ϵ_{pre} values along with the corresponding internal moment M and normal force F_n are shown in Figures 6.9.1 and 6.9.2 respectively that correspond to the antisymmetric and symmetric configurations derived analytically in [51]. Lastly, we plot a few classes of general multiple fold shapes in Figure 6.4.2. The advantage of our approach is that all of these shapes can also be obtained by using the same solver.

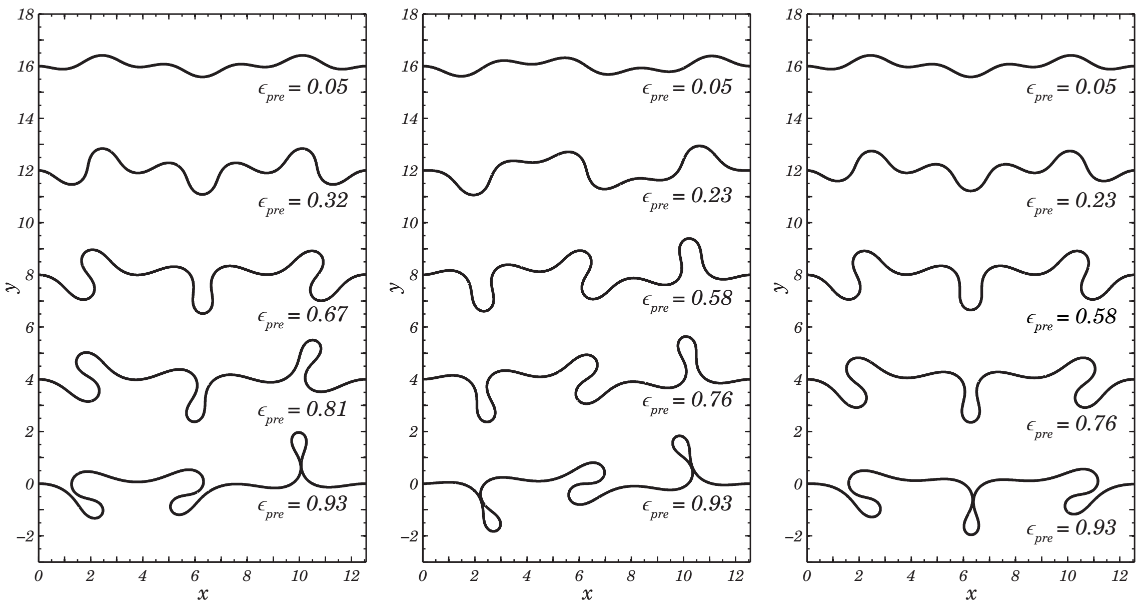


Figure 6.4.2: Other classes of equilibrium shapes that can be obtained by solving (6.4.1) through (6.4.8). The solutions were obtained by numerically continuing ϵ_{pre} from the folded solution to the low amplitude solution while varying L to keep $x(L/2)$ constant.

6.5 Fluid-structure interaction problem

Consider a planar elastic membrane suspended in an incompressible viscous fluid with viscosity μ. The fluid domain Ω is assumed to be periodic in the $x-$direction and unbounded in the $y-$direction. Let $\mathbf{x}$ be the position of the membrane γ, $\mathbf{v}$ the fluid velocity, and p the pressure. In the vanishing Reynolds number limit, the governing equations for the ambient fluid are given by

$$\begin{aligned} -\mu\Delta\mathbf{v} + \nabla p &= \mathbf{0} \text{ in } \Omega\backslash\gamma, \\ \nabla\cdot\mathbf{v} &= 0 \text{ in } \Omega\backslash\gamma. \end{aligned} \tag{6.5.1}$$

The fluid motion is coupled to the membrane motion via the kinematic boundary condition

$$\dot{\mathbf{x}} = \mathbf{v} \quad \text{on} \quad \gamma. \tag{6.5.2}$$

The local inextensibility constraint on the membrane can be expressed as [24]

$$\mathbf{x}_s \cdot \dot{\mathbf{x}}_s = 0 \text{ on } \gamma. \tag{6.5.3}$$

The stress balance at the membrane interface is given by

$$[\![-p\mathbf{n} + (\nabla \mathbf{v} + \nabla \mathbf{v}^T)\mathbf{n}]\!] = \mathbf{f}_{mem} \text{ on } \gamma, \tag{6.5.4}$$

where $\mathbf{n}$ is the unit normal to the membrane. The notation $[\![q]\!]$ denotes the jump of a quantity q across the membrane. The classical Helfrich energy model for the bilayer membrane and an augmented Lagrangian approach (to enforce the local inextensibility) lead to a bending force $\mathbf{f}_b$ and a tension force $\mathbf{f}_\sigma$ on the membrane so that $\mathbf{f}_{mem} = \mathbf{f}_b + \mathbf{f}_\sigma$. They are defined by [24]

$$\mathbf{f}_b = -\kappa_B \mathbf{x}_{ssss}, \quad \mathbf{f}_\sigma = (\sigma \mathbf{x}_s)_s, \tag{6.5.5}$$

where κ_B is the bending modulus, κ is the curvature, s is the arclength parameter, and σ is the tension which acts as a Lagrange multiplier to enforce the local inextensibility constraint.

Given an arbitrary initial shape, the membrane deforms until the elastic stress due to bending and tension balances out the hydrodynamic stress at the interface. The transient dynamics can be simulated by solving the the fluid equations subject to the no-slip, periodic, and kinematic boundary conditions along with the stress balance. We use potential theory to recast the governing equations for the membrane evolution in the form of coupled integro-differential equations:

$$\begin{aligned} \dot{\mathbf{x}} = \int_\gamma G(\mathbf{x}, \mathbf{y}) \left(\mathbf{f}_b(\mathbf{y}) + \mathbf{f}_\sigma(\mathbf{y})\right) \, \mathrm{d}\gamma(\mathbf{y}), \\ \mathbf{x}_s \cdot \dot{\mathbf{x}}_s = 0, \end{aligned} \tag{6.5.6}$$

where G is the periodic Green's function for the Stokes equations (6.5.1), which is known

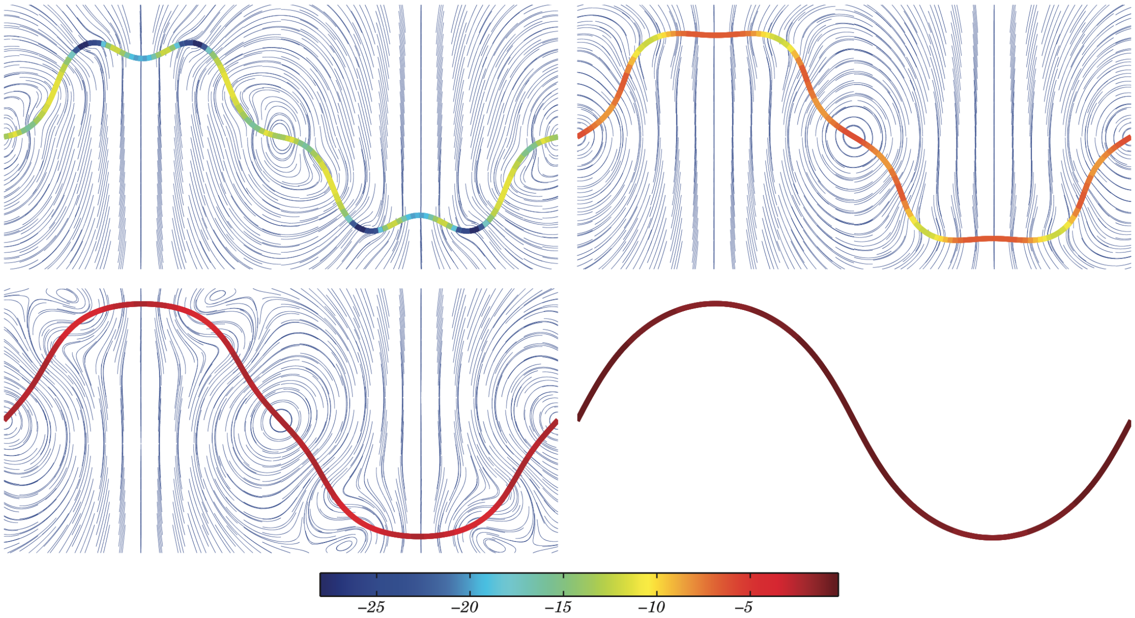


Figure 6.5.1: (Color online) Snapshots of a bilipid membrane evolving to an equilibrium shape starting from an arbitrary initial shape. The streamlines are determined by evaluating the surrounding fluid velocity using the relation $\mathbf{v}(\mathbf{x}) = \int_\gamma G(\mathbf{x}, \mathbf{y})\, \mathbf{f}_{\text{mem}}$ for any $\mathbf{x}$ in the fluid domain. The color on the membrane indicates the magnitude of the tension σ.

in closed analytic form[1]. The integral equation formulation (6.5.6) has two important advantages. First, the Green's function satisfies the fluid incompressibility condition and the far-field (periodic) boundary condition by construction. Second, the unknowns reside only on the membrane. This avoids the need for a volume mesh. We describe our numerical scheme to solve (6.5.6) next.

6.6 Numerical Scheme

We represent the membrane orientation as a parametrized curve. Each value of $\alpha \in [0, 2\pi)$ is a reference to a point $\mathbf{x}(\alpha, t) = [x(\alpha, t),\, y(\alpha, t)]^T$ on the membrane. Denoting by L the total arc length of the membrane, the x-component of the membrane position can be

[1] It is given by [66]:
$G(\hat{\mathbf{x}}) = \begin{bmatrix} -A - \hat{y}A_y + 1 & \hat{y}A_x \\ \hat{y}A_x & -A + \hat{y}A_y \end{bmatrix}$, with $\hat{\mathbf{x}} = \begin{bmatrix} \hat{x} \\ \hat{y} \end{bmatrix} = \mathbf{x} - \mathbf{y}$,
$A(\hat{\mathbf{x}}) = \frac{1}{2} \ln(\cosh \hat{y} - \cos \hat{x}) + \frac{1}{2} \ln 2$, $A_x = \frac{\partial A}{\partial \hat{x}}$, and $A_y = \frac{\partial A}{\partial \hat{y}}$.

expressed as the sum of a linear term and a function x_p that is 2π-periodic in α:

$$x(\alpha, t) = \frac{L}{2\pi}\alpha + x_p(\alpha, t). \tag{6.6.1}$$

We track the evolution of marker points on the membrane that are uniformly spaced in the parametric domain at $\{\alpha_j = 2\pi(j-1)/M\}_{j=1}^{M}$ for a given discretization size M. This allows fast and spectrally accurate computation of derivatives via the fast Fourier transform (FFT). For example, we can write

$$y_\alpha = \sum_{j=-M/2}^{M/2} (-ij)\hat{y}(j)e^{-ij\alpha}, \tag{6.6.2}$$

and use FFTs to switch between y and $\hat{y}$.

The periodic Stokes Green's function in (6.5.6), similar to its free-space counterpart, exhibits a logarithmic singularity to the leading order when the evaluation point $\mathbf{x}$ resides on the membrane [66]. Therefore, quadrature rules designed for smooth integrands such as the trapezoidal rule are not effective in evaluating the integral in (6.5.6), and a special treatment is required. We use the hybrid Gauss-trapezoidal quadrature rule outlined in [67] that is designed to yield arbitrarily high-order accuracy for integrals of this form. We evaluate the integrand at the small number of nonuniform Gaussian nodes this rule requires via Fourier interpolation.

6.6.1 Time-stepping scheme

We propose the following time-stepping scheme for the membrane orientation and inextensiblity equations (6.5.6):

$$\frac{\mathbf{x}^{k+1} - \mathbf{x}^k}{\Delta t} = \int_{\gamma^k} G(\mathbf{x}^k, \mathbf{y}^k)[(\sigma^{k+1}{\mathbf{y}^k}_s)_s - \kappa_B \mathbf{y}^{k+1}_{ssss}]\,\mathrm{d}\gamma, \tag{6.6.3}$$

$$0 = \mathbf{x}^k_s \cdot \left(\int_{\gamma^k} G(\mathbf{x}^k, \mathbf{y}^k)[(\sigma^{k+1}{\mathbf{y}^k}_s)_s - \kappa_B \mathbf{y}^{k+1}_{ssss}]\,\mathrm{d}\gamma \right)_s . \tag{6.6.4}$$

All derivatives are computed using the membrane orientation from the kth time step. This semi-implicit discretization is based on [24] and has been shown to have unconditional stability. If we let $\mathbf{v} = [x,\ y,\ \sigma]^T$, then it may be verified that the following are linear operators:

$$\mathcal{B}(\mathbf{v}) = \mathbf{x} - \Delta t \int_{\gamma^k} G(\mathbf{x}^k, \mathbf{y}^k)[(\sigma {\mathbf{y}^k}_s)_s - \kappa_B \mathbf{y}_{ssss}]\,\mathrm{d}\gamma, \tag{6.6.5}$$

$$\mathcal{D}(\mathbf{v}) = \mathbf{x}_s^k \cdot \left(\int_{\gamma^k} G(\mathbf{x}^k, \mathbf{y}^k)[(\sigma {\mathbf{y}^k}_s)_s - \kappa_B \mathbf{y}_{ssss}]\,\mathrm{d}\gamma \right)_s, \tag{6.6.6}$$

where $\mathbf{x} = \mathbf{y} = [x,\ y]^T$. This yields the linear system

$$\mathcal{B}(\mathbf{v}) = \mathbf{x}^k, \qquad \mathcal{D}(\mathbf{v}) = 0, \tag{6.6.7}$$

whose solution gives the functions $\mathbf{x}^{k+1}(\alpha)$ and $\sigma^{k+1}(\alpha)$ at the next time step. This linear system, as it is stated now, is not ideal for a number of reasons. First, x should maintain the form $x(\alpha) = L\alpha/2\pi + x_p(\alpha)$ throughout the entire computation. Any nonunitary scaling of x that happens within the linear solver would change the linear term and as a result, invalidate our assumption about the form of x. Another issue is that when x is rescaled by the linear solver, the logarithmic singularities in the integral operators $\mathcal{B}$ and $\mathcal{D}$ will move so that applying the quadrature will be nontrivial. We handle this by decomposing $\mathbf{v}$ as follows:

$$\mathbf{v} = \begin{bmatrix} x \\ y \\ \sigma \end{bmatrix} = \begin{bmatrix} L\alpha/2\pi \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} x_p \\ y \\ \sigma \end{bmatrix} = \begin{bmatrix} L\alpha/2\pi \\ 0 \\ 0 \end{bmatrix} + \mathbf{v}_p. \tag{6.6.8}$$

This gives us the system

$$\mathcal{B}(\mathbf{v}_p) = \mathbf{x}^k - \mathcal{B}_L, \qquad \mathcal{D}(\mathbf{v}_p) = -\mathcal{D}_L, \tag{6.6.9}$$

where

$$\mathcal{B}_L = \mathcal{B}([L\alpha/2\pi,\ 0,\ 0]^T), \qquad \mathcal{D}_L = \mathcal{D}([L\alpha/2\pi,\ 0,\ 0]^T). \tag{6.6.10}$$

With this change, all input functions x_p, y, and σ are periodic so that scaling $\mathbf{v}_p$ will no longer jeopardize the form of x. We use the GMRES method [68] to solve this linear system.

Note that, since $[L\alpha/2\pi, 0, 0]_s^T$ is periodic, computing $\mathcal{B}_L$ and $\mathcal{D}_L$ leads to a periodic integrand, as is the case with $\mathbf{v}_p$. Since the integrand is periodic, we may shift it so that the logarithmic singularity is always at the two ends of the interval $[0,\ 2\pi)$. We then apply the quadrature rule of [67] on the intervals $[0,\ \pi)$ and $[\pi,\ 2\pi)$ separately.

6.7 Computational results

In this section, we perform an error analysis on the numerical scheme. We begin the program with an arbitrary initial profile and allow it to evolve in time in order to verify that the inextensibility of the membrane and the incompressibility of the fluid are preserved. We check that as the number of spatial grid points is increased and the size of the time-step is decreased, the relative error approaches zero. Later, the program is tested on one of the analytic equilibrium shapes with periodic boundary conditions. We do this by computing the initial membrane velocity on the shape. If it is indeed an equilibrium solution, then the velocity should approach zero as the resolution of the numerical scheme is enhanced.

6.7.1 Error analysis

We test that the program preserves membrane inextensibility and fluid incompressibility by evolving the curve

$$x(\alpha) = \alpha, \quad y(\alpha) = \exp(\sin\alpha + 0.1\cos 5\alpha), \tag{6.7.1}$$

until $t = 0.32$. We let $\Delta t = 0.004/M$ where M is the number of spacial grid points. The evolution of the curve and its bending energy are shown in Figure 6.7.1. If membrane

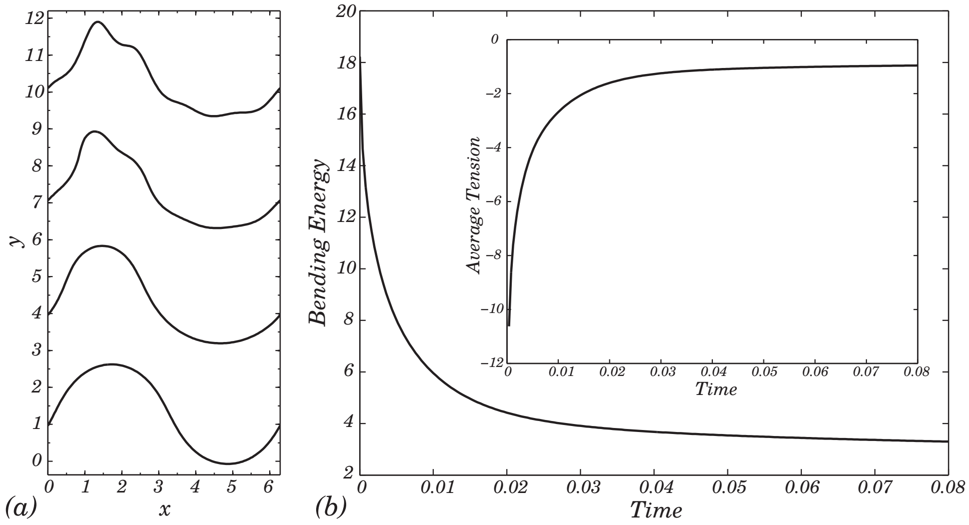


Figure 6.7.1: (a) Membrane shapes in time as it relaxes to equilibrium in a quiescent flow. (b) Plots of the membrane bending energy and average tension as a function of time.

Table 6.1: Convergence analysis of our numerical scheme measured on the basis of preserving the membrane inextensibility and fluid incompressibility constraints. Here, M is the number of spatial discretization points and the time-step size Δt is reduced by half as M is doubled. We report the relative errors in the length of the membrane L and area under it A measured at the end of the simulation in figure 6.7.1. We observe a first-order asymptotic convergence for the length and the area, which is consistent with our time-stepping scheme.

M	64	128	256
$\lvert L_0 - L_f\rvert/L_0$	$3.314 \cdot 10^{-4}$	$1.790 \cdot 10^{-4}$	$9.368 \cdot 10^{-5}$
$\lvert A_0 - A_f\rvert/A_0$	$3.939 \cdot 10^{-6}$	$1.746 \cdot 10^{-6}$	$9.111 \cdot 10^{-7}$

inextensibility is preserved, we expect the relative error between the initial membrane arc length L_0 and the final membrane arc length L_f to approach zero as M increases. Similarly, fluid incompresibility should imply that the area under the membrane remains constant. The area considered is the region contained by the x-axis and the membrane. The results of our error analysis are shown in Table 6.1. We observe a first order asymptotic convergence for the length and area. This is in accordance with our first order time-stepping scheme.

Table 6.2: The analytical solutions in figure 6.3.2 (I) are verified by feeding them as initial shapes to our membrane-fluid solver and checking that the resultant velocity field is zero. We report the maximum magnitude values for the initial velocity field evaluated on the membrane in the second row. We simulate the evolution to a fixed time horizon and report the max-norm error in the difference of initial and final positions in the third row. Clearly, both the errors converge to zero verifying that these are indeed the equilibrium shapes.

M	32	64
$\|\mathbf{v}(\mathbf{x})\|_\infty$	$3.043 \cdot 10^{-7}$	$2.797 \cdot 10^{-10}$
$\|\mathbf{x}_0 - \mathbf{x}_f\|_\infty$	$6.978 \cdot 10^{-9}$	$6.630 \cdot 10^{-11}$

6.7.2 Equilibrium shapes

We numerically verify that one of the analytic shapes derived above is an equilibrium solution. This will be done using two approaches. First, we check that the single-layer potential approaches zero as M, the number of spatial grid points, increases. This would indicate an initial velocity of zero. Second, we run the program until $t = 1$ using $\Delta t = 0.32/M$. As M increases, the difference between the initial and final membrane orientations should approach zero. Table 6.2 summarizes the results for the shape with one period and excess length equal to 5.09. For both approaches we obtain rapid convergence to zero. This is as expected, since the singular integrals were computed using the eigth-order Gauss-Trapezoidal scheme, and all derivatives have spectral convergence.

Finally, a natural question that arises is whether the multiple-fold shapes as discussed in Figure 6.3.2 are in fact stable equilibria. To verify this, we can take one of our analytical equilibrium shapes with multiple folds as the initial condition to the fluid-structure interaction problem and check if the shape changes over time. The result of one such numerical experiment is plotted in Figure 6.7.2. As is evident, the shape tends to remain the same until the numerical errors perturb it from this local equilibrium to a shape with lower number of folds. The timescales for these transitions from higher-to-lower energy states are found to be entirely dependent on the numerical parameters such as the discretization and time-step sizes.

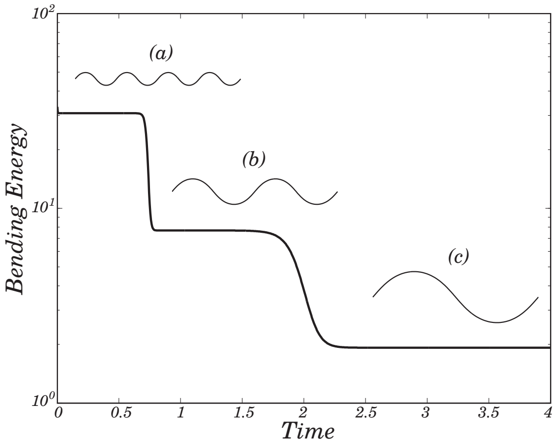


Figure 6.7.2: Evolution of a membrane with (a) four-fold shape to (b) two-fold shape to (c) a single-fold shape, in quiescent flow (see supplementary material for an animation). While the multiple-fold shapes have higher bending energy (as also pointed out in Figure 6.3.2), they correspond to stable equilibria.

6.8 Conclusions

We provided a rod theory based approach for determining the equilibrium profiles that arise when a thin elastic membrane is subjected to either clamped, simply supported or periodic boundary conditions. In the former case, we showed that a larger family of quasi-static equilibrium shapes exist. We derived a system of ODEs that when solved by the provided numerical routine, produced these more general multiple-fold solutions, along with the previously known shapes. The implementation of this routine is quick, and solutions can be generated in a matter of seconds.

In the case of periodic boundary conditions, we provided analytical solutions in terms of elliptic functions and solved the fluid-structure interaction problem using a boundary integral formulation. Our numerical scheme is unconditionally stable and converges spatially with spectral accuracy. We were able to verify our analytical solutions by showing that their initial velocity quickly converged to zero as the resolution of the solver was increased. In addition, we found that for any arbitrary initial profile, the membrane always relaxed back to one of the analytically obtained equilibrium shapes.

6.9 Appendix: Numerical solution of rod theory equations

The system consisting of (6.4.1) through (6.4.8) is numerically solved using a shooting method based routine. A shooting method works by reducing the solution of a boundary value problem to the solution of an initial value problem. This is done by determining a set of initial conditions that yields a solution that satisfies the desired boundary conditions. Since the first boundary condition relies on the difference between $x(L/2)$ and $x(-L/2)$, we may let $x(-L/2) = 0$ and then require

$$x(L/2) = \frac{L}{1+\epsilon_{pre}}. \tag{6.9.1}$$

Next, we let

$$C = \lambda_1, \quad M(-L/2) = \lambda_2, \quad F_n(-L/2) = \lambda_3. \tag{6.9.2}$$

For a given triplet of λ-values $(\lambda_1, \lambda_2, \lambda_3)$ we have the initial conditions

$$\begin{aligned} x(-L/2) &= 0, \\ y(-L/2) &= 0, \\ \theta(-L/2) &= 0, \\ M(-L/2) &= \lambda_2, \\ F_n(-L/2) &= \lambda_3. \end{aligned} \tag{6.9.3}$$

We may now step forward in time using a time-stepping routine, such as Matlab's `ode45` which uses a variable step Runge-Kutta method. At $s = L/2$ we obtain

$$x_{L/2}(\lambda_1, \lambda_2, \lambda_3) = x(L/2), \tag{6.9.4}$$

$$y_{L/2}(\lambda_1, \lambda_2, \lambda_3) = y(L/2), \tag{6.9.5}$$

$$\theta_{L/2}(\lambda_1, \lambda_2, \lambda_3) = \theta(L/2). \tag{6.9.6}$$

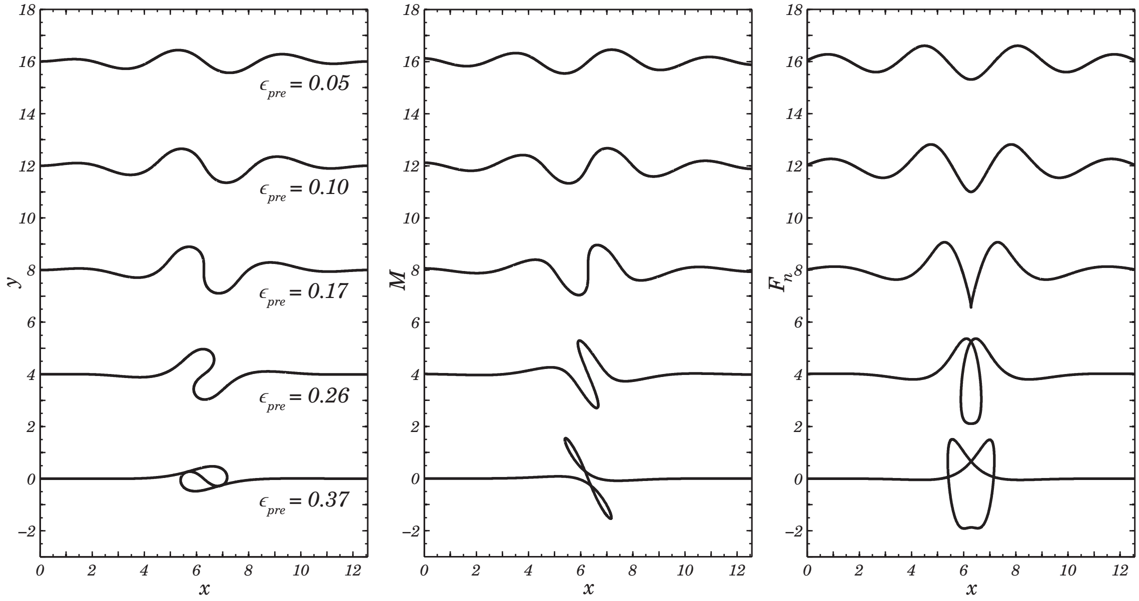


Figure 6.9.1: Odd type equilibrium shapes for different values of ϵ_{pre} along with the internal moments M and the normal component of the internal force F_n obtained by solving (6.9.7). Plots were shifted vertically by 4 for spacing.

If a given triplet of λ-values yields a solution that satisfies (6.4.7), (6.4.8), and (6.9.1), then

$$\mathcal{E}(\lambda_1, \lambda_2, \lambda_3) = \left[\begin{array}{ccc} x_{L/2} - \frac{L}{1+\epsilon_{pre}}, & y_{L/2}, & \theta_{L/2} \end{array}\right]^T \tag{6.9.7}$$

will equal the zero vector. That is, $\mathcal{E}$ is a mapping from $\mathbb{R}^3$ to $\mathbb{R}^3$ whose roots correspond to solutions of the BVP. These roots can be found using Newton's method. We use Matlab's `fsolve` routine to solve for them.

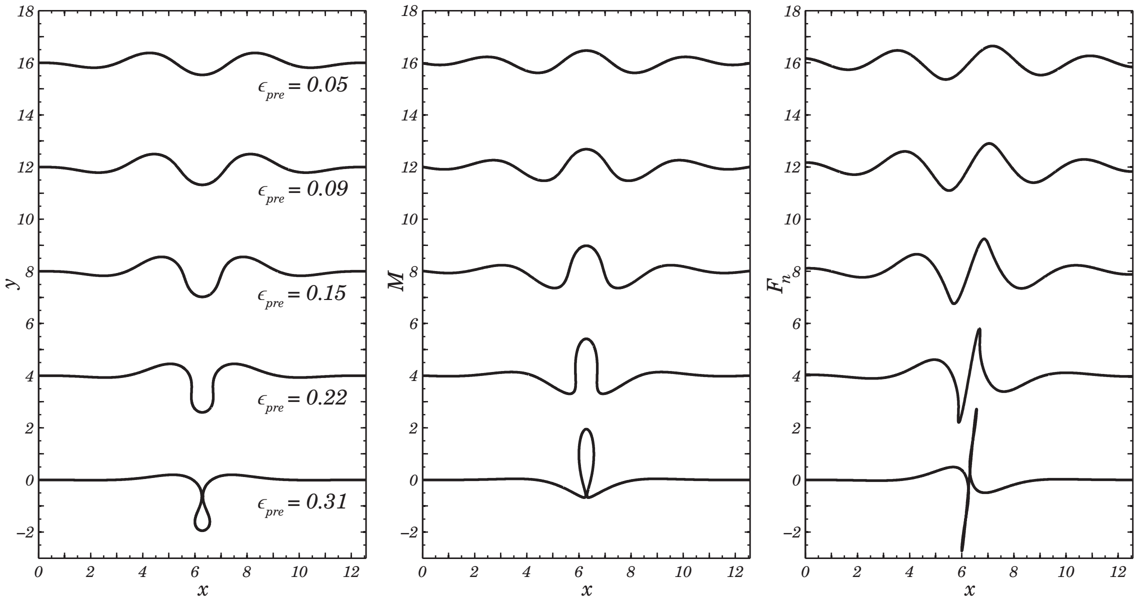


Figure 6.9.2: Even type equilibrium shapes for different values of ϵ_{pre} along with the internal moments M and the normal component of the internal force F_n obtained by solving (6.9.7). Plots were shifted vertically by 4 for spacing.

CHAPTER 7

Application: Gating of a mechanosensitive channel due to cellular flows

This work was first published in [29].

7.1 Introduction

Mechanosensitive (MS) channels are essential to mechanosensation and mechanotransduction in a wide range of cells [69, 70, 71]. Due to the great diversity of MS channels, the general gating mechanism is found to depend on combinations of the detailed molecular structures [72, 73, 74, 75], the gating-associated conformational changes [76, 77, 78], and coupling with the lipid bilayer membrane [79, 80, 81, 82]. It remains a great challenge to elucidate general mechanisms underpinning the gating of MS channels. In this spirit it is useful to have "simple" models to understand the complex response of MS channels and their associated biological functions [83].

Stretch-activated (SA) channels are a class of (relatively) simpler MS channels that are stretched open mainly by membrane tension (e.g. due to osmotic shock or stress from fluid flow) for nonselective intracellular transport of ions and macromolecules [84, 85, 86, 87]. Their gating mechanisms have been investigated by experiments [86], continuum modeling [88] and molecular dynamics (MD) simulations [89]. By exerting an unphysiologically large load onto a membrane patch with a single SA channel in the center, MD simulations show that a SA channel (several nanometers in size) responds to membrane tension within a few nanoseconds [89]. Due to computational limitations MD simulations are restricted to a small lipid patch and a short time scale (approximately microseconds). On the other hand, continuum modeling has been widely adopted in recent studies where the transduction of membrane tension to SA channels is found to depend on the molecular details of lipid binding in the channels [81]. Once stretched open, continuum modeling shows that the electrokinetic gating still depends on the protein surface charge and hydrophobicity [90].

Novel technological advancement in microfluidics has made it possible to construct artificial vesicles (of tens of μm in size) reconstituted with MS channels [91]. Furthermore recent experiments on dynamic conductance of MS channels in vesicles show that the oscillatory fluid stress in glass pipettes is sufficient to cause large tension in the membrane and the gating of MS channels [92]. Such idealized model system for gating of MS channels in cells is suitable for elucidating the detailed gating mechanism coupled with the membrane dynamics, which is intrinsically multiscale both in space and time. Coarse-grained MD simulations of such systems would take too long given the current computational power, and thus are not practical.

Motivated by recent success of continuum modeling of MS channels [90, 81, 88], in this paper we propose a multiscale continuum model to study the gating of a single MS channel by tension in a membrane exposed to fluid stress. In particular we aim to address how a nanosize MS channel can be activated by fluid stresses in a microfluidic channel by focusing on two typical physiological flows: a planar shear flow and flow through a narrowing constriction. The modeling work focuses on a SA channel, and for simplicity we refer to it generally as a MS channel for the rest of the paper.

In a planar shear flow, a vesicle membrane exhibits tank-treading, tumbling, breathing and ventilating modes of motion depending on the shear rate, viscosity mismatch (between inner and outer fluids), and vesicle excess area (or reduced volume). Numerical results show that the membrane tension varies both spatially and temporally as it deforms under

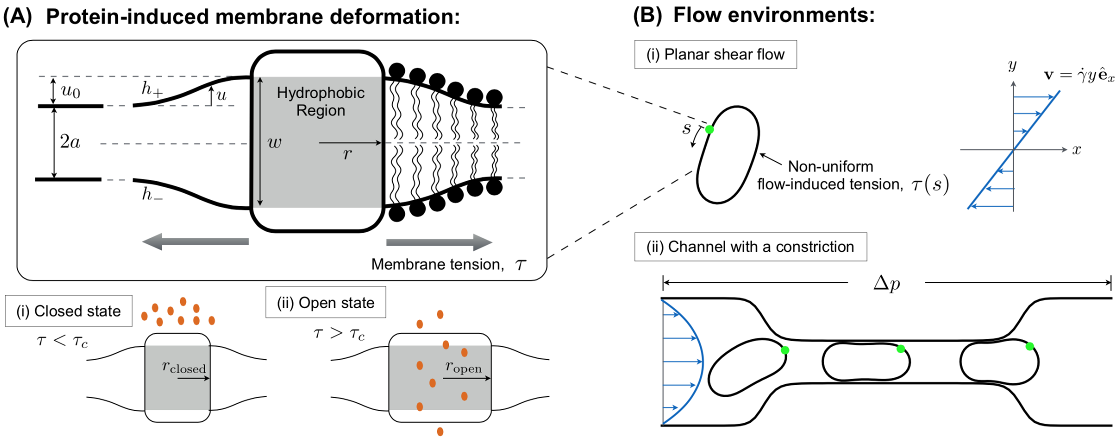


Figure 7.1.1: Multiscale characterization of a stretch-activated, membrane-bound channel in a flow. (A) Schematic representation of the protein-induced membrane deformation. Ions either are unable or allowed to cross the membrane depending, respectively, on whether the channel is closed (i) or open (ii) states. (B) (i) A 2D vesicle in a shear flow, with s as the arc length along the membrane. (ii) A vesicle entering and translating through a constriction due to a pressure-driven flow.

fluid stress. Since the MS channel size is comparable to the membrane thickness and they are both smaller than the size of the vesicle, the interaction between a single channel and the lipid bilayer membrane dynamics is a one-way coupling within the continuum modeling: the membrane hydrodynamics is not affected by the channel state, which depends on the membrane tension at the channel location. In addition, the diffusive transport of a transmembrane protein is negligible compared to the velocity on the vesicle under typical flow strengths in the microfluidic experiments. Therefore, we neglect the diffusive transport of the MS channel in the vesicle membrane. Instead, we will focus on the channel state (open or closed) as it moves along the vesicle (at the membrane velocity), experiencing the local membrane tension that results from vesicle hydrodynamics in two flow configurations: a planar shear flow and a pressure-driven flow across a narrow constriction. Results from these analyses and numerical simulations will shed light on hydrodynamically induced molecular transport across lipid bilayer membranes [93], which are relevant, for example, to the induced ATP release from red blood cells (RBCs) in blood flow [94].

7.2 Formulation

The complex interaction between a MS channel and the membrane has been simplified in continuum modeling [95, 88], where the total free energy of the membrane-protein system consists of a free energy for the membrane thickness variation, a free energy for the membrane shape, and a free energy due to the membrane tension. The model assumes a planar lipid bilayer membrane with a constant tension and has also been extended to the interaction between two MS channels [96]. For a constant membrane tension, the mechanical gating of a MS channel is consistent with experimental observations [88]. In reality, however, the tension of the cytoplasmic membrane varies in position and time as the cell encounters changes in hydrodynamic stress, which occurs as RBCs enter or exit narrowing constrictions similar to arterioles and capillaries [97]. In experiments, it is impossible to measure the tension along the cell membrane as the cell deforms when it flows through such geometries. On the other hand, theoretical analyses [98] and numerical simulations [99, 24, 100] of continuum models show that the membrane tension varies significantly as an RBC or vesicle undergoes large deformation under flow. Therefore in our multiscale modeling, we allow the tension τ (force/length) to vary with arc length s along a two-dimensional (2D) membrane of length l (Fig. 7.1.1B).

We denote the position of the lipid bilayer membrane leaflets as h_+ and h_- (Fig. 7.1.1). Without the MS channel the bilayer membrane is assumed to have a uniform thickness $2a$ at equilibrium. In the presence of a MS channel, the mismatch between the hydrophobic region of the protein w and the bilayer equilibrium thickness $2a$ (see Fig. 7.1.1 for notations) induces a thickness variation u along the membrane, denoted as $u = (h_+ - h_- - 2a)/2$. Following the development of Wiggins and Phillips [88] and Haselwandter and Phillips [96, 101], the free energy E of a cylindrical MS channel (of radius r, see Fig. 7.1.1) embedded in a lipid bilayer membrane can be formulated in the continuum framework as

$$E = G_h 2\pi r - \tau_0 \pi r^2, \tag{7.2.1}$$

where G_h corresponds to the free energy per unit length associated with the membrane thickness variation due to the hydrophobic mismatch[1], and $\tau_0 \pi r^2$ is the free energy of the loading associated with the local membrane tension $\tau_0 = \tau(s = 0, t)$ at the channel. Note that the channel radius can only vary within a range $r_{\text{closed}} \leq r \leq r_{\text{open}}$ due to steric constraints [88]. The channel is in the closed state when the minimum total free energy occurs at $r = r_{\text{closed}}$; similarly, the channel attains an open state when the minimum total free energy occurs at $r = r_{\text{open}}$.

As we next describe, the state of the channel is determined by the tension, which requires us to consider the membrane shape. The free energy per unit length G_h [88] is given by

$$G_h = \frac{1}{2} \int_0^l \left[K_b (\nabla^2 u)^2 + K_t \left(\frac{u}{a} \right)^2 + \tau \left(\frac{2u}{a} + (\nabla u)^2 \right) \right] ds, \tag{7.2.2}$$

where K_b is the bending rigidity of the lipid bilayer, K_t is the stiffness associated with membrane thickness deformation. The variation of Eq. 7.2.2 gives the governing equation for u:

$$\nabla^4 u + \frac{K_t}{K_b a^2} u - \frac{\nabla \cdot (\tau \nabla u)}{K_b} + \frac{\tau}{K_b a} = 0. \tag{7.2.3}$$

At the MS channel ($s = 0$ and $s = l$), the thickness variation u is constrained by the hydrophobic mismatch

$$u|_{s=0} = u|_{s=l} = u_0, \tag{7.2.4}$$

and we assume the bilayer membrane meets the protein at zero slope [102],

$$\left. \frac{\partial u}{\partial s} \right|_{s=0} = \left. \frac{\partial u}{\partial s} \right|_{s=l} = 0. \tag{7.2.5}$$

The MS channel is coupled to the membrane hydrodynamics via the membrane tension τ, which is determined from the force balance on an elastic inextensible 2D vesicle in a

[1] We follow the assumption in Wiggins and Phillips [88] that the energy per unit length can be used for the circular deformation field around a cylindrically symmetric ion channel, so that the total free energy due to membrane thickness variation is approximated by multiplying the energy per unit length G_h by the circumference of the channel, $2\pi r$.

Dimensional Parameters	Symbols	Values
Bending modulus	K_b	20 kT
Expansion modulus	K_t	30 kT/nm^2
Hydrophobic mismatch	$2u_0$	0.2 nm
Undeformed bilayer thickness	$2a$	4 nm
Channel radius at the closed state	r_{closed}	3 nm
Channel radius at the open state	r_{open}	5 nm
Equivalent vesicle radius	R_0	0.1 μm

Dimensionless Parameters	Definitions	Values
α	$K_t R_0^2/K_b$	15000
γ	R_0^2/a^2	2500

Table 7.1: Upper table: Summary of physical parameters of the bilayer membrane and the MS channel and their values used in this study. Lower table: Definitions of dimensionless parameters and their corresponding typical values.

Stokes flow as described next.

Vesicle dynamics. The dynamics of an elastic inextensible 2D vesicle suspended in a flow is governed by a balance between the nonlocal hydrodynamic forces and the elastic forces due to bending and tension along the membrane. We use the boundary integral formulation [103, 24] to solve for the vesicle shape evolution and dynamics. Our goal is to extract the membrane tension for given flow conditions via numerical simulations and use the tension distribution $\tau(s)$ to determine the thickness variation $u(s)$ by solving Eq. 7.2.3. Then we determine G_h and E using, respectively, Eq. 7.2.2 and 7.2.1.

Parameter values and scaling. The values of the model parameters depend on specific proteins and membrane properties. We illustrate our model using values that are of the same order of magnitude as typical measured values [95, 88] (Table 1, upper panel). We scale the variables to identify relevant dimensionless parameters in this system (Table 1, lower panel). The thickness deformation u is scaled by the undeformed thickness of a single leaflet a and the arc length along the 2D enclosed membrane s is scaled by $R_0 = l/2\pi$, which is the radius of a circle having the same perimeter as the vesicle, i.e. $s/R_0 \in [0, 2\pi]$. The membrane tension and free energy of a MS channel are scaled, respectively, by K_b/R_0^2 and $K_b a^2/R_0^2$. In the following we highlight the energy E (defined in Eq. 7.2.1), which dictates the state of the MS channel.

7.3 Results

We investigate the effects of different mechanical environments on the gating of a MS channel, which in general render the membrane tension nonuniform. Before presenting the results for different cellular-type flows, we first illustrate the features of the model by prescribing a uniform tension along the vesicle, which extends the results of Wiggins and Phillips [88] to the case of an enclosed membrane. In particular, we derive the critical tension for opening a MS channel.

Uniform membrane tension. The analysis is simplified when a uniform membrane tension $\tau(s)/(K_b/R_0^2) = \bar{\tau}$ is prescribed in Eq. 7.2.3. This case allows an analytical solution for the thickness variation $u(s)$ and hence the total free energy of a MS channel. Given the parameter values in Table 7.2 and a prescribed $\bar{\tau}$, the free energy E is only a function of the channel radius r (Eqs. 7.2.1), which is shown in Fig. 7.3.1 for different values of uniform membrane tension. Due to steric constraints, the channel radii outside the range $[r_{\text{closed}}, r_{\text{open}}] = [3, 5]$ nm are not accessible (shaded yellow). For $\bar{\tau} = 14$ (blue line), the minimum energy is attained at $r = r_{\text{closed}}$, i.e. the MS channel remains closed.

The channel is expected to open when the membrane tension is sufficiently large. The critical tension $\bar{\tau}_c$ at which the channel becomes open can be calculated by the condition that the energies of the open and closed states are equal, which results in a transcendental equation for $\bar{\tau}_c$

$$\frac{\bar{\tau}_c}{G_h(\bar{\tau}_c)} = \frac{2}{r_{\text{open}} + r_{\text{closed}}}, \tag{7.3.1}$$

where $\gamma = R_0^2/a^2$. Should G_h be independent of the membrane tension as in the results from [95, 88], the above expression is no longer transcendental and reduces to that given in [95, 88] for the planar case. With the parameter values in Table 7.2, we solve the equation numerically for the critical tension, $\bar{\tau}_c = 16.3$. The corresponding energy landscape at this

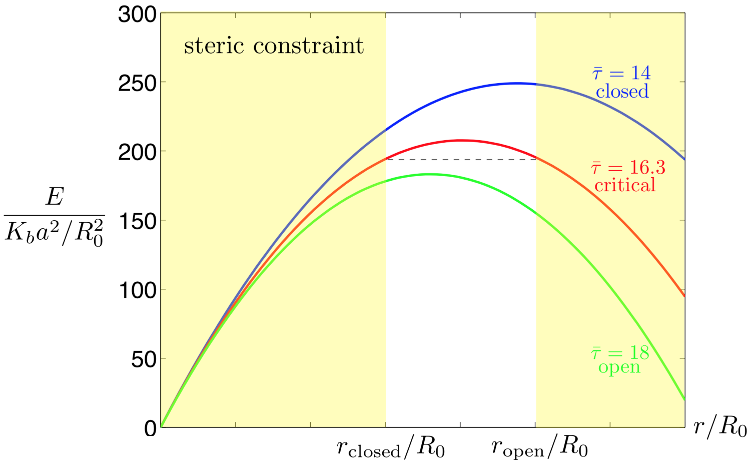


Figure 7.3.1: Energy landscape of a MS channel as a function of (dimensionless) channel radius r, for different values of uniform membrane tension: $\bar{\tau} = 14$ (blue line), $\bar{\tau} = 16.3$ (red line), and $\bar{\tau} = 18$ (green line), which correspond to a closed, critical, and open states respectively. The yellow-shaded regions ($r < r_{\text{closed}} = 3$ nm and $r > r_{\text{open}} = 5$ nm) are not accessible due to steric constraints.

critical membrane tension is shown in Fig. 7.3.1 (red line). Beyond this critical tension (e.g. $\bar{\tau} = 18$), the lowest energy is attained at $r = r_{\text{open}}$ (green line in Fig. 7.3.1), i.e. the MS channel is in the open state. Thus, in the spirit of [95, 88] we have demonstrated an algorithm for membrane tension regulating the opening and closing of a MS channel.

Planar linear shear flow. Shear stress-induced membrane poration has been utilized for intracellular transport of macromolecules [104]. Here we utilize our multiscale model to address if a steady shear flow can induce gating of MS channels in a vesicle. A vesicle in a planar shear flow deforms and the membrane tension varies spatially as a result of inextensibility. For simplicity, we assume that the fluid viscosities inside and outside the vesicle are the same, and hence the vesicle takes a steady equilibrium shape in shear flow with a tank-treading motion along the membrane (rolling motion of the membrane about its interior while maintaining a constant shape) [98]. The shear flow is characterized by a dimensionless shear rate (or a capillary number), given by Ca $= \dot{\gamma}\mu R_0^3/K_b$. For a vesicle with $R_0 \approx 0.1$–10 μm, physiological shear rates $\dot{\gamma} \approx 1$–10^3 s^{-1} and $\mu \approx 1$ m Pa s, then Ca ≈ 0.25–10. For a given Ca, the vesicle attains a steady-state shape with an inclination

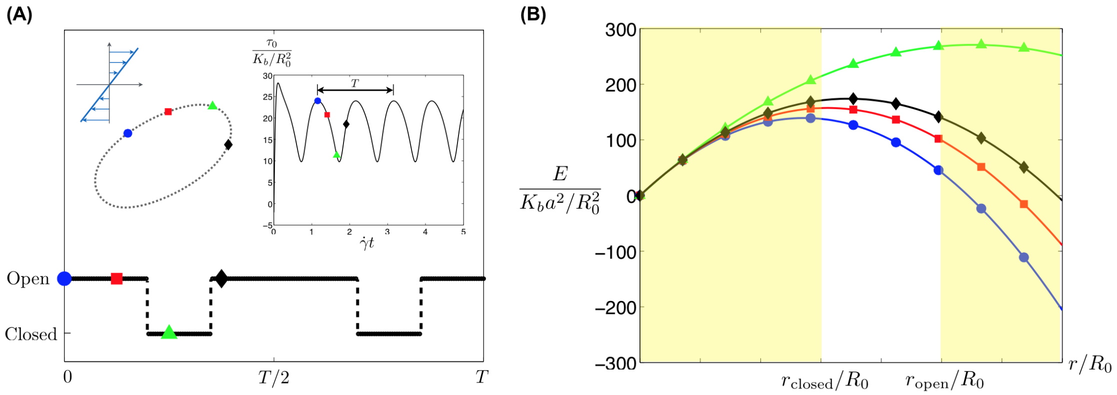


Figure 7.3.2: Gating of a mechanosensitive channel on a vesicle subject to a shear flow at Ca $= 8$. (A) The state (open/closed) of the MS channel as a function of time in a period T, where insets show the steady-state shape of the vesicle (left) and the local membrane tension at the MS channel location as a function of time. (B) Energy landscape of the MS channel as a function of channel radius r at different times (or locations on the membrane). The blue dot, red square, green triangle, and black diamond correspond to $\dot{\gamma}t = 0, T/8, T/4, 3T/8$, respectively.

relative to the horizontal axis (see Fig. 7.3.2A inset for the shape at Ca $= 8$).

We normalize all times by $\dot{\gamma}^{-1}$ and define a period T as the time it takes for a Lagrangian marker on the tank-treading membrane to complete one cycle and return to its original position. Ignoring in-plane protein diffusion relative to the effects of the tank-treading speed, we assume that the MS channel moves with the local fluid velocity along the membrane. In the left inset of Fig. 7.3.2A, we consider a MS channel in the location represented by the blue dot, where the local membrane tension is maximum (Fig. 7.3.2A, right inset), and its subsequent locations along the membrane, over a half period, at fixed time intervals ($T/8$) by the red square, green triangle, and black diamond; the dynamics for the other half period is similar. The channel protein moves along the membrane and hence samples the local tension τ_0 at different times (Fig. 7.3.2A right inset). The energy landscape of the MS channel is therefore a periodic function of time or location along the membrane as shown in the inset to Fig 7.3.2A. As a result of the variations in tension, the state of the MS channel depends on its location on the membrane. Our model reveals the time history of the state of the MS channel as it travels along the membrane (Fig. 7.3.2A).

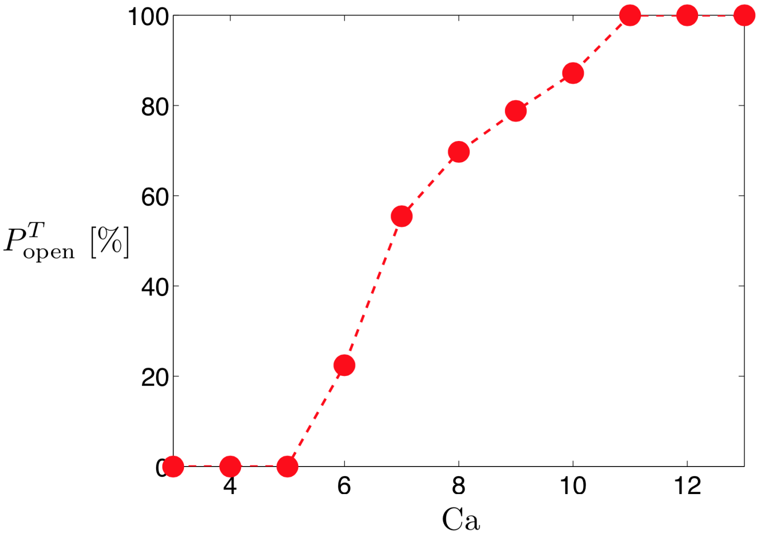


Figure 7.3.3: Percentage of time the MS channel spends in the open state over one period, P_{open}^T, as a function of the dimensionless shear rate, Ca.

At the location with maximum local membrane tension (blue dot in Fig. 7.3.2A), the MS channel is in the open state (see the corresponding energy landscape in Fig. 7.3.2B, blue line). The channel remains open until the protein approaches the elongated end of the vesicle (between the green triangle and black diamond in Fig. 7.3.2), where the free energy is minimum at $r = r_{\text{closed}}$ (Fig. 7.3.2B green line). As shown in the time history of local tension sampled by the moving MS channel (Fig. 7.3.2A right inset), the vesicle has low membrane tension around the elongated end, which is a main reason for the closed state being the more energetically favorable state in these regions. When the protein departs from the elongated end, it resumes the open state (black diamond in Fig. 7.3.2A and black line in Fig. 7.3.2B). For a low shear rate (e.g. Ca $= 3$), the lowest energy states always occur at the closed state, regardless of the location of the MS channel on the membrane. The MS channel therefore remains closed as the protein travels along the membrane. When Ca is sufficiently large, the MS channel can remain always open throughout its journey along the entire membrane. Such is the case, for instance, at Ca $= 13$. For a single MS channel traveling along the vesicle in one period, the percentage of time the MS channel spends in the open state in one period, P_{open}^T can be calculated as a function of Ca (Fig. 7.3.3).

With the parameter values in Table 7.2, the MS channel remains always closed when $\mathrm{Ca} \leq 5$ and becomes fully open when $\mathrm{Ca} \geq 11$ (Fig. 7.3.3). The results demonstrate quantitatively how varying the shear rate acts as a mechanism to switch on and off a MS channel, and control the membrane permeability.

Vesicle in a channel flow with a constriction. Finally, we consider how sudden changes in geometry affect the gating of MS channels. To do so we consider here a standard configuration where a vesicle with a MS channel is suspended in a pressure-driven channel flow with a constriction (Fig. 7.3.4A). Such geometries are common physiological and experimental environments; for instance, RBCs often go through constrictions 30–80% smaller than their diameters, and microfluidic flow channels with similar geometry have been utilized for intracellular delivery of macromolecules into vesicles [93]. Unlike the case of planar shear flows, the vesicle does not admit a steady-state shape but continuously deforms as it flows along the channel, especially when entering and leaving the constriction (see Fig. 7.3.4A for the vesicle shape at different locations in the channel). In this case, a control parameter is the pressure drop Δp across the constriction.

The vesicle therefore has different tension profiles along the membrane, depending on its location in the channel. For the vesicle locations in the channel displayed in Fig. 7.3.4A, the corresponding tensions $\tau(s)$ are shown in Fig. 7.3.4B and 7.3.4C. We observe that the membrane tension generally increases as the vesicle enters the constriction and decreases when it leaves the constriction. It is therefore expected that the constriction can serve as an external mechanism to control the ion channel gating on a vesicle. Due to the increased membrane tension, an ion channel may open during its time inside the constriction. For the case of a vesicle tank-treading in shear flows discussed earlier, a single MS channel travels along the membrane and experiences local membrane tension in one period. In contrast, for the case of a channel flow with a constriction, the MS channel does not travel around the entire membrane when the vesicle flows through the channel (the green dot in Fig. 7.3.4A). The initial location of the MS channel is therefore important because the channel

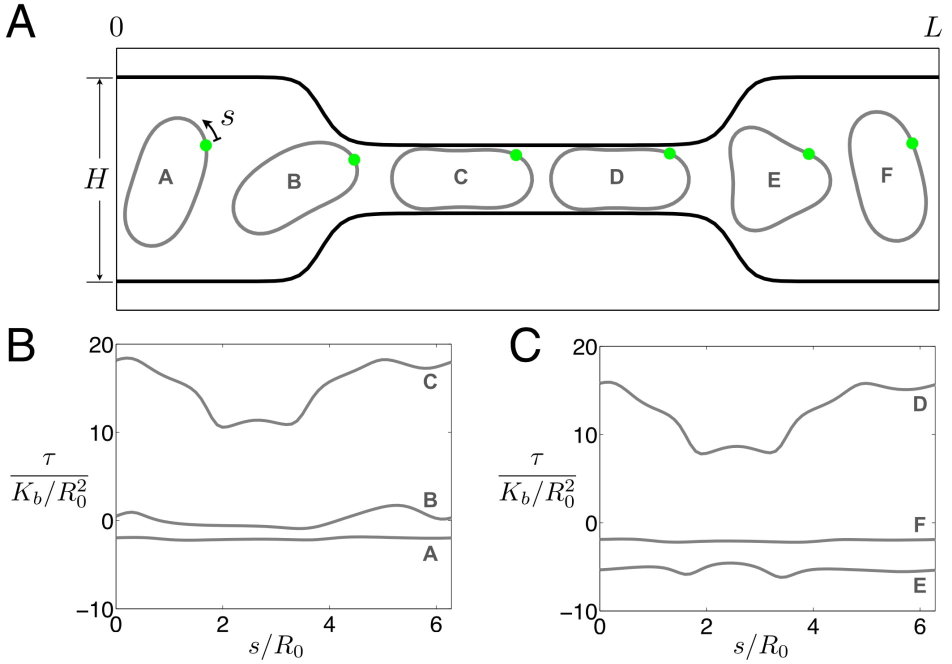


Figure 7.3.4: Dynamics of a 2D vesicle through a channel with a constriction. (A) Deformation of a vesicle at different locations inside the channel. The green dot represents the location of the MS channel ($s = 0$). (B,C) Membrane tension profiles along the vesicle at the corresponding locations inside the channel shown in (A). Geometric parameters of the channel and constriction (scaled by R_0): $L/R_0 = 15.56$, $H/R_0 = 3.86$. The length and width of the constriction are, respectively, $L/2$ and $H/3$. The nondimensionalized pressure difference is $\frac{\Delta p R_0^3 H}{K_b L} = 8.17$.

protein only samples a portion of the tension profile along the membrane. To quantify the MS channel dynamics using our multiscale model, we place the MS channel at one of 64 different locations on the membrane, represented by symbols along the vesicle in Fig. 7.3.5A. We then determine whether the MS channel would be in the closed or open state (represented by blue asterisks and red open circles respectively) at these locations on the vesicle as it passes through the constriction. When the vesicle enters the constriction, the MS protein in the front half of the membrane begins to open (Fig. 7.3.5A). The percentage of open MS channels, P^N_{open}, is calculated as a function of the vesicle position as it travels through the channel (Fig. 7.3.5B). A maximum of approximately 53% of MS channels on the membrane would be open when the vesicle is about midway in the constriction. Note that since the MS channel does not travel around the vesicle in this case, a MS channel initially located in the low membrane tension region may stay closed as the vesicle enters and leaves the constriction.

7.4 Discussion

We have presented a multiscale continuum formulation to couple the open/closed dynamics of a MS channel with the dynamics of a lipid bilayer membrane under flow. For a constant membrane tension the continuum model gives analytical results for the critical tension for opening a MS channel. When our multiscale model is applied to a vesicle in a planar shear flow our results indicate that, for a tank-treading vesicle, the embedded MS channel can be open for sufficiently large shear rate (Ca > 5), and an optimal shear rate for the highest membrane permeability is achieved for Ca > 11. Moreover, our model of the MS channel combined with direct numerical simulations of a vesicle going through a microfluidic channel with a constriction show that MS channel is open mostly right after the entry to the constriction where the most deformation is observed. As the vesicle adjusts itself inside the constriction, the tension reduces in magnitude and the MS channel is closed. We believe

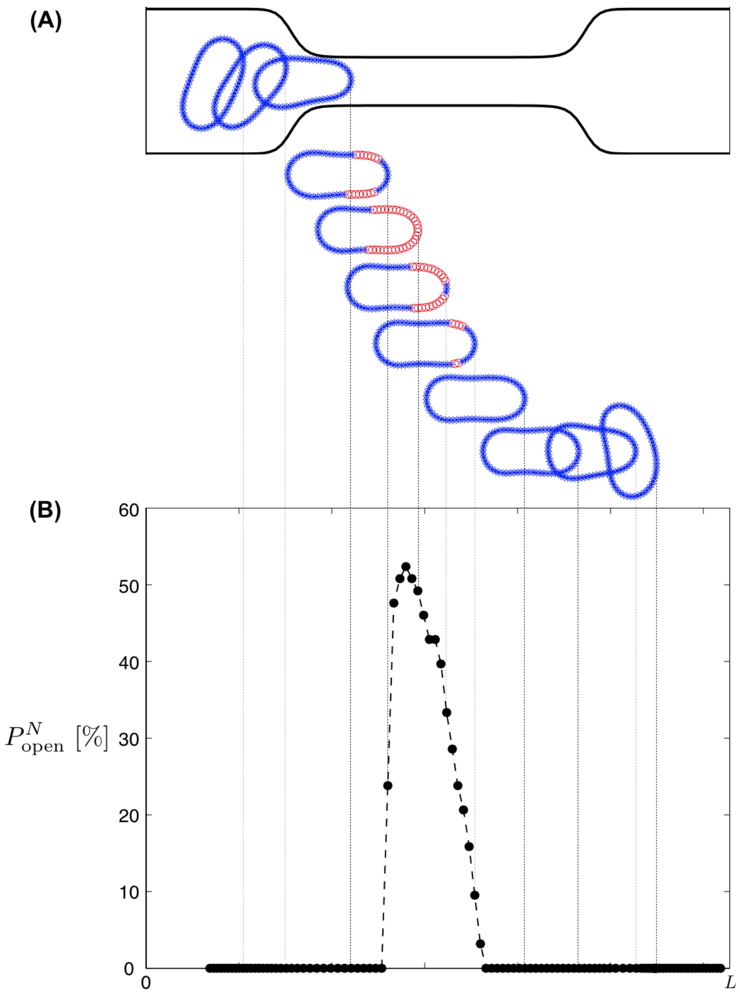


Figure 7.3.5: (A): Visualization of the state of the MS channel as the vesicle flows inside a channel. Symbols (blue asterisks and red open circles) on the membrane represent 64 different locations of the MS channel on the membrane. The closed state is represented by blue asterisks and the open state is represented by red open circles. (B): Percentage of the number of locations where the MS channel is at the open state as the vesicle passes through different locations along the channel.

these results and our basic approach will be useful for designing microfluidic flow channels for shear stress-induced intracellular delivery and may offer another interpretation of the extracellular ATP release of RBCs in such microfluidic flow configurations.

Many methods, such as electroporation, sonoporation, and shear stress-induced poration, have been utilized for intracellular delivery of drugs and transfected DNAs. Results from our work imply a different intracellular delivery method based on the gating of a transmembrane MS channel reconstituted in the lipid bilayer membranes: the MS channel is mechanically gated by the membrane tension, which reaches large values as the membrane deforms under fluid flow that is physiologically feasible and experimentally realizable in microfluidic flow configurations. These results are consistent with recent experimental findings in [92]. While membrane integrity (resealing of the membrane) is an important issue for externally enhanced membrane poration in other intercellular delivery methods, our proposed method retains the membrane integrity as the gating of a MS channel by membrane can be achieved before cell lysis under fluid stress (the lysis tension is much larger than the critical tension for opening a MS channel by fluid stress). In addition the cell dynamics and motion under fluid flow are easier to control than under an external electric field where fluid flow has been added to enhance the performance of electroporation in terms of cell vitality and rate of poration in a suspension of cells.

In our multiscale model, the MS channel is assumed to be a simple cylinder across the double leaflets. Once stretched open by tension in the lipid membrane under fluid stress, the intracellular transport of ions or macromolecules through the MS channel may depend on the detailed molecular structures of the MS channel (such as the protein surface charge) and its coupling with the surrounding lipids [77, 81]. We are working on incorporating these details into our multiscale continuum model to quantify the mechanosensitivity and efficiency of mechanotransduction of fluid stress by MS channels.

BIBLIOGRAPHY

[1] Barnett, A. H., Wu, B., and Veerapaneni, S., "Spectrally-accurate quadratures for evaluation of layer potentials close to the boundary for the 2D Stokes and Laplace equations," *SIAM J. Sci. Comput.*, Vol. 37, No. 4, 2015, pp. B519–B542.

[2] Pocivavsek, L., Dellsy, R., Kern, A., Johnson, S., Lin, B., Lee, K. Y. C., and Cerda, E., "Stress and Fold Localization in Thin Elastic Membranes," *Science*, 2008.

[3] Huang, R. B., Mocherla, S., Heslinga, M. J., Charoenphol, P., and Eniola-Adefeso, O., "Dynamic and cellular interactions of nanoparticles in vascular-targeted drug delivery (review)." *Molecular membrane biology*, Vol. 27, No. 4-6, Aug. 2010, pp. 190–205.

[4] Olbricht, W., "Pore-scale prototypes of multiphase flow in porous media," *Annual review of fluid mechanics*, Vol. 28, No. 1, 1996, pp. 187–213.

[5] Jin, C., McFaul, S. M., Duffy, S. P., Deng, X., Tavassoli, P., Black, P. C., and Ma, H., "Technologies for label-free separation of circulating tumor cells: from historical foundations to recent developments," *Lab on a Chip*, Vol. 14, No. 1, 2014, pp. 32–44.

[6] Decuzzi, P., Godin, B., Tanaka, T., Lee, S.-Y., Chiappini, C., Liu, X., and Ferrari, M., "Size and shape effects in the biodistribution of intravascularly injected particles." *Journal of controlled release : official journal of the Controlled Release Society*, Vol. 141, No. 3, Feb. 2010, pp. 320–7.

[7] Russom, A., Gupta, A. K., Nagrath, S., Di Carlo, D., Edd, J. F., and Toner, M., "Differential inertial focusing of particles in curved low-aspect-ratio microchannels." *New journal of physics*, Vol. 11, No. 7, July 2009, pp. 75025.

[8] Vlahovska, P. M., Barthes-Biesel, D., and Misbah, C., "Flow dynamics of red blood cells and their biomimetic counterparts," *Comptes Rendus Physique*, Vol. 14, No. 6, 2013, pp. 451–458.

[9] Helfrich, W., "Elastic properties of lipid bilayers: theory and possible experiments," *Zeitschrift für Naturforschung C*, Vol. 28, No. 11-12, 1973, pp. 693–703.

[10] K., Y. G. and Acrivos, A., " Stokes flow past a particle of arbitrary shape: a numerical method of solution," *Journal of Fluid Mechanics*, Vol. 69, May 1975, pp. 377–403.

[11] K., Y. G. and Acrivos, A., “ On the shape of a gas bubble in a viscous extensional flow,” *Journal of Fluid Mechanics*, Vol. 76, August 1976, pp. 433–442.

[12] Pozrikidis, C., “Interfacial dynamics for Stokes flow,” *Journal of Computational Physics*, Vol. 169, No. 2, 2001, pp. 250–301.

[13] Ewald, P. P., “Die Berechnung optischer und elektrostatischer Gitterpotentiale,” *Annalen der Physik*, Vol. 369, No. 3, 1921, pp. 253–287.

[14] Hasimoto, H., “On the periodic fundamental solutions of the Stokes equations and their application to viscous flow past a cubic array of spheres,” *Journal of Fluid Mechanics*, Vol. 5, No. 02, 1959, pp. 317–328.

[15] Pozrikidis, C., “Computation of periodic Green’s functions of Stokes flow,” *Journal of engineering Mathematics*, Vol. 30, No. 1-2, 1996, pp. 79–96.

[16] Darden, T., York, D., and Pedersen, L., “Particle mesh Ewald: An $N\log(N)$ method for Ewald sums in large systems,” *J. Chem. Phys.*, Vol. 98, 1993, pp. 10089.

[17] Lindbo, D. and Tornberg, A.-K., “Spectrally accurate fast summation for periodic Stokes potentials,” *Journal of Computational Physics*, Vol. 229, No. 23, 2010, pp. 8994–9010.

[18] Af Klinteberg, L. and Tornberg, A.-K., “Fast Ewald summation for Stokesian particle suspensions,” *International Journal for Numerical Methods in Fluids*, Vol. 76, No. 10, 2014, pp. 669–698.

[19] Wang, M. and Brady, J. F., “Spectral Ewald Acceleration of Stokesian Dynamics for polydisperse suspensions,” *arXiv preprint arXiv:1506.08481*, 2015.

[20] Gholami, A., Malhotra, D., Sundar, H., and Biros, G., “FFT, FMM, or MULTIGRID? A comparative study of state-of-the-art poisson solvers,” *arXiv preprint arXiv:1408.6497*, 2014.

[21] Barnett, A. H. and Greengard, L., “A new integral representation for quasi-periodic scattering problems in two dimensions,” *BIT Numer. Math.*, Vol. 51, 2011, pp. 67–90.

[22] Cho, M. H. and Barnett, A. H., “Robust fast direct integral equation solver for quasi-periodic scattering problems with a large number of layers,” *Opt. Express*, Vol. 23, No. 2, 2015, pp. 1775–1799.

[23] Liu, Y. and Barnett, A., “Efficient numerical solution of acoustic scattering from doubly-periodic arrays of axisymmetric objects,” 2015, submitted, *J. Comput. Phys.*; `arxiv:1506.05083`.

[24] Veerapaneni, S. K., Gueyffier, D., Zorin, D., and Biros, G., “A boundary integral method for simulating the dynamics of inextensible vesicles suspended in a viscous fluid in 2D,” *Journal of Computational Physics*, Vol. 228, No. 7, 2009, pp. 2334–2353.

[25] Kress, R., "Boundary Integral equations in time-harmonic acoustic scattering," *Mathl. Comput. Modelling*, Vol. 15, 1991, pp. 229–243.

[26] Marple, G. R., Barnett, A., Gillman, A., and Veerapaneni, S., "A Fast Algorithm for Simulating Multiphase Flows Through Periodic Geometries of Arbitrary Shape," *SIAM Journal on Scientific Computing*, 2016 (accepted).

[27] Barnett, A., Marple, G. R., Veerapaneni, S., and Zhao, L., "A Unified Integral Equation Scheme for Doubly-Periodic Laplace and Stokes Boundary Value Problems in Two Dimensions," 2016.

[28] Marple, G. R., Purohit, P., and Veerapaneni, S., "Equilibrium Shapes of Planar Elastic Membranes," *Physical Review E*, Vol. 92, No. 1, 2015.

[29] Pak, O., Young, Y., Marple, G. R., Veerapaneni, S., and Stone, H. A., "Gating of a mechanosensitive channel due to cellular flows," *Proceedings of the National Academy of Sciences*, Vol. 112, No. 32, 2015.

[30] Ojala, R. and Tornberg, A.-K., "An accurate integral equation method for simulating multi-phase Stokes flow," *Journal of Computational Physics*, Vol. 298, No. C, October 2015, pp. 145–160.

[31] Klöckner, A., Barnett, A. H., Greengard, L., and O'Neil, M., "Quadrature by expansion: a new method for the evaluation of layer potentials," *J. Comput. Phys.*, Vol. 252, No. 1, 2013, pp. 332–349.

[32] Ladyzhenskaya, O. A., *The Mathematical Theory of Viscous Incompressible Flow, revised 2nd edition*, Mathematics and Its Applications 2, Gordon and Breach, 1969.

[33] Hsiao, G. and Wendland, W. L., *Boundary Integral Equations*, Applied Mathematical Sciences, Vol. 164, Springer, 2008.

[34] Liu, Y., *Fast Multipole Boundary Element Method: Theory and Applications in Engineering*, Cambridge University Press, 2009.

[35] Bogomolny, A., "Fundamental Solutions Method for Elliptic Boundary Value Problems," *SIAM J. Numer. Anal.*, Vol. 22, No. 4, 1985, pp. 644–669.

[36] Barnett, A. H. and Betcke, T., "Stability and convergence of the Method of Fundamental Solutions for Helmholtz problems on analytic domains," *J. Comput. Phys.*, Vol. 227, No. 14, 2008, pp. 7003–7026.

[37] Gillman, A., Young, P., and Martinsson, P., "A direct solver with $O(N)$ complexity for integral equations on one-dimensional domains," *Frontiers of Mathematics in China*, Vol. 7, No. 2, 2012, pp. 217–247.

[38] Kress, R., *Numerical Analysis*, Graduate Texts in Mathematics #181, Springer-Verlag, 1998.

[39] Kress, R., *Linear Integral Equations*, Vol. 82 of *Appl. Math. Sci.*, Springer, 2nd ed., 1999.

[40] Gillman, A. and Barnett, A., “A fast direct solver for quasiperiodic scattering problems,” *J. Comput. Phys.*, Vol. 248, 2013, pp. 309–322.

[41] Marple, G. R., Barnett, A. H., Gillman, A., and Veerapaneni, S., “A fast algorithm for simulating multiphase flows through periodic geometries of arbitrary shape,” 2015, submitted, SIAM J. Sci. Comput.

[42] Sifuentes, J., Gimbutas, Z., and Greengard, L., “Randomized methods for rank-deficient linear systems,” *Electron. Trans. Numer. Anal.*, Vol. 44, 2015, pp. 177–188.

[43] Gimbutas, Z. and Veerapaneni, S., “A fast algorithm for spherical grid rotations and its application to singular quadrature,” *SIAM Journal on Scientific Computing*, Vol. 35, No. 6, 2013.

[44] Colton, D. and Kress, R., *Inverse acoustic and electromagnetic scattering theory*, Vol. 93 of *Applied Mathematical Sciences*, Springer-Verlag, Berlin, 2nd ed., 1998.

[45] Ascher, U. M. and Petzold, L. R., *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 1998.

[46] Tornberg, A. and Shelley, M., “Simulating the dynamics and interactions of flexible fibers in Stokes flows,” *Journal of Computational Physics*, Vol. 196, May 2004, pp. 8–40.

[47] Rahimian, A., Veerapaneni, S. K., and Biros, G., “Dynamic simulation of locally inextensible vesicles suspended in an arbitrary two-dimensional domain, a boundary integral method,” *Journal of Computational Physics*, Vol. 229, No. 18, 2010, pp. 6466–6484.

[48] Hou, T. Y., Lowengrub, J. S., and Shelley, M. J., “Removing the stiffness from interfacial flows with surface tension,” *Journal of Computational Physics*, Vol. 114, No. 2, 1994, pp. 312–338.

[49] Gimbutas, Z. and Greengard, L., “FMMLIB2D, Fortran libraries for fast multipole methods in two dimensions,” 2012, `http://www.cims.nyu.edu/cmcl/fmm2dlib/fmm2dlib.html`.

[50] Li, B., Cao, Y.-P., Feng, X.-Q., and Gao, H., “Mechanics of morphological instabilities and surface wrinkling in soft materials: a review,” *Soft Matter*, Vol. 8, No. 21, 2012, pp. 5728–5745.

[51] Diamant, H. and Witten, T. A., “Compression induced folding of a sheet: An integrable system,” *Physical review letters*, Vol. 107, No. 16, 2011, pp. 164302.

[52] Oshri, O., Brau, F., and Diamant, H., "Wrinkles and folds in a fluid-supported sheet of finite size," *arXiv preprint arXiv:1503.01674*, 2015.

[53] Rivetti, M., "Non-symmetric localized fold of a floating sheet," *Comptes Rendus Mécanique*, Vol. 341, 2013, pp. 333–338.

[54] Rivetti, M. and Neukirch, S., "The mode branching route to localization of the finite-length floating elastica," *Journal of the Mechanics and Physics of Solids*, Vol. 69, 2014, pp. 143–155.

[55] Lipowsky, R., "The conformation of membranes," *Nature*, Vol. 349, 1991, pp. 475.

[56] Shao-guang, Z. and Zhong-can, O., "Periodic cylindrical solution for fluid bilayer membranes," *Phys. Rev. E*, Vol. 53, No. 4, 1996, pp. 4206–4208.

[57] Deng, Y., Marko, M., Buttle, K. F., Leith, A., Mieczkowski, M., and Mannella, C. A., "Cubic membrane structure in amoeba mitochondria determined by electron microscope tomography," *J. Struc. Biol.*, Vol. 127, 1999, pp. 231–239.

[58] Almsherqi, Z., Hyde, S., Ramachandran, M., and Deng, Y., "Cubic membranes: a structure-based design for DNA uptake," *J. R. Soc. Interface*, Vol. 5, 2008, pp. 1023–1029.

[59] Zeineldin, R., Last, J. A., Slade, A. L., Ista, L. K., Bisong, P., O'Brien, M. J., Brueck, S. R. J., Sasaki, D. J., and Lopez, G. P., "Using bicellar mixtures to form supported and suspended lipid bilayers on silicon chips," *Langmuir*, Vol. 22, 2006, pp. 8163–8168.

[60] Fan, J., Han, T., and Haataja, M., "Hydrodynamic effects on spinodal decomposition kinetics in planar lipid bilayer membranes," *The Journal of chemical physics*, Vol. 133, No. 23, 2010, pp. 235101.

[61] Young, Y.-N., Veerapaneni, S., and Miksis, M. J., "Long-wave dynamics of an inextensible planar membrane in an electric field," *Journal of Fluid Mechanics*, Vol. 751, 7 2014, pp. 406–431.

[62] Veerapaneni, S. K., Raj, R., Biros, G., and Purohit, P. K., "Analytical and numerical solutions for shapes of quiescent two-dimensional vesicles," *International Journal of Non-Linear Mechanics*, Vol. 44, 2009, pp. 257–262.

[63] Rim, J. E., Purohit, P. K., and Klug, W. S., "Mechanical collapse of confined fluid membrane vesicles," *Biomechanics and modeling in mechanobiology*, 2014, pp. 1–12.

[64] Antman, S. S., *Nonlinear problems of elasticity*, Vol. 107, Springer, 2005.

[65] Agrawal, N. J., Radhakrishnan, R., and Purohit, P. K., "Geometry of mediating protein affects the probability of loop formation in DNA," *Biophysical journal*, Vol. 94, No. 8, 2008, pp. 3150–3158.

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

[67] Alpert, B. K., "Hybrid Gauss-Trapezoidal Quadrature Rules," *SIAM Journal on Scientific Computing*, Vol. 20, No. 5, 1999, pp. 1551–1584.

[68] Saad, Y., *Iterative methods for sparse linear systems*, Siam, 2003.

[69] Kung, C., Martinac, B., and Sukharev, S., "Mechanosensitive channels in microbes," *Annu. Rev. Microbiol.*, 2010.

[70] Martinac, B. and Kloda, A., "Mechanosensory transduction," *Compr. Biophys.*, 2012.

[71] Sackin, H., "Mechanosensitive channels," *Annu. Rev. Physiol.*, 1995.

[72] Betanzos, M., Chiang, C.-S., Guy, H. R., and Sukharev, S., "A large iris-like expansion of a mechanosensitive channel protein induced by membrane tension," *Nature Struc. Biol.*, 2002.

[73] P., B., Sukharev, S. I., Schroeder, M. J., Nagle, S. K., and Kung, C., "Single residue substitutions that change the gating properties of a mechanosensitive channel in escherichia coli." *Proceedings of the National Academy of Sciences*, 1996.

[74] Sukharev, S. and Sachs, F., "Molecular force transductino by ion channels - diversity and unifying principles," *J. Cell Sci.*, 2012.

[75] Sukharev, S., Durell, S. R., and Guy, H. R., "Structural models of the mscl gating mechansim," *Biophys. J.*, 2001.

[76] Kong, Y., Shen, Y., Warth, T. E., and Ma, J., "Conformational pathways in the gating of escherichia coli mechanosensitive channel," *Proceedings of the National Academy of Sciences*, 2002.

[77] Samuli Ollila, O. H., Risselada, H. J., Louhivuori, M., Lindahl, E., Vattulainen, I., and Marrink, S. J., "3d pressure field in lipid membranes and membrane-protein complexes," *Phys. Rev. Lett.*, 2009.

[78] Yoshimura, K., Usukura, J., and Sokabe, M., "Gating-associated conformational changes in the mechanosensitive channel mscl," *Proceedings of the National Academy of Sciences*, 2008.

[79] Mukherjee, N., Jose, M. D., Brikner, J. P., Walko, M., Ingolfsson, H. I., Dimitrova, A., Arnarez, C., Marrink, S. J., and Kocer, A., "The activation mode of the mechnosensitive ion channel, mscl, by lysophosphatidylcholine differs from tension-induced gating," *FASEB*, 2014.

[80] Phillips, R., Ursell, T., Wiggins, P., and Sens, P., "Emerging roles for lipids in shaping membrane-protein function," *Nature*, 2009.

[81] Vanegas, J. M. and Arroyo, M., "Force tranduction an lipid binding in mscl: A continuum-molecular approach," *PLoS one*, 2014.

[82] Sukharev, S., Akitake, B., and Anishkin, A., "The bacterial mechanosensitive channel mscs: Emerging principles of gating and modulation," *Curr. Top. Membr.*, 2007.

[83] Sukharev, S. and Anishkin, A., "Mechanosensitive channels: what can we learn from 'simple' model systems?" *Trends Neurosci.*, 2004.

[84] Belyy, V., Anishkin, A., Kamaraju, K., Liu, N., and Sukharev, S., "The tension-transmitting 'clutch' in the mechansensitive channel mscs," *Nat. Struct. Mol. Biol.*, 2010.

[85] Hamill, O. P. and Martinac, B., "Molecular basis of mechanotransduction in living cells," *Physiol. Rev.*, 2001.

[86] Martinac, B., Buechner, M., Delcour, A. H., Adler, J., and Kung, C., "Pressure-sensitive ion channel in escherichia coli," *Proceedings of the National Academy of Sciences*, 1987.

[87] Sachs, F., "Mechanical transduction by membrane ion channels: a mini review," *Mol. Cellular Biochem.*, 1991.

[88] Wiggins, P. and Phillips, R., "Analytic models for mechanotransduction: Gating a mechanosensitive channel," *Proceedings of the National Academy of Sciences*, 2004.

[89] Jeon, J. and Voth, G. A., "Gating of the mechanosensitive channel protein MscL: The interplay of membrane and protein," *Biophys. J.*, 2008.

[90] Bonthuis, D. J. and Golestanian, R., "Mechanosensitive channel activation by diffusionosmotic force," *Phys. Rev. Lett.*, 2014.

[91] Sarles, S. A. and Leo, D. J., "Regulated attachment method for reconstituting lipid bilayers of prescribed size with flexible substrates," *Anal. Chem.*, 2010.

[92] Najem, J., Dunlap, M., Sukharev, S., and Leo, D. J., "Mechanosensitive channels activity in a droplet interface bilayer system," *MRS Proceedings*, 2014.

[93] Sharei, A., Zoldan, J., Adamo, A., Sim, W. Y., Cho, N., Jackson, E., Mao, S., Schneider, S., Han, M.-J., Lytton-Jean, A., Basto, P. A., Jhunjhunwala, S., Lee, J., Heller, D. A., Kang, J. W., Hartoularos, G. C., Kim, K.-S., Anderson, D. G., Langer, R., and Jensen, K. F., "A vector-free microfluidic platform for intracellular delivery," *Proceedings of the National Academy of Sciences*, 2013.

[94] Forsyth, A. M., Wan, J., Owrutsky, P. D., Abkarian, M., and Stone, H. A., "Multiscale approach to link red blood cell dynamics, shear viscosity, and ATP release," *Proceedings of the National Academy of Sciences*, 2011.

[95] Phillips, R., Kondev, J., and Theriot, J., "Physical Biology of the Cell," *Garland Science, Taylor and Fransis Group*, 2012.

[96] Haselwandter, C. A. and Phillips, R., "Directional interactions and cooperativity between mechanosensitive membrane proteins," *Euro. Phys. Lett.*, 2013.

[97] Wan, J., Ristenpart, W. D., and Stone, H. A., "Dynamics of shear-induced ATP release from red blood cells," *Proceedings of the National Academy of Sciences*, 2008.

[98] Vlahovska, P. M., Podgorski, T., and Misbah, C., "Vesicles and red blood cells in flow: from individual dynamics to rheology," *Comptes. Redu. Phys.*, 2009.

[99] Bagchi, P. and Kalluri, R. M., "Dynamics of nonspherical capsules in shear flow," *Physical Review E*, 2009.

[100] Veerapaneni, S. K., Rahimian, A., Biros, G., and Zorin, D., "A fast algorithm for simulating vesicle flows in three dimensions," *Journal of Computational Physics*, 2011.

[101] Haselwandter, C. A. and Phillips, R., "Connection between oligomeric state and gating characteristics of mechanosensitive ion channels," *PLoS Comput. Biol.*, 2013.

[102] Huang, H. W., "Deformation free energy of bilary membrane and its effect on gramicidin channel lifetime," *Biophys. J.*, 1986.

[103] Pozrikidis, C., "Creeping flow in two-dimensional channels," *J. Fluid Mech.*, 1987.

[104] Yoshimoto, M., Tamura, R., and Natsume, T., "Liposome clusters with shear stress-induced membrane permeability," *Chem. Phys. Lipids.*, 2013.