Mathematical and Computational Studies of the Biomechanics of Biofilms

by

J.A. Stotsky

B.S. Chemical and Biological Engineering, Tufts University, 2013

A thesis submitted to the

Faculty of the Graduate School of the

University of Colorado in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

Department of Applied Mathematics

2018

This thesis entitled:
Mathematical and Computational Studies of the Biomechanics of Biofilms
written by J.A. Stotsky
has been approved for the Department of Applied Mathematics

Prof. David M. Bortz

Vanja Dukic

Keith Julien

Zachary Kilpatrick

Michael Solomon

Date

The final copy of this thesis has been examined by the signatories, and we find that both the content and the form meet acceptable presentation standards of scholarly work in the above mentioned discipline.

Stotsky, J.A. (Ph.D., Applied Mathematics)

Mathematical and Computational Studies of the Biomechanics of Biofilms

Thesis directed by Prof. David M. Bortz

Bacterial biofilms are communities of bacteria growing on a surface to which they have adhered, typically in an aqueous environment. The motivation to understand biofilm behavior arises from a variety of applications including the development of strategies to mitigate corrosion in industrial machinery, the treatment of bacterial infections, and process control in bioreactors. The focus in this thesis is on fluid-structure interaction and biomechanical properties of biofilms. Detailed studies of a mathematical biofilm model that includes the heterogeneous rheology observed in biofilms, a statistical model of biofilm microstructure, and an application of techniques from *a posteriori* numerical analysis to the Method of Regularized Stokeslets are explored. Key findings include the validation of a biofilm model with experimental data, an exploration of the effect that biofilm microstructure has on macroscopic properties, and an elucidation of how error propagates in a numerical method for biofilm simulation.

Dedication

To my grandfather, Bernard A. Stotsky and my grandmother, Miriam Itzkowitz. Their encouragement, integrity, and accomplishments are a source of inspiration.

Acknowledgements

Foremost, I acknowledge my advisor, Professor David Bortz for his advice and wisdom over the last four and a half years of my doctoral studies. His support was vital for the completion of this thesis. I also acknowledge the support and influence the Department of Energy's Computational Science Graduate Fellowship program, of which I have been a fellow since April 2014, had on my doctoral studies. The fellowship guided much of my graduate coursework, introduced me to a community of like-minded researchers, and spurred my interest in computational science. Another important influence on my thesis are my cohort in the MathBio group at the University of Colorado: Sabina Altus, Lewis Baker, Harry Dudley, Eric Kightley, Taisa Kushner, Inom Mirzaev, John Nardini, and Jaqueline Wentz. I also acknowledge the advice which co-authors Vanja Dukic, Jason Hammond, Leo Pavlovsky, Michael Solomon, Elizabeth Stewart, and John Younger provided during the writing and revision of two peer-reviewed articles of which I am an author. Furthermore, I acknowledge Phil Collela and Daniel Martin who advised me during the summer of 2016 when I spent a summer at Lawrence Berkeley National Laboratory in the Applied Numerical Algorithms Group working on algorithms to simulate the motion of drag-coupled particles in a viscous fluid. I also thank the faculty in Research Computing at the University of Colorado for providing facilities and access to computing resources that were needed in my research. Last but not least, I thank my mother, father, brother, grandparents, and extended family for the numerous circumstances in which they have continually supported and advised me over the years.

Contents

Chapter

Tables

Table

Figures

Figure

Chapter 1

Introduction

Bacterial biofilms are communities of bacteria growing on a surface, typically in an aqueous environment. Interest in understanding biofilm behavior arises from a variety of applications: strategies to mitigate corrosion in industrial machinery, the treatment of bacterial infections, and process control in bioreactors are just a few of many examples. Several other examples are overviewed in [40]. The diversity of situations where bioflims are relevant has led to many mathematical models, each designed to capture various aspects of biofilm behavior. Significant efforts have been devoted to modeling growth, infection and disinfection dynamics, nutrient transport, and fluid-structure interactions with flowing media. The focus in this thesis is on fluid-structure interaction and biomechanical properties of biofilms.

Biofilms are morphologically complicated and exhibit highly heterogeneous material properties. After the initial stages of adhesion to a substrate, the bacteria within a biofilm produce a viscous secretion of extracellular material, known as the extracellular matrix, which encases the bacteria. The extracellular matrix protects it from external harms, provides structural support, and helps to anchor the biofilm to substrate. Because they are embedded in an extracellular matrix, biofilms do not have sharply defined boundaries but are demarcated by diffuse interfaces where rheological properties vary drastically but continuously over short length scales. For instance, the viscosity near the surface of a bacteria may be up to 500 times that of the surrounding media. Modeling and simulation of such interfaces present significant challenges.

In addition, to realistically modeling biofilm material properties, the length and time scales of interest are an important consideration because of the multiscale nature of biofilms. Temporally, biofilms exhibit viscoelastic effects over a range of time scales and exhibit viscoplastic properties over long time scales. At the smallest relevant spatial scale, the discrete nature of the individual bacteria is an important consideration. Medium scale structures include channels permeating through the biofilm and streamers or strand-like pieces of biofilm that have detached from the larger biomass due to viscous stresses. Larger scale structures include ripples and mushroom shaped growths that are often observed in nature. One goal of the mathematical models that will be discussed is to find relations between structures on these different length scales. In the Chapter 2, I study how rheological parameters related to the bacteria effect the overall biomechanics, and in Chapter 3, the influence which the spatial configuration of bacteria have on the biomechanics is studied through the introduction of a novel statistical model.

Interest in modeling mechanical properties of biofilms has existed several decades. An overview of mathematical biofilm modeling covering biomechanics along with biochemistry and growth can be found in [70]. The first models treated biofilm-fluid interactions in a compartmentalized manner with disjoint biofilm and fluid domains [100, 82]. Over time various improvements to these models were made, and their now exist several more sophisticated types of models that can handle irregular biofilm geometries, topological changes in biofilm shape, biochemical reactions, and diffuse interface properties. For biomechanics, the state-of-the-art models are typically based upon phase-field formulations or the immersed boundary method. In a phase field model, the biofilm is treated as a continuum material, and a "phase variable" that tracks the mass fraction of biofilm is introduced[104, 105, 18]. In contrast, immersed boundary method models treat each bacteria as a discrete entity in the model. The positions of the bacteria are coupled to equations of fluid dynamics and their motion over time is tracked. Conversely, the equations of fluid dynamics are coupled to the bacteria locations through variable rheological parameters, and forces associated with each bacteria. Examples of immersed boundary method models are found in [2, 52, 89].

1.0.1 The Heterogeneous Rheology Immersed Boundary Method

In Chapter 2[1] the focus is on the validation of a biofilm model by comparison to experimental data. The model, called the **heterogeneous rheology Immersed Boundary Method** (hrIBM) is designed to accurately capture the diffuse interfaces typical of fluid-structure interactions in biofilms. It provides a high resolution description of the biofilm undergoing deformation, resolving the biofilm to the level of individual bacteria. The main results in this chapter demonstrate that with a suitable choice of parameters, the hrIBM model provides a quantitatively accurate description of the biomechanics of a biofilm. This result is significant because previous quantitative validation attempts were overly simplistic, whereas the results presented in Chapter 2 much more definitively show that immersed boundary method biofilm models are physically realistic. This work was completed with the help of the following collaborators: Jason Hammond (High Power Electromagnetic Division, AFRL, Kirtland AFB), Leo Pavlovsky (Department of Chemical Engineering, University of Michigan), Elizabeth Stewart (Department of Chemical Engineering, University of Michigan), John Younger (Akadeum Life Sciences), Michael Solomon (Department of Chemical Engineering, University of Michigan), and David Bortz (Department of Applied Mathematics, University of Colorado).

1.0.2 Statistical Modeling of the Configuration of Bacteria in a Biofilm

In Chapter 3[2] a statistical model of the positions of bacteria within a biofilm is described. There are two main motivations for this work. First, the hrIBM which will be discussed in Chapter 2, relies on data to provide the initial positions of bacteria in a biofilm. Obtaining such data is costly, and may be impossible in practical cases. The statistical model developed in Chapter 3 allows such data to be realistically generated *in silico*. The second motivation is to study how spatial aspects of biofilm microstructure impact biofilm properties. A number of studies have measured

[1] This chapter appeared in the Journal of Computational Physics in 2016 [89]

[2] The work described in this chapter has been accepted into the European Journal of Applied Mathematics and can be found on ArXiv [88]

statistical properties of bacteria positions in biofilms, but the work discussed in Chapter 3 is the first to combine many of these results into a single coherent model.

The model is a type of **pairwise interaction model** (PIM) derived from results on the statistical physics of fluids and mathematical literature on spatial point processes. Realizations of the model are collections of points in space that correspond to the locations of bacteria for *in silico* biofilm simulation. The PIM is parametrized by the empirical estimation of certain energy functions. Using a Metropolis-Hastings algorithm, realizations of the model are generated. Through simulations, it is demonstrated that statistical properties of the model closely match experimental data. A key result from this chapter is that nonuniformity in the positions of bacteria in a biofilm can increase the mechanical strength of the biofilm. The work done in this chapter was a collaboration with Vanja Dukic (Department of Applied Mathematics, University of Colorado) and David Bortz (Department of Applied Mathematics, University of Colorado)

1.0.3 *A posteriori* Analysis of the Method of Regularized Stokeslets

In Chapter 4, the **Method of Regularized Stokeslets** (MRS), a numerical method which can be used to simulate biofilms, is analyzed with techniques from *a posteriori* error analysis. In contrast to *a priori* error analysis, *a posteriori* error analysis leads to error bounds that depend only upon the numerical solution to a given problem, not the exact solution. Thus, they are very effective for global and local error control techniques. Suitable adjoint equations are derived and an error representation formula is obtained. Numerical simulations are used to illustrate the propagation of various types of numerical error in this method. The research described in this chapter was done under the guidance of David Bortz (Department of Applied Mathematics, University of Colorado).

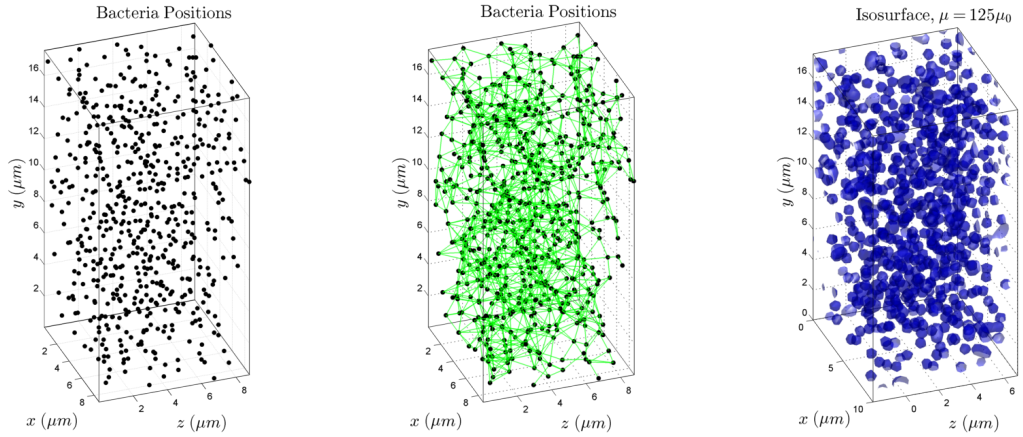
Chapter 2

Validation of an Immersed Boundary Model of Biofilm Rheology

2.1 Introduction

The goal of this chapter is to show that the model and corresponding simulation method originally developed in [52], can accurately capture the biomechanical response of bacterial biofilms. The underlying mathematical technique is an adaptation of the Immersed Boundary Method (IBM) that takes into account the finite volume of bacteria, and the widely variable material parameters anchored to the positions of the bacteria in a biofilm. We call this method the **heterogeneous rheology Immersed Boundary Method** (hrIBM). With the hrIBM model, biofilms are resolved to a scale at which the contributions of individual bacteria are distinguishable. This allows for detailed modeling of the highly viscous and spatially heterogeneous extracellular matrix (ECM) and, the polysaccharide chains that link bacteria together in biofilms.

A key feature of our simulations is the use of experimental data from live **S. epidermidis** biofilms to initialize the spatial location of each bacterium. This removes ambiguity about how to represent the biofilm computationally. Motivated by results from [43], we model the spatial heterogeneity of the ECM by treating the ECM as a "polymer brush" with radially dependent viscosity and density fields centered at each bacteria location. In Figure 2.1, images of the positions, connectivity, and a viscosity isosurface of bacteria in a sample biofilm are shown. When using experimental data to initialize the positions of bacteria, the bulk dynamic moduli and compliance modulus computed through simulation are consistent with experimental results found in [77]. We also show that upon running simulations with several different data sets that possess similar spatial



(a) (b) (c)

Figure 2.1: a) Shows the 3D locations of bacteria from experimental biofilm data. b) Each line represents a viscoelastic connection between two bacteria. Bacteria connected if they are within $1.62\mu m$ of each other. c) A viscosity isosurface of the same biofilm. The maximum viscosity is $250\mu_0$ where μ_0 is the viscosity of water. The isosurface is the surface defined by $\mu(\mathbf{x}) = 125\mu_0$.

statistics, the physical properties of the biofilm do not change significantly. Finally, we provide quantitative results on the periodic rotation of suspended aggregates of bacteria in shear flow.

In Section 2.2, we provide a brief review of the classical Immersed Boundary Method (IBM), a well known computational technique for the simulation of coupled fluid-structure interactions. Additionally, we discuss some other IBM-based biofilm models, and explain the adaptations of the IBM that lead to the hrIBM. In Section 2.3, we give a description of the numerical properties of the hrIBM model and provide results from numerical tests showing that the model is convergent. In Section 2.4, methods for computing relevant material properties from the hrIBM model are discussed, and the dynamic moduli and compliance moduli computed from the model are compared to experimental data from biofilms grown in a bioreactor. In Sections 2.5 and 2.6, we conclude this work and discuss future research directions.

To our knowledge, this work is the first to use a model that accounts for both the heterogeneous rheological properties (e.g., variable viscosity) and, the inter-bacterial connectivity to compute material properties of a biofilm. This work along with Part I, is also the first to incorporate variable viscosity into an IBM-based biofilm model, and the second to use variable viscosity with an IBM model in general. Matlab code used to produce the results obtained in this paper is available at https://github.com/MathBioCU/BiofilmSim.

2.2 The Biofilm Model

2.2.1 Previous IBM Based Biofilm Models

In recent years, a number of different approaches to IBM-based biological material models have been developed. In [68], an immersed viscoelastic structure is coupled to fluid flow in an immersed boundary type formulation. However, the fluid equations are solved separately from the equations governing the motion of the immersed viscoelastic solid and then coupled together at a sharp physical interface. This strategy works well for systems that have sharp fluid-structure interfaces. We deviate from this strategy since biofilms do not have sharp interfaces, and are permeated by

fluid. Thus we invoke a spatially variable viscosity and couple the viscoelastic structure and fluid throughout the entire computational domain.

Another approach to capturing the viscoelastic nature of biofilms with an IBM is through adapting the choice of viscoelastic model governing the forces generated by links between bacteria. This type of strategy was used first by [12] to model general viscoelastic connections in the actin cytoskeleton of ameboid cells. This strategy has also been used in [108] to model sperm motility, and in [101] to model immersed three dimensional viscoelastic networks. We pursue this strategy by testing our model with several viscoelastic force laws, and also build upon this class of models with the inclusion of rheologically variable fluid media.

Other IBM-based biofilm models can be found in [4] and in [30]. In these models, an IBM is used directly to couple the forces between connected bacteria with fluid motion. Additionally, some validation results are performed to show that properties such as the recovery and relaxation times of simulated biofilms fall within the range of realistic values. Although not an IBM-based model, we also note that similar results are computed and compared to experiments in [95] where a phase-field model is used. Other phase field models that the reader may find of interest can be found in [18, 104, 105, 106].

In a recent work [93], the IBM is used along with the von Mises stress criterion to provide detailed simulations of the detachment of a biofilm in shear flow. Although we do not currently employ a means of modeling detachment, it is possible to implement a model that allows the breakage of connections between bacteria in the hrIBM framework. If enough connections break, detachment of one or more bacteria from the biofilm can occur. A simple, strain based criterion was used in Part I [52] to model the rupture of links connecting bacteria as deformation occured in a biofilm.

As our biofilm model is based on the IBM, interested readers can find detailed explanations of the IBM in [78, 79, 107, 108], and additional IBM-based biofilm models can be found in [3, 30, 52].

2.2.2 The Biofilm Model

The model we use is comprised of two sets of equations; those that model fluid flow through the biofilm, and those that model motions and forces experienced by each bacteria cell in the biofilm. These equations are listed in (2.1)-(2.8).

$$\rho(\mathbf{x},\, t)\,(\mathbf{u}_t + (\mathbf{u}\cdot\nabla)\,\mathbf{u}) = -\nabla P + \nabla\cdot\left(\mu(\mathbf{x},\, t)\left(\nabla\mathbf{u} + (\nabla\mathbf{u})^T\right)\right) + \mathbf{f}(\mathbf{x},\, t) \tag{2.1}$$

$$\nabla\cdot\mathbf{u} = 0 \tag{2.2}$$

$$\mathbf{U}(s,\, t) = \int_\Omega \mathbf{u}(\mathbf{x},\, t)\,\hat{\delta}(\mathbf{X}(s,\, t) - \mathbf{x},\, \omega)\,d\mathbf{x}\; s = 1, 2, ...N \tag{2.3}$$

$$\frac{\partial \mathbf{X}(s,\, t)}{\partial t} = \mathbf{U}(s,\, t) \tag{2.4}$$

$$\mathbf{F}(s,\, t) = \mathcal{F}(\mathbf{X}(s,\, t),\, \mathcal{P}) \tag{2.5}$$

$$\mathbf{f}(\mathbf{x},\, t) = \frac{1}{d_0^3}\sum_{s=1}^{N}\mathbf{F}(s,\, t)\,\hat{\delta}(\mathbf{X}(s,\, t) - \mathbf{x},\, \omega) \tag{2.6}$$

$$\rho(\mathbf{x},\, t) = \rho_0 + (2\omega)^3\sum_{s=1}^{N}(\rho_b - \rho_0)\,\hat{\delta}(\mathbf{X}(s,\, t) - \mathbf{x},\, \omega) \tag{2.7}$$

$$\mu(\mathbf{x},\, t) = \mu_0 + (2\omega)^3\max_{1\le s\le N}(\mu_b - \mu_0)\,\hat{\delta}(\mathbf{X}(s,\, t) - \mathbf{x},\, \omega) \tag{2.8}$$

These equations are similar to those used in Part I [52] except for some small changes that were convenient for implementing the simulations described in Section 2.4.

In most IBM literature, the Equations 2.6 and 2.7 are written as integrals over Lagrangian coordinates. In this case, there is a fixed, finite number of bacteria in the domain regardless of refinement level, so we compute these as summations instead. In these equations, we use a scaled approximation of the Dirac δ-function, denoted $\hat{\delta}$, which has a region of support that depends on a radial parameter ω, defined such that 2ω is the hydrodynamic radius of a typical bacteria. Since only finitely many bacteria are considered, we also define a Lagrangian label s as a number associated with each bacteria.

The scale factor of $(2\omega)^3$ in Equations 2.7 and 2.8 is chosen to enforce that at the locations, $\mathbf{X}(s,t)$, the density and viscosity are equal to ρ_b and μ_b respectively. Finally, we define $\mathbf{F}$ as a force instead of a force density to allow for standard units to be associated with the constitutive

Table 2.1: Quantities assigned a value at each grid point and timestep in the discretized computational domain. Together they provide an Eulerian description of the fluid motion in the biofilm.

Eulerian Variables	
Symbol	Definition
$\mathbf{x}$	Eulerian position
t	time
$\mathbf{u}$	Eulerian velocity
P	pressure
$\mathbf{f}(\mathbf{x}, t)$	Eulerian force density
$\mu(\mathbf{x}, t)$	Eulerian viscosity
$\rho(\mathbf{x}, t)$	Eulerian density

parameters (i.e. spring constants) that will be used to model the viscoelastic links between neighboring bacteria. Complete listings of the quantities appearing in the model equations can be found in Tables 2.1, 2.2, and 2.3.

Since individual bacteria are not assumed to have infinitessimal volume at the scale of our simulations, the Lagrangian quantities; $\mathbf{X}$, $\mathbf{U}$, μ_b, ρ_b, and $\mathbf{F}$ correspond to measurements taken at the center of mass of each bacterium. As described in Section 2.2.3, the second argument, ω, of the smoothed Dirac δ function, $\hat{\delta}(\cdot, \omega)$ determines a region of support for the smoothed δ function. The choice of $\hat{\delta}(\cdot, \omega)$ govern how the mass density, viscosity, and force density vary around each bacterium.

The viscosity is computed using a different form than the density and force and is shown in Equation (2.8). This formula for viscosity is not intuitive, and an explanation for this choice of viscosity can be found in Part I. However, finding the most accurate form for $\mu(\mathbf{x})$ is still an area of active research. In particular, we experimented with approximations to $\mu(\mathbf{x}) = \mu_0 + \int_\Omega (\mu_b - \mu_0)\, \hat{\delta}(\mathbf{X}(t) - \mathbf{x}, \omega) d\mathbf{X}$. We made our ultimate choice for $\mu(\mathbf{x})$ so that the viscosity does not overshoot μ_b, the force and viscosity drop off at the same rate, and to maintain consistency with Part I.

The last part of the model is a force law, $\mathcal{F}$ needed to specify $\mathbf{F}$ based on the configuration of bacteria, $\mathbf{X}$. We use Hooke's law, Maxwell's law, and the Zener model in our simulations, and initialize the connectivity of the biofilm by using a cutoff radius, r_c. At the start of the simulation,

Table 2.2: Quantities assigned to each bacteria in the domain at each time step. They can be thought of as Lagrangian variables associated with each bacteria center of mass.

Quantities Associated with each Bacterium (Lagrangian variables)	
Symbol	Definition
s	label for each bacteria
$\mathbf{X}(s,\, t)$	center of mass of bacteria s at time t
$\mathbf{U}(s,\, t)$	velocity of bacteria s at time t
$\mathbf{F}(s,\, t)$	force on bacteria s at time t
μ_b	viscosity at bacteria center of mass
ρ_b	density at bacteria center of mass

Table 2.3: These quantities are parameters to the model. They are either based on physical data (i.e. density and viscosity of water) or are tuned through simulations to obtain realistic values.

Model Parameters	
Symbol	Definition
μ_0	viscosity of water
ρ_0	density of water
$\mathcal{F}(\mathbf{X},\, \mathcal{P})$	constitutive relation between bacteria configuration and force
$\mathcal{P}$	parameters in the viscoelastic law used (i.e. spring coefficients etc.)
N	number of bacteria in computational domain
d_0^3	volume of average bacterium
ω	hydrodynamic radius of a bacterium
r_c	cutoff radius for establishing which bacteria are linked
$\hat{\delta}(\cdot,\, \omega)$	discrete Dirac delta function with support related to parameter ω

any two bacteria that are separated by a distance less than the cutoff radius will be linked by a spring whose initial, resting length is their initial separation.

By using the IBM as a basis for our biofilm model, we avoid treating the biofilm as a two phase fluid with a distinct bulk fluid region and a distinct biofilm region. Instead, the use of variable rheological properties over the entire domain couples the biofilm and bulk fluid motions as a single viscoelastic material. Because the fluid permeates through the entire domain, even when the simplest force law, Hooke's Law, is used to determine forces between bacteria, the model still behaves viscoelastically, not elastically.

2.2.3 The Heterogeneous Rheology Immersed Boundary Method

In our model, we extend the IBM to account for the fixed, finite size of bacteria and allow for variable physical properties that are anchored to a moving Lagrangian mesh (i.e. the bacteria positions). We denote this approach the **heterogeneous rheology IBM** (hrIBM).

The original IBM was first developed as a means of solving fluid-structure interaction problems in cardiology and is applicable to problems with moving, irregularly shaped boundaries [78, 79]. With the IBM, the fluid velocity fields and pressure are usually solved for on a fixed, Eulerian grid and the movement of the boundaries due to fluid motion is tracked by a Lagrangian mesh that moves according to the underlying fluid motion. As the material boundaries are deformed, a constitutive model is used to determine the force density exerted by the immersed boundary on the fluid. The Lagrangian force density field is then transferred to an Eulerian force density field through a convolution integral with a δ function kernel. The Eulerian force density influences the fluid motion through its inclusion in the Navier-Stokes equations. Thus the fluid and boundary motions are coupled since the boundary motion is interpolated from the fluid velocity field, and the fluid velocity field is influenced by forces on the boundary. The couplings and solution process for the IBM are depicted in Figure 2.2.

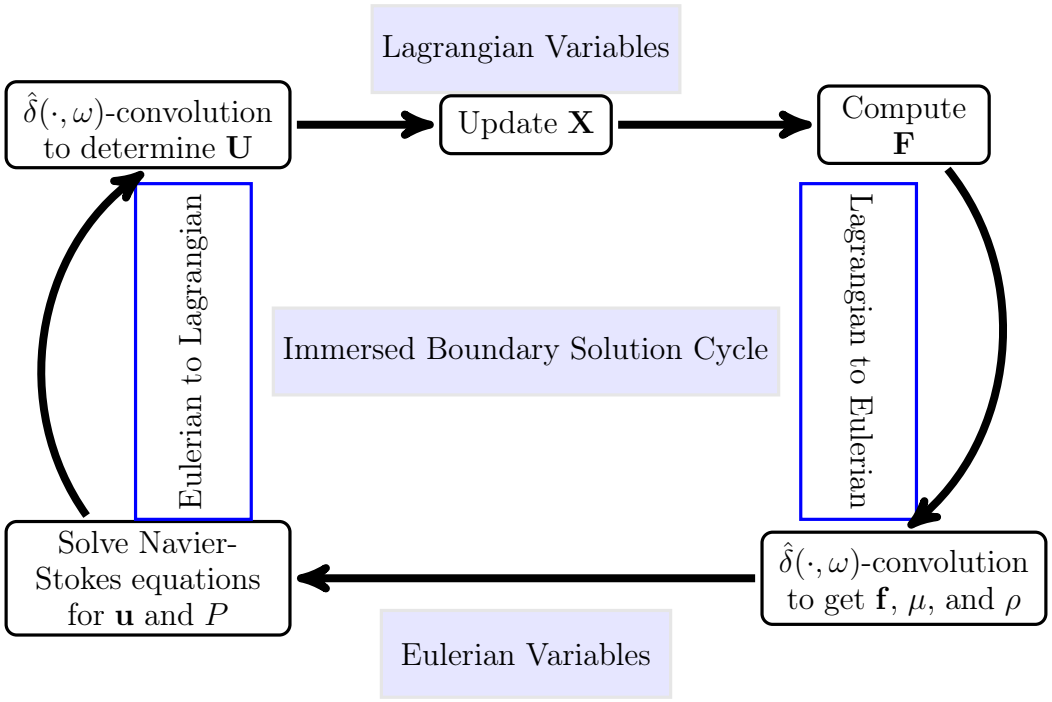


Figure 2.2: The coupling between the Eulerian and Lagrangian variables in the hrIBM is shown here. The Eulerian and Lagrangian variables are coupled by the computation of **U** from **u**, and the computation of **f**, μ, and ρ from **F** and **X**. The IBM is a widely applicable method in part because it allows for a great variety of fluid solvers and solid structural models to be coupled through δ function transfer identities.

2.2.3.1 A Dirac δ Approximation Based on Finite Volume Bacteria

An important step in the numerical implementation of the IBM is the construction of a discrete approximation of the Dirac δ function (denoted$\hat{\delta}(\cdot, h)$). This approximation is chosen such that as $h \to 0$, $\hat{\delta}(\mathbf{r}, h) \to \delta(\mathbf{r})$. This is accurate for fluid structure interactions involving fluid-solid boundaries that have infinitessimal thickness, and thus zero volume. In the hrIBM model, each Lagrangian point corresponds to the center of mass of a bacterium which has a finite volume. Therefore, we use a smoothed version of the standard discrete δ function whose region of support is controlled by a radial parameter, ω, that is independent of the grid spacing, h.

In our model, we use a smoothed discrete Dirac δ approximation of the form,

$$\hat{\delta}(\mathbf{x}, \omega) = \frac{1}{\omega^3}\phi\left(\frac{x}{\omega}\right)\phi\left(\frac{y}{\omega}\right)\phi\left(\frac{z}{\omega}\right) \tag{2.9}$$

with $\phi(r)$ as defined in [79] by

$$\phi(r) = \begin{cases} \frac{1}{8}\left(5 - 2|r| - \sqrt{-7 + 12|r| - 4|r|^2}\right) & 1 \le |r| \le 2 \\ \frac{1}{8}\left(3 - 2|r| + \sqrt{1 + 4|r| - 4|r|^2}\right) & 0 \le |r| \le 1 \\ 0 & |r| > 2 \end{cases}$$

Further discussion of this choice for $\hat{\delta}(\mathbf{x}, \omega)$ can be found in [52]. If $\omega = h$, the standard discrete δ function seen in IBM literature is obtained. For this work, we assume that the bacteria are spherical and thus ω is understood as a hydrodynamic radius. We also note that extensions to this formalism will allow for the treatment of nonspherical bacteria or unevenly sized bacteria. Thus, ω may be though of more generally as a shape parameter.

2.2.3.2 Variable Viscosity and Density in the hrIBM

Highly heterogeneous viscosity and moderately heterogeneous density are common characteristics of biofilms. Although IB methods with variable density have existed for some time (see [107]), the incorporation of spatially variable viscosity in the IBM is an area that has yet to be well developed. We note that in [36] and [37], an IBM capable of solving problems with variable

viscosity and density is used to model the motion of red blood cells flowing in capillaries. When modeling red blood cells, the viscosity exhibits a "jump" discontinuity between the blood plasma and the intracellular, hemoglobin-containing fluid of a red blood cell. Thus, their model is designed to capture the dynamics of two interacting fluids with different rheological properties separated by a deformable membrane. In our case, there do not exist well defined boundaries and thus, $\delta(\cdot, \omega)$ is adjusted to reflect this.

In biofilms, the spatial variance of material properties is localized around the position of each bacterium, while in fluid far away from any bacteria, the physical properties are those of the bulk fluid. This localization of the variation in material properties allows the spatial variation in density and viscosity to be found by using a smoothed δ-function integration similar to that used to compute the Eulerian force field. We define an viscosity, μ_b and density, ρ_b and assume that at the center of mass of each bacteria, the viscosity and density are $\rho(\mathbf{X}_i, t) = \rho_b$, and $\mu(\mathbf{X}_i, t) = \mu_b$. Defining μ_0 to be the viscosity of the bulk fluid (in this case water) the viscosity at any Eulerian grid point is calculated as:

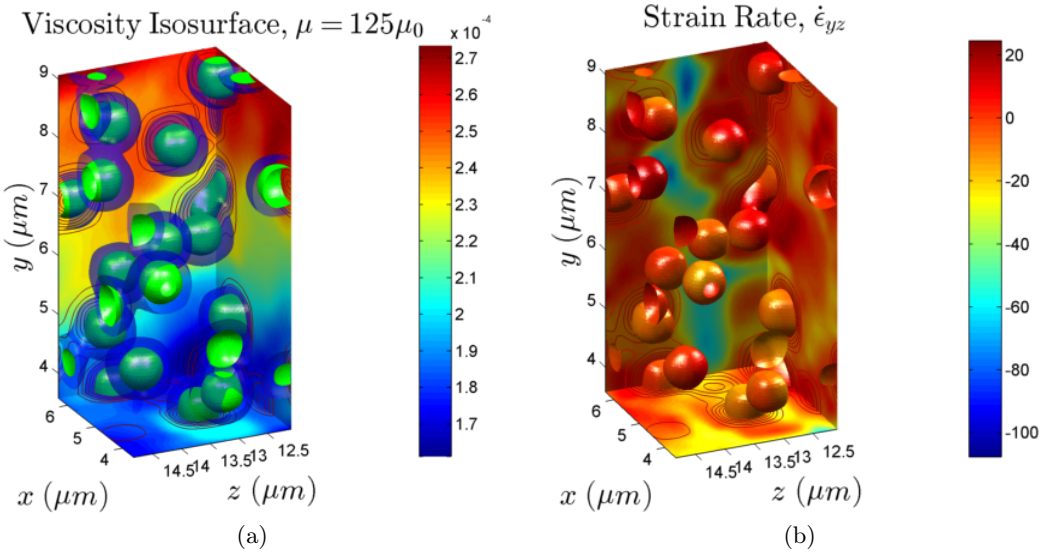
$$\mu(\mathbf{x}) = \mu_0 + (2\omega)^3 \max_{1 \le s \le N} \left[(\mu_b - \mu_0)\, \hat{\delta}(\mathbf{x} - \mathbf{X}(s), \omega) \right]. \tag{2.10}$$

The density can be computed from

$$\rho(\mathbf{x}, t) = \rho_0 + (2\omega)^3 \sum_{s=1}^{N} (\rho_b - \rho_0)\, \hat{\delta}(\mathbf{X}(s, t) - \mathbf{x}, \omega).$$

In the simulations, the number of bacteria, N, is fixed and independent of the mesh spacing h, thus a summation is used instead of an integral for the density computation.

In this model, we indirectly take into account the fluid volume displacement caused by the presence of the bacteria. We treat the localized high viscosity around each bacteria as an effective viscosity that accounts for both the displaced fluid volume and the increased viscosity near the bacteria surface [43]. Extensions based on changing our choice for $\hat{\delta}(\mathbf{x}, \omega)$ and $\mu(\mathbf{x})$ could possibly allow for a more precise computation of the volume displacement caused by the biofilm into the model. In Figure 2.3, observe that lower strain rates are found near the bacteria cells where high viscosity is present. This is indicative of the bacteria appearing as relatively solid objects compared



(a) (b)

Figure 2.3: a) A viscosity isosurface is shown for a small section of a biofilm used in simulation. The inner isosurface is $\mu = 125\mu_0$ and the outer transparent isosurface is at $\mu = 50\mu_0$. Slices of the $||\mathbf{u}||$ velocity field are shown as well. b) The $\dot{\epsilon}_{yz}$ component of the strain rate is plotted on the $\mu = 125\mu_0$ viscosity isosurface. Additionally, the strain rate and contours of viscosity are shown in slice planes. For a single phase fluid, Newton's viscosity law is $\sigma = \mu\dot{\epsilon}$. Although biofilms are not Newtonian fluids, we still see that in areas of low viscosity, higher strain rates are found and in areas of higher viscosity lower strain rates occur.

to the bulk fluid.

We make the assumption that the bacteria have a uniform diameter of $1.0\mu m$ based on [44] and [92] where bacteria diameters are measured and determined to have a tightly centered distributions around $0.64\,\mu$m $0.70\,\mu$m respectively. We choose a larger radius since we want to account for the influence of the polymeric ECM which extends into the fluid outside the cell. We also note another source that states that **S. epidermidis** typically have a diameter in the range of $0.5 - 1.0\mu m$ [41]. We also assume that the bacteria are uniformly spherical in shape. Although we have not been able to find data to quantify this assumption, examination of high-resolution images of **S. epidermidis** bacteria found in [87] and [76] appears to confirm the validity of this assumption.

2.3 Numerical Methods

The numerical methods we use are based on those originally discussed in [52]. We summarize them here for convenience and also provide convergence results. To approximate solutions to equations (2.1)-(2.8), we use an approximate projection method similar to that used in [107]. The solution scheme uses an implicit Euler solver to update an intermediate velocity profile at each time step and is expected to be $\mathcal{O}(\Delta t)$ convergent in the velocity field. To discretize the domain, we use a uniform finite difference discretization with equal spacings in the x, y, and z directions. The spatial derivatives are approximated with 2nd order, centered finite differences.

2.3.1 Numerical Algorithm

At each time step, the following quantities must be updated: $\mathbf{u}$, $\mathbf{U}$, P, $\mathbf{F}$, $\mathbf{f}$, $\mathbf{X}$, μ, and ρ. We nondimensionalize the problem with the following choices:

$$\hat{\mathbf{u}} = \frac{\mathbf{u}}{u_0} \quad \hat{P} = \frac{P}{\rho_0 u_0^2} \quad \hat{\mathbf{f}} = \frac{L\mathbf{f}}{\rho_0 u_0^2} \quad \hat{\mu} = \frac{\mu}{\mu_0} \quad \hat{\rho} = \frac{\rho}{\rho_0} \quad \hat{\mathbf{x}} = \frac{\mathbf{x}}{L} \quad \hat{t} = \frac{t}{t_0}$$

and also introduce the following nondimensional parameters:

$$Re = \frac{\rho_0 L u_0}{\mu_0} \quad St = \frac{L}{t_0 u_0}.$$

As is standard terminology, Re is the Reynold's number and St is the Strouhal number. Additionally, we define d_0^3 to be the average Lagrangian volume element as described in Part I [52]. For convenience, we will now assume that all quantities are nondimensional unless otherwise stated. The values of the constants we use are listed in Table 2.4 and the motivation for these values is discussed in Part I.

As is standard practice in IBM algorithms, we uncouple the Eulerian variable updates and Lagrangian variable updates for computational reasons. At each time step, we use a projection-based solver to solve the Navier-Stokes equation for $\mathbf{u}$ and P. We define $\mathbf{G}_h$ a discrete gradient operator, and $\mathbf{D}_h$ a discrete divergence operator, and use the following approximate projection method to obtain $\mathbf{u}$ and P:[1]

(1) Solve for $\mathbf{u}^*$

$$\rho^{(n)}\left(St\,\frac{\mathbf{u}^*-\mathbf{u}^{(n)}}{\Delta t}+\frac{1}{2}\left(\mathbf{u}^{(n)}\cdot\mathbf{D}_h(\mathbf{u}^{(n)})\ +\mathbf{D}_h\left(\mathbf{u}^{(n)}\mathbf{u}^{(n)}\right)\right)\right)$$
$$=\frac{1}{Re}\,\mathbf{D}_h\left(\mu^{(n)}\left(\mathbf{G}_h(\mathbf{u}^*)+(\mathbf{G}_h(\mathbf{u}^*))^T\right)\right)+\mathbf{f}^{(n)}$$

(2) Solve for $P^{(n)}$

$$\mathbf{D}_h\left(\frac{1}{\rho^{(n)}}\mathbf{G}_hP^{(n)}\right)=St\frac{\mathbf{D}_h(\mathbf{u}^*)}{\Delta t}$$

(3) Compute $\mathbf{u}^{(n+1)}$

$$\mathbf{u}^{(n+1)}=\mathbf{u}^*-\left(\frac{1}{St}\right)\frac{\Delta t}{\rho^{(n)}}\mathbf{G}_h(P^{(n)})$$

In steps 1 and 2, full multigrid solvers and multigrid preconditioned conjugate gradient solvers are used to find $\mathbf{u}^*$ and $P^{(n)}$[2] . After obtaining the updated velocity and pressure, the Lagrangian velocity and position updates follow,

$$\mathbf{U}^{(n+1)}=\sum_{h\in\mathcal{G}_h}\mathbf{u}^{(n+1)}\hat{\delta}(\mathbf{x}_h-\mathbf{X}^{(n)},\,h)\,h^3$$

[1] Numerically, products of the form, $\mathbf{D}_h a^{(n)}\mathbf{G}_h$ are discretized using standard finite difference stencils rather than as a product of the gradient and divergence operators.

[2] Strictly speaking, the pressure should be interpreted as being defined at the half-integer time steps as discussed in [13]. We use the notation $P^{(n)}$ for simplicity

Table 2.4: Values of Physical Parameters and Nondimensional constants used in simulations. The values of F_{s1}, F_{s2}, and F_{d1} vary depending on whether Hooke's Law, the Maxwell Model, or the Zener Model are used, so the values shown here are for the Zener model which appeared to perform most favorably compared to experimental results. The coefficients for the other two models are discussed in Section 2.4.2.

Quantity	Value	Explanation
μ_0	$1 \cdot 10^{-3}$ Pa·s	viscosity of water at $20°C$
ρ_0	$998\,\mathrm{kg}/m^3$	density of water at $20°C$
L	$10\,\mu$m	characteristic length scale
Re	0.0014 – 0.0175	Reynold's number. Varies based on the simulation
St	0.0075 – 0.0720	Strouhal number
t_0	1 s	characteristic time
u_0	$1.40 \cdot 10^{-4}$ – 0.0017m/s	characteristic velocity, chosen to match experimental strain conditions
d_0	$1.59 \cdot 10^{-6}$m	average volume of each bacterium, found by dividing the total volume by the number of bacteria
ρ_b	$0.12\rho_0$	added density contribution from ECM
μ_b	$450\mu_0$	added viscosity from ECM
F_{s1}	$1.64 \cdot 10^{-11}$ N	Hooke's law force constant
r_c	$1.62 \cdot 10^{-6}$m	maximum distance by which bacteria can be connected $t = 0$
ω	$2.5 \cdot 10^{-7}$m	scaling constant, equal to 1/2 the hydrodynamic radius of a bacteria
F_{d1}	$5.6 \cdot 10^{-12}$ N/s	damping coefficient in Maxwell and Zener models
F_{s2}	$1.64 \cdot 10^{-11}$ N	second force constant used in Zener model

$$\mathbf{X}^{(n)} = \mathbf{X}^{(n-1)} + \frac{\Delta t}{St}\mathbf{U}^{(n)}.$$

Next the Lagrangian force density is computed based on the new positions, $\mathbf{X}^{(n+1)}$ as $\mathbf{F}^{(n+1)} = \mathcal{F}(\mathbf{X}^{(n+1)})$ (see section 2.4.2 for more details). Finally, the Eulerian fields,$\mathbf{f}$ and ρ are computed using discrete δ function interpolation to the Eulerian grid through equations of the form:

$$\mathbf{f}^{(n+1)} = \sum_{s=1}^{N} \left(\mathbf{F}^{(n+1)}(s)/d_0^3\right)\, \hat{\delta}(\mathbf{x}_h - \mathbf{X}^{(n+1)}(s),\, \omega) d_0^3$$

$$\rho^{(n+1)} = \rho_0 + (2\omega)^3(\rho_b - \rho_0) \sum_{s=1}^{N} \hat{\delta}(\mathbf{x}_h - \mathbf{X}^{(n+1)}(s),\, \omega),$$

and μ is computed from

$$\mu^{(n+1)} = \mu_0 + (2\omega)^3 \max_{1\le s\le N} \left[(\mu_b - \mu_0)\, \hat{\delta}(\mathbf{x} - \mathbf{X}^{(n+1)}(s),\, \omega)\right]$$

In the simulations we conduct, the primary direction of fluid flow is in the z direction. The height is governed by the y coordinate and width by the x coordinate. In Figure 2.5, the motion of the bacteria in a simulation with these conditions is depicted.

2.3.2 Numerical Verification and Convergence Properties

In the first numerical verification result, we verify the accuracy of the numerical projection method solver with no biofilm present by comparing the numerical solution with an analytical solution. Since there is no immersed structure, this is a test of the fluid solver alone, and not the IBM method. For this test, the domain, Ω is chosen to be a rectangular solid that is periodic in the x and z directions. From [17], the following boundary conditions for $y = 0$ and y_L,

$$\frac{\partial P}{\partial y} = 0 \qquad \mathbf{u}|_0 = (0,\, 0,\, 0) \qquad \mathbf{u}|_{y_L} = (0,\, 0,\, \sin \nu t) \tag{2.11}$$

provide us with an analytic solution,

$$u_z(y,\, t) = \left|\frac{\sinh k\, y(1+i)}{\sinh k\, y_L(1+i)}\right| \sin\left(\nu\, t + \arg\left(\frac{\sinh k\, y(1+i)}{\sinh k\, y_L(1+i)}\right)\right) \quad k = \left(\frac{\nu\rho}{2\mu}\right)^{1/2}. \tag{2.12}$$

Table 2.5: Average convergence factors of the Fluid Solver. Spatial convergence tests were carried out with grid spacings, h, set to 1/32, 1/64, and 1/128 and a time step of $\Delta t = 1/2500/\nu$. Temporal convergence tests were done with $\nu\Delta t$ set to $\pi/125$, $\pi/250$, and $\pi/500$ and $h = 1/32$. Error is computed at $t = (0.5\pi/\nu)\,s$ for temporal convergence tests and after 1250 time steps for the spatial convergence tests. Convergence factors are computed as $\rho(\Delta t) = \log_2\left(\frac{||u(\Delta t) - u_{exact}||_2}{||u(\Delta t/2) - u_{exact}||_2}\right)$ in time and by $\rho(h) = \log_2\left(\frac{||u(h) - u_{exact}(h)||_2}{||u(h/2) - u_{exact}(h/2)||_2}\right)$ in space and $u_{exact}(h)$ is the exact solution on a grid with spacing h. For the spatial convergence tests, we found it necessary to increase the Reynold's number by a factor of 1000 to ensure that the error in the iterative solver and the temporal discretization was much smaller than the spatial discretization error. Before this change the discretization error was only $\mathcal{O}(10^{-9})$.

Frequency of Boundary Oscillation	Time	Space	Error $\|\|\mathbf{u}^h - \mathbf{u}\|\|_2$
$49.91\,Hz$	1.735	1.6928	$1.8363 \cdot 10^{-5}$
$499.1\,Hz$	1.246	1.9817	$5.6811 \cdot 10^{-5}$

The values of P, u_x and u_y are exactly zero in this case. The values of ρ and μ are set to $998\,\mathrm{kg}/m^3$and $1\,m\mathrm{Pa}\cdot s$ respectively and are homogenous across the domain since no analytic solutions with variable density and viscosity and the boundary conditions given above are known to the authors. Convergence tests were conducted with frequencies $\nu = 4.991$ Hz,49.91 Hz with the biofilm, and at 49.91 Hz and 499.1 Hz without the biofilm. In Table 2.5 the absolute error, temporal convergence factors, and spatial convergence factors are listed. For a full explanation and justification of these tables, see Part I [52].

Additionally, with the same boundary conditions as above, we tested the convergence rates for simulations with a biofilm that possesses variable density and viscosity. Temporal and spatial convergence factors are shown in Table 2.6. As is often seen with immersed boundary methods, discretization error induced by the presence of discrete δ functions, in the transferrence of density, viscosity, and forces between Eulerian and Lagrangian coordinates leads to a reduction to first order convergence [79, 61]. In some cases, second order accuracy can be recovered, however this generally requires predictor-corrector methods that are more computationally expensive than what we use here. In addition, variable density tends to lead to first order in time methods even when the velocity is discretized by a formally second order accurate scheme [36, 37]. More detailed numerical convergence results for this model with different boundary conditions are shown in [52]. In Table 2.6,

Table 2.6: Convergence factors of hrIBM with biofilm. For spatial convergence, h was set to 1/32, 1/64, and 1/128 with a time step of $\nu\Delta t = 1/500$. To measure the temporal convergence factors, $\nu\Delta t$ was set to 1/250, 1/500, and 1/1000. In both cases, the boundary conditions described in Section 2.3 were used. Note, that ν refers to oscillation frequency, not dynamic viscosity. Temporal convergence factors were computed as $\rho(\Delta t) = \log \frac{||u(\Delta t) - u(\Delta t/2)||}{||u(\Delta t/2) - u(\Delta t/4)||}$ and spatial convergence factors were computed as $\rho(\Delta x) = \log \frac{||u(\Delta x) - I^h_{h/2} u(\Delta x/2)||}{||u(\Delta x/2) - I^{h/2}_{h/4} u(\Delta x/4)||}$ where I^{2h}_h is an interpolation operator from a grid of spacing h to a grid of spacing $2h$. For spatial refinement, we also note that the number of Lagrangian nodes (bacteria) remains constant as we are not approximating a surface, but instead treating each Lagrangian node as a single bacteria which should have a constant volume independent of the grid refinement.

Frequency of Boundary Oscillation	Velocity, $\|\|\mathbf{u}\|\|$		Position, $\|\|\mathbf{X}\|\|$	
	Time	Space	Time	Space
$49.91\,Hz$	0.983	1.105	1.022	0.952
$4.991\,Hz$	0.991	0.910	1.007	1.054

temporal convergence factors for the same fluid conditions and domain as the analytical solution are listed. The pressure convergence is not shown here since we use a "pressure-free" projection method in our simulations (see [13] for more details), and because the effect of pressure on the velocity field is several orders of magnitude less than the viscous and elastic effects on the scale of our simulations.

2.4 Experimental Validation Results

The material characterization of bacterial biofilms is a difficult experimental task. It is usually not possible to grow biofilms large enough for use in standard testing devices and, attempts to move a biofilm from the environment it was grown in to a testing apparatus may alter its structure. In [77], a promising experimental method of testing material properties of biofilms was developed. In the experimental setup, a biofilm is grown in a parallel plate rheometer that is 40 mm in diameter and approximately 250 μm in height. As the biofilm grows, it adheres to both the top and bottom plate of the device. The top plate can then be rotated or repositioned vertically and the stress and strain induced in the biofilm can be monitored. These measurements can then be used to infer material properties of the biofilm. Using the hrIBM model, we set up a simulation to reproduce

experiments described in [77].

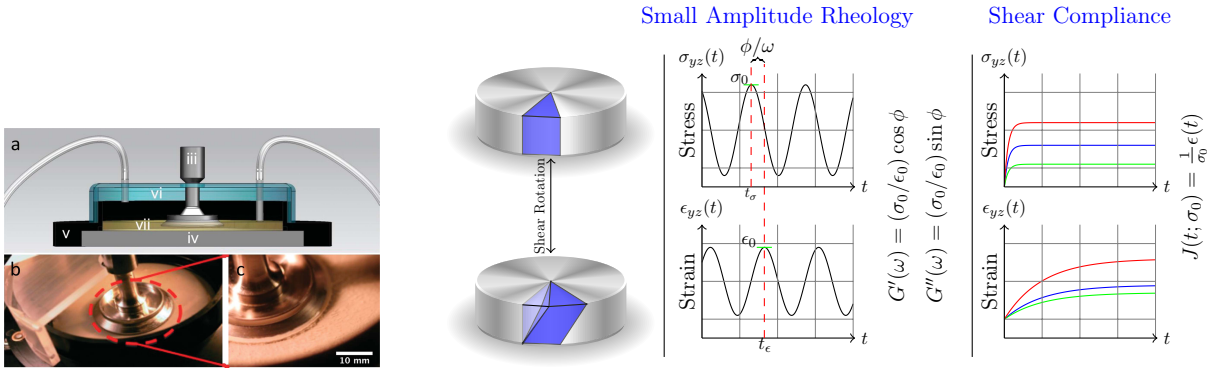
In order to reproduce the biofilm in simulation, 3D position data sets obtained by high resolution microscopy of live biofilms are used to initialize the positions of bacteria in the computational domain. The experimental setup used to obtain these data sets is described in [77, 76] and [87]. Although the biofilm position data sets that we use, which were obtained from the experiments described in [76], are not the ones grown and tested in the bioreactor, they are from biofilms grown under similar physical and nutrient availability conditions. A key result seen from our simulations is that the material properties computed by our model of the different data sets are similar to each other. This indicates that the material properties obtained through simulation must depend on larger scale structural properties of the biofilm and may be treated as bulk material properties of the biofilm. For validation we compared bulk properties computed by our model to those measured in experiments. The methods used to compute these quantities are discussed in the next subsections.

In [77], small amplitude rheometry (SAR) is used to characterize the viscoelastic behavior of **S. epidermidis** biofilms. In SAR experiments, the upper plate of the rheometer is rotated to induce a sinusoidal shear deformation such that the average strain amplitude at the top of the biofilm is a fixed value and the corresponding stress is measured. In [77], the strain amplitude was set to 0.13 at the outer radius of the rheometer since this strain amplitude is found to be in a regime of primarily linear and elastic mechanical behavior. In Figure (2.4),

we show images of the rheometer setup and an illustration of the SAR and compliance experiments.

Creep compliance testing is another characterization technique used in [77] and also in [86] to characterize biofilms. In a creep compliance test, a constant shear stress is applied to the biofilm through the top plate of the rheometer. This induces a time dependent strain which can be measured.

We will now heuristically show that for a small, box-shaped sample of the biofilm that is not near the rotational center or the outer edge of the rheometer, the effects of the rheometer's cylindrical geometry are negligible Thus, the rotational motion that occurs during SAR tests can be well approximated as linear shear. This assumption is pragmatic as it greatly reduces the



(a) (b)

Figure 2.4: a) A depiction of the experimental rheometer setup. Reproduced from [77] with permission from the Royal Society of Chemistry. b) An image highlighting the shear rotation of the rheometer. The typical stress and strain profiles expected from SAR and Compliance experiments are shown on the right.

computational expense our simulations and simplifies the discretization process.

If we temporarily approximate the biofilm as a homogenous, isotropic material, then for low amplitude rotational shear oscillations, the spatial variation in the shear stress and strain will be proportional to the radial distance from the center of the rheometer [19]. If we were to simulate a region of biofilm with dimensions $30 \times 30 \times 30\,\mu m^3$ positioned approximately half way between the center and the outer boundary of the rheometer, the ratio of shear strain and shear stress exerted at the inner and outer boundaries of the simulated section of biofilm is greater than 0.99. Thus the deviation from linear shear is minimal in the computational domain. Of course, biofilms are heterogeneous and may be anisotropic however, the previous rough calculation indicates that rectangular geometry and linear shear should provide an accurate approximation of the motion of the biofilm.

2.4.1 Computation of Rheological Properties

In order to compute the desired dynamic moduli, and compliance modulus results, the stress and strain experienced by the biofilm during simulation must be computed.

The stress $\boldsymbol{\sigma}$, is computed as the sum of stress due to the fluid motion, $\boldsymbol{\sigma}^f$, and stress induced by the straining of inter-bacteria connections within the biofilm $\boldsymbol{\sigma}^b$. Although each component of stress is computed separately during simulations, distinct simulations cannot be used to individually test $\boldsymbol{\sigma}^b$ and $\boldsymbol{\sigma}^f$ since they are coupled.

In order to calculate the strain $\boldsymbol{\epsilon}$, a set of tracer particles is tracked throughout the simulation. Spatial derivatives can then be calculated to obtain approximations of the strain. Additionally, since only small amplitude strains are observed, the linear relation, $\boldsymbol{\epsilon} = \frac{1}{2}(\nabla \mathbf{d} + \nabla^T \mathbf{d})$ is an accurate approximation of the strain for a displacement vector $\mathbf{d}$.

Viscoelastic materials are often characterized through their time dependent stress response to strain or their time dependent strain response to stress. For a general viscoelastic material, given that the stress and strain are sufficiently smooth functions of time, constitutive relations between

the stress and strain may be written in terms of a convolution with viscoelasticity tensors as:

$$\sigma_{ij}(\mathbf{x},\, t) = \int_{-\infty}^{t} G_{ijkl}(\mathbf{x},\, t-\tau)\, \frac{d}{d\tau}\epsilon_{kl}(\mathbf{x},\, \tau)\, d\tau \tag{2.13}$$

$$\epsilon_{ij}(\mathbf{x},\, t) = \int_{-\infty}^{t} J_{ijkl}(\mathbf{x},\, t-\tau)\, \frac{d}{d\tau}\sigma_{kl}(\mathbf{x},\, \tau)\, d\tau \tag{2.14}$$

where σ_{ij} is the stress tensor, ϵ_{ij} is the strain tensor, and G_{ijkl} and J_{ijkl} are fourth order viscoelasticity tensors (see [19] , §1 for a derivation). In the literature, $\boldsymbol{G}$ is often called the relaxation modulus and $\boldsymbol{J}$ is called the compliance modulus. For linear, isotropic materials, the expression for G_{ijkl} simplifies to $G_{ijkl} = \frac{1}{3}\left(G_1(t) - G_2(t)\right)\delta_{ij}\delta_{kl} + \frac{1}{2}G_1\left(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk}\right)$, where δ_{mn} is the Kronecker delta function and Einstein summation notation is used. The two functions, $G_1(t)$ and $G_2(t)$ correspond respectively to shear stress and dilatational stress. Analogous expressions exist for the compliance tensor.

Although the viscoelastic moduli are spatially heterogeneous, we believe that more meaningful results are obtained in the mean field, or spatially averaged, time (and frequency) dependent values for $\boldsymbol{\epsilon}$, $\boldsymbol{\sigma}$, $\boldsymbol{G}$, and $\boldsymbol{J}$. These quantities depend less on the exact configuration of bacteria in a biofilm and behave more like bulk material parameters that can be measured experimentally. Although the interconnected links used to model the connections between adjacent bacteria each individually introduce anisotropy into the model, under the conditions of our simulations, the overall behavior of the biofilm is not highly anisotropic.

As can be seen in Figure 2.5, fluid flow is primarily in the z direction and the y coordinate indicates height in the simulations. Thus the ϵ_{yz} and σ_{yz} components of the stress and strain must be computed at the top plate in order to determine the shear moduli.

2.4.1.1 Computation of Strain

Although a single phase Newtonian fluid will behave viscously (i.e., the stress only depends on the strain rate, not strain itself), in a biofilm the fluid component is influenced by the elastic

components of the biofilm and thus the stress state in a biofilm depends directly on the strain (along with the strain rate). In order to compute the strain, the displacement field must be computed. The displacement, $\mathbf{d}$ of a particle located at $\mathbf{x}_0$ at time, $t = 0$ in a material undergoing deformation can be found by solving the following ODE:

$$\frac{\partial}{\partial t}\mathbf{d}(\mathbf{x}_0,\, t) = \int_\Omega \mathbf{u}(\mathbf{x},\, t)\delta(\mathbf{x} - \mathbf{d}(\mathbf{x}_0,\, t) - \mathbf{x}_0)\, d\mathbf{x}; \quad \mathbf{d}(\mathbf{x}_0,\, 0) = \mathbf{0}. \tag{2.15}$$

In the biofilm simulations, "tracer" particles with positions denoted by $\mathbf{S}(x,\, y,\, z)$, are initialized at heights $y_L - \gamma$, $y_L - \gamma - h$, and $y_L - \gamma - 2h$, near the top of the biofilm at $t = 0$. At each time step, the positions of the tracers are updated using the same δ function interpolation used to update the bacteria positions. With these tracers, the deformation of the biofilm can be tracked throughout the simulation.

In the simulations, the ϵ_{yz} component of strain is needed at the upper boundary of the domain. Therefore, the tracers are initialized near the top of the domain in three vertically aligned layers. This is done to make the numerical approximation of derivatives of the form $\partial d_z/\partial y$ simpler to compute. With the initial arrangement of tracers in vertically aligned layers, the centered finite difference approximation

$$\epsilon_{yz}(\mathbf{S},\, t) \approx \frac{1}{2}\left(\frac{\partial d_z}{\partial y} + \frac{\partial d_y}{\partial z}\right) \approx \frac{1}{2}\left(\frac{\partial d_z}{\partial y}\right) \approx \frac{1}{2}\left(\frac{d_z(\mathbf{S}(y+h) - d_z(\mathbf{S}(y-h))}{2h}\right) \tag{2.16}$$

can be used to approximate the strain. Since the entire upper plate moves at a single velocity at any given time, $\partial d_y/\partial z$ is negligible in this case, whereas in general, this term is required to compute the shear strain. For our simulations, the strain was also approximated in terms of material coordinates.

2.4.1.2 Computation of Stress Induced by Fluid Motion

From Newton's viscosity law the σ^f_{yz} component of stress can be found as

$$\sigma^f_{yz} = \mu(\mathbf{x})\left(\frac{\partial u_z}{\partial y} + \frac{\partial u_y}{\partial z}\right). \tag{2.17}$$

Since the velocity field is already known from solving the Navier-Stokes equations at each time step, the relevant derivatives can be approximated by finite difference approximations. As with the

strain calculation, the second term, $\partial u_y/\partial z$, is zero since the y velocity on the entire top plate of the rheometer is zero. The reported value of σ^f_{yz} at each time step is then found by spatially averaging over the top $2.5\,\mu m$ of the domain. This is done instead of just averaging over the very top of the domain in case there are numerical boundary layers in the fluid flow field near the boundary. Although we did not observe any boundary layers, they are known to sometimes arise in projection method based fluid solvers [13, 74].[3]

2.4.1.3 Computation of Stress Induced by the Biofilm Configuration

In order to compute the force exerted by the biofilm connections on the top plate, we integrate the Eulerian force field induced by bacteria adhered to the top plate. To determine if a bacteria is adhered, we choose a distance, $\gamma = 0.4\,\mu m$ from the top plate, and assume that each bacteria with y coordinate in the interval $[y_L - \gamma,\, y_L]$ is adhered to the top, and that its z-component of velocity is fixed to be that of the upper plate. For these bacteria, any force applied on them by spring-like connections to other bacteria behaves like a force exerted by the biofilm on the upper plate instead of on the bulk fluid. The sum of these forces is used to compute the stress induced by the spring-like connections by means of Cauchy's traction law,

$$\boldsymbol{\sigma}^b \mathbf{n} = \frac{\mathbf{F}^b}{A}. \tag{2.18}$$

The outward unit normal, $\mathbf{n}$, is (0, 1, 0) in this case since the top plate is parallel to the xz plane. The force, $\mathbf{F}^b$ is found by integrating the Eulerian force density field that would be generated by the biofilm nodes adhered to the top plate. Additionally, since we are interested in the applied shear stress, σ^b_{zy}, this can be found as $-\frac{F^b_z}{A}$. Note that γ was chosen arbitrarily, however we observed that with $\gamma = 0.7\,\mu m$ the results were not significantly different.

[3] Boundary layers can also be mitigated through adjustments to the boundary conditions for $\mathbf{u}^*$ in conjunction with a Crank-Nicholson time stepping scheme.

2.4.2 Shear Moduli, $G'$and G''

When a nearly isotropic material is subjected to an oscillatory displacement field with frequency ν, we may write the strain as $\epsilon(t) = \epsilon_0 e^{-i\nu t}$, where i is the imaginary unit and ϵ_0 is the strain amplitude. For cases where the strain is primarily only shear strain equation (2.13) gives, $\boldsymbol{\sigma}(\nu) \approx G_1^*(\nu)\boldsymbol{\epsilon}(\nu)$ where $G_1^*(\nu)$ is the Fourier transform in time of $G_1(t)$. In general $G_1^*(\nu)$ is a complex valued function. Breaking the complex shear modulus into its real and imaginary components, $G_1^*(\nu) = G'(\nu) + iG''(\nu)$; and given a strain amplitude $\epsilon_0(\nu)$ and stress amplitude $\sigma_0(\nu)$ (in Pascals),

$$G'(\nu) = \frac{\sigma_0(\nu)}{\epsilon_0(\nu)} \cos\delta(\nu), \qquad G''(\nu) = \frac{\sigma_0(\nu)}{\epsilon_0(\nu)} \sin\delta(\nu). \tag{2.19}$$

Here, $\delta(\nu)$ is known as the loss angle, measured in radians at frequency ν. In the literature, $G'(\nu)$ and $G''(\nu)$ are often referred to as the storage and loss moduli. They correspond to the elastic and viscous components of a viscoelastic stress strain relationship.

Taking the domain to be a rectangular solid, oriented as shown in Figure 2.1, we assume that all fields are periodic in the x and z directions. We use the following boundary conditions:

$$\left.\frac{\partial P}{\partial y}\right|_{y=0,y_L} = 0 \quad \mathbf{u}(x,0,z,t) = 0 \quad \mathbf{u}(x,y_L,z,t) = (0\,,0\,,u_b(t)\,). \tag{2.20}$$

Along the top boundary, we set the z velocity to be

$$u_b(t) = \epsilon_0 \nu c(\nu) \frac{(e^{2\nu t}-1)((e^{4\nu t}-1)\cos\nu t + 8e^{2\nu t}\sin\nu t)}{(1+e^{2\nu t})^3}. \tag{2.21}$$

This particular function is chosen since it is continuous, at $t = 0$, $u_z = 0$, and because it converges to within 0.001 of $\epsilon_0 \nu c(\nu)\cos\nu t$ within half an oscillation, reducing the amount of time needed to run simulations. Experimentally, the strain amplitude, ϵ_0 was set to a fixed value over ν. In order to approximate this in simulation, we had to use a prefactor, $c(\nu)$ that can vary slightly in order to maintain similar strain amplitudes at each frequency we simulated. Slightly different values of G' and G'' can be found with different values of $c(\nu)$, however the change is small as long as $c(\nu)$ does not vary too much. In one test, a variation of 0.4957 in $c(\nu)$ yielded a variation in the moduli of approximately 0.4610 Pa in G' and 0.040 Pa in G'' at a frequency of 40 Hz.

To initialize the bacteria positions, we take a $9\,\mu m \times 27\,\mu m \times 9\,\mu m$ subset of a $30\,\mu m \times 30\,\mu m \times 10\,\mu m$ bacteria position data field obtained experimentally. This data is also used in the initialization of the viscosity and density fields present in the biofilm. We believe that setting the internal forces to zero at the start is reasonable since experimental results from SAR under both compression and tension yielded similar results in [77]. In Figure 2.5, the deformation induced by an oscillatory shearing motion is depicted.

In order to tune our model to the experimental data, we adjusted the biofilm viscosity, μ_b, the spring constant, k_{ij} used in Hooke's Law, and the cutoff radius with which we allow any two biofilm nodes to be connected by at time $t = 0$. For Maxwell's law, we gain an additional viscoelastic parameter that can be tuned, and with the Zener model, there are two additional parameters. In the simulations, we assume that any bacteria separated by a distance of less than $1.62\,\mu m$ are connected. This number was chosen as it leads to biofilms that appear to have similar connectivity to images of live biofilms, such as in [9].

For a spring connecting two points in space, Hooke's Law can be written as

$$\mathbf{F}_{ij} = k_{ij}\,\Lambda_{ij}(\mathbf{X},\,\mathbf{X}_0)\,(\mathbf{X}_i - \mathbf{X}_j), \tag{2.22}$$

with

$$\Lambda_{ij}(\mathbf{X},\,\mathbf{X}_0) = \frac{||\mathbf{X}_i(t) - \mathbf{X}_j(t)|| - ||\mathbf{X}_i(0) - \mathbf{X}_j(0)||}{||\mathbf{X}_i(t) - \mathbf{X}_j(t)||}. \tag{2.23}$$

Following [52], we choose each k_{ij} to be a force constant, F_{s1} divided by the initial separation of bacteria i and j. With Maxwell's law and the Zener model, we use an integral form of the standard viscoelastic relation and, integrate by parts to obtain a formula for $\mathbf{F}$. This is done since integration by parts obviates the need to compute any time derivatives of stress or strain. We then approximate the integral at each time step using the trapezoid method. Since the immersed boundary method requires a Lagrangian force density, we divide the values of $\mathbf{F}_{ij}$ determined from the viscoelastic force law by the Lagrangian volume element, d_0^3 to obtain an average force density associated with each bacteria.

In previous works [30, 79], identical constitutive relations to (2.22) are derived from the

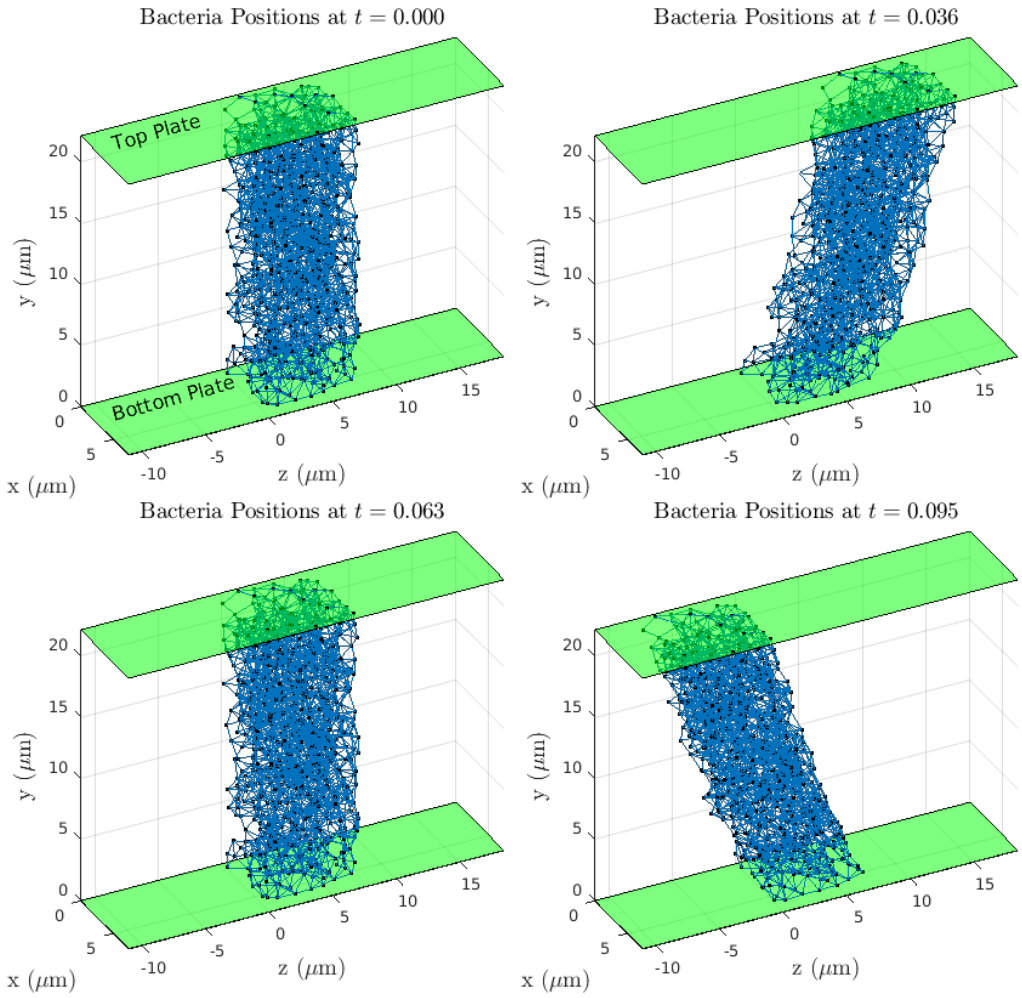


Figure 2.5: Starting at $t = 0$ on the left, the images show how the biofilm is moved as the top plate oscillates. Dots are bacteria locations and lines indicate viscoelastic connections. In the simulations the domain is periodic in the x and z directions. The periodicity is not shown here since it makes it difficult to visualize the effect of deformation on the biofilm.

starting point of energy functionals from which the force density is found by taking a Fréchet derivative of the energy functional. Additionally, in [101] and [31], similar viscoelastic force laws are studied in an immersed boundary method setting and in [101] corresponding dynamic shear moduli are computed.

In Figure 2.6 we depict the frequency dependence of G' and G'' as determined by the hrIBM model, and compare with experimental data. From these results, it is clear that our model fits experimental data on G' quite well. For G'' the fit is not as strong, although we still do see that for the Maxwell and Zener models, most of the results from simulation are within the range of experimental error. We also observe that even with Hooke's Law which is an elastic constitutive relation, the biofilm still behaves viscoelastically (i.e. $G'' \neq 0$) due to its immersion in a viscous fluid.

2.4.3 Creep Compliance Measurements $J(t)$

Creep compliance is a measure of how a material deforms over time in response to an applied stress. In a rheometer, the compliance is measured applying a step change to the shear stress on the upper plate and observing the resultant shear strain. A step change in stress can be written in terms of a Heaviside step function, $H(t)$ as, $\bar{\sigma}(t) = \sigma_0 H(t)$ where σ_0 is the magnitude of the step change. For a linear isotropic material with $\sigma_{yz}(t) = \bar{\sigma}(t)$, the integral in Equation (2.14) simplifies to

$$\epsilon_{yz}(t) = \sigma_0 J_1(t). \tag{2.24}$$

From a physical standpoint, most of the bacteria and the bulk of the fluid are only effected by the step change in stress after the stress propagates vertically through the biofilm. However, in the portion of the biofilm adjacent to the upper plate the effect of a change in stress is instantaneous. Thus, we can write a force balance between the forces in the biofilm, the acceleration of the top plate, and the applied force on the top plate. Using this balance of forces, we obtain a time dependent

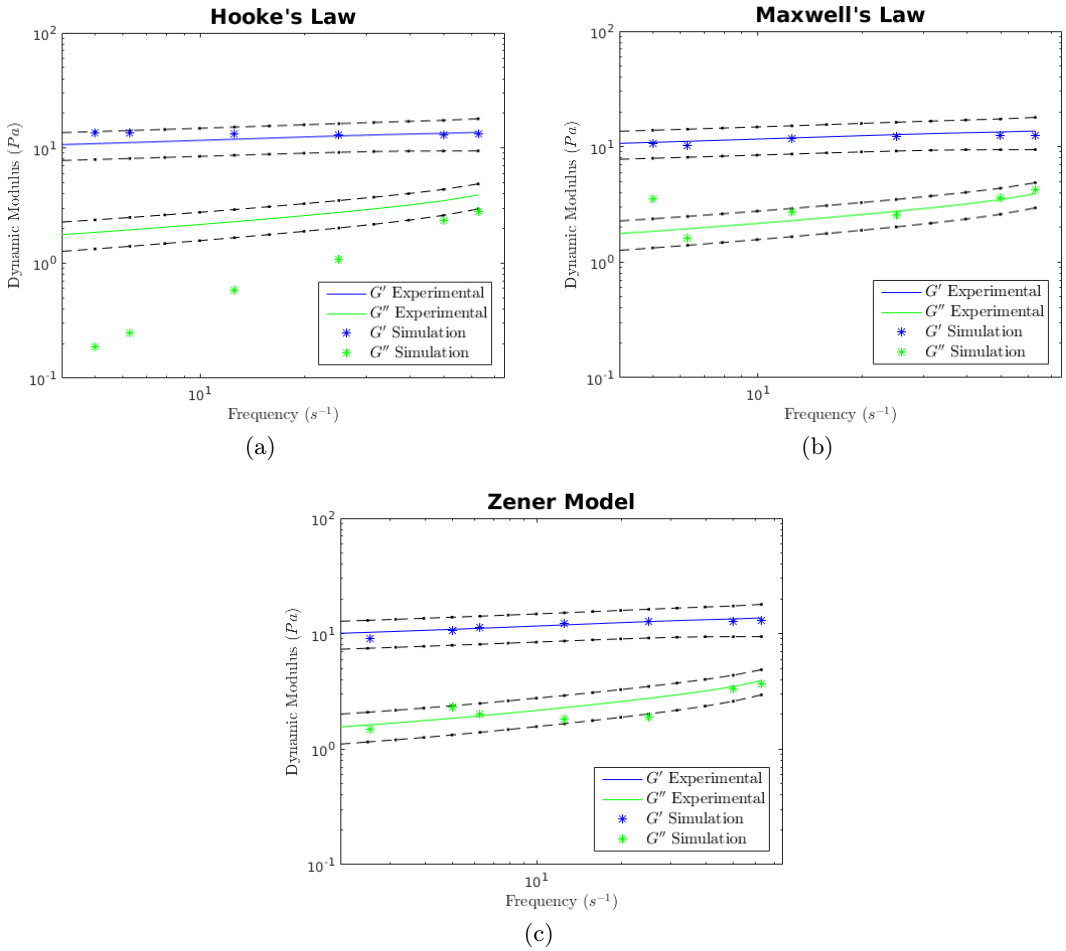


Figure 2.6: Comparison between experimentally measured results for G' and G'' and simulated results for three viscoelastic models are depicted. For these results, the force constant was $F_{s1} = 2.30 \cdot 10^{-11} N$, the connection distance between bacteria was $1.62 \mu m$, $\mu_b = 350 \mu_0$, and $\rho_b = 1.12 \rho_0$. The dashed lines indicate the experimental error ranges. Subfigure (a) uses Hooke's Law, Subfigure (b) uses Maxwell's law, and Subfigure (c) uses the Zener model. For the Maxwell model, a damping coefficient $F_{d1} = 1.03 \cdot 10^{-11} N/s$ was used, and for the Zener model, the spring coefficient was set to $F_{s1} = 1.64 \cdot 10^{-11}$N , the damping coefficient was set to $F_{d1} = 0.051 \cdot 10^{-11}$N/s and, a second spring coefficient of $F_{s2} = 1.64 \cdot 10^{-11} N$ was used.

boundary condition for velocity at the top plate. This boundary condition can be written as

$$\left.\frac{d}{dt}u_z\right|_{y=H} = (m_{plate})^{-1}(\sigma_0 - \sigma^b - \sigma^f)A_{plate} \tag{2.25}$$

where σ^b and σ^f are the stress exerted by the fluid and the springs in the biofilm at the top plate, A_{plate} is the area of the upper plate, and m_{plate} is the mass associated with the top plate of the rheometer. We found that using a mass of $4.22 \cdot 10^{-11}$ kg yielded results similar to those seen experimentally.

Numerically, this boundary condition can be written as

$$u_z^{(n+1)}|_{y=H} = u_z^{(n)}|_{y=H} + \left(\frac{A_{plate}\Delta t}{m_{plate}u_0}\right)(\sigma_0 - \sigma^b - \sigma^f). \tag{2.26}$$

In (2.26), ρ_0 and μ_0 are the density and viscosity of water, and L is the characteristic length (in this case $10\,\mu m$). In numerical experiments, rather than immediately impose a step in stress at time 0, we multiply the applied stress σ_0 by a mollifier, $\left(\frac{2}{1-e^{-\alpha t}} - 1\right)$ to mitigate any possible numerical instabilities associated with a discontinuous jump in the wall stress. In this case, $\alpha = 155$ is chosen to be large so that the applied stress approaches its equilibrium value within 0.1 seconds. We believe this is reasonable because the very short time scale compliance behavior is not experimentally measurable, and also because a true step change in the stress is not physically realizable. Results from simulations using this boundary condition and two values of σ_0 are shown in Figure 2.7.

2.4.4 Similarity of Material Properties Between Different Bacteria Position Data Sets

In [33], the spatial statistics of bacteria in a biofilm are studied. We show here that from data sets that have similar spatial distributions of nearest neighbor connections between bacteria, similar bulk property measurements are obtained. In our simulations, we take blocks of experimental data that are $18\,\mu m$ wide, $9\,\mu m$ long, and $27\,\mu m$ high and compute the dynamic moduli of each block at a fixed frequency . The graphs in Figure 2.8 show the stress and strain of four different biofilms over one period of oscillation.

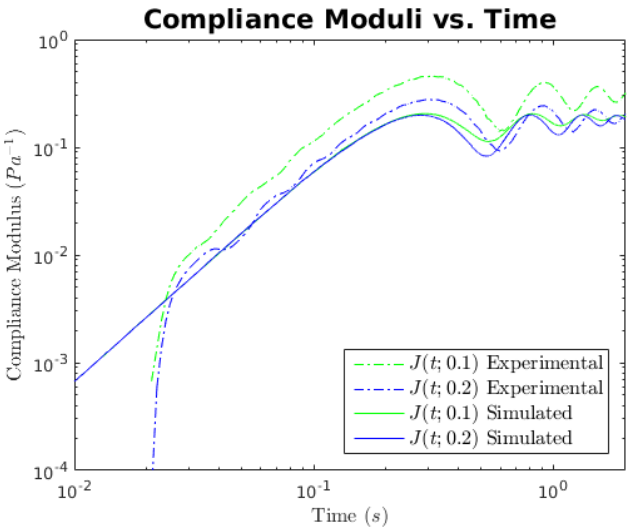
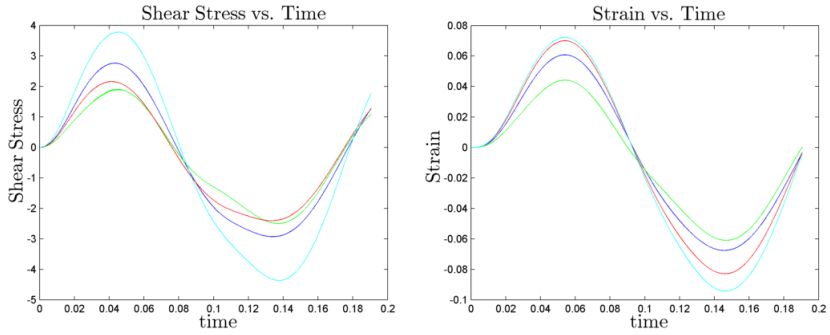


Figure 2.7: The time dependence of $J(t, \sigma_0)$ is shown above for $\sigma_0 = 0.1\,Pa$ and $\sigma_0 = 0.2\,Pa$. The Zener model as used in these simulations. For these levels of applied stress, the change in the compliance modulus is only slight indicating a nearly linear regime.



(a) (b)

Figure 2.8: These graphs show the stress vs. time and strain vs. time for 4 different biofilm samples. The samples are all $18\mu m \times 27\mu m \times 9\mu m$ size and contain approximately 2000 bacteria positions.

Table 2.7: Results for $G'(\nu)$ and $G''(\nu)$ are shown at two frequencies for 3 different biofilm coordinate data sets with $\nu = 49.91\,rad/s$. These results show that the physical properties measured here do not depend solely on the exact microstructure of the biofilm, but on some type of more large scale organization of the bacteria positions in space.

Simulation	1	2	3	Experimental	Experimental Std. Err.
G'	9.03	12.71	8.17	13.399	3.9791
G''	2.98	3.31	2.28	3.4812	0.8853

From Figure 2.8 and Table 2.7,

we see that in three of the four biofilm data sets, similar results are obtained. We also note that the mean standard error in the experimental results for this particular test was 3.9791 for G' and 0.8836 for G''. We also suspect that if larger data sets were used, even better agreement would be seen in the computed values of G' and G''.

The importance of this section is in verifying that the properties we are validating can be considered as bulk properties. Since three out of the four data sets provided results within the experimental error deviation, we believe this to be strong evidence the properties we measure are bulk properties.

2.4.5 In-Stream Tumbling of Biofilm Fragment

Bacterial structures exhibit a diverse range of behaviors when subject to fluid flow. One such behavior is the tumbling motion of aggregates in shear flow. To simulate this effect, we conduct simulations in which there are no bacteria anchored along the plates of the domain. Instead, an aggregate of bacteria is located near the middle of the computational domain and the upper and lower plates move in opposite directions as

$$u_z(x,\, y_L,\, z) = 10^{-3} = -u_z(x,\, 0,\, z). \tag{2.27}$$

The boundary velocities are scaled by 10^{-3} to provide an average shear rate of approximately $114 s^{-1}$. This in the range of shear rates examined in [15]. To ensure that the biofilm is not attached to the plates and is sufficiently far from the plate to induce a rotating, or tumbling motion, these simulations only include bacteria that are greater than $8.8\,\mu m$ from either plate at the start. With the physical parameters we use, the bacteria aggregation rotates and is deformed by the fluid shear forces exerted by the fluid [15]. Several snapshots from a simulation of an aggregate tumbling are shown in Figure (2.9).

In [11], analytical results on the frequency at which a solid ellipsoid will rotate in shear flow are provided. For an ellipsoid with axis aligned with the direction of fluid motion, the rotational

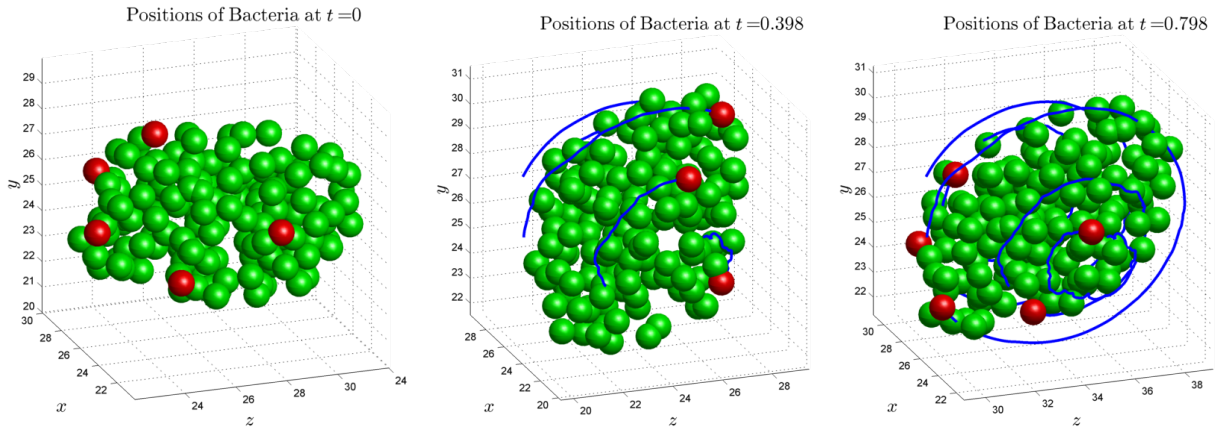


Figure 2.9: Biofilm aggregate suspended in shear flow rotate over time. Snap shots shown at 0, 0.4 and 0.8 seconds into the simulation. The flattening of the ellipse in response to the shear flow can be distinguished between the first and third figure. Several bacteria are marked red to help show the rotation of the aggregate. The blue lines indicates the trajectories of the marked cells relative to the center of mass of the aggregate. Distances are in micrometers.

Table 2.8: Comparison of theoretical and simulated rotational results

Major axis, a_1	First minor axis, a_2	Theoretical Period	Observed Period	Relative Error
$2.344\mu m$	$1.268\mu m$	$0.1317s$	$0.1579s$	+19.9%
$1.646\mu m$	$1.194\mu m$	$0.1116s$	$0.1542s$	+38.2%

frequency is found as

$$T = \frac{2\pi(a_1^2 + a_2^2)}{a_1 a_2 \tau}$$

where τ is the shear rate and $a_1$and a_2 are the principle axes of the ellipse undergoing rotation. In our simulation (see Figure 2.9), we show that a bacterial aggregate approximated as a hydrodynamically equivalent ellipse will rotate at a frequency similar to the theoretically expected result. The rotational frequency of the aggregate is found by computing the average frequency of rotation of bacteria in the yz plane about the center of mass of the aggregate. In the simulations, we observed a frequency of approximately 0.158 seconds for an aggregate approximated by an ellipse with major axis $a_1 = 2.344\,\mu m$ and first semimajor axis $a_2 = 1.268$, and with shear rate of $114\,s^{-1}$ containing 110 bacteria. The theoretical result in this case is 0.1317 seconds. For smaller aggregates, the elliptical assumption appears to become less accurate although it still provides results that are on the same order of magnitude. These results are shown in Table 2.8.

Although we do not currently have experimental results to compare our simulations with, rotational frequency measurement do provide a strong metric for model validation in fields such as red blood cell modeling (see [36]).

2.5 Conclusions

Based on the experimental results shown above and in Part I [52], the **heterogeneous rheology Immersed Boundary Method** model or hrIBM (Equations (2.1)-(2.8)), appears to be an accurate model of the biomechanical response of bacterial biofilms to fluid motion. With the hrIBM model, we resolve biofilms to the scale of individual bacteria. This enables the use of various viscoelastic stress-strain relationships to account for polysaccharide links between bacteria. The

spatially heterogeneous ECM also plays a prominent role in the hrIBM model and is treated as a variable viscosity and variable density fluid. We recreate a continuous spatially varying viscosity field, with increased viscosity in the vicinity of the center of mass of each bacteria, by adapting the interpolation procedure found in the immersed boundary method. Although variable viscosity was introduced into the IBM's in [36, 37], this work is the first to consider viscosity as a continuously varying field, not an approximation to a step change in viscosity. Additionally, the authors believe that this work and Part I are the first to consider variable viscosity in an IBM model of biofilms.

In Section 2.4, a major effort was spent towards validating the hrIBM model by comparing material properties computed by the hrIBM model to experimental data. In particular, we show that the model yields close agreement with experimental results from [77] in which the bacterium **S. epidermidis** was grown in a bioreactor and characterized using a parallel plate rheometer. Based on the simulations we have conduced it appears that viscoelastic models that contain a Maxwell element provide results comparable to experimental data whereas models that lack a Maxwell element become less accurate at low frequencies. Another development is in showing the similarity between the bulk dynamic moduli over different experimental data sets that possess similar spatial statistics. We also show that suspended aggregates of bacteria in shear flow rotate with a similar period as a hydrodynamically equivalent ellipse. To our knowledge, the hrIBM model is the first that can accurately compute bulk material properties of biofilms based on coupling the microscale connectivity of the biofilm with the heterogeneous rheology of the ECM.

2.6 Future Directions

Currently the scheme is $\mathcal{O}(\Delta t)$ and $\mathcal{O}(h)$ for the velocity field and bacteria positions. In future work, a predictor-corrector method as used in [36, 37], or the Crank-Nicholson time-stepping scheme may be used since, at least in the constant viscosity and density case, these methods can lead to $\mathcal{O}(\Delta t^2)$ convergent Navier-Stokes solvers as shown in [13]. However, even without heterogeneous material properties, obtaining $\mathcal{O}(\Delta t^2)$ convergence in the overall IBM is more complicated and also depends on properties of the discrete Dirac δ function as discussed in [65, 66].

Modelling and simulation of biofilm detachment is an area of interest in biofilm research. One way to model detachment in our biofilm model would be the inclusion of a stress or strain criterion for when the viscoelastic links in the biofilm should rupture. This approach is explored in [93] where the von Mises stress criterion is used. Another direction would be to follow along the lines of [12], and include model of the reconfiguration of the connectivity of the biofilm over time, especially as the biofilm is deformed due to fluid motion. It is also possible to model the viscoelastic properties of the biofilm through a more complex constitutive model of the Lagrangian force based on the nodal configuration of the bacteria.

Another area that could be explored is the shape of the discrete δ function used to approximate each bacteria and its associated viscosity halo. It is possible that adjusting this function may allow for more accurate modeling of the mass displacement induced by the bacteria bodies in the bulk fluid. Adjustments to the δ function may also allow for the inclusion of non-spherical bacteria into the model. It would be interesting to see if similar results are obtained for bacteria that are different shapes.

One possible difficulty in adjusting the discrete δ function is the preservation of mass in the model. In Equation (2.2), there is no density dependence as is often seen with the Navier-Stokes equations for a variable density system. For the original IBM, Equation (2.2) is in fact the correct statement of mass preservation as described in [79]. For the hrIBM, there is an error however, the error is expected to be small in our situation since $\rho(\mathbf{x}, t)$ only varies by $\approx 20\%$ over the domain (density of bacteria is not highly variable), all simulations are at low Reynold's numbers, and because biofilms are essentially incompressible even though they may have variable density.

2.7 Acknowledgements

This work was supported in part by the National Science Foundation grants PHY-0940991 and DMS-1225878 to DMB, and PHY-0941227 to JGY and MJS, and by the Department of Energy through the Computational Science Graduate Fellowship program, DE-FG02-97ER25308, to JAS. This work utilized the Janus supercomputer, which is supported by the National Science Foundation

(award number CNS-0821794), the University of Colorado Boulder, the University of Colorado Denver, and the National Center for Atmospheric Research. The Janus supercomputer is operated by the University of Colorado Boulder. We would also like to thank the reviewers for several useful suggestions that improved this chapter.

Chapter 3

A Point Process Model for Biofilm Rheology

3.1 Introduction

Biofilms are complex, multi-organism communities of bacteria. They are abundant in nature and grow readily in many industrial systems where they often cause maintenance and safety issues. Measures to mitigate or remove biofilms, though costly, are often necessary in the design and operation of many industrial systems [39, 49]. The demand for better biofilm control strategies and the causative role of biofilms in bacterial infections has inspired the development of numerous mathematical biofilm models [42, 49, 52, 63, 89, 93, 98, 104, 105, 106]. Despite a thirty year history, it is only recently that the validation of models by comparison with empirical data has become a focus in computational biofilm studies. Recent works have shown agreement between experiment and simulation regarding the frequency dependent dynamic moduli [89], the ranges of certain spring constants [98], and the rate of disinfection of biofilms in response to antimicrobial substances [106]. However, this recent progress has also led to new questions.

Due to the complex behavior of biofilms, even state-of-the-art models rely on many assumptions. Material parameter values, biofilm morphology, and properties of a viscoelastic network of fibers that link bacteria together (known as the extracellular polymeric substance) are often specified heuristically. Furthermore, even when parameters are informed by experimental results, there are difficulties. For instance, the experimental measurements of biofilm properties may differ by orders of magnitude between studies depending on the specifics of the conditions under which biofilms were grown and tested [77].

The assumptions noted before involve microstructural properties. Although, some microstructural properties have been measured [33, 87], little work has been done to elucidate the influence of microstructural properties on macroscale behaviors. Along these lines, there are two main goals to this chapter. The first is to introduce a microstructural description of **S. epidermidis** biofilms by estimating certain fundamental statistical characteristics from experimental data. The second is to numerically demonstrate the efficacy of first and second order summary statistics for generating data with similar material properties as experimental data sets. The similarity of material properties is tested through simulation using the **heterogeneous rheology Immersed Boundary Method** (hrIBM) [52, 89].

3.1.1 Biofilm Point Data: Description and Pre-processing

Here we provide a summary of the techniques used to 1) obtain data on the locations of bacteria **in situ** and 2) process such data to make it suitable for statistical analysis. Detailed descriptions of these steps can be found in [87] and [77] and the references therein.

Measuring properties of live biofilms is difficult due to the micrometer scale thickness of biofilms and empirical observations that they deform upon removal from an aqueous environment. In [87], high resolution confocal laser scanning microscopy (CLSM) is used to image biofilms because it is non-destructive, can be applied to live biofilms, and resolves the biofilm up to features of about $0.5\mu m$ in size - small enough to capture individual bacteria. With CLSM, a series of images, each vertically offset by a small amount from the previous, are layered together to provide a three dimensional data set.

The raw CLSM data forms a three dimensional gray-scale image of a biofilm. Using the image analysis methods described in [27], the centroid of each individual bacteria can then be obtained to very high accuracy compared to the size of a bacteria. We use the list of bacteria centroids as the data for our model. An image of a biofilm point data set is depicted in Figure 3.1.

This data set contains a list of three dimensional coordinates of the centers of mass of about

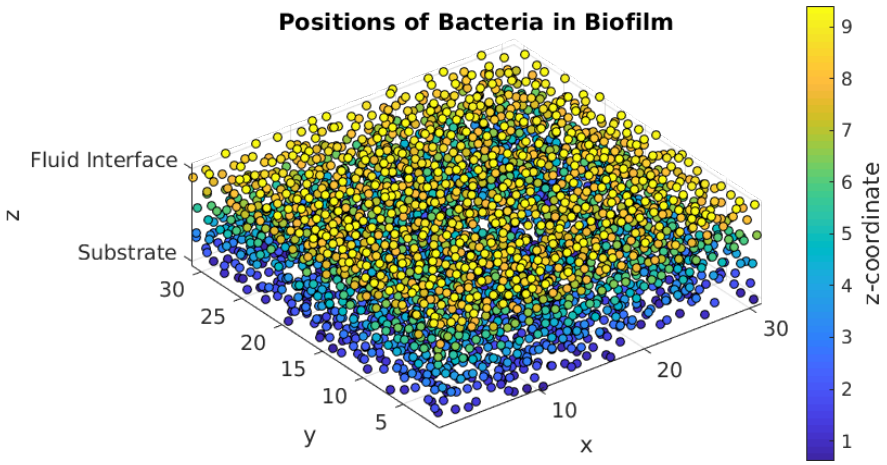


Figure 3.1: An experimental data set from a live biofilm. The data set is approximately $30\mu m \times 30\mu m \times 10\mu m$ in size and consists of the centers of mass of 3981 live bacteria. This data was obtained from high-resolution confocal microscopy images. The techniques used to obtain this data are discussed in [87]. The coloring is for ease of viewing and simply corresponds to the z coordinate of each bacterium.

4000 bacteria from live biofilms. Moving from $z = 0$ to $z \approx 10$, takes us from the biofilm-substrate interface to the biofilm-fluid interface. In this chapter, results from this data set and 3 additional data sets obtained through the same techniques serve as a basis on which our statistical analysis is conducted.

3.1.2 Overview of Mathematical Approach

To begin, we develop a point process model of the positions of bacteria in a biofilm that is parametrized by quantities derived from the experimental data. In particular, using estimators of the number density and pair correlation function, we use a numerical method to solve an integral equation, known as the Ornstein-Zernike equation, and employ an approximate relation known as the hypernetted-chain equation to obtain a pairwise potential energy function. This pair potential, along with an external potential, whose derivation from an integral relation is discussed in Section 3.2.2, is used to compute the value of an unnormalized probability density function associated with the configuration of points. We then generated biofilm realizations by using a Metropolis-Hastings algorithm with that is constructed to converge to a certain probability distribution discussed in Section 3.2. Finally, in Section 3.6 we use the heterogeneous rheology Immersed Boundary Method to compare material properties of the statistically generated biofilm samples to experimental data.

We also note that the overall procedure bears similarity to another recent article. In [38], a statistical model of red blood cell cytoskeletal structure parametrized by empirical data is introduced, and immersed boundary method simulations are used to estimate biomechanical properties of the model which are quantitatively compared to experimental data.

3.1.3 Organization of the chapter

In Section 3.2.1, we introduce the statistical model for the positions of bacteria in a biofilm. The model, known as a Pairwise Interaction Model (PIM), is designed to accurately replicate first and second order spatial statistics. Similar types of models are frequently studied in the statistical

mechanics of fluids, and much of our discussion of the model follows the classic texts [53, §5] and [71, §6].

In Section 3.3, we detail the non-parametric estimators of summary statistics of finite point processes that are necessary for the computation of energy functions in the PIM model. These estimators are based on those analyzed in [90, 91, 71, 56, 62, 94]. In Section 3.4, we discuss the numerical solution of the integral equations used to obtain the energy functions from estimable spatial statistics of the data.

From the PIM, an unnormalized probability density associated with the configuration of bacteria in the biofilm is obtained and used in Markov Chain Monte Carlo (MCMC) simulations to generate "artificial" biofilms [71, 103]. Details can be found in Section 3.5. Once realizations of the PIM have been obtained, the material properties of the artificial biofilms generated by the PIM, along with some previous biofilm generation models (e.g. [2, 93]), can be compared to those of the experimental data obtained through high resolution CLSM. This comparison is achieved through simulations using the hrIBM to estimate the dynamic moduli of the resulting biofilms. In Section 3.7, we discuss the results, motivate some future research directions, and suggest potential improvements.

3.2 A Pairwise Interaction Model of Bacterial Biofilms

In this section, we first introduce the statistical model and then discuss how parameters in the model are obtained through the use of statistical estimators and the solution of certain integral equations. In constructing a model of the positions of bacteria in a biofilm, we require that the model is physically realistic. In particular, the model should be able to accurately account for volume exclusion effects due to the finite size of bacteria, regularity and spatial coordination in the positions of bacteria, and possess statistical characteristics that match those computed from the experimental data. Furthermore, since relatively few attempts have been made to analyze such biofilm data, we require a model with parameters that may be estimated empirically.

A fairly general class of models that can account for these properties are the Markov point

process models [71, §6]. Among the various types of Markov point process models, pairwise interaction models (PIMs) are an attractive choice for this application. They are among the simplest Markov point process models, requiring only the specification of an external potential and a pairwise potential, and there is a well established theory for obtaining such potentials empirically from the data with minimal assumptions.

The probability density associated with a point configuration of n points, denoted $\boldsymbol{X}$, for a general PIM is of the form [53, §2]

$$f^{(n)}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) = \frac{1}{Z_n} \exp\left[-\sum_{i=1}^{n} \phi(\boldsymbol{x}_i) - \sum_{i=1}^{n} \sum_{j=i+1}^{n} v(\boldsymbol{x}_i, \boldsymbol{x}_j) \right] \tag{3.1}$$

where ϕ is known as an external potential, v a pairwise potential[1] , and the normalization factor, Z_n the canonical partition function. The probability density,
$f_n(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) d\boldsymbol{x}_1 \ldots d\boldsymbol{x}_n$ can be interpreted as the probability, given n distinct points $\{\boldsymbol{x}_1, \ldots \boldsymbol{x}_n\}$ of there being a single point in each of the regions of volume $d\boldsymbol{x}_i$ centered at $\boldsymbol{x}_i$.

Variations in the external potential are closely related to variations in the number density of bacteria in various parts of the domain. For instance, given data in a bounded region, outside of the bounded region, we could set $\phi(\boldsymbol{r}) = \infty$ so that $f^{(n)}$ tends to zero if any points are outside the domain. This property will be useful in Section 3.5.

For our case, the domain, denoted W is the smallest rectangular solid containing all data points. In particular, we set the minimum x, y and z coordinates to 0, and the maximum coordinates correspond to the maximum of each coordinate over all data points within the data sets. The assumption of W being a rectangular solid seems justified by the fact that the data is relatively homogeneously distributed throughout W (although there is some number density variation, see Section 3.3).

The form of the pair potential is crucial to accurately modeling interactions between bacteria. Since we do not know of previous studies on the statistical generation of biofilm data, we take

[1] In the statistical physics community, it is customary to include a temperature dependence on the energy as $\beta v(\boldsymbol{r}_1, \boldsymbol{r}_2)$ for the pair potential (or external potential) where β is an inverse temperature [53, §2]. In this chapter, we are not concerned with thermal effects, so we redefine $v \leftarrow \beta v$ and $\phi \leftarrow \beta\phi$. Furthermore, without some assumption on the "temperature" of the biofilm, β and v or ϕ are not independently estimable quantities.

an empirical approach to obtaining a pair potential. Other possible choices we could have made include Poisson point process models, Hard-Sphere models[2] , and Strauss Process models. In this chapter, we explore an empirical choice, and also provide some comparison with a Poisson point process. We favor the empirical model over the other models due to its ability to better match the pair correlation function shown in Figure 3.2a.

3.2.0.1 Empirical Choice for Potentials

The choice of $\phi(\boldsymbol{r})$ and $v(\boldsymbol{r}_1, \boldsymbol{r}_2)$ has a profound influence on the resulting model. Since, to the authors' knowledge, this work is the first to propose a PIM-type model for bacteria spatial positions, we have little intuition about the forms of ϕ and v, so would like to estimate them from data. Unfortunately, estimating $v(\boldsymbol{r}_1, \boldsymbol{r}_2)$, a function of six coordinates, empirically requires a vast amount of data; beyond what is available to us from experiment. Thus, we assume several conditions to reduce the dimension of the problem. These assumptions will be described in Section 3.2.3. The result is that the external potential becomes a function only of z, the coordinate which governs distance from the biofilm-substrate and biofilm-fluid interfaces, and the pairwise potential reduces to a function of z_1, z_2, and the planar separation $|\boldsymbol{r}_1 - \boldsymbol{r}_2|_{xy} \equiv \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2})$. For a non-negative, measurable function, $q(\boldsymbol{X})$ of finite point configurations, the expected value over all realizations of the PIM model, with an arbitrary number of points, can be written in the form

$$\mathbb{E}\left[q(\boldsymbol{X})\right] = \frac{1}{\Xi}\sum_{n=0}^{\infty}\frac{1}{n!}\underbrace{\int\cdots\int}_{n} q(\boldsymbol{r}_1,\ldots,\boldsymbol{r}_n) f^{(n)}(\boldsymbol{r}_1,\ldots,\boldsymbol{r}_n) d\boldsymbol{r}_1\ldots d\boldsymbol{r}_n, \tag{3.2}$$

where the normalization constant, Ξ is known as the grand-canonical partition function [53, 20]. The Markov property of the PIM comes from the assumption of $f^{(n)}(\cdot,\ldots,\cdot)$ being dependent on pairwise interactions between particles (or more generally on a Markov function as shown through the Clifford-Hammersley theorem [71, 85]).

[2] Hard-Sphere processes are point processes that have a minimum radius whereby the set of point configurations with a pair of points closer than the minimum radius is observed with probability 0.

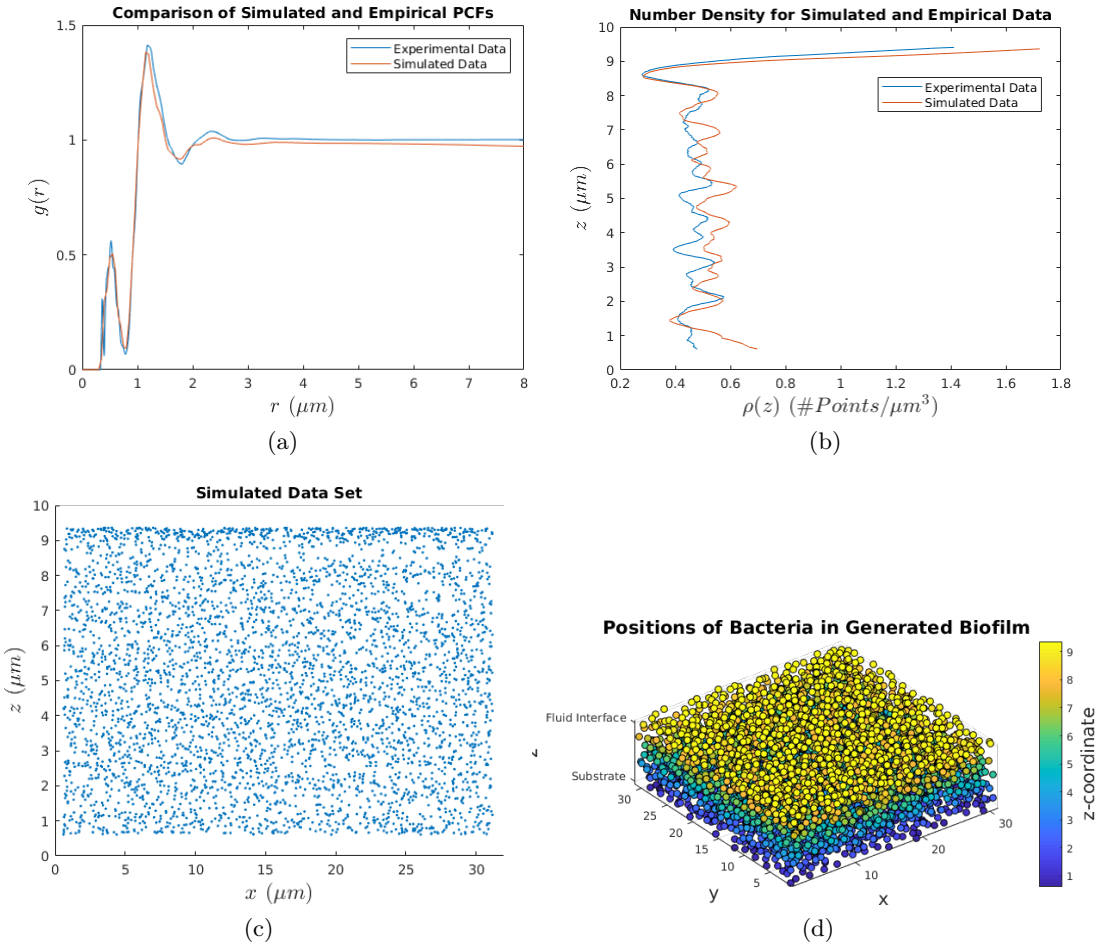


Figure 3.2: (a) Comparison between experimental and simulated data of the SOIRS estimator for a radially symmetric pair correlation function. The radially symmetric pair correlation exhibits peaks at $r = 0.525$, $r = 1.185$, and $r = 2.330$ micrometers. The relative heights of the first two peaks is an interesting finding, and has been observed in biofilm data in a previous study [87]. The first, smaller peak is likely due to a small number of bacteria that were undergoing cellular division at the time the data set was recorded although the possibility of some experimental error cannot be definitively ruled out [86]. (b) Comparison of number density between experimental data and simulated data. The simulation method used to generate data is discussed in Section 3.4. (c) Projection of the simulated data set into the xz-plane. (d) Image of a simulated biofilm generated form Algorithm 1.

3.2.0.2 Poisson Point Processes

Point processes are point processes with $v(\boldsymbol{r}_1, \boldsymbol{r}_2) = 0$, in other words, there are no interactions between pairs of points. An important property of a Poisson process is that given two disjoint Borel sets, B_1 and B_2, the cardinalities of $\boldsymbol{X}_{B_1}$ and $\boldsymbol{X}_{B_2}$ are independent, Poisson distributed random variables. Numerous characterizations of Poisson point processes exist, and they are frequently used as models for a wide range of random phenomena. The statistical properties of a Poisson point process do not agree with statistical properties of biofilm data because Poisson point processes do not exhibit any interactions between points. However, we will see in Section 3.6 that mechanical properties of data generated from a Poisson process with an average number density equal to that of the biofilm data do match experimental values.

In the next three subsections we discuss an important class of functions that arise in the study of point processes, show how the external potential and pairwise potential can be estimated from readily observable properties of the data, and provide more detail on the assumptions introduced to reduce the dimensionality of the PIM potentials.

3.2.1 Factorial Moment Measures, Product Densities, and the Pair Correlation Function

The theory of point processes [29, 71] provides a starting point from which to develop a model of the locations of bacteria in biofilms. Point processes are defined as measurable mappings from a probability space, to the space of locally finite point configurations. A locally finite point configuration, $\boldsymbol{X} \equiv \{\boldsymbol{x}_1, \boldsymbol{x}_2, \dots\}$, is a countable collection of distinct points in $\mathbb{R}^3$ such that for all bounded Borel sets, $B \subset \mathbb{R}^3$, the restriction,$\boldsymbol{X}_B \equiv \boldsymbol{X} \cap B$ is of finite cardinality. Point processes are typically characterized in terms of their probability distribution over locally finite point configurations [71, 91, §2] or as locally finite random counting measures [29, §9].

In the analysis of point processes, a set of measures known as the **factorial moment measures** (FMM) are of fundamental importance. For a general point process, under mild restrictions

(see [29, §9.2]), the FMMs can be written in terms of a density, known as a product density, with respect to the Lebesgue measure. Given a point process Φ, the first order FMM is defined on a Borel set, B as

$$\alpha^{(1)}(B) = \mathbb{E}\left[\sum_{\boldsymbol{x}\in\boldsymbol{X}} \mathbf{1}\left(\boldsymbol{x}\in B\right)\right] = \int \mathbf{1}(\boldsymbol{r}\in B)\rho^{(1)}(\boldsymbol{r})d\boldsymbol{r}$$

i.e. the expectation, over all locally finite point configurations, $\boldsymbol{X}$ distributed according to Φ (written $\boldsymbol{X}\sim\Phi$ henceforth), of the cardinality of $\boldsymbol{X}_B$. The second order FMM is defined by

$$\begin{aligned}\alpha^{(2)}(B_1\times B_2) &= \mathbb{E}\left[\sum_{\boldsymbol{x}\in\boldsymbol{X}}\sum_{\boldsymbol{y}\in\boldsymbol{X}\setminus\boldsymbol{x}} \mathbf{1}\left(\boldsymbol{x}\in B_1\right)\mathbf{1}\left(\boldsymbol{y}\in B_2\right)\right]\\ &= \int\int \mathbf{1}(\boldsymbol{r}_1\in B_1)\mathbf{1}(\boldsymbol{r}_2\in B_2)\rho^{(2)}(\boldsymbol{r}_1,\boldsymbol{r}_2)d\boldsymbol{r}_1 d\boldsymbol{r}_2.\end{aligned}$$

The first order product density, $\rho^{(1)}$, is called the **number density** or **intensity**. In this work, we favor the term **number density** over **intensity** since the raw experimental data has a grayscale intensity that does not correspond to the definition of intensity typically used in spatial point process literature. The density function, $\rho^{(2)}$ is called the **second order product density**.

The **pair correlation function** (PCF), defined as

$$g(\boldsymbol{r}_1,\boldsymbol{r}_2) = \frac{\rho^{(2)}(\boldsymbol{r}_1,\boldsymbol{r}_2)}{\rho^{(1)}(\boldsymbol{r}_1)\rho^{(1)}(\boldsymbol{r}_2)}$$

is also of great interest. It provides a characterization of the likelihood of two points $\boldsymbol{r}_1$ and $\boldsymbol{r}_2$ occurring simultaneously in a realization of Φ relative to a Poisson process with number density $\rho^{(1)}(\boldsymbol{r})$. Estimates of $g(\boldsymbol{r}_1,\boldsymbol{r}_2)$ are needed in Sections 3.4 and 3.6 in to compare the PIM model to solve certain integral equations and to compare the biofilm model to experimental data.

For PIMs, the FMMs and product densities can be related through functional differentiation techniques to Ξ, the grand canonical partition function.

3.2.2 Integral Equations for the External and Pair Potentials

With pairwise interactions, the relation between $\rho^{(1)}$ and ϕ is complicated, thus it is convenient to define a new function, $a(\boldsymbol{r}) \equiv \exp(-\phi(\boldsymbol{r}))$ (known as the **activity**, i.e. $z(\boldsymbol{r})$ in [53]). From

Symbol	Definition
W	Borel set containing experimental data
$\boldsymbol{r}$, (x, y, z)	generic point in $\mathbb{R}^3$
$\boldsymbol{x}$	point contained in a point process realization
$\boldsymbol{r}^n$	collection of n points in $\mathbb{R}^3$
$\boldsymbol{X}$	realization of a point process
Φ	Generic point process
$\nu(B)$	Lebesgue measure of a set, B
$\rho^{(2)}(\boldsymbol{r}_1, \boldsymbol{r}_2)$	2nd order factorial moment density
$g(\boldsymbol{r}_1, \boldsymbol{r}_2)$	2nd order correlation function
$h(\boldsymbol{r}_1, \boldsymbol{r}_2)$	indirect correlation function, $h^{(2)} = g^{(2)} - 1$
$f^{(n)}(\boldsymbol{r}^n)$	nth order probability density
$\rho(\boldsymbol{r})$	number density (also known as the intensity), equivalent to $\rho^{(1)}(\boldsymbol{r})$
$\hat{q}$	estimator for quantity q
$\tilde{q}$	Hankel transform of quantity q
$\mathbb{E}[q]$	expectation of q
$k_b(r)$	smoothing kernel with support of radius b
$c_{b,W}(z)$	edge correction factor
$c(\boldsymbol{r}_1, \boldsymbol{r}_2)$	direct correlation function
$\phi(\boldsymbol{r})$	external potential
$v(\boldsymbol{r}_1, \boldsymbol{r}_2)$	pair potential
$\boldsymbol{e}_x$	unit vector in x-coordinate direction
$H_k(r)$	nearest-neighbor distribution function

Table 3.1: Definitions of commonly used symbols. The nth order probability density when integrated over a product set, $B_1 \times B_2 \times \ldots B_n$, with $B_i \cap B_j = \emptyset$, gives the probability that for each $\boldsymbol{r}_i$, there exists exactly one B_k such that $\boldsymbol{r}_i \in B_k$ for $i \in [1, n]$.

data, we can immediately estimate $\rho^{(1)}$ and g, but not ϕ and v. Thus, we review how integral equations relating ϕ and v to $\rho^{(1)}$ and g are derived through functional differentiation techniques. The discussion here summarizes [53, §5], and numerical approximations methods used to obtain ϕ and v from the data are described in Section 3.4

Given an external potential and using the definition of the functional derivative[3] ,

$$\phi(\boldsymbol{r}_1 + \boldsymbol{s}) = \phi(\boldsymbol{s}) + \int \frac{\delta\phi(\boldsymbol{r}_1)}{\delta\rho^{(1)}(\boldsymbol{r}_2)} \left(\rho^{(1)}(\boldsymbol{r}_2 + \boldsymbol{s}) - \rho^{(1)}(\boldsymbol{s})\right) d\boldsymbol{r}_2. \tag{3.3}$$

To obtain $\delta\phi/\delta\rho^{(1)}$, it ends up being convenient to define a related function,

$$c(\boldsymbol{r}_1, \boldsymbol{r}_2) \equiv \frac{\delta \log\left(a(\boldsymbol{r}_1)/\rho^{(1)}(\boldsymbol{r}_1)\right)}{\delta\rho^{(1)}(\boldsymbol{r}_2)},$$

known as the **direct correlation function** (DCF). From the definition of $\phi(\boldsymbol{r}_1)$,

$$\frac{\delta\phi(\boldsymbol{r}_1)}{\delta\rho^{(1)}(\boldsymbol{r}_2)} = -\frac{\delta \log a(\boldsymbol{r}_1)}{\delta\rho^{(1)}(\boldsymbol{r}_2)} = -\frac{1}{\rho^{(1)}(\boldsymbol{r}_1)}\delta(\boldsymbol{r}_1 - \boldsymbol{r}_2) + c(\boldsymbol{r}_1, \boldsymbol{r}_2). \tag{3.4}$$

Using this result in Equation (3.3), and taking the limit as $\boldsymbol{s} \to \boldsymbol{0}$, an equation for ϕ is obtained,

$$\nabla \log \rho^{(1)}(\boldsymbol{r}_1) + \nabla\phi(\boldsymbol{r}_1) = \int c(\boldsymbol{r}_1, \boldsymbol{r}_2)\nabla_{\boldsymbol{r}_2}\rho^{(1)}(\boldsymbol{r}_2)d\boldsymbol{r}_2. \tag{3.5}$$

The integration is over all of space, thus, includes any regions where $\rho^{(1)}$ may exhibit a discontinuity due to domain boundaries [67]. Noting that the derivative of a jump discontinuity in the distributional sense leads to a Dirac delta, we may expand Equation (3.5) as

$$\nabla \log \rho^{(1)}(\boldsymbol{r}_1) + \nabla\phi(\boldsymbol{r}_1) = \int_V c(\boldsymbol{r}_1, \boldsymbol{r}_2)\nabla_{\boldsymbol{r}_2}\rho^{(1)}(\boldsymbol{r}_2)d\boldsymbol{r}_2 + \int_{\partial V} c(\boldsymbol{r}_1, \boldsymbol{r}_2)\rho^{(1)}(\boldsymbol{r}_2)d\boldsymbol{r}_2$$

where V is the domain where $\rho^{(1)}$ is positive, and ∂V is the boundary of V. As noted in [67], it is also possible to integrate by parts to eliminate the boundary terms, however, this requires approximation of the gradient of $c(\boldsymbol{r}_1, \boldsymbol{r}_2)$. However such an approach is more difficult numerically than approximating the number density derivative.

[3] Given a functional, $\Psi(f)$, the functional derivative [26, ch. 4 §3] with respect to $f(\boldsymbol{r})$ is defined by the relation

$$\int \frac{\delta\Psi}{\delta f}(\boldsymbol{r})q(\boldsymbol{r})d\boldsymbol{r} = \frac{d}{d\epsilon}\left[\Psi(f(\boldsymbol{r}) + \epsilon q(\boldsymbol{r}))\right]_{\epsilon=0}.$$

Now we must derive an expression for $c(\boldsymbol{r}_1, \boldsymbol{r}_2)$ in terms of computable quantities. To solve for $c(\boldsymbol{r}_1, \boldsymbol{r}_2)$, we use the functional differentiation identity,

$$\int \frac{\delta\phi(\boldsymbol{r}_1)}{\delta\rho^{(1)}(\boldsymbol{r}_3)} \frac{\delta\rho^{(1)}(\boldsymbol{r}_3)}{\delta\phi(\boldsymbol{r}_2)} d\boldsymbol{r}' = \delta(\boldsymbol{r}_1 - \boldsymbol{r}_2). \tag{3.6}$$

Defining $h(\boldsymbol{r}_1, \boldsymbol{r}_2) \equiv g(\boldsymbol{r}_1, \boldsymbol{r}_2) - 1$, it is possible to show that

$$\begin{aligned} \frac{\delta\rho^{(1)}(\boldsymbol{r}_1)}{\delta\phi(\boldsymbol{r}_2)} = \frac{\delta^2 \log \Xi}{\delta a(\boldsymbol{r}_1)\delta a(\boldsymbol{r}_2)} &= \rho^{(1)}(\boldsymbol{r}_1)\rho^{(1)}(\boldsymbol{r}_2) - \delta(\boldsymbol{r}_1 - \boldsymbol{r}_2)\rho^{(1)}(\boldsymbol{r}_1) - \rho^{(2)}(\boldsymbol{r}_1, \boldsymbol{r}_2). \\ &= \delta(\boldsymbol{r}_1 - \boldsymbol{r}_2)\rho^{(1)}(\boldsymbol{r}_1) - h(\boldsymbol{r}_1, \boldsymbol{r}_2). \end{aligned} \tag{3.7}$$

Substituting our results from Equations (3.7) and (3.4) into relation (3.6), we obtain an equation known as the Ornstein-Zernike equation [73]

$$h(\boldsymbol{r}_1, \boldsymbol{r}_2) = c(\boldsymbol{r}_1, \boldsymbol{r}_2) + \int \rho^{(1)}(\boldsymbol{r}_3) c(\boldsymbol{r}_1, \boldsymbol{r}_3) h(\boldsymbol{r}_3, \boldsymbol{r}_2) d\boldsymbol{r}_3. \tag{3.8}$$

This equation provides a definition for $c(\boldsymbol{r}_1, \boldsymbol{r}_2)$ in terms of known quantities, and thus an equation for the density variation can be solved once $c(\boldsymbol{r}_1, \boldsymbol{r}_2)$ is found.

Next, we seek a relation between $c(\boldsymbol{r}_1, \boldsymbol{r}_2)$, $h(\boldsymbol{r}_1, \boldsymbol{r}_2)$ and $v(\boldsymbol{r}_1, \boldsymbol{r}_2)$. This can be done using the functional expansion technique described in [53, §5]. The strategy is to assume a pairwise interaction system with a particle fixed at $\boldsymbol{x} = \boldsymbol{R}$. This particle induces a potential of the form

$$\phi(\boldsymbol{r}) = v(\boldsymbol{r}, \boldsymbol{R}).$$

Expanding the function $\log\left(\rho_\phi^{(1)}(\boldsymbol{r})/a_\phi(\boldsymbol{r})\right)$ in terms of $\rho_\phi^{(1)}(\boldsymbol{r})$ about the reference number density $\rho_0^{(1)}(\boldsymbol{r})$ (i.e. the number density with $\phi(\boldsymbol{r}) = 0$), we obtain

$$\begin{aligned} \log\left(\frac{\rho_\phi^{(1)}(\boldsymbol{r})}{a_\phi(\boldsymbol{r})}\right) &= \log\left(\frac{\rho_0^{(1)}(\boldsymbol{r})}{a_0(\boldsymbol{r})}\right) \\ &\quad + \int \left[\frac{\delta}{\delta\rho^{(1)}(\boldsymbol{r}')} \log\left(\frac{\rho_\phi^{(1)}(\boldsymbol{r})}{a_\phi(\boldsymbol{r})}\right)\right]_{\phi=0} \left(\rho_\phi^{(1)}(\boldsymbol{r}') - \rho_0^{(1)}(\boldsymbol{r}')\right) d\boldsymbol{r}' \\ &\quad + \mathcal{O}\left(\left|\rho_\phi^{(1)}(\boldsymbol{r}) - \rho_0^{(1)}(\boldsymbol{r})\right|^2\right). \end{aligned}$$

By noting that $\rho_{\phi}^{(1)}(\boldsymbol{r}) = \rho_0^{(1)}(\boldsymbol{r}) g_0(\boldsymbol{r}, \boldsymbol{R})$ and $a_\phi(\boldsymbol{r}) = a_0(\boldsymbol{r}) \exp(-\phi(\boldsymbol{r}))$, the lefthand side reduces to $\log g_0(\boldsymbol{r}, \boldsymbol{R}) + v(\boldsymbol{r}, \boldsymbol{R}) + \log\left(\rho_0^{(1)}(\boldsymbol{r})/a_0(\boldsymbol{r})\right)$ and, by ignoring higher order terms the righthand side is simplified using Equation (3.8) to yield

$$v(\boldsymbol{r}, \boldsymbol{R}) = h(\boldsymbol{r}, \boldsymbol{R}) - c(\boldsymbol{r}, \boldsymbol{R}) - \log g(\boldsymbol{r}, \boldsymbol{R}). \tag{3.9}$$

This result is known as the **hypernetted chain equation** (HNC) [53]. It relates the pair potential to the DCF and PCF.

It can be seen that the external potential and pair correlation can be found by first solving the the Ornstein-Zernike equation for $c(\boldsymbol{r}_1, \boldsymbol{r}_2)$, then computing $\phi(\boldsymbol{r})$ from Equation (3.5), and finally solving for $v(\boldsymbol{r}_1, \boldsymbol{r}_2)$ in terms of $c(\boldsymbol{r}_1, \boldsymbol{r}_2)$ and $g(\boldsymbol{r}_1, \boldsymbol{r}_2)$ through Equation (3.9).

3.2.3 Simplifying Assumptions

At the beginning of this Section 3.2.1, we mentioned that certain assumptions are made to reduce the dimensionality of the integral equations. We now describe several concepts that lead to the assumptions we make.

In [8], the concept of a **second order intensity reweighted stationary (**SOIRS) point process was introduced. They are defined for strictly positive number densities in terms of the measure,

$$M(B_1 \times B_2) = \mathbb{E}\left[\sum_{\boldsymbol{x} \in \boldsymbol{X} \cap B_1} \sum_{\boldsymbol{y} \in \boldsymbol{X} \cap B_2} \frac{1}{\rho(\boldsymbol{x})} \frac{1}{\rho(\boldsymbol{y})}\right].$$

This measure is closely related to the second order factorial moment measure, $\alpha^{(2)}(B_1 \times B_2)$. In particular,

$$M(B_1 \times B_2) = \int_{B_1} \int_{B_2} \frac{\rho^{(2)}(\boldsymbol{r}_1, \boldsymbol{r}_2)}{\rho(\boldsymbol{r}_1)\rho(\boldsymbol{r}_2)} d\boldsymbol{r}_1 d\boldsymbol{r}_2 = \int_{B_1} \int_{B_2} g(\boldsymbol{r}_1, \boldsymbol{r}_2) d\boldsymbol{r}_1 d\boldsymbol{r}_2.$$

If $M(B_1 \times B_2)$ is translation invariant, we say that the point process is SOIRS. This implies that the PCF of a SOIRS point process is translation invariant. Thus, $g(\boldsymbol{r}_1, \boldsymbol{r}_2) = g(\boldsymbol{r}_1 - \boldsymbol{r}_2)$. Our first assumption is that the biofilm data results from realizations of a SOIRS point process.

The SOIRS assumption leads to a PCF dependent on a single three-dimensional variable representing the vectorial difference between two points. However, the further assumption of isotropy, or that only the magnitude of the distance, but not the angular orientation between the pair of points matters, allows for the simplification, $g(\boldsymbol{r}_1 - \boldsymbol{r}_2) = g(|\boldsymbol{r}_1 - \boldsymbol{r}_2|)$.

Second, we assume that the density varies only as a function of distance from the substrate. We make this assumption due to symmetry considerations; the bacteria are grown on a uniformly planar plate. Thus, we do not expect any difference in the number density for any two points in space that are of equal distance from the plate.

We show in Appendix A.3, that a number density that varies along one dimension, and an isotropic, translation invariant PCF are consistent with a DCF (and pair potential) of the form $c(\boldsymbol{r}_1, \boldsymbol{r}_2) = c(|\boldsymbol{r}_1 - \boldsymbol{r}_2|, z_1, z_2)$. Defining $r_{xy} \equiv |\boldsymbol{r}|_{xy} = \sqrt{x^2 + y^2}$, Equation (3.5) reduces to

$$\begin{aligned}\frac{d}{dz_1}\left(\log\rho(z_1) + \phi(z_1)\right) = 2\pi \int_{z_0}^{z_{max}} \left(\int_0^\infty c(r_{xy}, z_1, z_2) r_{xy} dr_{xy}\right) \frac{d}{dz_2}\rho(z_2) dz_2 \\ + 2\pi \left[\rho(0) \int_0^\infty c(r_{xy}, z_1, 0) r_{xy} dr_{xy}\right. \\ \left. -\rho(z_{max}) \int_0^\infty c(r_{xy}, z_1, z_{max}) r_{xy} dr_{xy}\right].\end{aligned} \tag{3.10}$$

Formulas of this type appear to have first been derived in the context of understanding fluid interfaces in [67]. Applying a transformation $\boldsymbol{r}_3' = \boldsymbol{r}_3 - \boldsymbol{r}_2$, Equation (3.8) reduces to

$$\begin{aligned}h(|\boldsymbol{r}_1 - \boldsymbol{r}_2|) =& c(|\boldsymbol{r}_1 - \boldsymbol{r}_2|_{xy}, z_1, z_2) \\ &+ \iiint \rho(z_3' + z_2) h(|\boldsymbol{r}_3'|) c(|\boldsymbol{r}_1 - \boldsymbol{r}_2 - \boldsymbol{r}_3'|_{xy}, z_1, z_3' + z_2) dx_3' dy_3' dz_3'.\end{aligned} \tag{3.11}$$

Similarly, the HNC (Equation (3.9)) becomes

$$v(|\boldsymbol{r}_1 - \boldsymbol{r}_2|_{xy}, z_1, z_2) = h(|r_1 - \boldsymbol{r}_2|) - c(|\boldsymbol{r}_1 - \boldsymbol{r}_2|_{xy}, z_1, z_2) - \log g(|\boldsymbol{r}_1 - \boldsymbol{r}_2|).$$

With these assumptions, the nth order probability densities associated with the PIM are of the form

$$f^{(n)}(\boldsymbol{x}_1, \ldots \boldsymbol{x}_n) = \frac{1}{Z_n} \exp\left(-\sum_{i=1}^n \phi(z_i) - \sum_{i=1}^n \sum_{j>i}^n v(|\boldsymbol{x}_i - \boldsymbol{x}_j|, z_i, z_j)\right).$$

To summarize, the assumptions are listed below:

(1) The process is a **Second Order Intensity Reweighted Stationary** (SOIRS) point process [8, 71].

(2) The pair correlation function (PCF) is isotropic.

(3) The number density is only dependent on the distance from the fluid-biofilm interface, i.e. $\rho(z) \equiv \rho^{(1)}(\boldsymbol{r} = (x, y, z))$.

It remains to i) estimate the number density and pair correlation function, ii) numerically solve the OZ equation and density integral equation, and iii) justify our model assumptions. These are the subjects of the next two sections.

3.3 Estimation of Summary Statistics

In this section we discuss the estimation of $\rho(z)$ and $g(r)$, and the nearest neighbor distributions, defined in Section 3.3.3, denoted $H_k(r)$. In the case of empirical data, without prior intuition about the point process, non-parametric estimators are often the tool of choice for the estimation of these characteristics. This is the approach we take since we are not aware of any previous models of the spatial statistics of bacteria in biofilms. For notation, we use a hat (i.e. $\hat{\rho}$) to denote an estimator for a quantity.

In addition to being flexible, non-parametric estimates are often easy to implement and versatile, yielding suitable intensity approximations for many classes of point processes. However, a common source of concern with non-parametric estimates is the presence of statistical bias [83, 99, 51]. For ρ, g, and H_k; non-parametric estimates are rarely pointwise unbiased [83], but are often asymptotically unbiased. Heuristically, the bias is due to inaccuracy in the numerical approximation of a Radon-Nikodym derivative [10] and is analogous to the truncation error associated with finite difference approximations of differentiable functions. Furthermore, as finite difference approximations depend on a spatial increment, non-parametric number density estimators are parametrized[4] by a bandwidth, b. Smaller values of b reduce the bias, but increase the variance of the estimator

[4] The term **non-parametric** refers to the absence of an underlying assumption about the functional form of

[75, 83, 99]. To ensure accurate estimators, careful selection of b is needed to balance this tradeoff between bias and variance. Further discussion of this balancing of bias and variance is included in the Appendices A.1 and A.2.

3.3.1 Number Density Estimators

As discussed in Section 3.2.1, the number density of $\boldsymbol{X} \sim \Phi$ is defined through the relation

$$\mathbb{E}\left[\sum_{\boldsymbol{x}\in\boldsymbol{X}} 1(\boldsymbol{x} \in B)\right] = \int_B \rho^{(1)}(\boldsymbol{r})d\boldsymbol{r}.$$

Thus, we see that $\sum_{\boldsymbol{x}\in\boldsymbol{X}} \mathbf{1}(\boldsymbol{x} \in B)$ is an unbiased estimator for $\alpha^{(1)}(B)$. However, $\rho^{(1)}(\boldsymbol{r})$ cannot generally be estimated without bias since it is defined as the average over an infinitesimal region,

$$\rho^{(1)}(\boldsymbol{r}) \equiv \lim_{r\to 0} \frac{\alpha^{(1)}(B_r(\boldsymbol{r}))}{\nu(B_r(\boldsymbol{r}))},$$

where $\nu(\cdot)$ is the Lebesgue measure and B_r is a ball of radius r centered at $\boldsymbol{r}$. We see that this is also related to the Dirac delta distribution through the relation

$$\rho^{(1)}(\boldsymbol{r}) = \lim_{r\to 0} \int_B \frac{1}{4\pi r^3} 1(|\boldsymbol{r}' - \boldsymbol{r}| \le r)\rho^{(1)}(\boldsymbol{r}')d\boldsymbol{r}' = \int \delta(\boldsymbol{r} - \boldsymbol{r}')\rho^{(1)}(\boldsymbol{r}')d\boldsymbol{r}'.$$

This motivates an approximation of the form

$$\rho^{(1)}(\boldsymbol{r}) \approx \int k_b(\boldsymbol{r} - \boldsymbol{r}')\rho^{(1)}(\boldsymbol{r}')d\boldsymbol{r}' = \mathbb{E}\left[\sum_{\boldsymbol{x}\in\boldsymbol{X}} k_b(\boldsymbol{r} - \boldsymbol{x})\right]$$

and an estimator

$$\hat{\rho}(\boldsymbol{r}) = \sum_{\boldsymbol{x}\in\boldsymbol{X}} k_b(\boldsymbol{r} - \boldsymbol{x})$$

where $k_b(\cdot)$ is known as a kernel density and b is the bandwidth. For data in $\mathbb{R}^n$, $k_b(\cdot)$ is typically a radially symmetric function of the form $k_b(\cdot) = \frac{1}{b^n} k_1(\cdot/b)$. There exist many choices for $k_b(\cdot)$, however, as long as $\int k_1(\boldsymbol{r})d\boldsymbol{r} = 1$ and $k_1(\cdot)$ is compactly supported, the resulting estimators are typically relatively insensitive to the choice of $k_1(\cdot)$, but highly sensitive to the choice of b. Lastly,

$\rho(\boldsymbol{r})$, not the absence of any tuning parameters in the estimator. Aside from a few simple examples, non-parametric estimators almost always have some sort of bandwidth parameter which must be chosen based on the length scales over which variations are seen in the data.

because of the distributional convergence of $\int k_b(\boldsymbol{r}-\boldsymbol{r}')\phi(\boldsymbol{r}')d\boldsymbol{r}'$ to $\int \delta(\boldsymbol{r}-\boldsymbol{r}')\phi(\boldsymbol{r}')d\boldsymbol{r}'$ in the limit as $b \to 0$, we say that $\hat{\rho}(z)$ is asymptotically unbiased. This estimator is similar to the number density estimator discussed in [91, §4]. For what follows, we specify $k_b(z)$ as

$$k_b(z) = \left(\frac{3}{4b}\right)\left(1-(z/b)^2\right)\mathbf{1}(|z|<b).$$

This kernel is known as the **Epanechnikov kernel** [34], a commonly used density estimation kernel.

Due to the limited size of a data set, we consider the dependence of the intensity on x, y, and z coordinates separately (for instance, assume that $\rho(\boldsymbol{r})$ only depends on x, independent of y and z). Given the restriction $\boldsymbol{X}_W$, this leads to estimators of the form

$$\hat{\rho}(z) = \sum_{\boldsymbol{x}'=(x',y',z')\in\boldsymbol{X}_W} \frac{k_b(z-z')}{c_{b,W}(z)}. \tag{3.12}$$

Defining A as the area of a cross-section perpendicular to the z-axis and W to be a set in $\mathbb{R}^3$ containing the data, the denominator, $c_{b,W}(z)$ in Equation (3.12), is an edge correction factor defined as

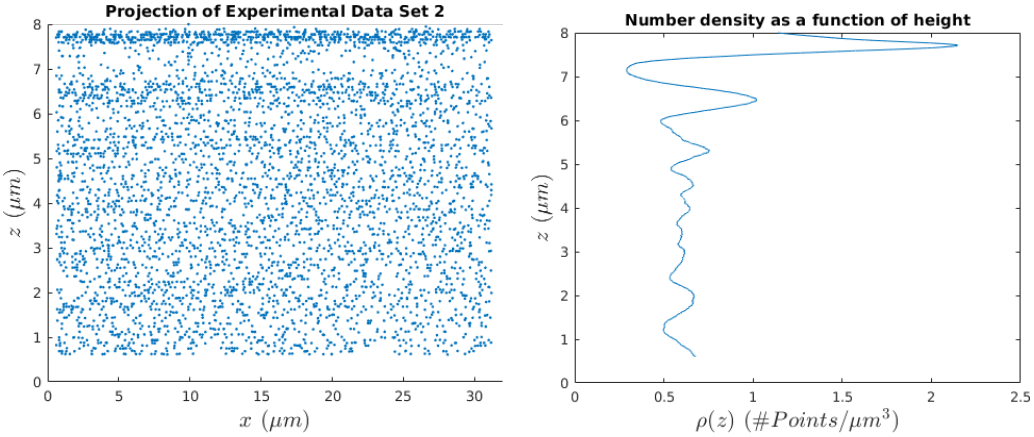
$$c_{b,W}(z) = A\int_{z-b}^{\min(z+b,z_{max})} k_b(z'-z)dz'. \tag{3.13}$$

where z_{max} is the largest value of the z-coordinate of points in W. We see that for $z_{max}-z<b$ given a differentiable number density,

$$A\int_{z-b}^{z_{max}} k_b(z-z')\rho^{(1)}(z')dz' - A\rho^{(1)}(z) = \\ A\rho^{(1)}(z)\left(\int_{z-b}^{z_{max}} k_b(z-z')dz'-1\right)+\mathcal{O}\left(Ab\frac{d}{dz}\rho^{(1)}(z)\right).$$

Thus, if we choose $c_{b,W}(z)$ as in Equation (3.13), the dominant, zeroth order error term is eliminated.

Upon computation, we found that although the bacteria locations are three dimensional coordinates, the number density only varies significantly along the z-axis, with changes in the distance from the fluid-biofilm interface. This justifies Assumption 1 of Section 3.2.3. As depicted in Figure 3.3a, it appears as though bacteria near the fluid-biofilm interface stratify into layers and pack more closely together than cells further interior. In Figure 3.3b, this phenomenon is clearly seen as bumps in the number density.



(a) (b)

Figure 3.3: (a): The projection of one of the data sets into the xz-plane is shown. The number density variation near the top of the domain can clearly be observed and some oscillatory intensity variation is discernible. (b): The height dependent intensity, $\rho(y)$, from the same experimental data set is depicted. We used a bandwidth, $b = 0.2$ for all four data sets.

The number density estimates obtained from data sets #1-4 exhibit some variation near the biofilm-fluid interface, thus in each simulation of Section 3.6, we parametrize our model on the number density from one of the data sets (e.g. from data sets #1-4). In Section 3.6, we separate the comparisons between model data and experimental data by data set. (see Figures 3.4a and 3.4b).

We also found that the precise form of the number density variation near the top of the biofilm did not have a substantial impact on the simulations conducted in Section 3.6. The sensitivity of those simulations to variability in the number density was examined by solving the integral equations of Section 3.2 with a constant number density (set to the number of points in W divided by $\mathrm{Vol}(W)$), and then conducting simulations equivalent to those in Section 3.6. We found that the dynamic moduli (discussed in Section 3.6) were very close to those of the variable density data. We expect that with more extreme variation in the number density, the similarity would disappear.

As previously mentioned, the value of the bandwidth, b, has a significant impact on the resulting number density estimate and should be carefully chosen to balance variance and bias. If b is too small, the estimate will be noisy, whereas if b is too large, key features of the data will be blurred. The estimation of a one dimensional intensity function is quite similar to the non-parametric estimation of a probability density up to the normalization condition required of probability densities. This motivates the adaptation of the **Least Squares Cross-Validation** (LSCV) technique, a common optimization strategy for non-parametric probability density estimators [99], to optimize our choice for b in Equation (3.12). We found that although the LSCV gave a reasonable estimate of b, the result could be improved upon by adapting the Bayesian technique in [45] with a prior centered about the value of b estimated by LSCV. We refer the interested reader to Appendix A.1 for a discussion of LSCV bandwidth selection and the Bayesian approach of [45].

In addition to the number density, Equation (3.5) depends on the gradient of the number density. In order to compute this quantity, we adhere to the strategy discussed in [99] and use the differentiable, triweight kernel density estimator, defined as

$$T_b(z) = \frac{35}{32b}(1-(z/b)^2)^3\mathbf{1}_{\{|z|\leq b\}}.$$

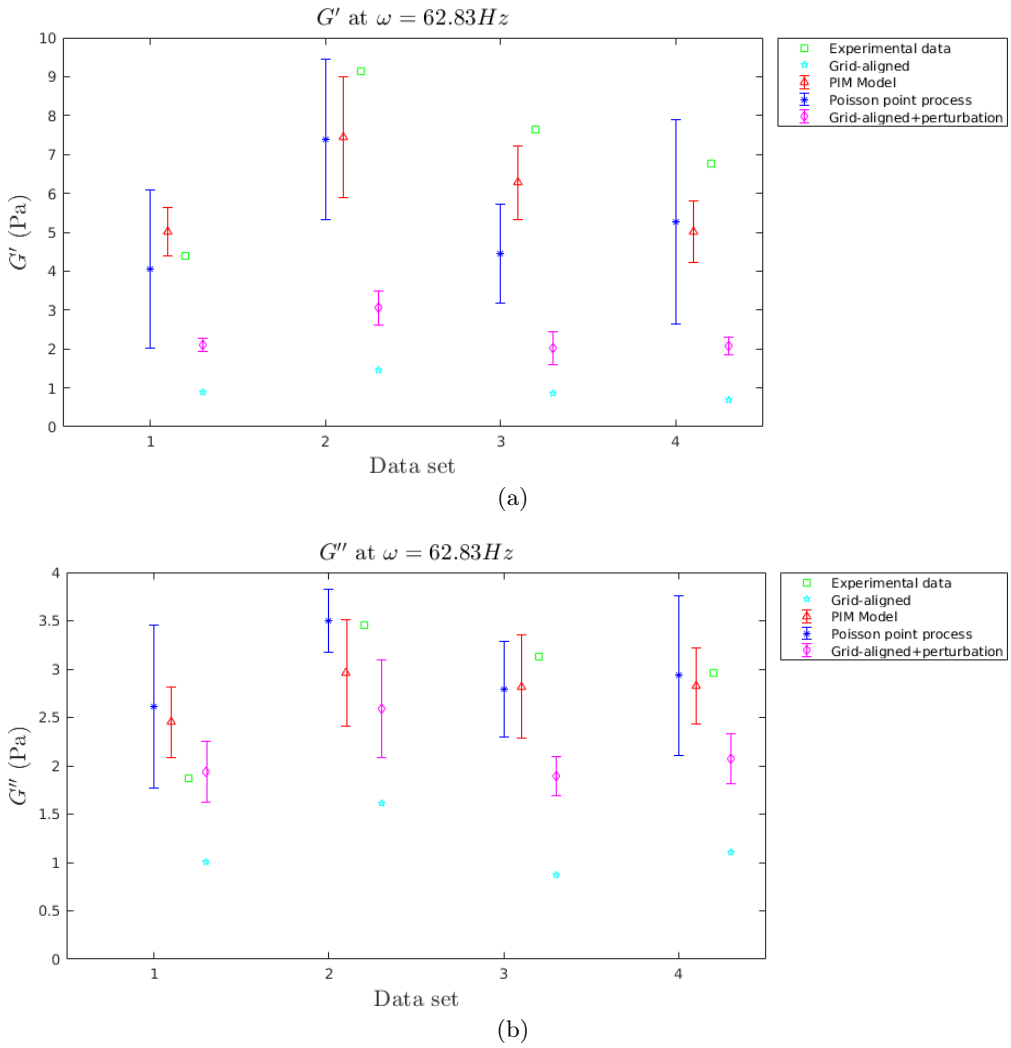


Figure 3.4: Comparison of Material Properties between experimental data and simulated data sets. Figure (a) shows the comparison between the statistical models and the experimental data for G' and Figure (b) shows the the comparison for G''. Both figures are from simulations at $\omega = 62.83Hz$. For the grid aligned data, the data falls on regular grid of the form $\boldsymbol{X} = (ih, jh, kh)$, for i, j, k being integers. For the Poisson point process data (e.g. random data) and PCF Model, each entry corresponds to the average of 5 samples containing the same number of points and domain as each of four experimental data sets. For the grid aligned plus perturbation, the results are from five trials with the grid aligned data and a perturbation drawn from a normal distribution of mean 0 and variance of $0.2\mu m$. The x axis numbering corresponds to each of the four data sets the statistical models were designed to compare with. Error bars in Figure 3.4 are 95% confidence intervals of the posterior distribution of dynamic moduli observed from each point process model. These are computed using Matlab's **normfit** function.

Similar to the Epanechnikov kernel, the triweight kernel is a symmetric probability density function. However, it is more useful for density derivative estimation since it is twice differentiable whereas the Epanechnikov kernel is not differentiable. The number density derivative, ρ', is approximated by

$$\hat{\rho}'(z) = \frac{1}{A} \sum_{\boldsymbol{x}' \in \boldsymbol{X}_W} \frac{\partial T_b(z - z')}{\partial z}.$$

As derivatives are more sensitive to noisy data, optimal balancing of variance and bias in number density derivative estimation tends to yield a larger value of b than that used in number density estimation [54]. By inspection of the resulting estimators, we found that a value of $b = 0.5$ seemed to give the best result. In particular, at $b = 0.5$, we found that $\hat{\rho}'$ was not overly noisy, and agreed well with finite difference estimates of the derivative of $\hat{\rho}$. However, in contrast to a finite difference estimate, it is very straightforward to compute $\hat{\rho}'$ for any value of z. To correct for bias near the boundaries of the domain, adjustments to the kernel constructed by the technique in [55, Equation (8.2)] were used.

3.3.2 Pair Correlation Function Estimates

Since the PCF is assumed to be stationary, and isotropic, $g(\boldsymbol{r}_1, \boldsymbol{r}_2)$, generally a function of six coordinates, is simplified to become $g(|\boldsymbol{r}_1 - \boldsymbol{r}_2|)$, a function of one coordinate. The advantages of this simplification for non-parametric estimation is substantial. Following [8, 47], an asymptotically unbiased estimator of the stationary PCF for a SOIRS point process with known $\rho(\boldsymbol{r})$ observed in a set, $W \subset \mathbb{R}^3$ is

$$\hat{g}_1(r) = \frac{1}{4\pi r^2 \bar{\gamma}_W(r)} \sum_{\boldsymbol{x} \in \boldsymbol{X}_W} \sum_{\boldsymbol{y} \in \boldsymbol{X}_W \setminus \boldsymbol{x}} \frac{k_b(r - |\boldsymbol{x} - \boldsymbol{y}|)}{\rho(\boldsymbol{x} \cdot \hat{\boldsymbol{e}}_z)\rho(\boldsymbol{y} \cdot \hat{\boldsymbol{e}}_z)}, \tag{3.14}$$

where $\bar{\gamma}_W(r)$ is the isotropized set covariance of W [91], defined by

$$\bar{\gamma}_W(r) = \int \nu(W \cap W_{\boldsymbol{t}}) d\boldsymbol{t}$$

with integration occuring over the surface of a sphere of radius r in $\mathbb{R}^3$. We also note that discussions of alternative choices for PCF estimates can be found in [94, 62, 56, 90]. We chose the estimator

in Equation (3.16) since it yields a continuous estimate of $g(r)$, and because it is relatively easy to implement (although the choice of b presents some difficulty).

Following [90], the expectation of $\hat{g}(r)$ is of the form:

$$\mathbb{E}\left[\hat{g}_1(r)\right] = \int g(|\boldsymbol{r}'|)k_b(|\boldsymbol{r}'| - r)d\boldsymbol{r}'. \tag{3.15}$$

Taking the limit as $b \to 0$ in Equation (3.15) shows that, under a few restraints on $k_b(r)$, $\hat{g}_1(r) \to g(r)$ at every point of continuity of $g(r)$. Thus, $\hat{g}_1(r)$ is asymptotically pointwise unbiased for continuous pair correlation functions. [5]

Since $\rho(\boldsymbol{r})$ is not known in our application, we use an estimator of the form

$$\hat{g}(r) = \frac{1}{4\pi r^2 \bar{\gamma}_W(r)} \sum_{\boldsymbol{x} \in \boldsymbol{X}_W} \sum_{\boldsymbol{y} \in \boldsymbol{X}_W \setminus \boldsymbol{x}} \frac{k_b(r - |\boldsymbol{x} - \boldsymbol{y}|)}{\hat{\rho}(\boldsymbol{x} \cdot \hat{\boldsymbol{e}}_z)\hat{\rho}(\boldsymbol{y} \cdot \hat{\boldsymbol{e}}_z)}. \tag{3.16}$$

Unlike $\hat{g}_1(r)$, the use of an approximate number density causes $\hat{g}(r)$ to fail to be asymptotically unbiased.

Equation (3.14) is valid for isotropic SOIRS point processes only. An estimator of the form

$$\hat{g}_a(\boldsymbol{r}) = \frac{1}{4\pi r^2 \nu(W \cap W_{\boldsymbol{r}})} \sum_{\boldsymbol{x} \in \boldsymbol{X}_W} \sum_{\boldsymbol{y} \in \boldsymbol{X}_W \setminus \boldsymbol{x}} \frac{k_b(|\boldsymbol{r} - (\boldsymbol{x} - \boldsymbol{y})|)}{\hat{\rho}(\boldsymbol{x} \cdot \hat{\boldsymbol{e}}_z)\hat{\rho}(\boldsymbol{y} \cdot \hat{\boldsymbol{e}}_z)}, \tag{3.17}$$

can be used to obtain an anisotropic PCF estimate.

In Figures 3.5a and 3.5b, $\hat{g}_a(\boldsymbol{r})$, averaged over four data sets is depicted. We display the averaged estimate since the anisotropic estimator is subject to higher variance. Data set averaging is employed as a variance reduction technique.

It can be seen that, although there is some anisotropy, the isosurfaces of $\hat{g}_a(\boldsymbol{r})$ are roughly spherical. Thus we believe the isotropic approximation of $g(\boldsymbol{r})$ is reasonable. In light of this result, we will assume in what follows that g is isotropic. In Figure 3.2 the SOIRS radially symmetric PCF estimator defined in (3.14), computed from one data set, is depicted.

An intriguing property of the biofilm data is the presence of two prominent peaks in the pair correlation function. It has been suggested that the first, smaller peak in Figure 3.2 is indicative

[5] For discontinuous pair correlation functions, such as those arising from hard-sphere processes, the estimate is not asymptotically pointwise unbiased at the point of discontinuity, but is asymptotically convergent to $g(r)$ in the least squares sense.

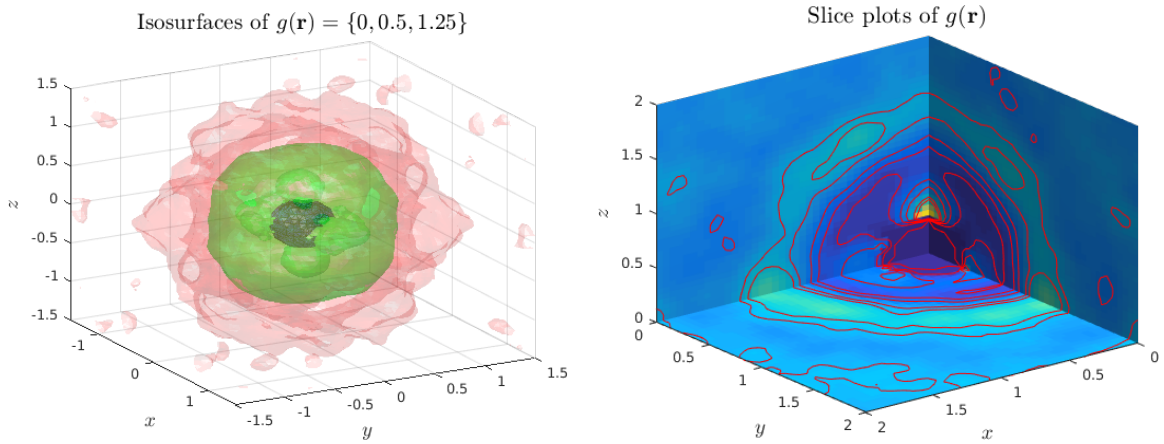


Figure 3.5: Both images are of PCFs averaged over the four experimental data sets. In (a): isosurfaces of the anisotropic pair correlation function at $g^{(2)}(\boldsymbol{r}) = 0$ (inner isosurface), 0.5 (middle), and 1.25 (outer) are depicted, and (b): contains slice plots and contours of the anisotropic pair correlation function. The anisotropic PCF is related to the isotropic PCF through the integral $g(r) = \frac{1}{4\pi} \int \int g_a(r, \phi, \theta) \sin\theta d\theta d\phi$. The anisotropic pair correlation function shows some vertical anisotropy, however most of the variation appears as a radially symmetric function.

of bacteria undergoing cell division at the time their position was measured [86]. The second peak occurs near the average diameter of a non-dividing bacterium. The third peak at $r = 2.330\mu m$ is indicative of local regularity in the positions of bacteria [53, §5]. Lastly, we also applied the mode detection test described in [84] to rule out the likelihood of further maxima in the interval $[0, 2.5]$. We found that for the bandwidth's deemed suitable by the LSCV procedure described in Appendix A.2, further peaks in the PCF are unlikely.

Recall from Figure 3.3 that there is evidence of variability in the number density along the z axis. This motivated our assumption of a SOIRS process in analyzing the data. However, we have not provided any evidence against an alternative hypothesis: that the PCF is not translation invariant. To make a convincing argument in favor of the SOIRS assumption (or at least that it is reasonable), it is necessary to estimate the magnitude of the variability of the pair correlation

To test for variability in the pair correlation function, its estimator, $\hat{g}(r)$ is calculated for various subsets of the full data set, $\boldsymbol{X}_W$. In particular, we compute $\hat{g}(r)$ over sets of the form $V_z = [z - \Delta z, z + \Delta z] \cap W$ with $z \in (\min(z_i), \max(z_i))$ where z_i are the z-coordinates of the bacteria, and $\Delta z = 1\mu m$. Computing $\hat{g}(r)$ on V_z as a function of z, we found only small variations in the resulting PCF. To fully utilize the available data in these computations, we use an altered estimator of $g(r)$, denoted $\hat{g}_{V_z}(r)$,

$$\hat{g}_{V_z}(r) = \frac{1}{4\pi r^2 \bar{\gamma}_{V_z,W}(r)} \sum_{\boldsymbol{x} \in \boldsymbol{X}_{V_z}} \sum_{\boldsymbol{y} \in \boldsymbol{X}_W \setminus \boldsymbol{x}} \frac{k_b(r - |\boldsymbol{x} - \boldsymbol{y}|)}{\hat{\rho}(\boldsymbol{x} \cdot \hat{\boldsymbol{e}}_z)\hat{\rho}(\boldsymbol{y} \cdot \hat{\boldsymbol{e}}_z)}$$

with

$$\bar{\gamma}_{V_z,W}(r) = \frac{1}{4\pi r^2} \int \nu(W \cap V_{z,\boldsymbol{t}}) d\boldsymbol{t}.$$

The integration is carried out over the surface of a sphere of radius r in $\mathbb{R}^3$. This altered estimator is used since it takes into account data that is in W but not V_z in the inner summation. This reduces the truncation errors that could occur if we used the estimator in Equation (3.14). It is in essence a generalization of our standard PCF estimator to allow for the case that $\boldsymbol{r}_i$ and $\boldsymbol{r}_j$ correspond to different types of points (or points in different subdomains).

Upon computation, we note that the mean square error [6] over z remained less than 0.05 for $r \in (0, 2)$. We do observe some variation with z in the pair correlation function, however, it seems to be a minor effect. In particular, we note that in some of the data sets, the height of the first peak in $\hat{g}_{V_z}(r)$ decreases from the bottom to the top of W. Although it would be ideal to compute a nonstationary estimate for $g(\boldsymbol{r}_1, \boldsymbol{r}_2)$, in practice, we found our calculations for such an estimator to be noisy, and unreliable for small values of $|\boldsymbol{r}_1 - \boldsymbol{r}_2|$ given the amount of data available. Thus, we have approximated the variable pair correlation function with a SOIRS form of $\hat{g}(r)$ for use in our computations.

As a final note, alternative approaches for estimating $g(r)$ by maximum likelihood estimation and Takacs-Fiksel estimation have been explored in several papers [7, 8, 71, 81]. With maximum likelihood estimation, the pair potential is assumed to be a member of a predetermined class of functions that differ through some set of parameters that can be optimized to fit the data. We did not proceed with this approach due to the unusual structure of the pair correlation function. With Takacs-Fiksel estimation, $g(r)$ is assumed to be a piecewise function (i.e. piecewise polynomial). The weights of the coefficients in the piecewise approximant are found by maximizing a nonlinear system of equations. Possible issues with such an approach here are the computational cost to accurately resolve $g(r)$ over a range of r and the attainment of a continuous (not just piecewise continuous) result.

3.3.3 Nearest Neighbor Distributions

Although not used in the PIM, the nearest neighbor distributions are useful for justifying assumptions made in the development of the PIM. As discussed in Section 3.2.2, the PIM is parametrized by first and second order statistics of the data (through $\rho^{(1)}(z)$ and $g(r)$). Thus, we would expect that the PCF and number density of the PIM will closely match those of the experimental data. However, it remains to be justified that such statistics provide a useful characterization of the data; it could be the case that higher order statistics or clustering phenomena are

[6] the mean square error is defined as $E = \int_{r_0}^{r_1} (\hat{g}_{V_z}(r) - \hat{g}(r))^2 dr$. It is approximated by a trapezoid rule quadrature.

crucial.

It is in general difficult to explicitly demonstrate that many body interactions are negligible since the estimation of higher order interaction functions, plagued by the curse of dimensionality, requires vast quantities of data. However, the validity of a pairwise interaction assumption may be evidenced by comparisons of lower order summary statistics that depend on many body interactions. In this capacity, the k-nearest neighbor distributions are useful.

The kth-nearest neighbor distribution is defined as the probability density function of distances between a point $\boldsymbol{x} \in \boldsymbol{X}_W$ and the kth closest point $\boldsymbol{y} \in \boldsymbol{X}_W \backslash \boldsymbol{x}$. Following [96, 97], the **nearest neighbor density** (NND), is defined as the conditional probability[7] probability over $\boldsymbol{X} \sim \Phi$, given $\boldsymbol{x} \in \boldsymbol{X}$,

$$H_k(r) = \lim_{\Delta r \to 0} \frac{\mathbb{P}\left[\exists \{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_k\} \in \boldsymbol{X} \text{ such that } r \leq \max_{[1,\ldots k]} |\boldsymbol{x} - \boldsymbol{y}_k| \leq r + \Delta r | \boldsymbol{x} \in \boldsymbol{X}\right]}{\Delta r}$$

Along with the intensity and pair correlation function, we use the NNDs as a means of comparing experimental data to realizations of point processes generated through simulation. As shown in [96, 97], $H_k(r)$ depends on the product density, $\rho^{(k+1)}(\boldsymbol{r}_1, \ldots, \boldsymbol{r}_{k+1})$, and not lower order densities. Although equivalence in NNDs does not guarantee equivalence of two point processes in probability, if two point processes are equal in probability, they must have the same nearest neighbor distributions. In practice, testing for equivalence in probability of a general point process is computationally infeasible, and would require an inordinate amount of data, however the nearest neighbor distributions are a useful summary statistic due to their low dimensionality, and the ease with which they can be estimated.

Two disadvantages of nearest neighbor distributions as a means of comparing point processes are their lack of directional information, and that, as defined here, they are spatially homogeneous. Thus, the NNDs we compute are best understood as homogenized variants of a more general nearest neighbor distribution that may depend on location. Since only four data sets are available, the

[7] For the point processes we consider, $\forall \boldsymbol{x} \in W$, $\mathbb{P}[\boldsymbol{x} \in \boldsymbol{X}_W] = 0$ since $\boldsymbol{X}_W$ is finite and W, being a continuum, is uncountable. The nearest neighbor distribution is defined rigorously through the use of Palm distribution theory [91, 20]. Roughly speaking, $\mathbb{P}\left[\cdot | \boldsymbol{x} \in \boldsymbol{X}_W\right]$ is the Palm distribution of Φ with respect to the constraint $\boldsymbol{x} \in \boldsymbol{X}_W$.

computation of a spatially variable NND would be a formidable difficulty, and subject to greater variance than the homogenized NND.

To estimate the nearest neighbor distributions, we use **minus sampling** [91]. Minus sampling is the technique of constructing estimators that "leave out" points near the edges of the domain to mitigate edge effects. Minus sampling estimators are less efficient than other estimators because they do not utilize all available data. However, for statistics that are strongly influenced by edge effects, the benefit in reducing edge effects can be well worth the inefficient use of data. Using the symbol, $\ominus$, to denote Minkowski subtraction [91, §1], the nearest neighbor distribution is estimated as,

$$\hat{H}_P(r) = \frac{1}{\Phi\left(W \ominus \mathcal{B}_{r_c}(0)\right)} \sum_{\boldsymbol{x} \in \boldsymbol{X}_W \ominus \mathcal{B}_{r_c}(0)} k_b \left(r - \left| \boldsymbol{x} - \underset{\boldsymbol{y} \in \boldsymbol{X}_W \setminus \boldsymbol{x}}{\arg\min} |\boldsymbol{y} - \boldsymbol{x}| \right| \right).$$

Minus sampling is suitable in our application since the box-shaped geometry of the domain dictates that most of the data is located away from the edges. Thus, the loss of data due to minus sampling is not severe. Even though there is a variation in number density near the top of the domain, we found that computations of the NNDs that left out this portion of the data did not have a pronounced effect on the resulting estimate.

Because $H_k(r)$ is a one dimensional probability density, kernel density estimation methods of choosing b based on the cardinality of $\boldsymbol{X}_W$ are used [75, 83]. In particular, we use the **ksdensity** function in Matlab to compute $H_k(r)$ once r_c is known. To specify r_c, we first compute a preliminary estimate, $\hat{H}_{P,0}(r)$, without minus sampling (e.g. assuming $r_c = 0$), and then take r_c equal to the supremum,

$$r_c = \underset{r}{\arg\sup} \left\{ \hat{H}_{P,0}(r) > 0 \right\}.$$

This choice ensures that, with high probability, the nearest neighbor of each point of $\Phi_W \ominus \mathcal{B}_{r_c}(0)$ is contained in Φ_W. Specifically, the expected error is the probability $\Pr[\min_{\boldsymbol{r}_j \in \Phi} |\boldsymbol{r}_i - \boldsymbol{r}_j| > r_c,\ \boldsymbol{r}_i \in \Phi \ominus \mathcal{B}_{r_c}(0)]$. Since each data set contains $\sim$ 4000 bacteria locations, this probability should be on the order of 10^{-3}. A second bias term arises from the use of non-parametric density estimators that have finite support for nonzero b. We expect this source of bias to be small since

the asymptotic bias of kernel smoothing is typically $\mathcal{O}(n^{-4/5})$ [75] where n is size of the data set ($\sim$4000 for each biofilm data set). For $k > 1$, analogous estimators are used with r_c replaced by $r_c^{(k)} = \arg\sup_r \left\{ \hat{H}_{P,0}^{(k)}(r) > 0 \right\}$. For larger values of k, the minus sampling technique is expected to become inaccurate as the subset of W from which points may be chosen shrinks. However, we found that $r_c^{(k)}$ does not rapidly increase with k, and at least for $k \leq 20$, the amount of data that must be disregarded due to minus sampling is small compared to the amount of data in each data set. In Figure 3.6

we show the nearest neighbor functions computed for the experimental data (from Data Set #1) and a realization of the PIM model of Section (3.2).

3.4 Numerical Solution of Integral Equations

After obtaining estimates for $h(|\boldsymbol{r}_1 - \boldsymbol{r}_2|)$ and $\rho(z)$ using the methods of Section 3.3, Equations (3.11) and (3.10) must be solved numerically. In the homogeneous case, the integral term in the OZ equation becomes a convolution that can be efficiently and accurately handled by Fourier transform methods. However, due to the inhomogeneous number density, the integral over z is not a convolution in this present application. In fact, with a variable number density, the OZ equation implies that the pair correlation and direct correlation functions cannot both simultaneously be translation invariant. A proof of this fact is straightforward and shown in the Appendix. Since the standard Fourier transform methods for convolutional integrals will not apply to our application, we use a two dimensional Hankel transform in the radial direction. To quantify variation in z and z' variation in the DCF, we solve a Fredholm integral equation of the 2nd kind.

As discussed in the Section 3.2.3, to simplify Equation (3.8), we assume that the xy dependence of the pair correlation and direct correlation functions is homogeneous with regard to within-plane translations and is radially symmetric. This assumption makes the xy integration in Equation (3.11) a two dimensional radially symmetric convolution. The radially symmetric convolution of two functions can be found through use of the two dimensional Hankel transform, denoted

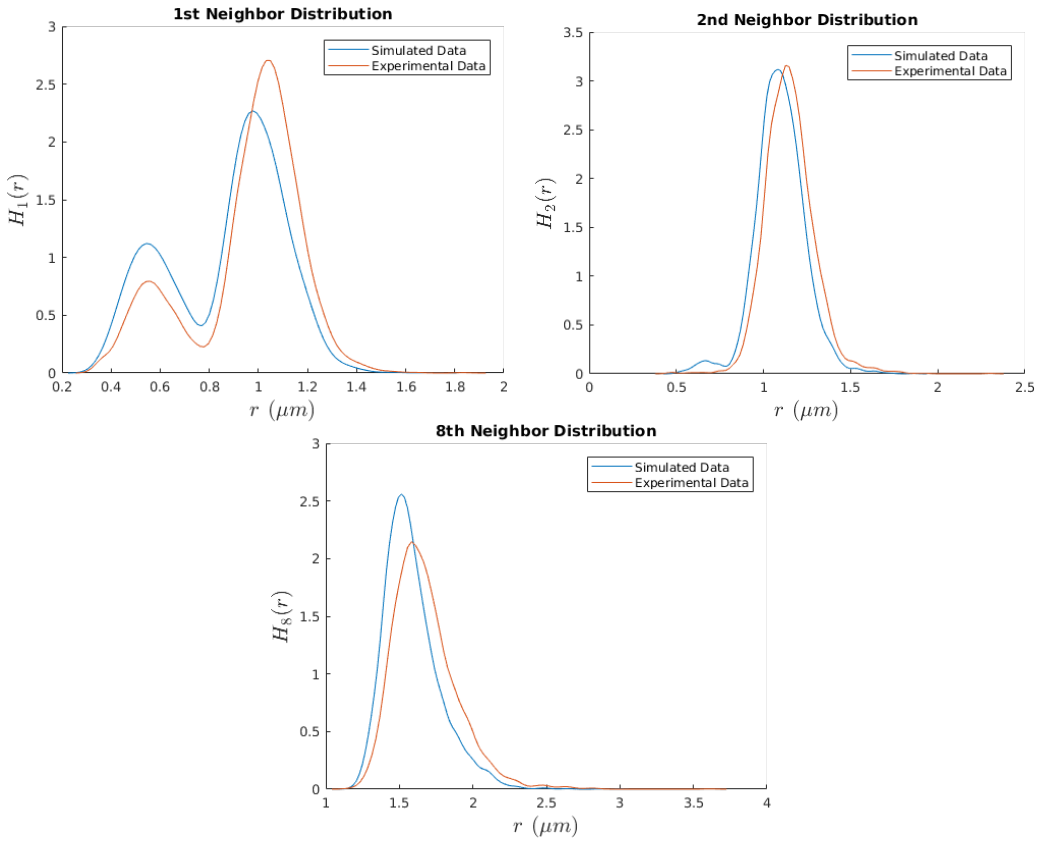


Figure 3.6: Comparison of nearest neighbor density functions between a simulated biofilm, and the experimental data. The bimodal behavior in the 1st neighbor distribution is not completely captured by the model, however, higher NNDFs (2 and 8 depicted here) are very accurately matched by the model.

$\mathcal{H}[\cdot]$, defined as the involutory transform

$$F(k) = \mathcal{H}[f(r)] = 2\pi \int_0^\infty J_0(2\pi kr) f(r) r dr$$

where $J_0(kr)$ is the 0th order Bessel J-function and $k \in [0, \infty)$. The two dimensional Hankel transform can be applied to the OZ equation to obtain

$$\mathcal{H}[h](k, z_1, z_2) = \mathcal{H}[c](k, z_1, z_2) + \int \rho(z_3)\mathcal{H}[h](k, z_1, z_2)\mathcal{H}[c](k, z_1, z_3) dz_3.$$

In the discrete analog, we discretize z, r, and k by setting $z_m = m\Delta z$, $m = 1, \ldots, M$, $r_\ell = \ell \Delta r$, $\ell \in 1, \ldots, K$ and $k_\ell = j_\ell^{(0)} R / j_{K+1}^{(0)}$, where $j_\ell^{(0)}$ is the ℓth root of the zeroth order Bessel J function, and R is the maximum value at which the estimator, $\hat{h}$ from Equation (3.14) is computed. We set $R = 3\mu m$ since beyond this distance, the radially symmetric pair correlation function showed little variation. We found that setting the discretization parameters, $M = 100$ and $K = 256$ yielded a sufficiently fine discretization. A discrete analogue of the Hankel transform is numerically computed in Matlab using the algorithm devised in [50] for each pair of z_m and z_n with $m, n \in [1, M]$.

Approximating the z-integral with the trapezoid method, the discrete approximation, C_{k_ℓ, z_m, z_n} of $\mathcal{H}[c](k_\ell, z_m, z_n)$ for each fixed value of k_ℓ and z_m, is found by solving

$$H_{k_\ell, z_m, z_n} = C_{k_\ell, z_m, z_n} + \sum_{n'=1}^{M} \hat{\rho}(n'\Delta z) H_{k_\ell, z_n, z_{n'}} C_{k_\ell, z_m, z_{n'}} w_{n'} \Delta z. \tag{3.18}$$

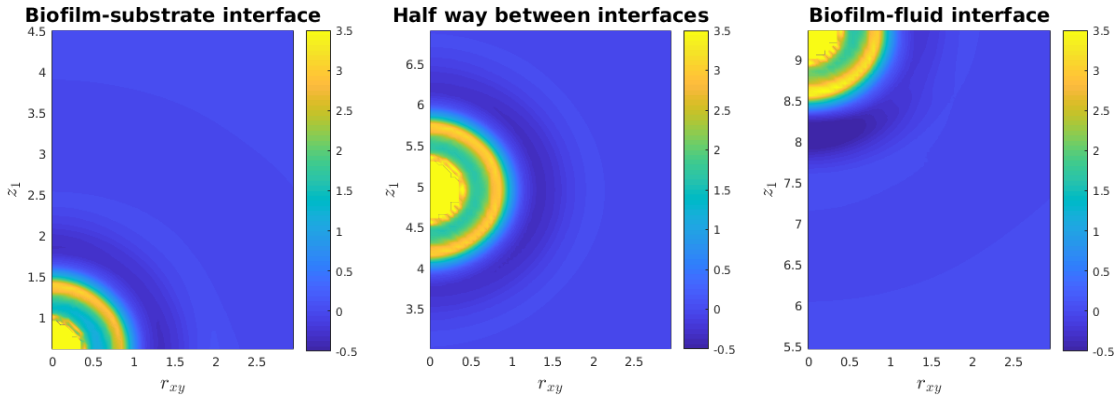
The symbol $w_{n'}$ are the trapezoid rule quadrature weights (i.e. $w_n = 1$ for $n = 2, \ldots, M-1$ and $w_1 = w_M = 1/2$). Defining $\boldsymbol{\mathcal{I}}$ as the $M \times M$ identity matrix, and the matrix $\boldsymbol{\mathcal{M}}$ by

$$[\mathcal{M}_k]_{i,j} \equiv \hat{\rho}(i\Delta z) H_{k, z_i, z_j} \Delta z, \quad i, j \in [1 : M]$$

Equation (3.18) can be written as a set of matrix equations,

$$\boldsymbol{H}_{k_\ell, z_m, z_n} = (\boldsymbol{\mathcal{I}} + \boldsymbol{\mathcal{M}}_{k_\ell}) \boldsymbol{C}_{k_\ell, z_m, z_n} \ \ \forall k_\ell, \forall z_m. \tag{3.19}$$

These matrix equations were solved using LU-decomposition implemented in Matlab. In principle, the matrix $\boldsymbol{\mathcal{I}} + \boldsymbol{\mathcal{M}}$ could be ill-conditioned. However, in the simulations we conducted, the condition number of $\boldsymbol{\mathcal{I}} + \boldsymbol{M}$ ranges only from 1 to 100. We also note that the trapezoid rule approximation



(a) (b) (c)

Figure 3.7: The potential, $v(r_{xy}, z_1, z_2)$ as it is computed from the hypernetted-chain equation. The three subfigures show $v(r_{xy}, z_1, z_2)$ forthree different values of z_2. Although the pair potential at all values is similar, they are not equivalent; the peaks and valleys in the pair correlation in (a) are slightly more pronounced than in (b) and (c). The differences in the pair potential at different heights influences the height-dependent number density trends we observed in the biofilm data.

is second order accurate, although this does not imply second order accuracy of the solution since it does not account for errors in the estimation of h by $\hat{h}$. [8]

Finally, after solving (3.19) for each value of ℓ and m, the Hankel transform can be applied to obtain c_{r_ℓ, z_m, z_n}, the discrete approximation of $c(r_\ell, z_m, z_n)$. If there are K radial nodes, and M vertical nodes, KM equations must be solved, and $2M^2$ transforms and inverse transforms must be computed. Although this leads to poor scaling, since we only need to compute the direct correlation function once, and can then store its value, the cost is not prohibitive, and, for the values of K and M we use, can be found in under a minute. In fact, the computation of $\hat{g}(r)$ is, by a substantial amount, the most time consuming step, followed by the discrete Hankel transformations of h and c for each z_m, z_n pair. The resulting pair potential for data set #3 is shown in Figure 3.7.

After computing c, a discrete approximation of the singlet energy now can be obtained. Discretizing Equation (3.5) with the trapezoid rule, and using the estimate, c_{r_ℓ, z_m, z_n}, we arrive at a forward Euler approximation of $\phi(z)$,

$$\frac{\left(\phi_{z_n} + \log \hat{\rho}(z_n)\right) - \left(\phi_{z_{n-1}} + \log \hat{\rho}(z_{n-1})\right)}{\Delta z} = 2\pi \left(\sum_{m=1}^{M} \left(\sum_{\ell=1}^{K} c_{r_\ell, z_m, z_n} \Delta r_\ell \right) \hat{\rho}_z(m\Delta z) w_m \Delta z \right)$$

[8] The matrix norm used here is the 2-norm. This norm is also used in the computation of the condition number.

with trapezoid rule weights, w_m defined as, $w_m = 1$ for $2 \leq m \leq M-1$ and $w_m = 0.5$ for $m = 1$ and $m = M$. It turns out that the initial value, $\phi(z_0)$ is arbitrary, so is set to 0.

Given $\hat{\rho}'(z)$, the discretization should be second order accurate in Δz and Δr_ℓ. As with the estimation of $c(|\boldsymbol{r}_1 - \boldsymbol{r}_2|_{xy}, z_1, z_2)$, a rigorous error derivation must take into account error terms due to approximations used in estimating $\rho(z)$ by $\hat{\rho}(z)$ and $c(|\boldsymbol{r}_1 - \boldsymbol{r}_2|_{xy}, z_1, z_2)$ by c_{r_ℓ, z_m, z_n}. For the approximation of $\phi(z)$, there is no advantage to using Hankel transforms for the radial integral since the real-space computation is explicit in $\phi(z)$. This differs from the pair energy computation where the real space integral equation is a three dimensional integral equation but, the transformed equation is a set of decoupled one dimensional integral equations. In that case, transforming in the r coordinate is highly beneficial. We also note that other numerical quadrature methods could be applied to compute the values of the integrals, however, we expect that the dominant error source is from the approximation of the pair correlation function and spatially variable number density. Thus, we do not expect significant differences resulting from different quadrature choices.

3.5 A Metropolis-Hastings Algorithm for Generating Biofilm Realizations

The pair energy function and singlet potential form the basis for a Metropolis Hastings (MH) algorithm to generate "artificial" biofilms. The practical aspect of MH (and more generally Markov Chain Monte Carlo) algorithms that makes them so useful is that they rely only on unnormalized probability densities. Given two realizations of the PIM, denoted $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$, each containing n points in W, and an unnormalized probability density, $q^{(n)}(\boldsymbol{X}_1) \propto f^{(n)}(\boldsymbol{X}_1)$, the ratio

$$\frac{q^{(n)}(\boldsymbol{X}_1)}{q^{(n)}(\boldsymbol{X}_2)} = \frac{f^{(n)}(\boldsymbol{X}_1)}{f^{(n)}(\boldsymbol{X}_2)} = \frac{\exp\left[-\sum_i \phi(z_i^{(1)}) - \sum_{i<j} v(|\boldsymbol{r}_i^{(1)} - \boldsymbol{r}_j^{(1)}|, z_i, z_j)\right]}{\exp\left[-\sum_i \phi(z_i^{(2)}) - \sum_{i<j} v(|\boldsymbol{r}_i^{(2)} - \boldsymbol{r}_j^{(2)}|, z_i, z_j)\right]} \tag{3.20}$$

can be computed efficiently in comparison to evaluating or approximating the probability density, $f^{(n)}(\boldsymbol{X})$. If $\boldsymbol{X}_2$ differs from $\boldsymbol{X}_1$ by the location of only one point such that $\boldsymbol{X}_1 = \boldsymbol{X}_2 \backslash \{\boldsymbol{r}\} \cup \boldsymbol{r}'$ then the quantity in Equation (3.20) simplifies and can be computed in $\mathcal{O}(n)$ operations as

$$\frac{f^{(n)}(\boldsymbol{X}_1)}{f^{(n)}(\boldsymbol{X}_2)} = \exp\left[-(\phi(z') - \phi(z)) - \sum_{i=1}^{n} v(|\boldsymbol{r}_i - \boldsymbol{r}|_{xy}, z_i, z) + \sum_{i=1}^{n} v(|\boldsymbol{r}_i - \boldsymbol{r}'|_{xy}, z_i, z')\right]. \tag{3.21}$$

Equation (3.21) can be used to generate realizations of the PIM conditional on there being n points in the domain. Such realizations would belong to the so-called canonical ensemble. However, in practice, it is often more favorable to generate realizations that are not conditional on n. Such a method samples from the grand-canonical ensemble. A Metropolis-Hastings algorithm with both move, and birth-death steps [71, 46] is a practical method of obtaining such realizations. Given $q^{(n)}$ (or $f^{(n)}$), the MH algorithm proceeds with the steps detailed in Algorithm 1.

Because $\phi(z)$ and $v(|\boldsymbol{r}_1 - \boldsymbol{r}_2|_{xy}, z_1, z_2)$ are computed using the methods of Section 3.4 on a grid and are not known analytically, interpolation is used to approximate their values at the points, $\boldsymbol{r}_i \in \Phi$, which are arbitrary points in space. Linear interpolation was the chosen interpolation method because it is computationally inexpensive and accurate in our case since the grid of points on which the energy is computed on in the previous section has a small spatial step. Potential improvements could apply the ideas in [22] obtain Markov chains such that have reduced dependence on the choice of interpolation method. For boundary conditions, we imposed periodicity in the two horizontal directions, and an impenetrable boundary at the top and bottom of the domain, by setting $\phi(z) = \infty$ for points when a proposed move sets a point outside of W.

The birth-step we employ is somewhat unconventional, but still well-defined in the context of Metropolis-Hastings algorithms. We found that the mean number of points in realizations of the algorithm depend strongly on p. Setting $p = 0.82$ and generating points in the region greater than $0.35\mu m$ away from points in $\boldsymbol{X}$ (due to the hard-sphere property of $v(\cdot, \cdot)$) lead to point processes with mean approximately equal to the number of points in the experimental data set. The factor $(n+1)/\int \rho^{(1)}(\boldsymbol{r})d\boldsymbol{r}$ is the ratio of probabilities of there being $n+1$ to n points in a Poisson point process. For PIM processes, this factor is intractable, thus we use the easily computable Poisson probability ratio in its place. Note that although we simulate in the grand canonical ensemble, we still assume a fixed volume based on the experimental data.

In step 17 of Algorithm 1, a halting criterion must be used to determine whether to continue finding $\boldsymbol{X}_{k+1}$ or to exit. Such criteria are usually based on easily computable quantities such as the total unnormalized density, $q^{(n)}(\boldsymbol{X})$, or, a characteristic such as the empirical pair correlation

Algorithm 1 MH algorithm for generating realizations of a point process with spatial characteristics similar to those of the experimental data in a domain W, of finite size. As a convergence criterion, we compute the norm of the difference in the PCF of $\boldsymbol{X}_k$ and the experimental data.

Generate n_0 randomly placed points in a window, W. Denote this set of points $\boldsymbol{X}_0$.
Generate a random number $q \sim \text{Uniform}([0,1])$
if $q < 0.5$ **then**
  Displace one point, $\boldsymbol{x}_i \in \boldsymbol{X}_k$, chosen at random, by a uniformly random displacement, $\delta\boldsymbol{x}$ and set

$$\tilde{\boldsymbol{X}} = \{(\boldsymbol{X}_k \backslash \{\boldsymbol{x}_i\}) \cup \{\boldsymbol{x}_i + \delta\boldsymbol{x}\}\}.$$

  Compute

$$\alpha_m = \min\left(\frac{f^{(n)}(\tilde{\boldsymbol{X}})}{f^{(n)}(\boldsymbol{X}_k)}, 1\right)$$

  according to Equation (3.21) using interpolation as needed.
  if $\tilde{\boldsymbol{X}} \subset W$ **then**
    With probability α_m, set $\boldsymbol{X}_{k+1} = \tilde{\boldsymbol{X}}$.
  end if
else
  Generate a random number $p \sim \text{Uniform}([0,1])$
  if $p < 0.8$ **then**
    Generate a point $\boldsymbol{\xi} \sim \text{Uniform}(W_{\boldsymbol{X}_k})$ in region $W_{\boldsymbol{X}_k} = \{\boldsymbol{\xi} | \min|\boldsymbol{\xi} - \boldsymbol{X}_k| > 0.35\}$
    Compute

$$\alpha_b = \min\left(\frac{f^{(n+1)}(\boldsymbol{X}_k \cup \boldsymbol{\xi})}{f^{(n)}(\boldsymbol{X}_k)} \frac{\int \rho^{(1)}(\boldsymbol{r}) d\boldsymbol{r}}{n+1}, 1\right)$$

    With probability α_b, $\boldsymbol{X}_{k+1} = \boldsymbol{X}_k \cup \boldsymbol{\xi}$
  else
    Choose a number $m \sim \text{Uniform}([1:n])$
    Compute

$$\alpha_d = \min\left(\frac{f^{(n-1)}(\boldsymbol{X}_k \backslash \{\boldsymbol{x}_m\})}{f^{(n)}(\boldsymbol{X}_k)} \frac{n}{\int \rho^{(1)}(\boldsymbol{r}) d\boldsymbol{r}}, 1\right)$$

    With probability α_d, set $\boldsymbol{X}_{k+1} = \boldsymbol{X}_k \backslash \{\boldsymbol{x}_m\}$.
  end if
end if
Set $k \leftarrow k+1$ and repeat steps 2-4 unless a **convergence criterion** has been reached. If a convergence criterion has been reached, output $\boldsymbol{X}_{k+1}$ and exit.

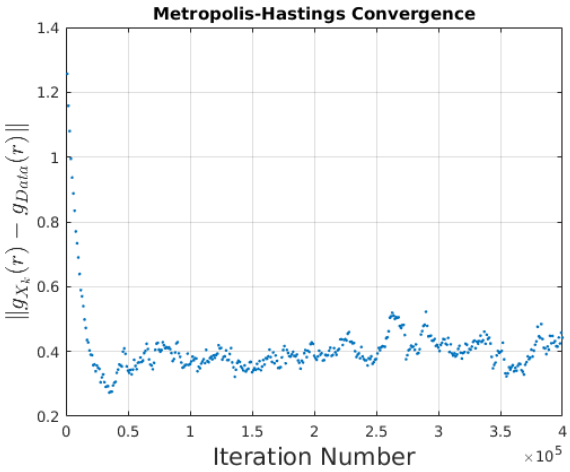


Figure 3.8: Convergence of the Metropolis Hastings Algorithm described in Algorithm 1.The convergence diagnostic used was the norm of the difference in the pair correlation computed from $\boldsymbol{X}_k$ and the data set. This diagnostic was computed every 1000th iteration. Even after the burn-in phase, there is still variability in the energy between iterations, but there is no downward trend diagnostic

function [71, §8]. In our case, we observed that, with an update step drawn from a uniform distribution, $\delta\boldsymbol{X} \propto U([-0.5, 0.5]^3)$, the total energy leveled off after $\mathcal{O}(10^5)$ iterations, and the average number of acceptances was approximately $1/2$ the number of attempts. The computed value of $g(r)$ for each sample also stabilized by this point in each case. Thus, as a halting criterion, we compute the total energy of the sample every 10000 iterations and stop after it has leveled off. In practice, this lead to convergence within 500000 iterations in each simulation that we conducted.

The convergence in total energy of one particular realization is shown in Figure 3.8.

There is a clearly distinguishable initial period of decreasing energy followed by a valley where the energy remains within a small range.

3.6 Comparison of Material Properties

Although experimental results on biofilms often range drastically between different studies, it is generally agreed that over short time scales and moderate mechanical stresses, biofilms behave as viscoelastic materials [77]. One way to characterize biofilm rheology is through measurement of the dynamic moduli discussed below.

Under mild restrictions [19], the stress, σ in a linear viscoelastic material undergoing small-amplitude oscillatory shear strain deformation at frequency, ω and amplitude, ϵ_0 can be written as

$$\sigma(\omega, t) = \epsilon_0 G^*(\omega) e^{i\omega t}.$$

The symbol G^* is known as the complex modulus. The storage and loss moduli (collectively known as the dynamic moduli), $G'(\omega)$ and $G''(\omega)$ are the real and imaginary parts:

$$G^*(\omega) = G'(\omega) + iG''(\omega).$$

Writing $\sigma(\omega) = \sigma_0(\omega) e^{i\omega t - i\delta(\omega)}$, where the $\delta(\omega)$ is the phase lag between the stress and strain, we arrive at,

$$G'(\omega) = \frac{\sigma_0(\omega)}{\epsilon_0} \cos(\delta(\omega)) \quad G''(\omega) = \frac{\sigma_0(\omega)}{\epsilon_0} \sin\left(\delta(\omega)\right).$$

Given $\boldsymbol{X} \sim PIM(\phi, v)$, the stress and strain in a biofilm undergoing oscillatory shear deformation [77] can be measured as functions of time using the hrIBM discussed in [52, 89]. This method yields a finite difference approximation of the velocity field, and an approximation of the forces exerted by a biofilm. The strain can be estimated by considering the motion of tracer particles in the fluid, and the stress is measured by averaging the forces over the top part of the biofilm, and then dividing by the area of the top of the biofilm. The phase lag is approximated by observing when the maxima in the stress and strain occur in time,

$$\delta(\omega) = \arg \max_{t \in [0, 2\pi/\omega]} \sigma(\omega, t) - \arg \max_{t \in [0, 2\pi/\omega]} \epsilon(\omega, t).$$

In Figure 3.4, we depict the dynamic moduli, for four different types of point processes and the experimental data. Since we have four experimental data sets available to us, we repeat the comparison for each data set. The "random" data sets are realizations of a Poisson process of constant number density, equal to the average number density of the experimental data sets. The "model" data is based on realizations of the model in Section 3.4, the grid-aligned data use a regular grid of points in simulation, and the grid-aligned plus perturbation uses the same grid-aligned data

with an added perturbation drawn from a normal distribution of mean 0 and variance $0.2\mu m$ added to each point.

We see that the Poisson point process and PIM models yield dynamic moduli in closer agreement to the experimental data than a grid aligned approximation, although some amount of improvement is seen when the grid aligned data is randomly perturbed. In each case, the comparisons are made between point processes with approximately the same number of points as the experimental data. (In the uniformly random, and model based simulations, the number of points is exactly the same as the experimental data)

To estimate the variability in the dynamic moduli for each of the point processes we have mentioned, the dynamic moduli are computed for five different realizations of each point process. Although it would be beneficial to observe a larger number of trials over a range of frequencies, the computational cost of such an endeavor is currently prohibitive. Each hrIBM simulation on a computer with an Intel i7-7600U CPU with clock rate 2.8GHz and 16GB of RAM takes 6 to 8 hours.

3.7 Discussion

In [101, 2, 52, 89], systems of points immersed in a viscous fluid and connected by a network of viscoelastic fibers are used as models of complex microbiological fluids. The points in each application are distributed according to different procedures. By placing such point distributions in the context of point process theory, we strive to elucidate the effect that the positions of points have on the rheology of the resulting system. In particular, in this work we show that for biofilms, i) empirical data yields biofilms that are exhibit larger dynamic moduli than biofilms artificially generated through simple point process models, ii) the empirically informed PIM leads to realizations with statistical characteristics that strongly match empirical data, and dynamic moduli that closely match data, and iii) the Poisson point process model leads to close agreement with empirical dynamic moduli, but fails to match statistical characteristics of the empirical data.

In contrast to our results showing variation in the dynamic moduli depending on the point

process model, we note that in a intriguing article [101], the network topology of an immersed boundary method model was not observed to have strong effects on bulk viscoelastic properties of the system. We believe that this difference with our findings is due to the magnitude of the number density of the point process being simulated and how it compares to the average connectivity of each point. In our simulations, a bacterium is, on average, connected to approximately 9 other bacteria through viscoelastic linkages, whereas [101] used two regular networks of points with 14 and 27 linkages per point respectively. We conjecture that for high connectivity networks, the rheological properties of the material are less sensitive to the spatial positioning of points, whereas lower connectivity networks exhibit more sensitivity. In a future work we plan to further study this idea.

Another future avenue of research would be to integrate a model for the network topology (i.e. given bacteria at $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$, what is the chance that there is a viscoelastic linkage connecting them?) into the point process model. The bacteria in a biofilm are typically interconnected through a complicated network of viscoelastic linkages whose properties effect the bulk material properties of the biofilm. Currently, to model this network, we use a deterministic rule specifying that if a pair of bacteria is separated by a distance less than a cutoff radius, r_c they are linked, and those separated by more than r_c are not linked. This network topology was originally proposed in [2], and was again used in [52, 89]. Although simple, it yields realistic results when combined with appropriate viscoelastic models of the linkages [89]. The reason that we do not introduce a statistical model for the network topology is that current state of the art experiments allow for determination of bacteria locations, but to the extent of our knowledge, there is no data on the connectivity of bacteria in a biofilm. We are very interested in analyzing connectivity statistics should such data become available.

The results in Section 3.6 show how different models of the positions of bacteria in a biofilm can influence the mechanical properties of the simulated biofilm. It was found that the Poisson process model, and grid-aligned model do not yield results that are consistent with statistical characteristics (e.g. Figure 3.2a and 3.2b) of the experimental data. However, we were surprised to find that

despite this difference from the experimental data, the Poisson model yields biofilms with similar dynamic moduli, performing as well as the PIM in terms of recreating the mechanical properties of a biofilm. In contrast, the grid-aligned model, as shown in Figure 3.4 is not a close match. These findings indicate that nonuniformity can lead to stronger biofilms in comparison to the grid-aligned case. We also show that the model introduced in Section 3.4, with first and second order characteristics informed by experimental data, yields agreement in the material properties and agreement in statistical properties of the data.

Although the Poisson point process model exhibits similar dynamic moduli as experimental data, from a physical interpretation, it is clear that the bacteria positions cannot conform to a Poisson process. The lack of correlations between point locations, a defining feature of Poisson processes, implies that for any radius, r, there is a nonzero probability that within a given realization, two points of the process are separated by a distance less than r. In contrast, bacteria have finite radii and are impenetrable, thus the centers of mass of two bacteria cannot be separated by less than some hard-sphere radius. The hard-sphere property of the experimental data is readily apparent upon computation of the nearest neighbor distribution and the pair correlation function. However, we note that the presence of an initial low magnitude peak in the PCF is not typical of hard-sphere processes, and requires a more complicated model such as the empirically informed PIM model we have proposed.

Although not the focus of our work, an interesting result that arose in the course of this study was the observation of variations in the number density of bacteria near the fluid-biofilm interface. It would be interesting to discover whether this phenomenon is a passive effect due to fluid motion at the biofilm-fluid surface or other experimental conditions, or whether it is a strategy employed by biofilm forming bacteria to improve the biological fitness of a given biofilm.

3.8 Acknowledgments

This work was supported in part by the National Science Foundation grants PHY-0940991 and DMS-1225878 to DMB, and by the Department of Energy through the Computational Sci-

ence Graduate Fellowship program, DE-FG02-97ER25308, to JAS. The authors would also like to thank Mike Solomon (University of Michigan) and John Younger (Akadeum Life Sciences) for insightful discussions and suggestions concerning this work. Furthermore, we would like to thank the anonymous reviewers for many helpful comments and suggestions.

Chapter 4

A Posteriori Error Analysis of Fluid Stucture Interactions: Time Dependent Error

4.1 Introduction

The method of regularized Stokeslets (MRS) is a commonly used method for studying viscous flow phenomena. The method is grid-free and based on the numerical discretization of integrodifferential equations involving singularity solutions of the Stokes equation. In this paper we develop and apply **a posteriori** error estimation techniques described in [21] to the ordinary differential equations that arise from the spatial discretization of the singular integral operators associated with the MRS. Such systems frequently arise in the study of quasi-steady state phenomena in biofluids [6, 23, 24, 25, 102, 101]. However, little work has been applied towards understanding the error in specific simulations that use the MRS (although see [25] for some examples of numerical convergence). The main contributions of this paper are the derivation of adjoints for the quasi-steady state MRS and computational results based on a decomposition of the numerical error into terms specifically tied each step in the discretization. Furthermore, our approach towards computing the regularization error during MRS simulations by treating the regularized equations as a perturbation to a set of non-regularized equations is novel.

As explained in [21], we see through **a posteriori** analysis that numerical error is composed of quadrature error, extrapolation (or explicit) error, residual error, and regularization error. The residual and regularization errors are due to the use of a finite dimensional function space where the solution is approximated and the use of regularization to remove singularities from the governing

equations. On the other hand, the quadrature and extrapolation errors are due to the choice of numerical method. Quadrature arises from the numerical approximation of integrals over each time interval. For explicit methods, extrapolations of the numerical solution across each time interval introduce error. Since the standard time-stepping methods for the MRS are explicit Runge-Kutta methods, we use the **nodally equivalent finite element method** formalism introduced in [21] to derive polynomial solutions correspond to the Runge-Kutta solutions over each time step in the solution.

Section 4.2 provides a brief overview of the method of regularized Stokeslets. For readers familiar with the MRS, it may be skimmed with the exception of Table 4.1 which contains essential notation.Similarly, readers familiar with **a posteriori** error estimation techniques applied to systems of ODEs may skim Section 4.3.

4.2 Review of the Method of Regularized Stokeslets

In this section we briefly overview the the singular integral formulation of Stokes flow. The notation used in the rest of this section and the proceeding sections is listed in Table 4.1.

4.2.1 Stokes Equations and the Method of Regularized Stokeslets

Stokes equations are an approximation of the Navier-Stokes equations valid for viscous fluids at small length scales and low velocities. Being an approximation method for solving Stokes equations, the Method of Regularized Stokeslets, is often applied to problems in biofluids which typically satisfy Stokes flow conditions.

The partial differential equations governing Stokes flow can be written as

$$\begin{aligned} \nabla P &= \Delta \boldsymbol{u} + \boldsymbol{f} \\ \nabla \cdot \boldsymbol{u} &= 0. \end{aligned}$$

There exist fundamental velocity and pressure solutions, $U_{ij}(\boldsymbol{x} - \boldsymbol{y})$ and $P_i(\boldsymbol{x} - \boldsymbol{y})$, that solve the

Symbol	Definition
t	time
$\boldsymbol{u}$	velocity
P	pressure
$\boldsymbol{\alpha}$	Lagrangian coordinate
k	discrete Lagrangian index
r	radial distance, $r = \sqrt{x_i x_i}$.
$\zeta_\delta(r)$	radially symmetric regularization kernel
$\boldsymbol{U}(\boldsymbol{x})$	fundamental solution of Stokes equation
$\boldsymbol{U}_\delta(\boldsymbol{x})$	smooth regularization of $\boldsymbol{U}(\boldsymbol{x})$ defined by $\boldsymbol{U}_\delta(\boldsymbol{x}) = \int \zeta_\delta(\boldsymbol{y})\boldsymbol{U}(\boldsymbol{x}-\boldsymbol{x})d\boldsymbol{y}$
$\boldsymbol{x}(\boldsymbol{\alpha}, t)$	spatially continuous flow map
$\boldsymbol{f}(\boldsymbol{x}, t)$	force
$\boldsymbol{x}_k(t)$	spatially discrete, time-continuous flow map
$\boldsymbol{X}_k(t)$	spatially discrete numerical approximation of $\boldsymbol{x}_k(t)$
$\boldsymbol{e}_k(t)$	numerical error: $\boldsymbol{x}_k(t) - \boldsymbol{X}_k(t)$
$(\cdot,\cdot)$	finite dimensional inner product e.g. $(f,g) = \sum_k f_k g_k$
$\lvert\cdot\rvert$	norm over a finite dimensional vector space $\lvert x\rvert = (x,x)^{1/2}$
$\langle\cdot,\cdot\rangle$	Inner product over time defined as $\int_0^t (\boldsymbol{u},\boldsymbol{v})\,dt$
$\lVert\cdot\rVert$	Norm over time and space
$\langle\cdot,\cdot\rangle_d$	discrete inner product approximation through a quadrature rule $\langle f,g\rangle_d = \sum_q^Q w_q f(t_q) g(t_q)$ with $\{w_q, t_q\}$ specified by the quadrature rule
$\mathcal{V}^q(I)$	space of qth degree polynomials over an interval $I = [a,b]$.
$\mathcal{V}^q([0,T])$	space of piecewise qth degree polynomials on $[0,T]$.
$\mathcal{P}$	a projection operator related to Runge Kutta methods
π	projection operator from a function space $\mathcal{V}$ to $\mathcal{V}^q(I)$
A^*	adjoint, defined by $\langle A\boldsymbol{x},\boldsymbol{y}\rangle = \langle \boldsymbol{x}, A^*\boldsymbol{y}\rangle$
$\bar{A}$	trajectory-averaged operator, $\bar{A} = \int_0^1 A(s\boldsymbol{x} + (1-s)\boldsymbol{y})ds$

Table 4.1: Definitions of frequently used terms..

following equations written in index notation[1]

$$\begin{aligned} P_{k,i}(\boldsymbol{x}-\boldsymbol{y}) &= U_{ki,jj}(\boldsymbol{x}-\boldsymbol{y}) + \delta(\boldsymbol{x}-\boldsymbol{y})\hat{e}_i^{(k)} \\ U_{kj,j}(\boldsymbol{x}-\boldsymbol{y}) &= 0 \end{aligned}$$

The basis vector, $\hat{\boldsymbol{e}}^{(k)}$ with $k \in \{1,2,3\}$ is defined such that $\hat{\boldsymbol{e}}^{(1)}$ is understood to be the basis vector that points along the x-axis, $\hat{\boldsymbol{e}}^{(2)}$ along the y-axis, and $\hat{\boldsymbol{e}}^{(3)}$ along the z-axis. The symbol, $\delta(\cdot)$ represents the Dirac delta distribution. Due to the linearity of Stokes equations, the solution for an arbitrary vector, $\boldsymbol{g} \in \mathbb{R}^3$, is $u_i(\boldsymbol{r}) = U_{ij}(\boldsymbol{r})g_j$. The tensor-valued fundamental velocity solution denoted, U_{ij} or $\boldsymbol{U}$ depending on the context, can be written in two dimensions as

$$U_{ij}(\boldsymbol{r}) = \frac{1}{4\pi}\left(-\delta_{ij}\log r + \frac{r_i r_j}{r^2}\right),$$

and in three dimensions,

$$U_{ij}(\boldsymbol{r}) = \frac{1}{8\pi}\left(-\frac{1}{r}\delta_{ij} + \frac{r_i r_j}{r^3}\right)$$

where δ_{ij} the Kronecker delta.

It can be shown that a generalized solution of the form

$$\boldsymbol{u}(\boldsymbol{\alpha}) = \int \boldsymbol{U}(\boldsymbol{x}(\boldsymbol{\alpha}) - \boldsymbol{x}(\boldsymbol{\alpha}')) \cdot \boldsymbol{f}(\boldsymbol{x}(\boldsymbol{\alpha}'))d\boldsymbol{\alpha}' \tag{4.1}$$

is valid for $\boldsymbol{f}(\boldsymbol{x})$ in a wide variety of function spaces [60].

In situations where $\boldsymbol{f}(\boldsymbol{x})$ is concentrated on a lower dimensional manifold within a domain, Equation (4.1) remains valid [24]. For instance, if $\boldsymbol{f}(\boldsymbol{x})$ is concentrated on some closed surface, S, the integral equation is equivalent to the single-layer hydrodynamic potential on the surface S of density $\boldsymbol{f}(\boldsymbol{x})$ [60].

Of interest here and in many applications [102, 101, 6], is the quasi-steady state situation where the fluid velocity is found from Stokes equation, but the boundary or particle positions and forces may vary over time. This quasi-steady state assumption is applicable for slowly moving

[1] For a vector quantity, $u_{i,k} \equiv \frac{\partial u_i}{\partial x_k}$. Summation over repeated indices is implied, i.e. in $\mathbb{R}^3$, $u_{i,kk} \equiv \frac{\partial^2 u_i}{\partial x_1^2} + \frac{\partial^2 u_i}{\partial x_2^2} + \frac{\partial^2 u_i}{\partial x_3^2}$

immersed structures and small length scales. In particular, the fluid velocity at any specific instant in time must remain close to equilibrium velocity due to a boundary with the same shape and force distribution as the moving boundary frozen at that point in time. In such cases, the boundary position is typically updated through the Eulerian-Lagrangian velocity equivalence:

$$\dot{\boldsymbol{x}}(\boldsymbol{\alpha}, t) = \boldsymbol{u}(\boldsymbol{x}(\boldsymbol{\alpha}, t), t) = \int \boldsymbol{U}(\boldsymbol{x}(\boldsymbol{\alpha}, t) - \boldsymbol{x}(\boldsymbol{\alpha}', t)) \cdot \boldsymbol{f}(\boldsymbol{x}(\boldsymbol{\alpha}', t), t) d\boldsymbol{\alpha}'.$$

The numerical discretization proceeds in two steps. The singular fundamental solution is regularized by convolution with a smooth, radially symmetric function that satisfies certain moment conditions (e.g. [24, 25]), and the integral is approximated as a summation over a finite number of points[2] , $\{\boldsymbol{x}_k\}_{k=1}^N$,

$$\dot{\boldsymbol{x}}_k(t) = \sum_{j=1}^{N} \boldsymbol{U}_\epsilon(\boldsymbol{x}_k(t) - \boldsymbol{x}_j(t)) \boldsymbol{f}_j(\boldsymbol{x}_j(t), t) \Delta\alpha_j(t). \tag{4.2}$$

The time dependence of $\Delta\alpha_j(t)$ occurs when the force is concentrated on a lower dimensional manifold. In such cases, $\Delta\alpha_j(t)$ is treated as a surface area element which may vary in time as deformation occurs. For the case of a collection of particles, $\Delta\alpha_j$ is generally a constant for each particle (e.g. the surface area, or a drag-coefficient) such that $\boldsymbol{f}_j \Delta\alpha_j$ is equal to the force that the particle exerts on the fluid. Likewise, for cases where the MRS is used to model fluid flow around a surface, the regularization parameter ϵ typically should be dependent on the mean surface area element size. On the other hand, for particle simulations, ϵ typically corresponds to a physical length scale related to the size of the particles [25].

Discussions of the well-posedness of Stokes equations and numerical convergence results for the MRS can be found in [24, 25, 60, 57, 80]. However, it appears that the well-posedness of the quasi-steady state MRS for general forces has not been studied (though two recent studies on specific examples in two dimensions exist [72, 64]). On the other hand, for the spatially discrete systems considered in this article, well-posedness is guaranteed by the Picard existence and uniqueness theorem (for short times) as long as the force operator is a Lipschitz continuous function of $\boldsymbol{x}(\boldsymbol{\alpha}, t)$ and t [59, §3].

[2] For the quasi-steady state Stokes equations, the numerical discretization procedure is quite similar to that used in the vortex method literature [5, 69].

4.3 A Posteriori Error Estimation

A posteriori error estimation is a counterpart to the **a priori** error estimation techniques of classical numerical analysis. With **a priori** error estimation, bounds on the error in a numerical method generally depend on derivatives of the exact solution to an ODE or PDE and stability factors related to the numerical method. A major difficulty with **a priori** bounds is that the exact solution and its derivatives are typically unknown. This often prevents accurate measures of the error in specific simulations from being obtained. In contrast, **a posteriori** error bounds do not depend upon the analytical solution to a given problem. Instead, they typically depend on finite differences of the numerical solution and stability factors derived from the continuous ODE or PDE [35]. We provide a brief overview of some of the results from [35] regarding **a posteriori** error estimation for ODEs.

To start, we detail the finite element discretizations that will be employed throughout the rest of the paper in Section 4.3.1. In Section 4.3.2, we describe how the Fréchet derivative of a nonlinear operator is obtained. This is needed to form the linearized adjoint equations required by the **a posteriori** error estimation procedure to quantify how numerical errors propagate over time. In Section 4.3.3, we derive error representation formulas for finite element methods that give the numerical error in terms of computable quantities. In Section 4.3.4, we show how to derive such error representation formulas for Runge-Kutta methods. Finally, in Section 4.3.5, we discuss some of the error components that we must monitor in the MRS.

4.3.1 Continuous Galerkin Finite Element Discretizations

Consider the weak form of a system of ODEs of the form

$$\begin{gathered} \langle \dot{\boldsymbol{x}}(t) - \boldsymbol{F}(t, \boldsymbol{x}(t)), \boldsymbol{V}(t) \rangle = 0 \quad \forall \boldsymbol{V} \in \mathcal{V} \\ (\boldsymbol{x}(0), \boldsymbol{V}(0)) = 0 \end{gathered} \tag{4.3}$$

where $\mathcal{V}$ is a Hilbert space, and $\boldsymbol{F}(t, \boldsymbol{u})$ is assumed to be sufficiently smooth such that a solution, $\boldsymbol{x}(t)$, to Equation (4.3) exists in some Hilbert space $\mathcal{U}$. The symbol $\langle \cdot, \cdot \rangle$ is a duality pairing on $\mathcal{V}$,

and $(\cdot,\cdot)$ is an inner product on $\mathbb{R}^n$. A Galerkin finite element method is obtained by choosing finite dimensional spaces $\mathcal{V}_n \subset \mathcal{V}$ and $\mathcal{U}_n \subset \mathcal{U}$ such that there exists a function $\boldsymbol{X} \in \mathcal{U}_n$ that satisfies

$$\left\langle \dot{\boldsymbol{X}}(t) - \boldsymbol{F}(t, \boldsymbol{X}(t)), \boldsymbol{V}(t) \right\rangle = 0 \quad \forall \boldsymbol{V} \in \mathcal{V}_n$$
$$(\boldsymbol{X}(0), \boldsymbol{V}(0)) = 0$$

Functions in the space $\mathcal{V}_n$ are called **test functions**, and functions in $\mathcal{U}_n$ are called **trial functions**. Typically, for finite element methods, the domain, $[0,T]$ is partitioned into a finite number of subintervals, $[0,T] = \cup_{i=1}^{N} \mathcal{I}_i$ where $\mathcal{I}_i = [t_{i-1}, t_i)$. Refinements of this partition then involve the addition of partition points t_i into the domain. For a continuous Galerkin method of order q, we then construct $\mathcal{U}_N$ and $\mathcal{V}_N$ as spaces of piecewise continuous polynomials. In particular, we define $\mathcal{P}^q(\mathcal{I}_i; \mathbb{R}^n)$ as the set of polynomials of order less than or equal to q with domain $\mathcal{I}_i$ and range $\mathbb{R}^n$. Then we have:

$$\mathcal{U}_N = \left\{ \boldsymbol{u}(t) | \forall i \in [1,N],\ \boldsymbol{u}|_{\mathcal{I}_i} \in \mathcal{P}^q(\mathcal{I}_i; \mathbb{R}^n) \text{ and } \boldsymbol{u} \in C^0([0,T]; \mathbb{R}^n) \right\}$$

$$\mathcal{V}_N = \left\{ \boldsymbol{V}(t) | \forall i \in [1,N],\ \boldsymbol{V}|_{\mathcal{I}_i} \in \mathcal{P}^{q-1}(\mathcal{I}_i; \mathbb{R}^n) \right\}.$$

where $\boldsymbol{u}|_{\mathcal{I}}$ and $\boldsymbol{V}|_{\mathcal{I}}$ are the restrictions of $\boldsymbol{u}$ and $\boldsymbol{V}$ to an interval $\mathcal{I} \subseteq [0,T]$.

4.3.2 Fréchet Derivatives of Operators

Let $F : Y \to Z$ be an operator on $Y \subset X$ where X and Z are Banach spaces and Y is an open subset of X. This operator is Fréchet differentiable if there exists a bounded linear operator DF such that

$$F[u + \delta u] - F[u] = (DF[u])\delta u + o(\|\delta u\|).$$

For any particular choice of $u(x) \in Y$ the form of this operator can often be found explicitly by considering the limit

$$DF[u]v = \lim_{\epsilon \to 0} \frac{F[u + \epsilon v] - F[u]}{\epsilon \|v\|}$$

for some arbitrary $v \in Y$.

4.3.3 Derivation of Error Representation Formulas

Follwing [35], the starting point for both **a priori** and **a posteriori** analysis is to subtract the exact and discrete equations and linearize about the range of trajectories from $\boldsymbol{x}(t)$ to $\boldsymbol{X}(t)$. We define the error $\boldsymbol{e}(t) = \boldsymbol{x}(t) - \boldsymbol{X}(t)$ and introduce the operator,

$$B(\boldsymbol{W}, \boldsymbol{V}) \equiv \left\langle \dot{\boldsymbol{W}}(t) - \int_0^1 D\boldsymbol{F}(t, s\boldsymbol{x}(t) + (1-s)\boldsymbol{X}(t))ds \cdot \boldsymbol{W}(t), \boldsymbol{V}(t) \right\rangle$$

where $D\boldsymbol{F}(\cdot)$ is the Fréchet derivative of $\boldsymbol{F}(t, \boldsymbol{x})$ with respect to variations in $\boldsymbol{x}$. For short-hand, we denote the linearized operator, $D\boldsymbol{F}(\cdot)$ averaged over all trajectories between the continuum and numerical solutions as

$$\boldsymbol{A}(t) = \int_0^1 D\boldsymbol{F}(t, s\boldsymbol{x}(t) + (1-s)\boldsymbol{X}(t))ds,$$

and integrating by parts, we obtain

$$B(\boldsymbol{W}, \boldsymbol{V}) = \left\langle \boldsymbol{W}(t), -\dot{\boldsymbol{V}}(t) - \boldsymbol{A}^*(t)\boldsymbol{V}(t) \right\rangle - (\boldsymbol{W}(t_n), \boldsymbol{V}(t_n)) \quad \forall \boldsymbol{V} \in \mathcal{V}^{q-1}([0,T]).$$

Choosing $\boldsymbol{Z}(t)$ that solves

$$B(\boldsymbol{W}, \boldsymbol{Z}) = (\boldsymbol{W}(T), \boldsymbol{Z}(T)) \quad \forall \boldsymbol{W} \in \mathcal{V}^q([0,T]),$$

and substituting $\boldsymbol{e}$ for $\boldsymbol{W}$, we obtain the **a priori** error representation formula,

$$B(\boldsymbol{e}, \boldsymbol{Z}) = |\boldsymbol{e}(T)|^2.$$

Using Galerkin orthogonality, and the linearity of $B(\cdot, \boldsymbol{Z})$,

$$B(\boldsymbol{e} - \pi\boldsymbol{e}, \boldsymbol{Z}) = |\boldsymbol{e}(T)|^2 \tag{4.4}$$

where $\pi\boldsymbol{e}$ is the L^2 projection of the error into $\mathcal{U}_N$. Hence, $\boldsymbol{Z}(t)$ is the continuous Galerkin approximation in $\mathcal{V}^q([0,T])$ of the weak solution of the continuous adjoint problem

$$\begin{aligned} -\dot{\boldsymbol{z}}(t) - \boldsymbol{A}^*(t)\boldsymbol{z}(t) &= 0 \\ \boldsymbol{z}(T) &= \boldsymbol{e}(T)/\left|\boldsymbol{e}(T)\right|. \end{aligned} \tag{4.5}$$

Further analysis of Equation (4.4) leads to bounds on the error that depend on derivatives of $\boldsymbol{x}(t)$ and the stability properties of the numerical scheme (through $\boldsymbol{Z}(t)$). On the other hand, if we solve Equation (4.5) for $\boldsymbol{z}(t)$, apply Galerkin orthogonality, and compute

$$\left\langle \dot{\boldsymbol{X}}(t) - \boldsymbol{F}(t, \boldsymbol{X}(t)), \boldsymbol{z}(t) - \pi \boldsymbol{z}(t) \right\rangle = |\boldsymbol{e}(t_n)|^2$$

we obtain bounds that depend on $\boldsymbol{X}(t)$ and stability factors which are functionals of $\boldsymbol{z}(t)$ [35].

For finite element method discretizations, where the numerical solution is defined at every point, **a posteriori** analysis can be conducted in a well-defined way using weak formulations over continuous and discrete function spaces. On the other hand, for finite difference methods, where the solution is only defined on a finite set of points, more work must be done to define a suitable adjoint equation and residual operator. One way of approaching this issue is through the construction of special finite element methods that are related to the finite difference method in question. This is described in the next subsection.

4.3.4 Nodally Equivalent Finite Element Methods

Because Runge-Kutta methods do not fall into the scope of typical finite element analysis, the first step to obtaining an error representation formula is the development of a **nodally equivalent finite element method** (neFEM)[21] that allows us to consistently extrapolate the pointwise Runge-Kutta solution (defined at time steps t_n) to a globally defined, piecewise polynomial solution defined on $[0, T]$. This is accomplished by starting with a standard continuous Galerkin method, and then applying various projection operators and quadrature rules so that at the time nodes, t_n, the solution of the modified continuous Galerkin method is equivalent to that of the Runge Kutta method.

In general, an L stage Runge Kutta method can be written in the form

$$\begin{aligned} \boldsymbol{k}_\ell &= \boldsymbol{F}\left(t_n + c_\ell \Delta t_n, \boldsymbol{X}_n + h_n \textstyle\sum_{j=0}^{L} a_{\ell j} \boldsymbol{k}_j\right) \quad \forall \ell \in [1, 2, \ldots, L] \\ \boldsymbol{X}_{n+1} &= \boldsymbol{X}_n + \Delta t \textstyle\sum b_\ell \boldsymbol{k}_\ell \end{aligned} \tag{4.6}$$

For an explicit Runge-Kutta method, $a_{\ell j} = 0$ whenever $j \geq \ell$. In this paper, only explicit Runge-Kutta methods are considered.

The construction of neFEM is discussed in [21]. To summarize, a neFEM is formed by introducing certain projection operators and numerical quadratures to the weak formulation of the problem.The projection operators, $\mathcal{P}_\ell$, correspond to the extrapolations done at each stage of an explicit Runge-Kutta method and can be written in the form

$$\mathcal{P}_\ell[\boldsymbol{X}](t) = \boldsymbol{X}_n(t_n) + \boldsymbol{k}_\ell(t - t_n)$$

where $\boldsymbol{k}_\ell$ is the ℓth stage of the Runge-Kutta method. With this formulation, $\boldsymbol{k}_\ell$ can be written in terms of $\boldsymbol{F}(t, \mathcal{P}_k\boldsymbol{X})$ with $k < \ell$ in the explicit Runge-Kutta case. This leads to a modified variational formulation,

$$\left\langle \dot{\boldsymbol{X}} - \sum_\ell \boldsymbol{F}\left(t, \mathcal{P}_\ell\left[\boldsymbol{X}\right](t)\right), \boldsymbol{V} \right\rangle = 0 \quad \forall \boldsymbol{V} \in \mathcal{V}^{q-1}([0,T]).$$

Next, quadrature rules are used to approximate integrals of the form

$$\int_{t_n}^{t_{n+1}} \boldsymbol{F}(t, \mathcal{P}_\ell[\boldsymbol{X}](t))dt.$$

As an example, we may use the midpoint rule,

$$\int_{t_n}^{t_{n+1}} \boldsymbol{F}(t, \mathcal{P}_\ell[\boldsymbol{X}](t))dt \approx \Delta t \boldsymbol{F}\left[\frac{1}{2}\left(t_n + t_{n+1}\right), \mathcal{P}_\ell[\boldsymbol{X}]\left(\frac{1}{2}\left(t_n + t_{n+1}\right)\right)\right].$$

Combining the multiple stages of a Runge-Kutta method together, we write

$$\left\langle \dot{\boldsymbol{X}}, \boldsymbol{V} \right\rangle = \sum_\ell \langle \boldsymbol{F}(t, \mathcal{P}_\ell\boldsymbol{X}), \boldsymbol{V}\rangle_{d_\ell} \quad \forall \boldsymbol{V} \in \mathcal{V}^{q-1}(I_n)$$

where $\langle \cdot, \cdot \rangle$ represents an integral, and $\langle \cdot, \cdot \rangle_{d_\ell}$ a numerical quadrature rule. Choosing a basis (e.g. Legendre polynomials) for $\mathcal{V}^{q-1}(I_n)$ allows us to solve for some polynomial $\boldsymbol{X}(t) = \sum_{k=0}^{q-1} \boldsymbol{a}_k(t-t_n)^k$ over each interval $[t_n, t_{n+1}]$ which defines the solution at all time points.

4.3.5 Residual, Quadrature, and Explicit Errors

In [21], **a posteriori** estimation techniques are applied to explicit multistep and Runge-Kutta time stepping methods. With implicit finite element methods, the error terms that result from **a**

posteriori analysis involve residuals,

$$\boldsymbol{R}[\boldsymbol{X}] \equiv \dot{\boldsymbol{X}} - \boldsymbol{F}(t, \boldsymbol{X}). \tag{4.7}$$

However, when explicit methods are used, there are two additional sources of error. In our formulation of Runge-Kutta methods as neFEM methods, discrete quadrature approximations introduce quadrature errors, and the approximation of $\boldsymbol{X}(t)$ by various projections introduces an extrapolation error. We briefly summarize Theorem 3 from [21] which shows why these error terms appear.

With the introduction of the projection operators described in the previous section, the "continuous" residual of Equation (4.7) is modified as in [21] with b_ℓ as in Equation (4.6)

$$\boldsymbol{R}_P[\boldsymbol{X}] = \dot{\boldsymbol{X}} - \sum_\ell b_\ell \boldsymbol{F}(t, \mathcal{P}_\ell \boldsymbol{X}).$$

Along with the introduction of numerical quadrature and a choice of finite dimensional function space, this leads to a modified version of the standard Galerkin orthogonality,

$$\left\langle \dot{\boldsymbol{X}}, \boldsymbol{V} \right\rangle - \sum_{\ell=1}^{L} b_\ell \left\langle \boldsymbol{F}(t, \mathcal{P}_\ell \boldsymbol{X}), \boldsymbol{V} \right\rangle_{d_\ell} = 0 \quad \forall \boldsymbol{V} \in \mathcal{V}^{q-1}(I_i),\ i \in [1, 2, \ldots, m].$$

Setting $\boldsymbol{V}(t) = \boldsymbol{z}(t)$ where $\boldsymbol{z}(t)$ is the solution to Equation (4.5), and adding and subtracting $\sum_{\ell=1}^{L} b_\ell \boldsymbol{F}(t, \mathcal{P}_\ell \boldsymbol{X})$, we obtain

$$\langle \boldsymbol{R}[\boldsymbol{X}], \boldsymbol{z} \rangle = \langle \boldsymbol{R}_P[\boldsymbol{X}], \boldsymbol{z} \rangle + \left\langle \sum_\ell^L b_\ell \boldsymbol{F}(t, \mathcal{P}_\ell \boldsymbol{X}) - \boldsymbol{F}(t, \boldsymbol{X}), \boldsymbol{z} \right\rangle$$

Now, applying the modified Galerkin orthogonality,

$$\langle \boldsymbol{R}[\boldsymbol{X}], \boldsymbol{z} \rangle = \boldsymbol{E}_R[\boldsymbol{X}, \boldsymbol{z}] + \boldsymbol{E}_E[\boldsymbol{X}, \boldsymbol{z}] + \boldsymbol{E}_Q[\boldsymbol{X}, \boldsymbol{z}]$$

with

$$\boldsymbol{E}_R[\boldsymbol{X}, \boldsymbol{z}] \equiv \langle \boldsymbol{R}_P[\boldsymbol{X}], \boldsymbol{z} - \pi \boldsymbol{z} \rangle \tag{4.8}$$

$$\boldsymbol{E}_E[\boldsymbol{X}, \boldsymbol{z}] \equiv \left\langle \sum_\ell^L b_\ell \boldsymbol{F}(t, \mathcal{P}_\ell \boldsymbol{X}) - \boldsymbol{F}(t, \boldsymbol{X}), \boldsymbol{z} \right\rangle \tag{4.9}$$

$$\boldsymbol{E}_Q[\boldsymbol{X}, \boldsymbol{z}] \equiv \sum_{\ell=1}^{L} b_\ell \left(\langle \boldsymbol{F}(t, \mathcal{P}_\ell \boldsymbol{X}), \pi \boldsymbol{z} \rangle_{d_\ell} - \langle \boldsymbol{F}(t, \mathcal{P}_\ell \boldsymbol{X}), \pi \boldsymbol{z} \rangle \right) \tag{4.10}$$

Equation (4.8) is a residual error that measures how well the ODE can be approximated in the finite dimensional space, $\mathcal{V}^q([0,T])$, Equation (4.9) is an explicit error term resulting from the extrapolation of $\boldsymbol{X}$ across each interval, and Equation (4.10) is a quadrature error term. A key point in this last step is that the operator $(\mathcal{I}-\pi)$ can often be bounded in terms of derivatives. For instance, bounds of the form $|\boldsymbol{z}-\pi\boldsymbol{z}| \leq Ch\,|\dot{\boldsymbol{z}}|$ are frequently found [35].

4.4 Application to the Method of Regularized Stokeslets

Recalling that $\Delta\alpha_j$ is either a surface area element, or a conserved physical quantity, such as the mass of a particle in a fluid, we introduce the shorthand,

$$\mathcal{S}_\epsilon[\boldsymbol{x}]_k \equiv \sum_j \boldsymbol{U}_\epsilon(\boldsymbol{x}_k-\boldsymbol{x}_j)\boldsymbol{F}_j[\boldsymbol{x}]\Delta\boldsymbol{\alpha}_j$$

so that the method of regularized Stokeslets equations can be written as $\dot{\boldsymbol{x}}_k = \mathcal{S}_\epsilon[\boldsymbol{x}]_k$.

4.4.1 The Adjoint of the Spatially Discrete MRS Operator

For the MRS applied to a network of particles the Fréchet derivative is of the form

$$D\boldsymbol{\mathcal{S}}^h_\epsilon[\boldsymbol{x}](\boldsymbol{y}) = \sum_j \left(\nabla_x \boldsymbol{U}_\epsilon\right)(\boldsymbol{x}_k-\boldsymbol{x}_j) : \left(\boldsymbol{F}_j[\boldsymbol{x}]\otimes(\boldsymbol{y}_k-\boldsymbol{y}_j)\right)h^2 + \sum_j \boldsymbol{U}_\epsilon(\boldsymbol{x}_k-\boldsymbol{x}_j)\cdot\nabla_x\boldsymbol{F}_j[\boldsymbol{x}]\cdot\boldsymbol{y}_j h^2.$$

where $\otimes$ is a dyadic product[3] , and : denotes a second order tensor contraction. The adjoint, can be obtained by considering the inner product

$$\left\langle D\boldsymbol{\mathcal{S}}^h_\epsilon[\boldsymbol{x}](\boldsymbol{y}), \boldsymbol{\phi}\right\rangle =$$

$$\sum_k\sum_j \left(\nabla_x \boldsymbol{U}_\epsilon\right)(\boldsymbol{x}_k-\boldsymbol{x}_j) : \left(\boldsymbol{F}_j[\boldsymbol{x}]\otimes(\boldsymbol{y}_k-\boldsymbol{y}_j)\right)h^2 + \sum_j \boldsymbol{U}_\epsilon(\boldsymbol{x}_k-\boldsymbol{x}_j)\cdot\nabla_x\boldsymbol{F}_j[\boldsymbol{x}]\cdot\boldsymbol{y}_j h^2\boldsymbol{\phi}_k$$

where $\boldsymbol{\phi}$ is contained in the same function space as $\boldsymbol{y}$. To obtain an adjoint operator, we must isolate all terms that depend on $\boldsymbol{y}_k$ for some particular k. We note that in this case, $\nabla_x\boldsymbol{K}(\boldsymbol{x}_k-\boldsymbol{x}_j) = -\nabla_x\boldsymbol{K}(\boldsymbol{x}_j-\boldsymbol{x}_k)$ and that the product adjoint formula: $(AB)^T = B^TA^T$ should be applied when

[3] The dyadic product is defined as $\boldsymbol{a}\otimes\boldsymbol{b} = \boldsymbol{a}\boldsymbol{b}^T$ with $\boldsymbol{a}$ and $\boldsymbol{b}$ in $\mathbb{R}^3$

necessary. The adjoint operator is of the form

$$\left(D\boldsymbol{\mathcal{S}}_\epsilon^h[\boldsymbol{x}]\right)_k^*(\boldsymbol{\phi}) =$$

$$\sum_j \left(\nabla_x \boldsymbol{U}_\epsilon\right)^T (\boldsymbol{x}_k - \boldsymbol{x}_j) : (\boldsymbol{F}_j[\boldsymbol{x}] \otimes \boldsymbol{\phi}_k + \boldsymbol{\phi}_j \otimes \boldsymbol{F}_k[\boldsymbol{x}])h^2 + \nabla_x \boldsymbol{F}_k[\boldsymbol{x}]^T : \left(\sum_j \boldsymbol{U}_\epsilon(\boldsymbol{x}_k - \boldsymbol{x}_j) \cdot \boldsymbol{\phi}_j h^2\right).$$

As an example, if each pair of attached points are connected by an elastic spring, we can write

$$\boldsymbol{F}_j[\boldsymbol{x}] = \sum_{\mathcal{N}_i} k\left(\frac{|\boldsymbol{x}_i - \boldsymbol{x}_j|}{r_{ij}^0} - 1\right)\frac{\boldsymbol{x}_i - \boldsymbol{x}_j}{|\boldsymbol{x}_i - \boldsymbol{x}_j|},$$

where the summation is over points connected to $\boldsymbol{x}_j$, and r_{ij}^0 is the equilibirium separation of $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$. The Fréchet derivative of this operator is

$$D\boldsymbol{F}[\boldsymbol{x}](\boldsymbol{y}) =$$

$$\sum_{\mathcal{N}_i} k\left[\left(\frac{1}{r_{ij}^0} - \frac{1}{|\boldsymbol{x}_i - \boldsymbol{x}_j|}\right)\left(\boldsymbol{\mathcal{I}} - \frac{\boldsymbol{x}_i - \boldsymbol{x}_j}{|\boldsymbol{x}_i - \boldsymbol{x}_j|} \otimes \frac{\boldsymbol{x}_i - \boldsymbol{x}_j}{|\boldsymbol{x}_i - \boldsymbol{x}_j|}\right) + \frac{1}{r_{ij}^0}\frac{\boldsymbol{x}_i - \boldsymbol{x}_j}{|\boldsymbol{x}_i - \boldsymbol{x}_j|} \otimes \frac{\boldsymbol{x}_i - \boldsymbol{x}_j}{|\boldsymbol{x}_i - \boldsymbol{x}_j|}\right] \cdot (\boldsymbol{y}_i - \boldsymbol{y}_j).$$

If the connectivity matrix describing the connections between $\boldsymbol{x}_j$ and $\boldsymbol{x}_k$ is symmetric (as it typically will be due to physical considerations), then this operator is self-adjoint.

4.4.2 Regularization Error

In addition to the residual, quadrature, and explicit errors, there is a regularization error term associated with the MRS. This term arises from the use of regularized Stokeslets instead of singular Stokeslets in Equation (4.2). It can be written as

$$\boldsymbol{E}_{Re}[\boldsymbol{X}, \boldsymbol{z}] \equiv \left\langle \boldsymbol{\mathcal{S}}_\epsilon^h[\boldsymbol{X}] - \boldsymbol{\mathcal{S}}_0^h[\boldsymbol{X}], \boldsymbol{z} \right\rangle. \tag{4.11}$$

We also note that consideration of regularization error effects how the adjoint is defined. If the continuous problem involves no regularization, then the adjoint is defined with respect to a modified problem (due to the use of Runge-Kutta methods), but without regularization as

$$\dot{\boldsymbol{x}} - \sum_\ell b_\ell \boldsymbol{\mathcal{S}}_0^h[\mathcal{P}_\ell \boldsymbol{x}]. \tag{4.12}$$

In [21, §5], the effect of instabilities caused by numerical approximation of an operator on the choice of adjoint is discussed. In this case, since regularization leads to a more stable system, it does

not introduce instabilities that are artifacts of the numerics. Thus, the complicated "dual-adjoint" procedure discussed in [21, §5] is not needed here.

Furthermore, although the operator, $\boldsymbol{\mathcal{S}}_0^h[\mathcal{P}_\ell \boldsymbol{x}]$ is an unbounded operator on $\boldsymbol{x}$, once $\boldsymbol{x}$ has been found, the linearized adjoint of $\boldsymbol{\mathcal{S}}_0^h[\mathcal{P}_\ell \boldsymbol{x}]$ may be treated as a bounded linear operator acting on the adjoint solution.

4.4.3 Numerical Error Estimation and Adjoint Equation Solution Algorithm

With the error representation formula and adjoint equation of the previous section, we now discuss how the error terms are computed once a numerical approximation is found and the adjoint equation solved. All three error terms, Equations (4.8)-(4.10) of Section 4.3.5 depend upon integration over time of a complicated function. These integrals are not analytical in general, and are approximated by numerical quadrature, or by deriving a bound on the size of the operator.

Even with quadrature approximations no reference to the continuous solution is needed to obtain bounds on the error in the quadrature used to approximate the error terms. The use of quadrature risks introducing unreliability into the error bounds, since quadrature may underestimate the integral[4] . However, in most situations, the use of high order quadrature is likely to be accurate so long as the numerical solution is sufficiently smooth, and not severely under-resolved. Furthermore, since the error in quadrature approximations can be bounded in terms of derivatives of the integrand, which in this case is a function of the numerical solution, this approach preserves the **a posteriori** nature of the bounds.

In practice, given some explicit Runge-Kutta method of order p, we use Gaussian quadrature formulas to approximate the time integration. Since the projection operators of the **neFEM** are defined for any time, it is possible to evaluate $\boldsymbol{\mathcal{S}}_\epsilon^h[\mathcal{P}_\ell \boldsymbol{x}]$ at any time, thus any standard quadrature method may be employed. We use sufficiently high order Gaussian quadrature formulas so that the error in estimating the integrals is much smaller than the value of the integral.

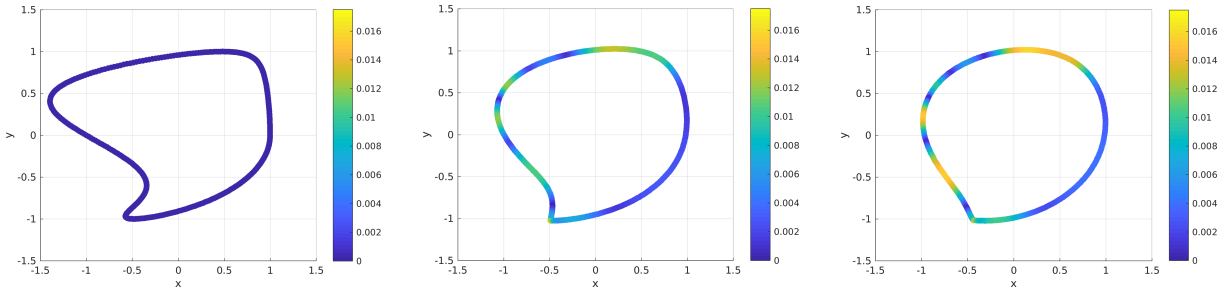
[4] Quadrature may also cause overestimation of the integral. In this case, error bound remains valid, but loses its sharpness

Now that we have discussed how the error terms can be estimated, it remains to discuss how to solve the adjoint equation for $\boldsymbol{z}(t)$. For the forward problems, we use Runge-Kutta methods of order 4 or less. In the **a posteriori** it is customary to use a higher order solver for the adjoint equation in order to ensure that the error terms are accurately computed. . Thus, we use a 6th order Runge-Kutta method [14]. In many practical applications, $\boldsymbol{z}$ is only needed for controlling the error and is not as important in terms of physical insight as the numerical approximant, $\boldsymbol{x}$. Thus, it may be the case that crude estimates for $\boldsymbol{z}$ are sufficient.

The "correct" initial condition (or final condition since data is specified at $t = T$) is to set $\boldsymbol{z}(T) = \boldsymbol{e}(T)/\|\boldsymbol{e}(T)\|$. This is problematic since $\boldsymbol{e}(T)$ is unknown. However, approximations of the initial condition for the adjoint equation were studied in [16]. It was shown that randomized initial conditions for the adjoint equation perform well for error estimation, and that taking the maximum error approximation over several initial values will lead, with increasingly high probability, to nearly optimal results. Since we lack information about what $\boldsymbol{z}(T)$ should be, we chose $\boldsymbol{z}(T)$ to be a random vector of unit norm. For the case where the ODE is a spatial discretization of a PDE, we attempt to more accutely capture spatial correlations in the error by drawing initial data from a Gaussian spatial process. Such initial data can be obtained efficiently using Fourier transform techniques discussed in [58].

4.5 Numerical Results

In computing the forward solution, we use several common Runge-Kutta methods. In particular, we use the RK4 method and the second order accurate Heun's method. We track over each interval, the values of $\boldsymbol{x}(t_n)$ and also $\boldsymbol{k}_i$ $i = 1, \ldots \ell$, the result from each stage of the RK method. These values are sufficient to reconstruct a polynomial FEM solution over each interval in a partition of $[0, T]$. For the adjoint equation we obtain a solution through the use of a sixth order Runge-Kutta method [14]. It is important to note that although the Runge Kutta method may exhibit a certain order of convergence at the time nodes, the extrapolated polynomial solution may be of lower degree over the interval. For instance with the RK4 method the extrapolated polynomial



(a) (b) (c)

Figure 4.1: The regularization error term and position of boundary shown at $t = 0.0s$ (a), $t = 0.5s$ (b), and $t = 1.0s$ (c).

solution is accurate to second order. This fact is verified through our numerical results; when the time step is halved, the error is quartered, but superconvergence occurs at the time nodes due to special cancellations of error [21]. As is done in **a priori** numerical estimation, to ensure that the Runge Kutta methods obtain the theoretically expected convergence rates, we compute convergence factors of the form,

$$\rho = \frac{\log |\boldsymbol{X}_h(T) - \tilde{\boldsymbol{x}}(T)|}{\log |\boldsymbol{X}_{h/2}(T) - \tilde{\boldsymbol{x}}(T)|}$$

where $\tilde{\boldsymbol{x}}$ is a highly refined numerical approximation to $\boldsymbol{x}$, and $\boldsymbol{X}_h$ is the numerical approximation with time step h. For the RK4 method, it was observed that $\rho \approx 4$ and for the RK6 method $\rho \approx 6$ for the problems we tested.

4.5.1 Method of Regularized Stokeslets Example

In this case we consider a deformed circle as the initial starting position and consider the errors developed for a tethered boundary and an elastic boundary. We set the resting configuration of the boundary to be a circle of radius 1, and the initial position of the boundary is parametrized as

$$\boldsymbol{x}_0(s) = \left(\cos(\pi s) + \frac{1}{2}\sin\left(2\pi\cos\left(\pi(s-1)\right)\right), \sin(\pi s)\right) \quad s \in [0, 1)$$

and simulate the boundary motion as it deforms towards its resting position.

In Figure 4.1, we see the residual error and adjoint solution to the problem described above.

We see that the initially deformed shape relaxes towards its resting configuration as a circle. We also observe some concentration in the error where the curvature is highest. In Figure 4.2, the time dependence of the explicit error, residual error, and adjoint solution norms are shown. The quadrature error is not shown since it behaves very similarly to the explicit error term. The error terms grow rapidly at first, followed by a period of slow growth as the boundary approaches its resting configuration. The adjoint solution remains small even at $t = 0$ and gradually decays across the time interval.

As a second example, we consider a circular boundary that is deformed in a background shear velocity field of the form $\boldsymbol{u} = (y, 0)$. In Figure 4.3, we see the extension and rotation of a circular elastic body in response to a shear flow field. The coloring of the deforming object corresponds to the magnitude of the regularization error. Figure 4.4 depicts the time dependence of the ℓ_2 norm over space during this deformation. We see that the adjoint solution remains small, but that given the numerical parameters of our simulation, the regularization error dominates the residual and explicit errors and increases roughly linearly in time.

4.5.2 Method of Regularized Stokeslets applied to an Elastic Network of Fibers

In [101], the method of regularized Stokeslets was used to model a three dimensional network of fibers immersed in a fluid. This is of interest as there are number of application in biology where the fluid structure interactions of such materials are of importance. The application in [101] was geared towards understanding how a spermatocyte swims through a material known as the zona pellucida that surrounds ooctyes. Another potential application is in the study of biofilms growing in a slowly moving fluid. In this case, the particles of the method of regularized Stokeslets represent bacteria cells, and are not elements of a discretized surface. In this case, the ODE system is exact no longer an approximation of a PDE, but forms the governing equations. Regularization in this context is typically used as a means of ensuring stability of the resulting numerical simulations by limiting the velocity when bacteria approach each other. The choice of length scale then depends on the level of resolution needed in the simulation.

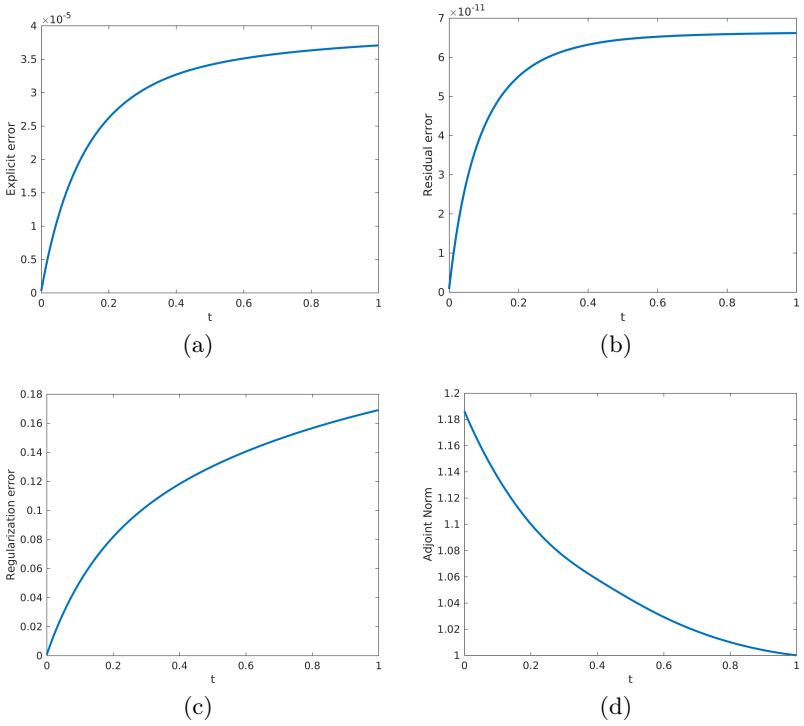


Figure 4.2: Explicit Error, Residual Error, Regularization Error, and the norm of $\boldsymbol{z}(\cdot, t)$ as functions of time.

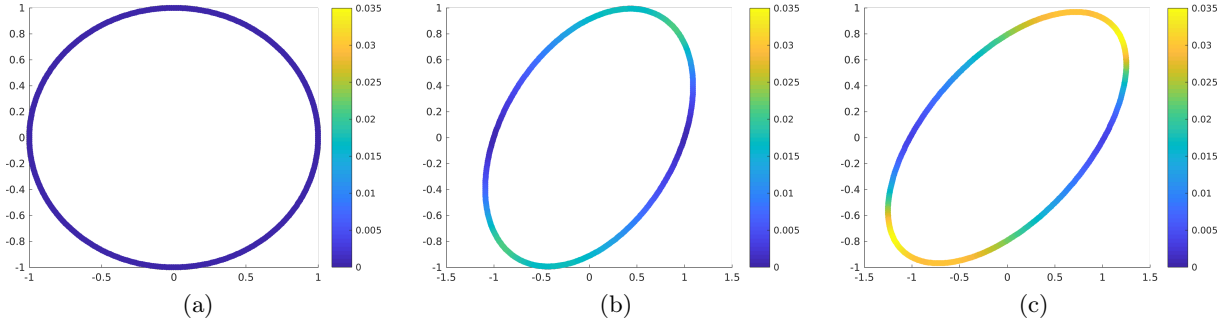


Figure 4.3: The regularization error term and position of boundary shown at $t = 0.0s$ (a), $t = 0.5s$ (b), and $t = 1.0s$ (c).

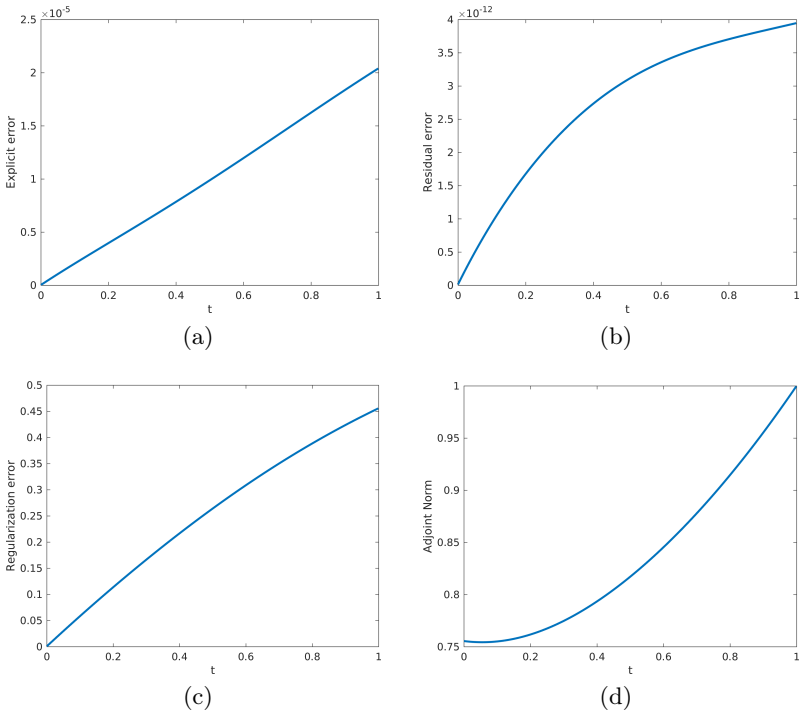
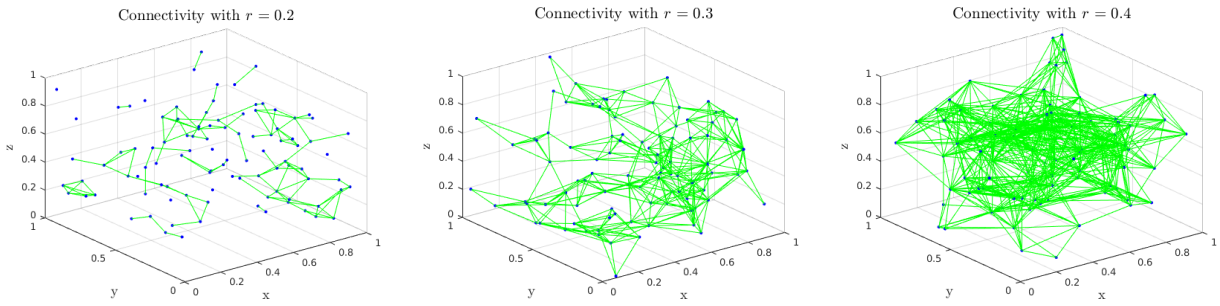


Figure 4.4: Explicit Error, Residual Error, Regularization Error, and the norm of $\boldsymbol{z}(\cdot, t)$ as functions of time.



(a) (b) (c)

Figure 4.5: A network of elastic fibers with 100 points for different values for $r_{connect}$.

Following [101, 102], we write

$$\begin{aligned} \dot{\boldsymbol{x}}_k(t) &= \sum_j \boldsymbol{U}_\epsilon(\boldsymbol{x}_k - \boldsymbol{x}_j)\boldsymbol{f}_{kj}(t)m_j \\ \boldsymbol{f}_{kj}(t) &= \ell_{kj}^2 E_{kj}\left(\frac{|\boldsymbol{x}_k(t) - \boldsymbol{x}_j(t)|}{\ell_{kj}} - 1\right)\frac{\boldsymbol{x}_k(t) - \boldsymbol{x}_j(t)}{|\boldsymbol{x}_k(t) - \boldsymbol{x}_j(t)|} \end{aligned}$$

Note that $f_{kj}(t) = -f_{jk}(t)$. In [102, 101], the resting length is allowed to change in order to incorporate viscoelastic effects, but for simplicity, we consider only the elastic (constant resting length) case here.

As a test problem, we consider 100 points that form a network of fibers with a connectivity rule that if $|\boldsymbol{x}_i - \boldsymbol{x}_j| \leq r_{connect}$, at time 0, then $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are attached by an elastic spring, and if $|\boldsymbol{x}_i - \boldsymbol{x}_j| > r_{connect}$, then there is no attachment. In Figure 4.5,

an image of such a network is depicted.

As an example problem, we place the elastically connected bacteria in a linear flow field, $\boldsymbol{u}(\boldsymbol{x}) = \boldsymbol{C} \cdot \boldsymbol{x}$ where C is a second order tensor. The velocity of each point is then set to

$$\dot{\boldsymbol{x}}_k(t) = \sum_j \boldsymbol{U}_\delta(\boldsymbol{x}_k - \boldsymbol{x}_j)\boldsymbol{f}_{kj}(t)m_j + \boldsymbol{u}(\boldsymbol{x}_k(t)).$$

The linear background velocity field appears in the error representation formula and adjoint equation. The ODE operator is augmented from $\boldsymbol{\mathcal{S}}_\delta^h[\boldsymbol{F}[\boldsymbol{x}], \boldsymbol{x}]$ to become

$$\boldsymbol{F}(t, \boldsymbol{x}) = \boldsymbol{\mathcal{S}}_\epsilon^h[\boldsymbol{x}] + \boldsymbol{u}(\boldsymbol{x}).$$

The addition of $\boldsymbol{u}(\boldsymbol{x})$ is then carried through all of the error components Equations (4.8)-(4.10). In the adjoint equation, we obtain an additional term of the form

$$\boldsymbol{C}^T \cdot \boldsymbol{\phi}.$$

For a general, nonlinear velocity field, the resulting term in the linearized adjoint equation would depend on $\boldsymbol{x}$, e.g. $\boldsymbol{C}^T(\boldsymbol{x}) \cdot \boldsymbol{\phi}$. Various choices for $\boldsymbol{C}$ lead to shear flow, straining flow, and rotational flow [80].

Alternatively, it is also possible to model situations where flow is driven by forces on the bacteria, e.g. gravitational settling of particles. In this case, the force on each particle would be of the form

$$\boldsymbol{F}[\boldsymbol{x}] + \boldsymbol{g}.$$

Since gravitational force is independent of position and time, the adjoint equation is not changed, and the error formulas are only modified through a different force relation.

Other examples of interest include cases where the forces depend on position, or when the positions of a subset of the particles is predetermined. The former case may occur with charged particles in an external field, and the latter may occur if some of the particles are assumed to be adhered to a moving boundary. Furthermore, the methods discussed here are directly applicable kernels aside from the free-space Green's function. For instance, the techniques developed here may be applied to half-plane flows, or flows in a sphere where Green's functions may be obtained through the method of images [80].

In Figure 4.6,

the accumulation of explicit error, and the magnitude of the components of $\boldsymbol{\phi}$ are plotted versus time. We observe that the magnitude of $\boldsymbol{\phi}(t)$ grows approximately linearly over time and that the explicit error accumulation seems to grow most rapidly at the beginning of the simulation.

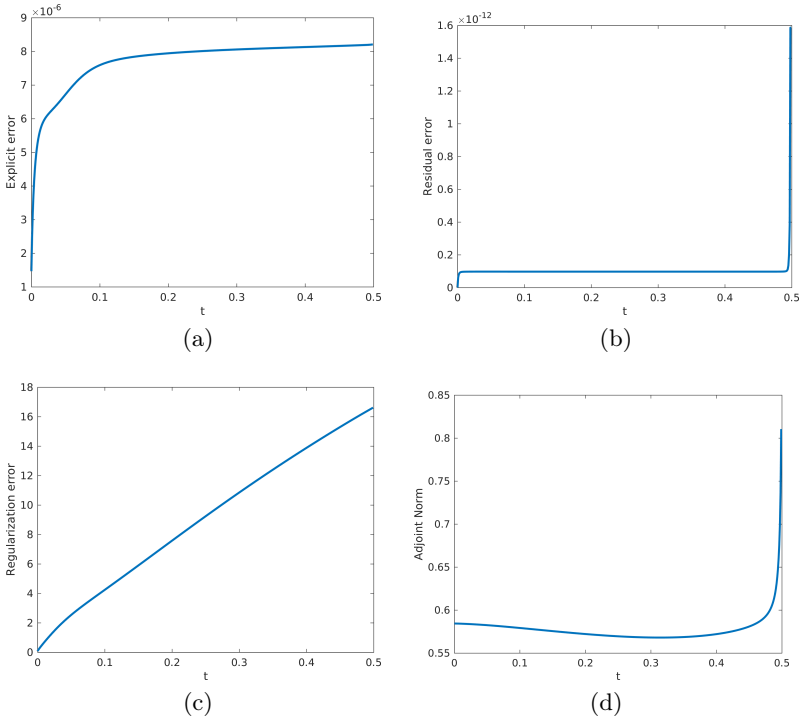


Figure 4.6: Explicit Error, Residual Error, Regularization Error, and the norm of $\boldsymbol{z}(\cdot, t)$ as functions of time for a network of elastically connected particles.

4.6 Discussion

We have implemented the **a posteriori** error estimation techniques, originally developed in [21] in application to a spatially discretized integrodifferential equation. Such equations are relevant to various models in biological fluid dynamics. In particular, the Method of Regularized Stokeslets is a popular technique for simulated low-speed small length scale biological fluid-structure interactions. Although the method is widely used due to its accuracy, ease of implementation, and the fact that results from the MRS can often be compared to experimental results, little theory exists about the well-posedness, and error accumulation in the case of dynamically moving boundaries. Furthermore, we are not aware of any previous studies that have looked at error control in conjunction with the MRS.

One ambiguity in this work is on the choice of method for extrapolating a finite difference solution to a continuous function. We believe there exists potential for improvements if extrapolations that are of the same order of accuracy as the numerical solution are used instead of lower order approximants. This seems especially likely for higher order Runge-Kutta methods where the extrapolation leads to second order accuracy in general, but superconvergence at the time nodes. One way obtain such approximants may be to use a spectral discretization in time, similar to those used in spectral deferred correction methods. Such discretizations lead to natural choices for extrapolants that are accurate up to the order of the discretization. We also note that using an extrapolant that agrees at the time nodes, t_n, but does not satisfy the property that its projection leads to $\boldsymbol{k}_i$ of Equation (4.6) might allow for choices of polynomial that exhibit higher order accuracy than those currently employed by the **neFEM** scheme.

Another interesting direction that may be helpful for practioners hoping to applied the methods discussed here would be an analysis of the impact of numerical errors in the adjoint equation solution on the estimators. An initial investigation on this topic was conducted in [21], but there is certainly room for further developments. For instance, it is agreed upon that a CFL condition exists for the explicit discretization of the motion of an elastic boundary in a fluid. However, it is

not clear that the **a posteriori** methods we have used can necessarily capture the moments when a simulation becomes unstable. Instability should be accompanied by rapid growth of the adjoint solution however, it is possible that the nonlinear behavior of the operators involved may fail to be captured by the linearized adjoint used to obtain stability factors.

A natural next step is to combine the spatial and temporal analysis to develop **a posteriori** estimation algorithms for MRS simulations of an elastic surface. The further extension to viscoelastic surfaces is of interest in many applications, but poses additional, nontrivial challenges due to the more complicated dynamic behavior of viscoelastic materials. We are pursuing these ideas in a follow-up paper. As a further application, we also hope to extend the methods here to quantify the accuracy of biofilm simulations such as those described in [89] and [52].

Chapter 5

Conclusions

In the previous chapters, a continuum biofilm model, a statistical biofilm model, and the numerical analysis of a numerical method relevant to microorganism studies have been discussed extensively. With the exception of Chapter 4, on *a posteriori* error analysis, the research discussed in this thesis can be found in two publications: [89] and [88]. The results of Chapter 4 will be published in the near future.

In Chapter 2, the hrIBM model was introduced and used to conduct a number of biofilm simulations. Key features of this model are that it captures the highly heterogeneous nature of biofilm microstructure, and that the initial positions of bacteria in the model are obtained from high-resolution confocal microscopy images. This work to lead to several key results. First, the heterogeneous rheological properties of the biofilms play a crucial role in determining the biomechanical response of a biofilm to stress and strain. Furthermore, quantitative agreement between the dynamic moduli and compliance moduli to experimental data was found with a suitable choice of viscoelastic model. The development of a computationally efficient means of solving the modeling equations was a significant challenge throughout the research described in this chapter.

In Chapter 3, a spatial statistical modeling approach based on point process theory and statistical physics was used to develop a generative statistical model of the positions of bacteria in a biofilm. Such a model is important for conducting biofilm simulations in conditions where ex-

perimental data is not available to describe the positions of bacteria in a biofilm. Additionally, although a number of experimental results on biofilm spatial statistics may exist, they had never been aggregated into a model that yields realizations consistent with the experimentally observed properties. Our key results in this chapter were to develop the statistical model, and also to highlight the importance of non-uniformity in bacterial positions on the biomechanical response of the biofilm.

In Chapter 4, simpler models were used and the focus was instead on the analysis of such models using techniques from *a posteriori* error analysis. In particular, we showed how different error components propagate over time in some simple biofilm simulations. The key developments in this chapter were the derivation of appropriate adjoint equations to model error growth, and also the derivation of an error representation formula for the method of regularized Stokeslets as applied to interacting particles in a viscous fluid.

In each of the three previous chapters, numerous opportunities exist for further development. In Chapter 2, the results could be extended by parallelization of of the code using MPI based routines. This would allow for simulations of larger biofilms, providing a more concrete comparison of the model to experiment. Furthermore, if combined with a biochemical model, simulation of corrosion due to biofilms may be possible. In Chapter 3, a useful next step would be to conduct simulations of biofilms in more complicated geometry. In the literature, it appears that only periodic or "square" geometries have been studied. However, the interaction between biofilms and their geometries can lead to interesting phenomena that have important practical consequences [32]. Finally, Chapter 4 can be extended in several ways. The first would be to include regularization error in the time-dependent *a posteriori* estimates. In many cases regularization error is a dominant error source. The further extension to spatiotemporal error is also an obvious next step, although it involves many technical difficulties.

The takeaway from this thesis is that biofilm modeling and simulation is first of all a useful endeavor with significant applications. I hope the reader also finds that this thesis also presents

convincing arguments that with careful modeling and parameter choices, quantitatively accurate descriptions of biomechanics can be obtained from mathematical models. Lastly, although full numerical analysis of most biofilm models is not yet possible, preliminary results indicate that such models are convergent and achieve high levels of accuracy with reasonable numerical parameter values.

Bibliography

[1] I. S. Abramson, On bandwidth variation in kernel estimates-a square root law, The Annals of Statistics, (1982), pp. 1217–1223.

[2] E. Alpkvist and I. Klapper, Description of mechanical response including detachment using a novel particle model of biofilm/flow interaction, Water Science and Technology, 55 (2007), pp. 265–273.

[3] E. Alpkvist and I. Klapper, A Multidimensional Multispecies Continuum Model for Heterogeneous Biofilm Development, Bulletin of Mathematical Biology, 69 (2007-02-22), pp. 765–789.

[4] E. Alpkvist, C. Picioreanu, M. C. van Loosdrecht, and A. Heyden, Three-dimensional biofilm model with individual cells and continuum EPS matrix, Biotechnology and Bioengineering, 94 (2006-08-05), pp. 961–979.

[5] C. Anderson and C. Greengard, On Vortex Methods, SIAM Journal on Numerical Analysis, 22 (1985), pp. 413–440.

[6] V. Aranda, R. Cortez, and L. Fauci, A model of stokesian peristalsis and vesicle transport in a three-dimensional closed cavity, Journal of biomechanics, 48 (2015), pp. 1631–1638.

[7] A. Baddeley and R. Turner, Practical maximum pseudolikelihood for spatial point patterns, Australian & New Zealand Journal of Statistics, 42 (2000), pp. 283–322.

[8] A. J. Baddeley, J. Møller, and R. Waagepetersen, Non-and semi-parametric estimation of interaction in inhomogeneous point patterns, Statistica Neerlandica, 54 (2000), pp. 329–350.

[9] D. Balestrino, J.-M. Ghigo, N. Charbonnel, J. A. J. Haagensen, and C. Forestier, The characterization of functions involved in the establishment and maturation of Klebsiella pneumoniae in vitro biofilm reveals dual roles for surface exopolysaccharides, Environmental Microbiology, 10 (2008), pp. 685–701.

[10] P. Billingsley, Probability and measure, John Wiley & Sons, 2008.

[11] S. Blaser, Forces on the surface of small ellipsoidal particles immersed in a linear flow field, Chemical Engineering Science, 57 (2002-02), pp. 515–526.

[12] D. C. Bottino, Modeling Viscoelastic Networks and Cell Deformation in the Context of the Immersed Boundary Method, Journal of Computational Physics, 147 (1998-11), pp. 86–113.

[13] D. L. Brown, R. Cortez, and M. L. Minion, Accurate projection methods for the incompressible navier-stokes equations, Journal of Computational Physics, 168 (2001-04), pp. 464–499.

[14] J. Butcher, On fifth and sixth order explicit runge-kutta methods: order conditions and order barriers, Canadian Applied Mathematics Quarterly, 17 (2009), pp. 433–445.

[15] E. Byrne, S. Dzul, M. Solomon, J. Younger, and D. M. Bortz, Postfragmentation density function for bacterial aggregates in laminar flow, Physical Review E, 83 (2011).

[16] Y. Cao and L. Petzold, A Posteriori Error Estimation and Global Error Control for Ordinary Differential Equations by the Adjoint Method, SIAM Journal on Scientific Computing, 26 (2004), pp. 359–374.

[17] H. S. Carlslaw and J. C. Jaegar, Conduction of Heat in Solids, Oxford University Press, 2nd Edition ed., 1959.

[18] C. Chen, M. Ren, A. Srinivansan, and Q. Wang, 3-D Numerical Simulations of Biofilm Flows, East Asian Journal on Applied Mathematics, (2011).

[19] R. M. Christensen, Theory of viscoelasticity: an introduction, Academic Press, New York, 2nd ed ed., 1982.

[20] J. F. Coeurjolly, J. Møller, and R. Waagepetersen, Conditioning in spatial point processes, arXiv preprint arXiv:1512.05871, (2015).

[21] J. B. Collins, D. Estep, and S. Tavener, A posteriori error analysis for finite element methods with projection operators as applied to explicit time integration techniques, BIT Numerical Mathematics, 55 (2015), pp. 1017–1042.

[22] P. R. Conrad, Y. M. Marzouk, N. S. Pillai, and A. Smith, Accelerating asymptotically exact MCMC for computationally intensive models via local approximations, Journal of the American Statistical Association, 111 (2016), pp. 1591–1607.

[23] R. Cortez, On the Accuracy of Impulse Methods for Fluid Flow, SIAM Journal on Scientific Computing, 19 (1998), pp. 1290–1302.

[24] ——, The Method of Regularized Stokeslets, SIAM Journal on Scientific Computing, 23 (2001), pp. 1204–1225.

[25] R. Cortez, L. Fauci, and A. Medovikov, The method of regularized Stokeslets in three dimensions: Analysis, validation, and application to helical swimming, Physics of Fluids, 17 (2005), p. 031504.

[26] R. Courant and D. Hilbert, Methods of mathematical physics, vol. i, Physics Today, 7 (1954), pp. 17–17.

[27] J. C. Crocker and D. G. Grier, Methods of Digital Video Microscopy for Colloidal Studies, Journal of Colloid and Interface Science, 179 (1996), pp. 298–310.

[28] O. Cronie and M. N. M. van Lieshout, Bandwidth selection for kernel estimators of the spatial intensity function, arXiv preprint arXiv:1611.10221, (2016).

[29] D. J. Daley and D. Vere-Jones, An introduction to the theory of point processes: volume II: general theory and structure, Springer Science & Business Media, 2007.

[30] G. Dan Vo, E. Brindle, and J. Heys, An experimentally validated immersed boundary model of fluid-biofilm interaction, Water Science & Technology, 61 (2010-06), p. 3033.

[31] R. H. Dillon, L. J. Fauci, C. Omoto, and X. Yang, Fluid Dynamic Models of Flagellar and Ciliary Beating, Annals of the New York Academy of Sciences, 1101 (2007), pp. 494–505.

[32] K. Drescher, Y. Shen, B. L. Bassler, and H. A. Stone, Biofilm streamers cause catastrophic disruption of flow with consequences for environmental and medical systems, Proceedings of the National Academy of Sciences, 110 (2013), pp. 4345–4350.

[33] S. P. Dzul, M. M. Thornton, D. N. Hohne, E. J. Stewart, A. A. Shah, D. M. Bortz, M. J. Solomon, and J. G. Younger, Contribution of the Klebsiella pneumoniae capsule to bacterial aggregate and biofilm microstructures, Applied and environmental microbiology, 77 (2011), pp. 1777–1782.

[34] V. A. Epanechnikov, Non-parametric estimation of a multivariate probability density, Theory of Probability & Its Applications, 14 (1969), pp. 153–158.

[35] D. Estep, A Posteriori Error Bounds and Global Error Control for Approximation of Ordinary Differential Equations, SIAM Journal on Numerical Analysis, 32 (1995), pp. 1–48.

[36] T. G. Fai, B. E. Griffith, Y. Mori, and C. S. Peskin, Immersed Boundary Method for Variable Viscosity and Variable Density Problems Using Fast Constant-Coefficient Linear Solvers I: Numerical Method and Results, SIAM Journal on Scientific Computing, 35 (2013-01), pp. B1132–B1161.

[37] ——, Immersed Boundary Method for Variable Viscosity and Variable Density Problems Using Fast Constant-Coefficient Linear Solvers II: Theory, SIAM Journal on Scientific Computing, 36 (2014-01), pp. B589–B621.

[38] T. G. Fai, A. Leo-Macias, D. L. Stokes, and C. S. Peskin, Image-based model of the spectrin cytoskeleton for red blood cell simulation, PLOS Computational Biology, 13 (2017), p. e1005790.

[39] H. C. Flemming, Microbial biofouling: unsolved problems, insufficient approaches, and possible solutions, in Biofilm highlights, Springer, 2011, pp. 81–109.

[40] H.-C. Flemming, J. Wingender, and U. Szewzyk, eds., Biofilm highlights, Springer series on biofilms, Springer, Heidelberg ; New York, 2011. OCLC: ocn729346877.

[41] T. Foster, Staphylococcus, in Medical Microbiology, S. Baron, ed., University of Texas Medical Branch at Galveston, Galveston (TX), 4th ed., 1996.

[42] F. Gaboriaud, M. L. Gee, R. Strugnell, and J. F. L. Duval, Coupled Electrostatic, Hydrodynamic, and Mechanical Properties of Bacterial Interfaces in Aqueous Media, Langmuir, 24 (2008), pp. 10988–10995.

[43] F. Gaboriaud, M. L. Gee, R. Strugnell, and J. F. L. Duval, Coupled Electrostatic, Hydrodynamic, and Mechanical Properties of Bacterial Interfaces in Aqueous Media, Langmuir, 24 (2008-10-07), pp. 10988–10995.

[44] M. Ganesan, E. J. Stewart, J. Szafranski, A. E. Satorius, J. G. Younger, and M. J. Solomon, Molar Mass, Entanglement, and Associations of the Biofilm Polysaccharide of Staphylococcus epidermidis, Biomacromolecules, 14 (2013), pp. 1474–1481.

[45] A. Gangopadhyay and K. Cheung, Bayesian approach to the choice of smoothing parameter in kernel density estimation, Journal of Nonparametric Statistics, 14 (2002), pp. 655–664.

[46] C. J. Geyer and J. Møller, Simulation procedures and likelihood inference for spatial point processes, Scandinavian journal of statistics, (1994), pp. 359–373.

[47] Y. Guan, A least-squares cross-validation bandwidth selection approach in pair correlation function estimations, Statistics & Probability Letters, 77 (2007), pp. 1722–1729.

[48] ——, On consistent nonparametric intensity estimation for inhomogeneous spatial point processes, Journal of the American Statistical Association, 103 (2008), pp. 1238–1247.

[49] T. Guélon, J. D. Mathias, and P. Stoodley, Advances in biofilm mechanics, in Biofilm Highlights, Springer, 2011, pp. 111–139.

[50] M. Guizar-Sicairos and J. C. Gutiérrez-Vega, Computation of quasi-discrete hankel transforms of integer order for propagating optical wave fields, Journal of the Optical Society of America A, 21 (2004), pp. 53–58.

[51] P. Hall and J. S. Marron, Local minima in cross-validation functions, Journal of the Royal Statistical Society. Series B (Methodological), (1991), pp. 245–252.

[52] J. F. Hammond, E. J. Stewart, J. G. Younger, M. J. Solomon, and D. M. Bortz, Variable Viscosity and Density Biofilm Simulations using an Immersed Boundary Method, Part I: Numerical Scheme and Convergence Results, CMES, 98 (2014), pp. 295–340.

[53] J. P. Hansen and I. R. McDonald, Theory of simple liquids, Elsevier, 1990.

[54] W. Hardle, J. S. Marron, and M. P. Wand, Bandwidth choice for density derivatives, Journal of the Royal Statistical Society. Series B (Methodological), (1990), pp. 223–232.

[55] M. C. Jones, Simple boundary correction for kernel density estimation, Statistics and Computing, 3 (1993), pp. 135–146.

[56] M. Kerscher, I. Szapudi, and A. S. Szalay, A comparison of estimators for the two-point correlation function, The Astrophysical Journal Letters, 535 (2000), p. L13.

[57] S. Kim and S. J. Karrila, Microhydrodynamics: principles and selected applications, Butterworth-Heinemann series in chemical engineering, Butterworth-Heinemann, Boston, 1991.

[58] D. P. Kroese and Z. I. Botev, Spatial process generation, arXiv preprint arXiv:1308.0399, (2013).

[59] G. E. Ladas and V. Lakshmikantham, Differential equations in abstract spaces, Elsevier, 1972.

[60] O. A. Ladyzhenskaya, The mathematical theory of viscous incompressible flow, vol. 12, 3, Gordon & Breach New York, 1969.

[61] M.-C. Lai and C. S. Peskin, An Immersed Boundary Method with Formal Second-Order Accuracy and Reduced Numerical Viscosity, Journal of Computational Physics, 160 (2000), pp. 705–719.

[62] S. D. Landy and A. S. Szalay, Bias and variance of angular correlation functions, The Astrophysical Journal, 412 (1993), pp. 64–71.

[63] C. S. Laspidou and B. E. Rittmann, Modeling the development of biofilm density including active bacteria, inert biomass, and extracellular polymeric substances, Water Research, 38 (2004), pp. 3349–3361.

[64] F. Lin and J. Tong, Solvability of the Stokes Immersed Boundary Problem in Two Dimensions, arXiv, 1703.03124 (2017).

[65] Y. Liu and Y. Mori, Properties of Discrete Delta Functions and Local Convergence of the Immersed Boundary Method, SIAM Journal on Numerical Analysis, 50 (2012-01), pp. 2986–3015.

[66] ——, L^p Convergence of the Immersed Boundary Method for Stationary Stokes Problems, SIAM Journal on Numerical Analysis, 52 (2014-01), pp. 496–514.

[67] R. Lovett, C. Y. Mou, and F. P. Buff, The structure of the liquid-vapor interface, J. Chem. Phys, 65 (1976), p. 2377.

[68] H. Luo, R. Mittal, X. Zheng, S. A. Bielamowicz, R. J. Walsh, and J. K. Hahn, An immersed-boundary method for flow-structure interaction in biological systems with application to phonation, Journal of Computational Physics, 227 (2008-11), pp. 9303–9332.

[69] A. Majda and A. L. Bertozzi, Vorticity and incompressible flow, Cambridge texts in applied mathematics, Cambridge University Press, Cambridge ; New York, 2002.

[70] I. Modeling, Mathematical Modeling of Biofilms, Scientific and Technical Report Series, IWA Publishing, 2006.

[71] J. Moller and R. P. Waagepetersen, Statistical inference and simulation for spatial point processes, CRC Press, 2003.

[72] Y. Mori, A. Rodenberg, and D. Spirn, Well-posedness and global behavior of the Peskin problem of an immersed elastic filament in Stokes flow, arXiv, 1704.08392 (2017).

[73] L. S. Ornstein and F. Zernike, The influence of accidental deviations of density on the equation of state, Koninklijke Nederlandsche Akademie van Wetenschappen Proceedings, 19 (1914), pp. 1312–1315.

[74] S. A. Orszag, M. Israeli, and M. O. Deville, Boundary conditions for incompressible flows, Journal of Scientific Computing, 1 (1986-03-01), pp. 75–111.

[75] E. Parzen, On estimation of a probability density function and mode, The annals of mathematical statistics, 33 (1962), pp. 1065–1076.

[76] L. Pavlovsky, R. A. Sturtevant, J. G. Younger, and M. J. Solomon, Effects of Temperature on the Morphological, Polymeric, and Mechanical Properties of *Staphylococcus epidermidis* Bacterial Biofilms, Langmuir, 31 (2015), pp. 2036–2042.

[77] L. Pavlovsky, J. G. Younger, and M. J. Solomon, In situ rheology of Staphylococcus epidermidis bacterial biofilms, Soft Matter, 9 (2013), p. 122.

[78] C. S. Peskin, Numerical analysis of blood flow in the heart, Journal of Computational Physics, 25 (1977-11), pp. 220–252.

[79] C. S. Peskin, The immersed boundary method, Acta Numerica, 11 (2002-01).

[80] C. Pozrikidis, Boundary integral and singularity methods for linearized viscous flow, Cambridge University Press, 1992.

[81] B. D. Ripley, Statistical inference for spatial processes, Cambridge university press, 1991.

[82] B. E. Rittmann and P. L. McCarty, Model of steady-state-biofilm kinetics, Biotechnology and Bioengineering, 22 (1980), pp. 2343–2357.

[83] M. Rosenblatt et al., Remarks on some nonparametric estimates of a density function, The Annals of Mathematical Statistics, 27 (1956), pp. 832–837.

[84] B. W. Silverman, Using Kernel Density Estimates to Investigate Multimodality, Journal of the Royal Statistical Society, 43 (1981), pp. 97–99.

[85] K. Sobczyk and D. J. Kirkner, Stochastic modeling of microstructures, Springer Science & Business Media, 2012.

[86] E. J. Stewart, M. Ganesan, J. G. Younger, and M. J. Solomon, Artificial biofilms establish the role of matrix interactions in staphylococcal biofilm assembly and disassembly, Scientific Reports, 5 (2015), p. 13081.

[87] E. J. Stewart, A. E. Satorius, J. G. Younger, and M. J. Solomon, Role of Environmental and Antibiotic Stress on Staphylococcus epidermidis Biofilm Microstructure, Langmuir, 29 (2013), pp. 7017–7024.

[88] J. A. Stotsky, V. Dukic, and D. M. Bortz, A point process model for generating biofilms with realistic microstructure and rheology, arXiv preprint arXiv:1707.05739, (2017).

[89] J. A. Stotsky, J. F. Hammond, L. Pavlovsky, E. J. Stewart, J. G. Younger, M. J. Solomon, and D. M. Bortz, Variable viscosity and density biofilm simulations using an immersed boundary method, part II: Experimental validation and the heterogeneous rheology-IBM, Journal of Computational Physics, 317 (2016), pp. 204–222.

[90] D. Stoyan, U. Bertram, and H. Wendrock, Estimation variances for estimators of product densities and pair correlation functions of planar point processes, Annals of the Institute of Statistical Mathematics, 45 (1993), pp. 211–221.

[91] D. Stoyan, W. S. Kendall, and J. Mecke, Stochastic geometry and its applications. 1995, Akademie-Verlag, Berlin, (1995).

[92] V. R. Stull, Size distribution of bacterial cells, J. Bacteriol., 109 (1972), pp. 1301–1303.

[93] R. Sudarsan, S. Ghosh, J. M. Stockie, and H. J. Eberl, Simulating biofilm deformation and detachment with the immersed boundary method, Communications in Computational Physics, 19 (2016), pp. 682–732.

[94] I. Szapudi and A. S. Szalay, A new class of estimators for the n-point correlations, The Astrophysical Journal Letters, 494 (1998), p. L41.

[95] G. Tierra, J. P. Pavissich, R. Nerenberg, Z. Xu, and M. S. Alber, Multicomponent model of deformation and detachment of a biofilm under fluid flow, Journal of The Royal Society Interface, 12 (2015), pp. 20150045–20150045.

[96] S. Torquato, Random heterogeneous materials: microstructure and macroscopic properties, vol. 16, Springer Science & Business Media, 2013.

[97] T. M. Truskett, S. Torquato, and P. G. Debenedetti, Density fluctuations in many-body systems, Physical Review E, 58 (1998), p. 7369.

[98] G. D. Vo, E. Brindle, and J. Heys, An experimentally validated immersed boundary model of fluid–biofilm interaction, Water Science and Technology, 61 (2010), pp. 3033–3040.

[99] M. P. Wand and M. C. Jones, Comparison of smoothing parameterizations in bivariate kernel density estimation, Journal of the American Statistical Association, 88 (1993), pp. 520–528.

[100] K. Williamson and P. L. McCarty, A model of substrate utilization by bacterial films, Journal (Water Pollution Control Federation), (1976), pp. 9–24.

[101] J. K. Wróbel, R. Cortez, and L. Fauci, Modeling viscoelastic networks in stokes flow, Physics of Fluids (1994-present), 26 (2014), p. 113102.

[102] J. K. Wrobel, S. Lynch, A. Barrett, L. Fauci, and R. Cortez, Enhanced flagellar swimming through a compliant viscoelastic network in Stokes flow, Journal of Fluid Mechanics, 792 (2016), pp. 775–797.

[103] C. L. Y. Yeong and S. Torquato, Reconstructing random media, Physical Review E, 57 (1998), p. 495.

[104] T. Zhang, N. G. Cogan, and Q. Wang, Phase Field Models for Biofilms. I. Theory and One-Dimensional Simulations, SIAM Journal on Applied Mathematics, 69 (2008), pp. 641–669.

[105] ——, Phase-Field Models for Biofilms II. 2-D Numerical Simulations of Biofilm-Flow Interaction, ComCompPhys, 4 (2008), pp. 72–101.

[106] J. Zhao, Y. Shen, M. Haapasalo, Z. Wang, and Q. Wang, A 3d numerical study of antimicrobial persistence in heterogeneous multi-species biofilms, Journal of Theoretical Biology, 392 (2016), pp. 83–98.

[107] L. Zhu and C. S. Peskin, Interaction of two flapping filaments in a flowing soap film, Physics of Fluids, 15 (2003), p. 1954.

[108] J. Zhuo and R. Dillon, Using the immersed boundary method to model complex fluids-structure interaction in sperm motility, Discrete and Continuous Dynamical Systems - Series B, 15 (2010-12), pp. 343–355.

Appendix A

A.1 Bandwidth Selection for the Estimation of the Number Density

In Section 3.4, it was necessary to estimate the number density of biofilms samples. In order complete this task, we used a kernel density based estimator of the form

$$\hat{\rho}(z;b) = \frac{1}{A} \sum_{\boldsymbol{r}_i \in \Phi \cap W} \frac{k_b(z - \hat{\boldsymbol{e}}_z \cdot \boldsymbol{r}_i)}{c(z;b)}.$$

As is typical with kernel density estimation, a choice of the bandwidth, b, must be made. Several typical techniques are discussed in [99]. The general idea is to minimize the **mean integrated square error** (MISE),

$$MISE(\hat{\rho}(z;b)) = \int \left(\hat{\rho}(z;b) - \rho(z)\right)^2 dz.$$

The difficulty is of course the lack of knowledge of $\rho(z)$. Although the choice of b is partly intuitive, (e.g. too large a value leads to an overly smooth estimate, and too small a value leads to an overly jagged estimate), it is difficult to judge the best value among reasonable values of b by mere qualitative observation. Although there are numerous methods of bandwidth selection, we choose to use the Least Squares Cross-Validation (LSCV) method described in [48] as a first estimate. We also found that by visual examination values of b in the range (0.13, 0.3) seem to provide reasonable results, and thus expect any optimization method to yield a value in this range. From [48], we optimize,

$$LSCV(b) = \int_W \hat{\rho}(z;b)^2 dz - 2 \sum_{\boldsymbol{r}_i \in \Phi \cap W} \hat{\rho}(z_i;b) - k(0;b)/(Ac(z_i;b))$$

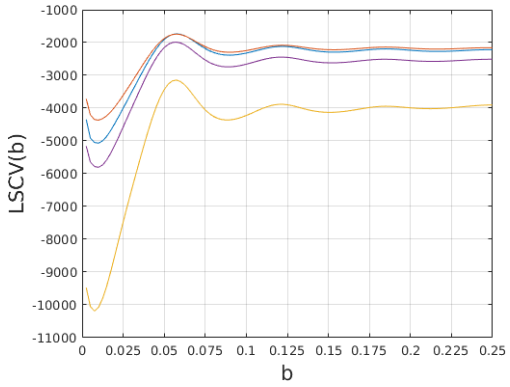


Figure A.1: The Least Squares Cross Validation value for selected values of b over four data sets.

$$= \frac{1}{A^2} \sum_{\boldsymbol{r}_i \in \Phi \cap W} \sum_{\boldsymbol{r}_j \in \Phi \cap W} \int_W \frac{k_h(z - z_i) k_h(z - z_j)}{c_b(z)^2} dz - \frac{2}{A} \sum_{\boldsymbol{r}_i \in \Phi \cap W} \sum_{\boldsymbol{r}_j \in \Phi \cap W \setminus \boldsymbol{r}_i} \frac{k_b(z_i - z_j)}{c_b(z_i)}.$$

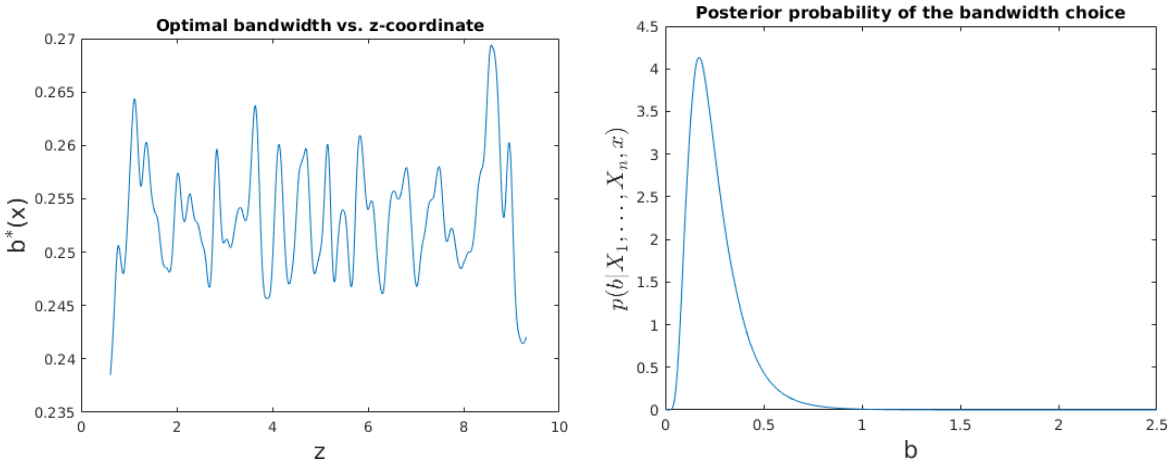
With the Epanechnikov kernel, and ignoring edge effects for simplicity,

$$I(z_i; b) \equiv \int_W k_b(z - z_i) k_b(z) dz = \begin{cases} \frac{3}{160b} \left(32 - 40(z_i/b)^2 + 20|z_i/b|^3 - |z_i/b|^5\right) & |z_i| \leq 2b \\ 0 & |z_i| > 2b \end{cases}$$

$$LSCV(b) = \frac{1}{A} \left(\frac{3}{5b} \Phi(W) + \sum_{\boldsymbol{r}_i \neq \boldsymbol{r}_j} I(z_i - z_j; b) \right) - \frac{2}{A} \sum_{\boldsymbol{r}_i \neq \boldsymbol{r}_j} k_b(z_i - z_j)$$

A plot of $LSCV(b)$ versus b is shown in Figure A.1. It can be seen that there exists several minima in each case, and the question is how to choose the "best" minima. In each case, the first minima, which is also the global minimum, is clearly too small leading to density estimates with unacceptably high variance. We found that the shallow local minimum at $b \approx 0.21$ worked best in practice. We additionally implemented the log-likelihood estimator described in [28] and found very similar results. It is unclear if there exists a method that possess a unique minimum in our case. There is also some evidence that spurious local minimizers of LSCV functionals tend to occur at smaller values than the optimal b, thus choosing the largest local minimizer seems a suitable strategy [51].

To augment the LSCV estimation, we use the fact that over a finite domain, the number density is a multiple of a probability density function, and adapt the Bayesian estimation technique discussed in [45]. To estimate the bandwidth, they choose a prior distribution $\pi(b)$, and compute



(a) (b)

Figure A.2: Results of the bandwidth estimation technique [45] applied to number density estimation. On the left, the variability of $b(z)$ over the domain. On the right, the posterior distribution for b for a particular value of z.

the posterior distribution,

$$\pi(b|\{z_1,\ldots,z_n\},z) = \frac{\hat{\rho}_b(z)\pi(b)}{\int_{b=0}^{\infty}\hat{\rho}_{b'}(z)\pi(b')db'}.$$

This estimator has the property of being optimized for each particular value of z over which we evaluate $\hat{\rho}_b(z)$. To determine b,

$$\mathbb{E}\left[b|\{z_1\ldots z_n\},z\right] = \int_0^{\infty} b\pi\left(b|\{z_1,\ldots,z_n\},z\right)db$$

is computed. An important ambiguity in this method is the choice of prior, $\pi(b)$. We choose to use a log-normal distribution of the form

$$\pi(b) = \frac{1}{b\sigma\sqrt{2\pi}}e^{-(\log b-\mu)^2/(2\sigma^2)}$$

with $\sigma = 0.5$, and $\mu = -1.5$. Although these choices are clearly somewhat arbitrary, they yield a distribution $\pi(h)$ that has mass where we expect b to be located and tends towards zero rapidly away from that region.

We found that, although the Bayesian estimate seems to do a good job of adapting b for various choices of x, it has a strong tendency to oversmooth if some sort of informed prior estimate for b is absent.

A.2 Bandwidth Selection for Estimation of the Pair Correlation Function

Similar to the estimation of a number density, the estimation of a pair correlation function bandwidth has an important effect upon the resulting estimator. From a theoretical perspective, it is easiest to analyze estimators of the form

$$\hat{g}(r) = \frac{1}{4\pi r^2 \bar{\gamma}_W(r)\rho^2} \sum_{\boldsymbol{r}_i \in \Phi \cap W} \sum_{\boldsymbol{r}_j \in \Phi \cap W \setminus \{\boldsymbol{r}_i\}} k_b(r - |\boldsymbol{r}_i - \boldsymbol{r}_j|)$$

where $\bar{\gamma}_W(r)$ is known as the **isotropized set covariance** [91, 90]. It can be computed as the following integral over $\boldsymbol{t} \in \left\{\boldsymbol{s} \in \mathbb{R}^3 |\ |\boldsymbol{s}| = r\right\}$,

$$\bar{\gamma}_W(r) = \frac{1}{4\pi r^2} \int \nu(W \cap W_{\boldsymbol{t}}) d\boldsymbol{t}.$$

Although generally intractable to integrate analytically, for boxes, the integral can be computed:

$$\begin{aligned}
\bar{\gamma}_W(r) =& \frac{1}{4\pi} \int\int \nu(\mathrm{W} \cap W_{\boldsymbol{t}=(r,\theta,\phi)}) \sin\theta \mathrm{d}\phi \mathrm{d}\theta \\
=& \frac{2}{\pi} \int_0^{\pi/2} \int_0^{\pi/2} (l - r\sin\phi\cos\theta)(w - r\sin\phi\sin\theta)(h - r\cos\phi)\sin\theta \, d\theta d\phi \\
=& hlw - \frac{1}{2}(hl + lw + wh)r + \frac{2}{3\pi}(h + l + w)r^2 - \frac{1}{4\pi}r^3
\end{aligned}$$

for $r < \min(h, l, w)$. Then, following [81] and [90]

$$\mathrm{Var}(\hat{g}(r)) \approx \frac{0.6 g(r)}{4\pi b \rho^2 r^2 \bar{\gamma}_W(r)}.$$

It is noted in [90, §5] that this approximation is particularly accurate for hard-core processes. The bias can be computed as

$$\mathrm{Bias}(\hat{g}(r)) = \left(\int k_b(r - r') g(r') dr' - g(r) \right)^2$$

Since the pair correlation function may exhibit a jump discontinuity near the hard-sphere radius, it is helpful to examine the integrated bias

$$\int \mathrm{Bias}(\hat{g}(r)) dr = \int \left(\int k_b(r - r') g(r') dr' - g(r) \right)^2 dr$$

Where $g(r)$ is twice differentiable, the bias and variance can be combined to obtain an approximation for the mean square error as a function of r and b,

$$\mathbb{E}\left[(g(r)-\hat{g}(r;b))^2\right] = \frac{0.6g(r)}{4\pi b\rho^2 r^2 \bar{\gamma}_W(r)} + \int k_b(r-r')r'^2 dr' g''(r).$$

Assuming $r > b$ since the hard-sphere diameter of bacteria is fairly large in comparison to the range over which we compute $g(r)$,

$$\mathbb{E}\left[(g(r)-\hat{g}(r;b))^2\right] = \frac{0.6g(r)}{4\pi b\rho^2 r^2 \bar{\gamma}_W(r)} + \frac{1}{5}(b^2+r^2)g''(r). \tag{A.1}$$

Equation (A.1) can be minimized for each value of r to yield an analogous result to those typically derived in the case of probability density estimators (c.f. [75, 83]),

$$b(r) = \left(\frac{3}{8}\frac{g(r)}{\rho^2 r^2 \bar{\gamma}_W(r) g''(r)}\right)^{1/3}.$$

As is the case with density estimation, this expression is of limited usefulness since it depends on the unknown quantities, $g(r)$ and $g''(r)$. However, bandwidth selection methods, such as **least squares cross validation** (LSCV), and **biased cross validation** (BCV) [48, 99] can be applied to the integrated MSE to estimate an optimal value of b across the entire interval. For instance, the LSCV estimator for $g(r)$ can be formulated as in [47, Equation 4]. In addition, to the minimization method introduced in [47], we also employ a "binning" technique to estimate optimal values of b over disjoint portions of the overall interval of computation. The motivation for this adaptation is that the behavior of $g(r)$ is quite different near the hard-sphere radius in comparison to the asymptotic behavior for $g(r)$ as r grows. It seems sensible that different bandwidths should be applied in these different regimes. Thus, we employ LSCV functionals of the form

$$LSCV(h;[r_0,r_1]) = 4\pi\int_{r_0}^{r_1}\hat{g}(r;b)^2 r^2 dr - 2\sum_{r_0\le|\boldsymbol{r}_i-\boldsymbol{r}_j|\le r_1}^{\neq}\frac{\hat{g}^{-(\boldsymbol{r}_i,\boldsymbol{r}_j)}(|\boldsymbol{r}_i-\boldsymbol{r}_j|;h)}{\bar{\gamma}_W(|\boldsymbol{r}_i-\boldsymbol{r}_j|)\rho(\boldsymbol{r}_i)\rho(\boldsymbol{r}_j)}. \tag{A.2}$$

The symbol $\hat{g}^{-(\boldsymbol{r}_1,\boldsymbol{r}_j)}(r;h)$ indicates the computation of the pair correlation function with points $\boldsymbol{r}_i$ and $\boldsymbol{r}_j$ ignored. The expectation of the summation in Equation (A.2) is shown in [47] to converge in the limit of a large domain to

$$\int \hat{g}(r;h)g(r;h)4\pi r^2 dr.$$

Table A.1: Values of b determined through LSCV optimization are reported for intervals of r.

b	r_{min}	r_{max}
0.050	0.3	0.7
0.3700	0.7	1.6
0.2150	1.6	2.5
0.3960	2.5	5.0

One can see from this the similarity to classical LSCV estimation of bandwidth for kernel density estimation [99].

In practice, we found that the summation of $\hat{g}^{-(\boldsymbol{r}_1,\boldsymbol{r}_2)}(\boldsymbol{r}_1,\boldsymbol{r}_2)$ was too expensive, leading to extremely lengthy computations. To alleviate this issue, we approximated $\hat{g}^{-(\boldsymbol{r}_1,\boldsymbol{r}_2)}(\boldsymbol{r}_1,\boldsymbol{r}_2)$ as

$$\hat{g}^{-(\boldsymbol{r}_1,\boldsymbol{r}_2)}(|\boldsymbol{r}_{12}|) \approx \mathcal{I}\hat{g}(|\boldsymbol{r}_{12}|) - \frac{2}{4\pi|\boldsymbol{r}_{12}|^2\bar{\gamma}(|\boldsymbol{r}_{12}|)} \sum_{\boldsymbol{r}_i \neq \{\boldsymbol{r}_1,\boldsymbol{r}_2\}} (k_b(|\boldsymbol{r}_{12}| - |\boldsymbol{r}_{1i}|) + k_b(|\boldsymbol{r}_{12}| - |\boldsymbol{r}_{2i}|))$$

where $\mathcal{I}\hat{g}(\cdot)$ is the linear interpolant of $\hat{g}(\cdot)$ to some value (i.e. $|\boldsymbol{r}_{12}|$). The factor of 2 in the numerator of the second term is due arises since the terms involving $\boldsymbol{r}_{1i}$,$\boldsymbol{r}_{2i}$, $\boldsymbol{r}_{i1}$, and $\boldsymbol{r}_{i2}$ must be subtracted from $\hat{g}(r)$. However, since $|\boldsymbol{r}_{ji}| = |\boldsymbol{r}_{ij}|$, there are only $\Phi(W)$ unique terms in the sum. With this fix, once $\hat{g}(r)$ is known, $\hat{g}^{-(\boldsymbol{r}_1,\boldsymbol{r}_2)}(|\boldsymbol{r}_{12}|)$ can be approximated in $\mathcal{O}(\Phi(W))$ computations as opposed to $\mathcal{O}(\Phi(W)^2)$ computations. As long as $\hat{g}(\cdot)$ is initially computed on a sufficiently dense set of points, the interpolant will be quite accurate. The resulting optimal values of b for different ranges of r are shown in Table A.1. In practice, the value of $b(r)$ is assumed to be piecewise linear in r taking on the reported value in A.1 at the midpoint of each interval. This prevents artificial discontinuous at the end points of the intervals over which b was estimated. Of course, if the estimator is accurate, one would expect such discontinuities to be small. Indeed, when $b(r)$ is a piecewise constant, discontinuities are difficult to discern by sight. We also use numerical integration to compute the relevant integrals in Equation (A.2).

One final issue with LSCV bandwidth selection is the presence of multiple minima. For the pair correlation function LSCV functional, spurious minima near $b = 0$ were observed. For the probability density estimation, it has been suggested that spurious minima are usually less than the ideal bandwidth [51]. Thus, when multiple minima are present, we choose the minima that is

largest over the range of values we consider for b.

An alternative approach to variable bandwidth estimation is use techniques such as that introduced in [1]. We believe such an approach would be effective as well.

A.3 Inhomogeneity of the Direct Pair Correlation Function in Nonstationary Processes

The motivation for using a transversely anisotropic pair correlation and direct correlation functions was attributed to properties of the Ornstein-Zernike equation. In this section we demonstrate why the Ornstein-Zernike equation implies a loss of translation invariance in the pair correlation function and direct correlation function when the density is variable.

Consider the inhomogeneous Ornstein-Zernike equation as shown in Equation (3.8). Let the two pairs of points $\{\boldsymbol{r}_1, \boldsymbol{r}_2\}$ and $\{\boldsymbol{r}_1', \boldsymbol{r}_2'\}$ satisfy $\boldsymbol{r}_1 - \boldsymbol{r}_2 = \boldsymbol{r}_1' - \boldsymbol{r}_2' = \delta\boldsymbol{r}$, and assume that the pair correlation and direct correlation functions are translation invariant. Then, using the transformation $\boldsymbol{r}_1' = \boldsymbol{r}_1 + \boldsymbol{x}$, $\boldsymbol{r}_2' = \boldsymbol{r}_2 + \boldsymbol{x}$,

$$\begin{aligned} h(\boldsymbol{r}_1 - \boldsymbol{r}_2) =& c(\boldsymbol{r}_1 - \boldsymbol{r}_2) + \int \rho(\boldsymbol{r}_3) c(\boldsymbol{r}_1 - \boldsymbol{r}_3) h(\boldsymbol{r}_2 - \boldsymbol{r}_3) d\boldsymbol{r}_3 \\ - \; h(\boldsymbol{r}_1' - \boldsymbol{r}_2') =& c(\boldsymbol{r}_1' - \boldsymbol{r}_2') + \int \rho(\boldsymbol{r}_3) c(\boldsymbol{r}_1' - \boldsymbol{r}_3) h(\boldsymbol{r}_2' - \boldsymbol{r}_3) d\boldsymbol{r}_3 \\ \hline 0 =& \int \rho(\boldsymbol{r}_3) c(\boldsymbol{r}_1 - \boldsymbol{r}_3) h(\boldsymbol{r}_2 - \boldsymbol{r}_3) d\boldsymbol{r}_3 - \int \rho(\boldsymbol{r}_3) c(\boldsymbol{r}_1' - \boldsymbol{r}_3) h(\boldsymbol{r}_2' - \boldsymbol{r}_3) d\boldsymbol{r}_3 \end{aligned}$$

In order for this to be true for all translations, $\boldsymbol{x}$, it must be the case that $\rho(\boldsymbol{r}_3) = const$ almost everywhere. Thus, for a smoothly varying number density, it cannot be the case that $\boldsymbol{c}(\boldsymbol{r}_1, \boldsymbol{r}_2)$ and $h(\boldsymbol{r}_1, \boldsymbol{r}_2)$ are both simultaneously translation invariant.