Brownian dynamics simulation of soft matter with hydrodynamics: methods for constrained systems and shear processing of 2D materials

by

William Tian Funkenbusch

BS, Chemical Engineering, University of Rochester (2019)

Submitted to the Department of Chemical Engineering
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY IN CHEMICAL ENGINEERING

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2024

Authored by: William Tian Funkenbusch
Chemical Engineering
July 25, 2024

Certified by: James W. Swan
Associate Professor of Chemical Engineering, Thesis Supervisor

Certified by: Patrick S. Doyle
Robert T. Haslam (1911) Professor of Chemical Engineering, Thesis Supervisor

Accepted by: Hadley D. Sikes
Willard Henry Dow Professor of Chemical Engineering
Chairman, Department Committee on Graduate Theses

Brownian dynamics simulation of soft matter with hydrodynamics: methods for constrained systems and shear processing of 2D materials

by

William Tian Funkenbusch

Submitted to the Department of Chemical Engineering on July 25, 2024,
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy in Chemical Engineering

Abstract

2D materials are a rising class of soft matter material with a promising set of unique characteristics. The most ubiquitous 2D material, graphene, for example, possesses large surface areas, tunability, and unique electrical, optical, and catalytic properties while being lightweight, strong, and flexible. This has led to graphene seeing use in separations, biomedical applications, flexible electronics, and more. Meanwhile, synthetic 2D polymers, a relatively new field of study, represent a massive expansion of the design space for 2D materials and their applications. Solution processing of these materials is often an important step for synthesizing or applying them, necessitating knowledge of their behavior in suspensions and in flows. As these materials become more viable, our fundamental understanding of them must increase in tandem. This will inform us in designing these materials for our desired applications. However, especially when compared to their 1D counterparts, our understanding of 2D materials is lacking. It is the goal of this thesis to help fill this gap in knowledge.

In Chapter 1, we discuss the basics of soft matter and methods for simulating them which are the basis for understanding the work in this thesis. We present and discuss the governing equations for the movement of soft matter particles. We then discuss the simulation methodology and mobility tensor approximations used in this thesis along with some additional considerations.

In Chapter 2 we study methods for simulating constrained Brownian systems. We compare the current state-of-the-art method for these simulations, GMRES, to a different method called the projected conjugate gradient (PrCG) method. In particular, we compare PrCG and GMRES for rigid bodies, freely jointed chains, and immobile systems. We find that both methods exhibit the same linear computational complexity. We find that PrCG, however, exhibits some notable advantages over GMRES including lower precomputational and storage burdens, a guaranteed feasible iterate, and trivial extension to new constraint types due to the lack of a preconditioner. We use PrCG to solve a mixed constraint problem with rigid body and immobile particles, comparing to the analytical solution at large separations.

The remainder of this thesis studies the effects of self-attraction on self-avoiding, semi-flexible, athermal 2D materials (sheets) in shear flow. In Chapter 3, we give a background on rheology and 2D materials which are necessary for understanding the remaining chapters. We begin by discussing non-Newtonian fluids, specifically their applications and affect on the momentum balance presented in Chapter 1. Then, we give a brief introduction on simple shear and discuss how it is implemented in simulations. Finally, we give a brief introduction to 2D materials, their applications, as well as

previous experimental, theoretical, and computational work.

In Chapter 4, we model self-interacting, self-avoiding, semi-flexible, athermal sheets in shear flow. We find a rich conformational landscape of 4 different behaviors — flat, tumbling, 1D folded, and 2D folded — which are well-delineated by several dimensionless groups representing the ratios between shear strength and interaction strength, and bending rigidity and interaction strength. We use these dimensionless groups to explain the observed behaviors, explain the folding behavior of 1D folded sheets, and calculate and explain the viscosity of a dilute suspension of these sheets. We use the conformational and rotational properties of the sheet simulations to explain this behavior, demonstrating a new explanation for the non-monotonic rheological properties of 2D materials which does not involve sheet-sheet interactions (which are rare in dilute suspensions) or thermal energy (which is often small in sheet systems). We also study systems with two initially stacked sheets in order to model, for example, shear exfoliation of 2D materials. We find three behaviors — separating, waltzing, and flipping — which are characterized by the same dimensionless groups as single sheets. We again explain these behaviors and calculate the viscosity of these sheets which again shows interesting non-monotonic rheological properties which we also explain using the conformational and rotational properties of the sheets.

In Chapter 5, we use simple time-dependent flow protocols to show how the properties of sheets can be controlled. Specifically, we use linear shear annealing simulations to show that the final conformational properties of a sheet suspension can be tuned continuously by varying the quench time. We also use our knowledge of the phase map of sheets to design flow protocols with step changes in shear rate to produce a target state of highly aligned, 1D folded sheets which represents, among other things, our predicted lowest possible viscosity for a sheet suspension.

In Chapter 6, we discuss potential future directions for the sheet model applied in Chapters 4 and 5. Specifically, we discuss loose ends from Chapter 4 and potential extensions of the model. We discuss potential benefits of and complications in exploring these directions.

Finally, in Chapter 7, we summarize the discoveries presented in this thesis and provide concluding remarks.

Thesis Supervisor: James W. Swan
Title: Associate Professor of Chemical Engineering

Thesis Supervisor: Patrick S. Doyle
Title: Robert T. Haslam (1911) Professor of Chemical Engineering

Professor Patrick S. Doyle has certified this thesis on behalf of
the late Professor James W. Swan (1982 - 2021).

Dedicated to James W. Swan, mentor, role model, and friend,
who passed away in November 2021.

Acknowledgements

When people hear that I go to MIT, they say all sorts of different things, often referencing how smart I must be. Like many MIT students, I've been told those sorts of things for my entire life. But even as a kid, no older than ten, I hated being told these things. I never asked to be "special" and, especially when I was young, I felt that the things that made me special were the same things that caused me to feel isolated and lonely. I wanted to be "normal" more than anything. Of course, my perspective on things has changed in the sixteen plus years since then. I think far less now about the distinction between normal and special and where I lie in that spectrum. Instead, I'll just be me. The people who I have the privilege of calling my friends these past five years are not the people I consider the smartest or coolest or nicest, but the people I feel like I can be myself around, who will accept me for who I am. I will doubtless miss many amazing people in the individual acknowledgements to follow, so I'll say this. Whether you are one of my closest friends or just a passing acquaintance, whether I have known you for a few months or longer than I can remember, thank you for being you.

The following are in no particular order.

Pat. My transition into your lab was bound to be difficult, but you made it as smooth as I could have hoped. You quickly accommodated me into your busy schedule and helped me with research, teaching, finishing my PhD on schedule, and planning my future. Thank you for all of your kindness.

Greg, Gareth, Gwen, Martin, Joey, Brandon, Javit, Jim, Janet, Kathleen, Matt, Melanie, Katie, Jennifer. To the many MIT faculty and staff members who have been an integral part of my PhD experience, from research to teaching to my other MIT-related endeavors. There is no part of the PhD experience which is easy, but you all made it feel much less difficult.

Griffin, Blake, Lucas, Aristotle, Bhavish. You have been my friends since the very beginning of my PhD (well, most of you). Your friendship has been one of the consistencies in my life that I've come to rely on. To many more sessions of DnD and trips to DTD, to many more movie viewings and much more bird watching, to board game nights, good pizza, bad puns, and endless bits.

Amanda, Shubhro, Jim, Rob, Kaitlyn, Gwyn, Noa, Erin, Ani, Kabir, Bridget. I'm not entirely sure when I became a part of this group, but I sure am glad I am. Thanks for bringing me along on your various adventures — ski trips, hiking, concerts, beer olympics, and otherwise just having a good time.

Jason, Areeb, Sagar, Sam, Susan, Hongru, Andrew. Many people had a difficult time during the pandemic, and I was no exception. In a time where in-person interactions were rare, having a group of friends online that I could spend time with was incredibly valuable for my mental health. I'm so grateful that even now you continue to be some of my closest friends.

Devashish, Adam, K'yal, Katie and Derek, Leslie, Daniel, Eric, Kat, Matt, Debbie, Pradeep, Jennifer. You are just a few of the many people who have made my PhD experience better just by being around. I can always count on you to let me sit next to you at seminars or lunch, stop in the hallway for a quick chat, and overall just be the best of company.

Merjema, Hayden, Gino, Elaine, Gambit. Merjema, it feels like you've been around since the beginning. Whether I'm succeeding or making mistakes, being smart or stupid, feeling happy or sad, I can always count on you to support me. Thank you for your friendship these past nine years. It's always a blast hanging out with you, Hayden, Gino, and Elaine, whether it's tennis, escape rooms, Overcooked, or just a simple dinner. Take good care of Gambit, he deserves the world.

Yakub, Shivani, Anna, Haley, Mandy. I need to do a better job keeping up with you all. It's always great catching up and watching you do such amazing things as you go along your own paths in life. Let's continue to keep in touch as we all enter new, exciting chapters in our lives.

Dan and Dylan. You two didn't have it easy begin accepted into the family, but I hope you know it's because we care about MayLin and LiLu so much. As long as you make them happy, I have no complaints. Thank you Dan for always answering my call when I have questions about things that most people would think an engineer would know how to do. Thank you Dylan for being one of the only industry voices in this family who can give me advice from a different perspective.

George. Sometimes, I feel like a bad big brother. Even though you are three years younger than me, I find I'm often the one looking up to you. You are always blazing a new path in life in a way that I admire so much, and I'm not just saying that because you didn't choose chemical engineering. It's been such a joy watching you grow into such an amazing adult. While we and our relationship have changed so much over the years, you have always been the best brother anyone could ever ask for. Love you, George.

MayLin. As my fellow middle sibling constantly vying for attention, we've butt heads more times than I can remember. You've always been strong, confident, and unapologetic, traits that I've often been jealous of. You're always pushing me outside of my comfort zone into doing things that I'm reluctant to do, but which are probably best for me. That's a force I've definitely needed in my life. Love you, MayLin.

LiLu. I don't envy the role of the eldest in a family of six. You always had to be the first in the family to reach a new milestone and deal with the growing pains associated with that. All the while, you kept looking back, being a role model for us to follow and a gentle hand to help guide us forward and catch us when we fall. I know I can always give you a call to ask for advice, tell you about my day, or just complain about anything, big or small. Love you, LiLu.

Mom. You have always taken such good care of me. When I'm home, you always bring me way too much food to make sure I'm eating enough. When I head back to Boston, you make sure I have enough paper towels, toilet paper, hand sanitizer, instant ramen, and more (they have these things in Boston, I promise!). Every Sunday, you make sure to text the family to make sure we are all doing well. I don't always appreciate these things as much as I should, but I do appreciate them. Love you, Mom.

Dad. It's hard to overstate the impact you've had on my life. I still remember our car rides to my various extracurricular activities, listening to various educational podcasts. I felt like I could ask you any question because, if you didn't know the answer, you would just say so and we would try to find it together. Even now I feel like I can just call you and ask you anything. Whenever I was nervous about a piece of writing, be it an essay or email, you would always read it and give me

comments. More than all of that, though, you've always made sure I know that you will love me, no matter what. Me too. Love you, Dad.

The Doyle Lab. Having the change labs was an extremely difficult experience, especially once the Swan Lab lost its lab space. It means so much that on the other side, you all were there with open arms and smiling faces to welcome me into the Doyle Lab. From group dinners to getting distracted in lab to the infamous one slide group meetings, you made me feel not only welcome, but like I had been part of the lab from the beginning.

Sam. As the oldest member of the Swan Lab when I first joined, I was always intimidated by you. You spoke with amazing confidence and clarity in group meetings and were a constant, calming presence in lab. I still remember attending your virtual defense, the first one I ever attended, and bawling during your acknowledgements.

Kevin. You were an entity I had heard about long before I came to MIT, from my sister who gushed endlessly about how helpful you were to her. She was right. From the very beginning of my PhD, you have been extremely helpful both in research and as a friend. I am so grateful for your friendship and your kind spirit, both of which have continued far past your graduation.

Emily. You have always been such a caring person to me. You always ask how I am doing and are constantly looking for ways to support me in lab and on a personal level. Thank you especially for continuing to keep the lab together by organizing group meet-ups like the Cinco de Mayo celebration.

Kyle. In research, you seem to make incredible, elegant work appear out of seemingly nowhere. I feel like our friendship also formed like this. I can't really pinpoint the moment we went from an awkward conversation talking about the lab over CAVA to where we are now. Thanks for teaching me about baseball, going on so many adventures in Boston with me, and not holding my hidden music tastes too hard against me. Go Birds!

Kelsey. You're another person I was extremely intimidated by when I first met you, talking about the lab by a building 66 window. You have this air about you that made me think that you have everything figured out, despite being only a year my senior. It's been fun peeling back some of those layers and seeing some of the mess and the silliness underneath. You are so thoughtful and caring towards all of your friends and I'm glad to count myself among them.

Eliza. You are the kind of person that seems to make everyone's day better just by being in it. When we're discussing geometry, cooking metaphors, our favorite Twilight quotes, or just life, in lab or outside 66-110 before a seminar, you never fail to bring a smile to my face.

Nick. I've always been a little jealous of you. The energy you bring with you seems to emanate outwards and make others excited as well. It's a skill not many people have, and it's great to see you utilizing it by being such an active part of so many communities — both of your labs, the greater MIT community, and your many friends. You always make me want to go out and do something good for those around me.

Madhu. Wow is there so much to say about you. You are experienced in so much more of life than

most people, and yet you are so humble, always trying to bring up and encourage those around you. I know that any time you come into the office, I won't be getting any work done that day because we can just talk for hours and hours about basically anything. You make Simona and Iris so happy and it's been so amazing seeing you become such a great father. You are so curious about the world and attentive to what's being said, traits that will make you an amazing professor. But most of all, you are just such a kind friend. I'm glad you're an honorary member of the fifth years.

Abby. We've been through a lot together. Although we took very different paths after coming to MIT, I'm glad that you were there for every moment of mine. Thanks for inviting me to all of your events, Super Bowl watch parties and group baking sessions. Thanks for being my eternal climbing buddy, even though you far outclass me in skill now. Thanks for Honeycomb Creamery and the conversations we have on our walks over. I look forward to much more of the same, great you.

Mary. You were a defining presence in my PhD from the very beginning and until the very end. I've already made a non-exhaustive list of our various different activities in the congratulations letter for your own defense, so I won't make another one here, but you know very well how long that list is. From our first moments trying to integrate ourselves into the lab while both being awkward disasters, to our last moments in the office cleaning everything out, to our continued correspondence at our weekly lunches, you've been there for it all, and I glad it was you. I'm so glad we ended up joining the same lab and becoming friends and I hope we'll continue to be friends many, many years from now.

Laura and Henry. You two are incredible. Laura, it has been such a joy getting to know you during my PhD. Thank you for the dinner parties, board games, reminiscing, and for being a great new addition to our climbing crew. Henry, you're an awesome kid. It's hard to believe how much you've grown since I first met you, and I can't wait to see how much more you grow from here on out. I've enjoyed our talks on DnD, your custom Minecraft mods, calculus, and the like. I hope we keep getting to hang out in the future.

Jim. When people asked me how MIT was going, I would always say pretty much the same thing. MIT is hard. Sometimes things really suck. But, I have the best advisor anyone could ask for. You were a shining beacon in the scientific community and your local community, for your lab, your students, and everyone you spoke to. You were a brilliant researcher, a fantastic teacher, but more importantly, just a genuinely kind and caring soul. You have shaped the way that I do research, think about teaching, and conduct myself in my day-to-day life. Most significantly for me, your unwavering belief in me, slowly but surely, filled me with confidence and made me want to believe in others as well. You've been such an invaluable role model to me and I hope to help spread some of the joy, curiosity, and compassion you brought to the world.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In this chapter, we discuss the basics of soft matter and soft matter simulations which will be needed to understand the rest of this thesis. We present and discuss the governing equations for the movement of soft matter particles. We then present the simulation methodology and mobility tensor approximation used in this thesis along with some additional considerations.

1.1 Introduction to soft matter

This thesis deals with soft matter, which is a particular class of material that deals with, well, soft things. If you are already familiar with soft matter, its equations of motion, and simulations with hydrodynamics, you can move on past the introduction without losing much. If not, this will serve as an introduction to soft matter simulations which will be needed to understand the work later in this thesis. I will state a more formal definition for soft matter a little later, but for now I'll start with a few examples. Food like milk and jello, care products like shampoo and creams, polymers which compose many of the things around you at this very moment, even the cells in your body, are all classified as soft matter. There are many more examples, but the point is just that soft matter is ubiquitous in our everyday lives. But to understand what soft matter really is and why behaves the way it does, it isn't enough to look at them with our naked eyes. We have to zoom in.

If we look closely at soft matter, we see that it is really composed of many small pieces immersed

in a fluid[a]. These pieces are larger than individual atoms, but much smaller than what we can perceive normally, on the order of nanometers to micrometers[b]. This is really small, these materials are typically tens, hundreds, or even thousands of times smaller than the width of a single human hair. If we want to understand how soft materials behave in our macroscopic world, it's necessary to understand what's going on in their microscopic world.

Our goal is to understand how particles[c] like this move around so that we can simulate them. Simulating them allows us to do many things. First, it informs understanding. If our simulations can be trusted, it allows us to peer in and see physics that might otherwise be very difficult to measure experimentally. Second, it saves resources, both physical and temporal. We can quickly and freely vary different parameters in our system without needing to go through possibly painstaking synthesis[d]. Third, it allows us to do things we couldn't otherwise do with real materials. For example, we could turn off a particular piece of physics and see how the system behaves. If the behavior is drastically different it means that this piece of physics is relevant for our system and we should pay attention to it in the future. In order to run simulations, we'll first need to gain an understanding of the forces that these particles undergo.

1.1.1 Drag forces as a function of particle size

Perhaps the most significant difference between the microscopic world and our world is the way we experience drag. When we are running or swimming through a fluid like air or water, we feel it push back against us. Physically, this is because in order to move to a new location, we need to displace the fluid which previously occupied it. The drag force that we feel is proportional to the amount, specifically the mass, of fluid that we displace per unit time, $\dot{m}$. To displace this mass, we need to accelerate it from rest (the fluid molecules are moving, but on average they are stationary)

[a] Fluids which consist of multiple components like this are called complex fluids. The particular case of a solid suspended in a fluid is called a suspension.

[b] "On the order of" means about the same order of magnitude (factor of 10). If two numbers are within about a factor of 3, they are the same order of magnitude. Mathematically, if L is the characteristic size of these materials, we could say $L \sim \mathcal{O}(\text{nm} - \mu\text{m})$. This is called "Big O" notation.

[c] By particle, we just mean some element in the fluid. This is a pretty broad definition. It could refer to an individual atom or molecule, but also can represent larger collections of these things. In essence, particles in simulations represent the smallest "thing" we are interested in observing. We will go over examples throughout this thesis.

[d] If our simulations are faster than the time it takes to run experiments.

to the velocity that we are moving, u. For an object with characteristic size L ($L \sim \mathcal{O}(1\text{ m})$ for humans), $\dot{m} \sim \rho L^2 u$[e], where ρ is the fluid density and L^2 is proportional to our cross-sectional area perpendicular to the direction of movement. Thus, the drag force we feel is $F_{DI} \sim \rho L^2 u^2$, and we expect smaller things to experience a much smaller drag force compared to larger things.

But this is not the only drag force that we experience. Described above is the *inertial* drag and is what most people think of when they think of drag. The other type of drag is called *viscous* drag. Viscous drag comes from the fact that, when we move, a small layer of our surrounding fluid comes along with us. This fluid layer shears with the fluid surrounding it, that is, slides along it, causing friction[f]. This friction is the origin of viscous drag. The equation for this drag was derived for a sphere by Stokes nearly two centuries ago[1]:

$$\mathbf{F}_{Dv} = -6\pi\eta a\mathbf{u}, \tag{1.1}$$

where a is the sphere's radius, η is the fluid viscosity, and the negative sign indicates that the force opposes movement. We'll remove the coefficient for now and just discuss how the magnitude of this drag changes (for an object of any shape) with different parameters: $F_{Dv} \sim \eta L u$, where we've replaced the sphere radius with L, the characteristic length of the object. Notably, when we halve our size or velocity, the viscous drag only halves, unlike the inertial drag, which decreases by a factor of 4. This means that the smaller we get, the larger viscous drag becomes relative to inertial drag.

We quantify the relative strengths of inertial and viscous drag by taking the ratio between the two quantities to obtain a dimensionless number called the Reynolds number:

$$\mathrm{Re} \equiv \frac{\rho L u}{\eta}. \tag{1.2}$$

[e]We would say this as *$\dot{m}$ goes as $\rho L^2 u$*. This is not an equality, but instead means that we expect that, under certain limits, if we change the right-hand-side by a specific factor, we expect the left-hand-side to change by the about the same factor. For example, here we expect that doubling L will quadruple $\dot{m}$.

[f]This layer doesn't really have a well-defined thickness. Rather, the closer the fluid is to the surface of the object, the more strongly it is dragged along. The further the fluid is from the surface of the object, the more its behavior resembles the bulk fluid. The velocity of the fluid changes continuously from the surface of the object to the bulk fluid and friction occurs due to this change in velocity. The larger the viscosity, the larger the friction, meaning a sharper change in velocity.

When the Reynolds number is large, inertial forces dominate. When we are swimming, our Reynolds number is about 10^6, so inertial forces are about a million times stronger than viscous forces. In other words, for us, we hardly feel the effects of viscous forces. For a soft matter particle however, which is a million times smaller and moves a hundred thousand times slower, the Reynolds number is about 10^{-5}. These particles hardly feel any inertial forces relative to their viscous forces. This means that we can approximate, with a high degree of accuracy, the drag they feel using just the viscous drag:

$$\mathbf{F}_D = \mathbf{F}_{Dv} + \mathbf{F}_{DI} \approx \mathbf{F}_{Dv}. \tag{1.3}$$

This will have very important consequences for us in a little bit.

1.1.2 Stokes equations and their implications

The Reynolds number can be derived more formally using the equations of motion for the fluid, which are obtained through a momentum balance. If we assume our fluid is incompressible (constant density) and is Newtonian (constant viscosity)[g], then the momentum balance reduces to the incompressible Navier-Stokes equation:

$$\rho\left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}\cdot\boldsymbol{\nabla})\mathbf{u}\right) \equiv \rho\frac{D\mathbf{u}}{Dt} = \eta\nabla^2\mathbf{u} - \boldsymbol{\nabla}p + \mathbf{f}_b. \tag{1.4}$$

The units for each term of this equation are a force per unit volume. $\frac{D\mathbf{u}}{Dt}$ gives the rate of change of the velocity for a tracer particle following the flow and its term in the above equation is a measure of the *inertia* of the system. This derivative is called by several names such as the material derivative or convective derivative. $\eta\nabla^2\mathbf{u}$ gives the *diffusion* or *dispersion* of momentum. The higher the viscosity, the faster momentum is dispersed. $\boldsymbol{\nabla}p$ is the pressure gradient and gives the *driving force* for flow. $\mathbf{f}_b$ captures all the other external forces on the fluid (e.g. gravity). These are 3 equations (one for each velocity component) with 4 unknowns (the 3 velocity components and the pressure), so to complete them we add a mass balance with the continuity equation, which for an

[g]This is usually a good assumption for our systems. Newtonian here specifically refers to a constant viscosity with respect to the strain rate, which is a measure of the rate at which the fluid is being deformed. We go into more detail in Chapter 3.

incompressible fluid is

$$\boldsymbol{\nabla} \cdot \mathbf{u} = 0. \tag{1.5}$$

The Reynolds number can be derived through *scaling analysis* of these equation. Scaling analysis involves finding how the size of each term changes ("scales") with the different system parameters. Scaling analysis on the Navier-Stokes equation shows that the relative size of the left-hand-side (the inertial term) to $\eta\nabla^2\mathbf{u}$ (the viscous term) is given by the Reynolds number. When the Reynolds number is small (such as for soft matter particles), the inertial term is small compared to the viscous term and can be neglected. This gives us the Stokes equations:

$$\mathbf{0} = \eta\nabla^2\mathbf{u} - \boldsymbol{\nabla}p + \mathbf{f}_b. \tag{1.6}$$

These equations are the governing equations for the fluid our soft matter particles exist in and determine the forces that our particles experience. As a notable example, these equations are what Stokes used to derive the viscous drag on a sphere.

These equations have some very interesting implications for soft matter systems. Because there is no time-dependence to these equations, the fluid responds *instantaneously* to changes in system parameters (boundary conditions, applied forces, etc.). This means that, in order to understand what a system will do, we only need to understand its current state, not anything about its history. If we apply a change to the system, it doesn't matter if we do so quickly or slowly, the resulting flow will be the same during each point in the change. These equations are also *time-reversible*. This means that, if we apply a change, then apply the same change in reverse (at any rate), the net result is no change. This has the interesting consequence that any reciprocal motion of a particle will not allow it to swim. The quintessential example is the scallop, which can swim in our world by slowly opening its shell then closing it quickly, expelling water. In a low Reynolds number system, the scallop would move back to its original location in the process of opening its shell again, no matter how slowly it does it.

For those interested, EM Purcell gave a great, high-level lecture on low Reynolds number flows

which has since been transcribed[2].

1.1.3 Diffusion of a single soft matter particle

There is one more thing we should understand about the fluid before we try to understand the way a small particle moves in it. Particles immersed in a fluid experience random motion called Brownian motion. This motion is due to collisions with fluid molecules which are moving around at high speeds. The average velocity of these particles is constant, but individual particles move around at different speeds and in different directions, causing spatial and temporal variation in the velocity, and thus temperature (which is the average kinetic energy), of the fluid. This variation, called thermal fluctuations, causes Brownian motion. For human-sized things these fluctuations are so small and fast that we experience an essentially constant force from the fluid around us, called pressure. For small things, however, these forces cause significant, random movements relative to their size. It is the defining characteristic of soft matter that its constitutive particles are small enough that thermal fluctuations are relevant. We might call such a particle a Brownian particle.

The diffusivity, D, is a measure of how quickly a particle moves, or diffuses, via random motion:

$$2nD\Delta t \equiv \langle \|\mathbf{x} - \mathbf{x}_0\|^2 \rangle, \tag{1.7}$$

where n is the number of dimensions in which the particle is diffusing (in this case 3), $\mathbf{x}_0$ is the initial position, Δt is the time that has passed since the particle was at its initial position (the time lag), $\|\cdot\|$ is the Euclidean norm, and $\langle \cdot \rangle$ indicates an average over many realizations. For spherical particles with small Reynolds numbers, like soft matter particles, the diffusion coefficient is given by the Stokes-Einstein-Sutherland relation:

$$D = \frac{k_B T}{6\pi\eta a}, \tag{1.8}$$

where k_B is the Boltzmann constant, T is the temperature, and their product is the thermal energy. Because the mean squared displacement of a particle goes as its diffusivity times the time lag, a particle moving over a fixed time step Δt will move a distance proportional to the square root of the product of its diffusivity with the time step: $\Delta x \sim \sqrt{6D\Delta t}$.

1.2 Introduction to Brownian dynamics simulations with hydrodynamics

1.2.1 Deriving the equation of motion for a single particle

The equations for the Stokes (viscous) drag and Stokes-Einstein-Sutherland diffusivity were both given for spherical particles. This is because spherical particles look the same moving in any direction and it makes deriving equations easier. For this reason, we usually simulate even complex systems as collections of spheres, sometimes rods. We can then link them up in various ways, some of which we'll discuss later. For the remainder of this thesis, "particle" or "bead" will refer specifically to spherical particles, although many of the non-numerical arguments will apply to non-spherical particles as well. The radius of these particles is the hydrodynamic radius, which means that their radius, a, is defined such that they experience Stokes drag $6\pi\eta a$.

Let's consider a single particle to start in the reference frame of the bulk fluid flow[h]. As with many physics problems, we start by writing a force balance. The sum of the forces on the particle is equal to its mass times its acceleration. We'll break the force into several pieces:

$$m\mathbf{a} = \mathbf{F}_H + \mathbf{F}_B + \mathbf{F}_P, \tag{1.9}$$

where m is the particle's mass, $\mathbf{a}$ is the particle's acceleration, $\mathbf{F}_H$ is the hydrodynamic (drag) force, $\mathbf{F}_B$ is the Brownian force, and $\mathbf{F}_P$ is the potential force.

We'll go over each force separately. Because we are at small Reynolds number, the hydrodynamic force is just the viscous drag we discussed before: $\mathbf{F}_H = -6\pi\eta a\mathbf{u}$. The Stokes drag coefficient, $6\pi\eta a$, will appear a lot in these systems, so we write it instead as the *resistance* to translation: $\mathcal{R} = 6\pi\eta a$. In a resistance problem, the particle velocities are known and the resulting forces must be calculated. Often, we know the particle forces and are interested in the velocities instead. In this case, it is often more convenient to use the inverse of the resistance, the *mobility*:

[h]That is to say, our reference frame is moving at the same rate as the bulk fluid so that it appears stationary.

$$\mathcal{M} \equiv \mathcal{R}^{-1} = \frac{1}{6\pi\eta a}. \tag{1.10}$$

So,

$$\mathbf{F}_H = -\mathcal{R}\mathbf{u} = -\mathcal{M}^{-1}\mathbf{u}. \tag{1.11}$$

In a mobility problem, the particle forces are known and the resulting velocities must be calculated (via $\mathbf{u} = -\mathcal{M}\mathbf{F}_H$).

Because the number of molecular collisions is large, the central limit tells us that the Brownian force should be normally distributed. The distribution should have mean $\langle \mathbf{F}_B(t) \rangle = \mathbf{0}$ (there's no preference for a force in any direction as we're in a reference frame stationary to the bulk fluid[i]) and autocorrelation[j] $\langle \mathbf{F}_B(t)\mathbf{F}_B(0)^T \rangle = 2k_BT\mathcal{M}^{-1}\mathbf{I}\delta(t)$ (to satisfy the Stokes-Einstein-Sutherland relation and instantaneous response of the fluid under Stokes equations[k]), where $\mathbf{I}$ is the identity matrix. In practice, we sample the velocity the Brownian force imparts directly using

$$\mathbf{u}_B = \sqrt{2k_BT/\Delta t}\mathcal{M}^{1/2} \cdot d\mathbf{W}, \tag{1.12}$$

where $\mathbf{W}$ is the standard Wiener process, which produces a random variable that is normally distributed. We can relate this Brownian velocity to the force that caused it with the Stokes drag:

$$\mathbf{F}_B = \mathcal{M}^{-1}\mathbf{u}_B. \tag{1.13}$$

Finally, $\mathbf{F}_P$, the potential force, covers all the conservative forces in our system. Conservative (path-independent) forces can be written as a potential gradient: $\mathbf{F}_P(\mathbf{x}) = -\boldsymbol{\nabla} V(\mathbf{x})$, where V is

[i] Even if we were not in this reference frame, the bulk motion of the fluid would be included in a separate term.

[j] The autocorrelation function gives a measure for how a property at one time correlates with that same property at a different time. One way to think about it is that large values of the autocorrelation (in terms of magnitude) mean that we can use the value of the property at one time to predict very well its property at another time. Properties tend to change continuously, so the autocorrelation will tend to be large at nearby times and decay over time. For purely cyclic motions, the autocorrelation can be cyclic as well, although there are usually deviations that cause it to decay in this case as well.

[k] The $\delta(t)$ is the delta function, which is 0 for all inputs other than 0. Its appearance in this equation tells us that the autocorrelation depends only on the specific time, t, we are looking at. This means that Brownian forces are memoryless, that is, the force at one time does not depend on the force at another. In reality, this is true up to extremely small time lags.

the potential energy at position $\mathbf{x}$. These potentials provide much of the richness of our simulations, so common examples will constitute their own section in this chapter.

This gives us the first version of our force balance, the Langevin equation:

$$m\frac{\mathrm{d}^2\mathbf{x}}{\mathrm{d}t^2} = -\mathcal{M}^{-1}\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} + \mathcal{M}^{-1}\mathbf{u}_B - \boldsymbol{\nabla}V. \tag{1.14}$$

Scaling analysis can be used again to simplify this equation. It tells us that the relative size of the left-hand-side of the equation (the inertial term) to the right-hand-side of the equation (aside from the potential force, which could be anything depending on the system) is given by the inverse Schmidt number, which measures the rate at which the particle diffuses relative to the rate at which its momentum is dispersed (it is slowed down) by the fluid:

$$\mathrm{Sc}^{-1} \equiv \frac{mk_BT}{\eta^2 a^4} \sim \frac{\rho_p k_B T}{\eta^2 a}, \tag{1.15}$$

where ρ_p is the density of the particle. The Schmidt number is typically very large in these systems ($\mathcal{O}(10^6 - 10^8)$), so the inertial term can be neglected. Adding this assumption to Equation 1.14 results in the overdamped Langevin equation:

$$\mathbf{u} = \frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} = -\mathcal{M}\boldsymbol{\nabla}V + \mathbf{u}_B, \tag{1.16}$$

which we can use to find the particle velocities. In the simulations in this thesis, we choose a finite time step and integrate via an Euler-Maruyama scheme[3]:

$$\Delta\mathbf{x} = \Delta t(-\mathcal{M}\boldsymbol{\nabla}V + \mathbf{u}_B). \tag{1.17}$$

This equation is stochastic (random) due to the presence of the Brownian velocity term.

The velocity of the particles depends only on the forces currently being applied to the particle and not any forces applied previously. This is a direct consequence of the low Reynolds number, which causes the drag force to be linear in the velocity, and the high Schmidt number, which causes the inertial term of the Langevin equation to disappear. So, if a particle is not receiving a force at a given time, it is not moving. This is in stark contrast to our daily experience. When

a human stops their stroke while swimming in water, they glide, typically for a few feet and for a few seconds, depending on their initial speed, before coming to a stop. That is, the stopping distance for humans in water is around the size of a human and the stopping time is around our reaction time. For soft matter, on the other hand, the stopping distance is about 10^5 times smaller than the size of the particle, with a stopping time less than 10^{-6} seconds. For reference, this is equivalent to a swimming human gliding less than the width of a human hair before coming to a complete stop. This glide would take a hundred thousand times less time than it takes for us to blink.

1.2.2 Incorporating more particles

At first glance, we might think that we can add more particles very easily by just stacking the velocities of all the particles and calculating them separately. This is called the freely-draining model because it assumes that a particle will not affect the fluid flow (i.e. the fluid will freely drain through the particles). Unfortunately, this is often not a good assumption because of viscous drag. As we discussed earlier, a particle moving in a fluid drags some fluid along with it. This fluid can affect other particles, dragging them along with it. So, if we apply a force to only one particle, other nearby particles will be influenced as well; the particle velocities are coupled through the fluid. The effective interactions between particles through the fluid are called hydrodynamic interactions and are illustrated in Figure 1.1. In essence, the existence of the particle causes a disturbance in the fluid flow field which affects the local fluid flow around other particles. This means that we cannot calculate the drag on a given particle without taking into account the movement of all the other particles. Similarly, this affects the strength of the Brownian forces particles experience.

Luckily, the only change we need to make is to replace the constant mobility, $\mathcal{M}$, with the mobility tensor[1], $\boldsymbol{\mathcal{M}}$, which takes into account hydrodynamic interactions. Unluckily, this tensor isn't easy to formulate. Consider N particles in a fluid. From now on, we use $\mathbf{u} \in \mathbb{R}^{3N\times 1}$ to refer to the stacked vector of all particle velocities. Similarly, the stacked Brownian velocities are $\mathbf{u}_B \in \mathbb{R}^{3N\times 1}$. The gradient is now with respect to the 3 component position of each particle, and is thus also a $3N \times 1$ vector. So, $\boldsymbol{\mathcal{M}} \in \mathbb{R}^{3N\times 3N}$. The ij'th 3×3 box of $\boldsymbol{\mathcal{M}}$ thus gives how the

[1]We will also refer to this as the mobility matrix in this thesis.

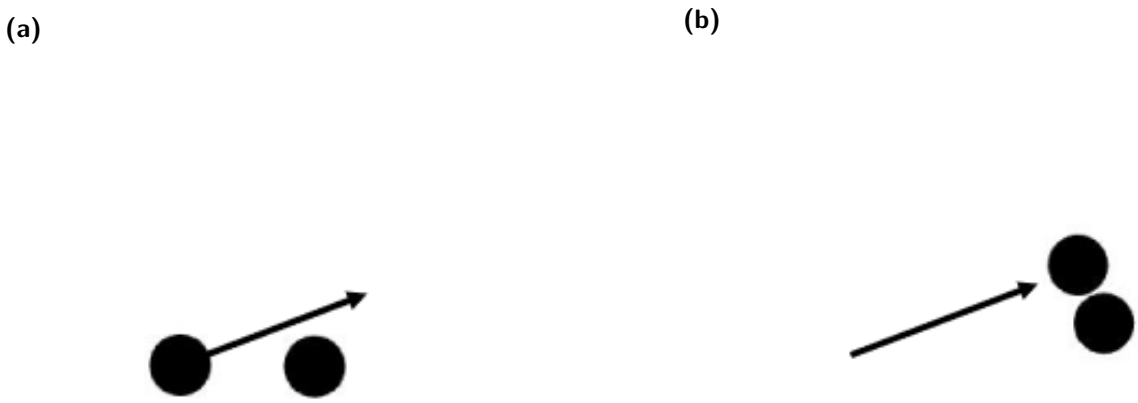

Figure 1.1 Snapshots of two particles. A force is applied up and to the right to the leftmost particle and no external force is applied to the rightmost particle. (a) Shows initial conditions and (b) shows after dragging. The arrow indicates the direction of dragging. After dragging only the leftmost particle, the rightmost particle has also moved.

forces on particle j affect the movement of particle i.

There are a few things that $\boldsymbol{\mathcal{M}}$ must satisfty. First, the diagonal elements, which correspond to the hydrodynamic interactions of a particle with itself, should correspond to the single particle diffusion:

$$\boldsymbol{\mathcal{M}}_{ii} = \frac{1}{6\pi\eta a}\mathbf{I}_3, \tag{1.18}$$

where $\boldsymbol{\mathcal{M}}_{ii}$ is the i'th 3×3 diagonal block of $\boldsymbol{\mathcal{M}}$ and $\mathbf{I}_j$ is the identity matrix of size j.

Second, $\boldsymbol{\mathcal{M}}$ must be symmetric. This is called *reciprocity* and implies that swapping particles will not affect the hydrodynamic interactions. Reciprocity is a direct result of the Lorentz reciprocal theorem for Stokes flow and is proved in Kim and Karilla[4].

Finally, $\boldsymbol{\mathcal{M}}$ must be positive-definite (i.e. all of its eigenvalues must be positive). This is due to the 2nd law of thermodynamics. If there was a negative eigenvalue of the mobility tensor, then there is a set of forces that we can apply on the particles such that their motions are in the opposite directions. This would imply that the fluid is not dissipating energy, which violates the second law. A more rigorous proof is also provided in Kim and Karilla[4].

So, the mobility tensor must be symmetric positive definite (SPD). This has a few important implications. First, it means that the mobility tensor is invertible, so we can obtain the resistance

tensor, $\boldsymbol{\mathcal{R}}$, by inverting $\boldsymbol{\mathcal{M}}$. The corresponding resistance tensor must also be SPD. Second, it means that, for any non-zero vector $\mathbf{x}$, $\mathbf{x}^T\boldsymbol{\mathcal{M}}\mathbf{x} > 0$, and equals 0 if and only if $\mathbf{x} = \mathbf{0}$. This will come in handy in the next chapter.

Groups of particles can experience something called Brownian drift, which must be added to the equations of motion[5]:

$$\mathbf{u} = -\boldsymbol{\mathcal{M}}\boldsymbol{\nabla}V + \mathbf{u}_B + k_BT\boldsymbol{\nabla}\cdot\boldsymbol{\mathcal{M}}. \tag{1.19}$$

In the case where the mobility tensor is divergence-free, as will be the case for the particular mobility tensor approximation we will use, this additional term is zero, so we will drop it in the future. Overall, an integration scheme for a simulation with many particles is

$$\Delta\mathbf{x} = \Delta t(-\boldsymbol{\mathcal{M}}\boldsymbol{\nabla}V + \mathbf{u}_B), \tag{1.20}$$

with

$$\mathbf{u}_B = \sqrt{2k_BT/\Delta t}\boldsymbol{\not{\mathcal{M}}}\cdot d\mathbf{W}, \tag{1.21}$$

where the slash here and in the rest of this thesis refers to the matrix satisfying $\boldsymbol{\not{\mathcal{M}}}\boldsymbol{\not{\mathcal{M}}}^\dagger = \boldsymbol{\mathcal{M}}$, with $\boldsymbol{\mathcal{M}}^\dagger$ indicating the adjoint (conjugate transpose) of $\boldsymbol{\mathcal{M}}$. It is a type of square root.

1.2.3 Calculating the mobility tensor

The mobility tensor (and its square root) is difficult to formulate and depends on the system's boundary conditions. A common approximation is the Rotne-Prager-Yamakawa (RPY) tensor[6], which is a form of regularized Stokeslet. Stokeslet means that it is indeed a solution to the Stokes equations and regularized means that the applied forces are spread over a small domain (to remove a singularity). The RPY tensor includes only long-range hydrodynamics interactions and therefore does not include lubrication, which sharply resists the fast approach or separation of two nearby surfaces (lubrication is short-ranged in that its strength scales with the separation between the surfaces to the minus third power, $F_{lub} \sim r^{-3}$, so it decays rapidly as the separation increases).

Notably, the RPY tensor is guaranteed to be SPD and divergence-free ($\boldsymbol{\nabla} \cdot \boldsymbol{\mathcal{M}} = \mathbf{0}$), fulfilling our necessary conditions for the mobility tensor and eliminating Brownian drift. In an unbounded domain (infinite fluid) with identical particles, the ij'th 3×3 block of the RPY tensor is

$$\boldsymbol{\mathcal{M}}_{ij}^{RPY} = \begin{cases} \frac{1}{6\pi\eta a}\mathbf{I} & \text{if } i = j \\ \frac{1}{8\pi\eta r}\left(1 + \frac{a^2}{3}\nabla^2\right)(\mathbf{I} + \hat{\mathbf{r}}\hat{\mathbf{r}}^T) & \text{if } i \neq j \text{ and } r > 2a \\ \frac{1}{6\pi\eta a}\left[\left(1 - \frac{9r}{32a}\right)\mathbf{I} + \frac{3r}{32a}\hat{\mathbf{r}}\hat{\mathbf{r}}^T\right] & \text{if } i \neq j \text{ and } r \leq 2a, \end{cases} \tag{1.22}$$

where a is the radius of the particles, r is the distance between their centers, and $\hat{\mathbf{r}}$ is the unit vector pointing from the center of the i'th particle to the center of the j'th particle[6]. There exist formulations of this tensor for different-sized particles[7] and particles which can rotate rigidly[8], but in this thesis all simulations are performed with a mono-disperse (single-size) suspension of particles with only translational degrees of freedom.

The strength of the hydrodynamic interactions between two particles decays with their distance to the first power, r^{-1}. This means that hydrodynamic interactions are long-range and thus cannot be neglected even for far away particles (roughly, this is because $\int_0^\infty 1/r \to \infty$). This means that the condition number of $\boldsymbol{\mathcal{M}}$ grows with system size. Consequently, matrix operations involving the mobility tensor become more difficult with system size, making it difficult to simulate large numbers of particles without clever calculation of the mobility tensor and its square root[9].

Often, simulations are run using periodic domains, where the simulation box is duplicated in all directions and interactions are calculated for the simulation box with all of its periodic images. This boundary condition is used to approximate the behavior of an infinite system, as often the number of particles we can simulate is much smaller than the number of particles that would exist in a real application[m]. This is especially relevant here, where the long-range nature of the hydrodynamic interactions means that they are relevant potentially for even distant images. For a cubic simulation box with side length L_{box}, the periodic RPY tensor is

[m]Because the fluid looks roughly the same everywhere, we expect the behavior in the simulation box to be representative of a much larger sample.

$$\boldsymbol{\mathcal{M}}_{ij}^{RPY} = \frac{1}{\eta L_{box}^3} \sum_{\mathbf{k}\neq\mathbf{0}} e^{i\mathbf{k}\cdot\mathbf{r}} \frac{1}{k^2} \left(\frac{\sin(ka)}{ka}\right)^2 (\mathbf{I} - \hat{\mathbf{k}}\hat{\mathbf{k}}), \tag{1.23}$$

where $\mathbf{k}$ is the set of all reciprocal lattice vectors, $k = |\mathbf{k}|$, and $\hat{\mathbf{k}} = \mathbf{k}/k$[10].

Practically, the mobility tensor is never stored (its size goes as the square of the number of particles, which is prohibitive). Instead, functions are written to compute the matrix-vector product of the mobility tensor with an arbitrary vector, $\mathbf{a}$: $\boldsymbol{\mathcal{M}}^{RPY}\mathbf{a}$. Importantly, methods such as the Force Coupling Method (FCM)[11] and Positively Split Ewald (PSE)[9,12] method have been developed for calculating this product as well as $\boldsymbol{\mathcal{M}}^{RPY}\mathbf{a}$ (for the Brownian velocities).

In the next chapter, we use the PSE method in particular for calculating mobility tensor-vector products in linear time with the number of particles. Conceptually, this algorithm works by splitting the mobility tensor into a near-field (short-ranged) and a far-field (long-ranged) component. Because the near-field component is short-ranged, it can use a cut-off radius for interactions. The far-field component, on the other hand, can take advantage of the periodicity of the system[n]. The PSE method splits the mobility tensor in a clever way such that both components are SPD. This allows for efficient calculation of both components (to a controllable error tolerance), which can be summed to produce the overall mobility tensor. The PSE method (as well as the FCM) is $\mathcal{O}(N \log N)$ time (although the $\log N$ dependence is difficult to observe in practice) and can be accelerated using GPU's to take advantage of its massively parallel structure[9,11,12]. Overall, this means that, in practice, the cost to simulate a system scales linearly with the number of particles, which is the same cost as for systems without hydrodynamics. This allows systems with many (millions) of particles to be simulated in reasonable time.

1.2.4 Exploring common conservative potentials

Conservative potentials are where many of the distinguishing details between soft matter systems is encoded in simulations. There are several common examples.

[n]Mathematically, a Fourier transform is performed to transform the far-field component to wave space, where large distances become small wave vectors whose sum converge quickly.

External fields

External potentials, such as gravity, electric fields, or magnetic fields can be implemented here. In the simplest case, the forces these fields cause are the same on each particle. For example, for gravity:

$$\mathbf{F}_g = \mathbf{g} V_p (\rho_p - \rho), \tag{1.24}$$

where g is the acceleration due to gravity and V_p is the particle volume. In many cases, soft matter particles have a similar density to water and are thus *neutrally buoyant* and will not experience a net gravitational force[o]. Other fields can be implemented in a similar manner. However, just as particles can disturb the flow field of a fluid, they can disrupt potential fields. For example, polarizable particles perturb applied electric/magnetic fields. In this case, the local electric/magnetic field a particle experiences is coupled with the locations of all of the other particles in the system, and it must be treated in a similar way to flow field perturbations with a different version of the mobility tensor. In this thesis, we do not deal with such external fields. Perturbations such as these and their applications is the topic of a thesis by Kelsey M. Reed, so those interested in learning more should look there[13].

Hard-sphere interactions

The other class of potentials are inter-particle potentials which occur between two or more particles. One of the most common inter-particle potentials is the hard-sphere potential. In many cases, a soft matter particle in a simulation with hydrodynamic radius a represents a real, roughly rigid (non-deforming) particle with radius a. In this case, particles cannot overlap. This corresponds to a potential:

$$V_{HS} = \begin{cases} 0, & r \geq a_1 + a_2 \\ \infty, & r < a_1 + a_2, \end{cases} \tag{1.25}$$

where r is the distance between two particles of hard-sphere radius a_1 and a_2 (usually but not nec-

[o]The gravitational force is balanced by the buoyant force.

essarily equal to the hydrodynamic radius). This potential is difficult to implement in simulation[p], so we must implement a "softer" hard-sphere potential. A common choice is the Heyes-Melrose hard-sphere (HMHS) potential[14]. Under this potential, non-overlapping particles again receive no force, but overlapping particles receive a force which, in the absence of other forces in the system, places the particles tangent to each other at the next time step. For monodisperse particles with hard-sphere radius a interacting under RPY hydrodynamics, the HMHS potential is[15]:

$$V_{HSper} = \begin{cases} \frac{16\pi\eta a^2}{\Delta t}(2a\ln\frac{2a}{r} + r - 2a), & r < 2a \\ 0, & r \geq 2a. \end{cases} \tag{1.26}$$

Because this potential depends only on the distance, r, between particles, it is symmetric, meaning the particles will experience the same force but in opposite directions. Thus, they do no net work on the system. This is common in inter-particle potentials and will be true for the rest of the potentials discussed in this chapter.

Soft matter particles in simulation are not necessarily hard-spheres. They may represent soft, deformable particles or a collection of units[q] which has holes and thus can overlap with other particles (with respect to their hydrodynamic radii). In this case, the potential for two overlapping particles can take many different forms (or take no form at all).

Harmonic potentials

The other most common inter-particle potential is a harmonic potential, which models a spring between two particles:

$$V_{har} = \frac{k}{2}(r - r_0)^2, \tag{1.27}$$

where k is the spring constant and r_0 is the equilibrium distance between the particles. This

[p]Remember that for the purposes of integration, we are interested in the gradient of the potential. Because the potential here is discontinuous, we run into an issue. The potential gradient is infinite right at contact and zero elsewhere. In simulations, we must take steps of finite size, so two particles are extremely unlikely to overlap exactly and be affected by the potential. If they were, the gradient would be infinite, which would cause an infinite displacement due to the finite time step.

[q]The process of taking a collection of units and turning them into a single piece is called coarse-graining. It is often useful for reducing the number of particles which need to be simulated, but great care needs to be taken such that the overall physics of the system are still represented. Coarse-graining is common in, for example, polymer simulations[16].

potential is very general, as the region around any local minimum looks quadratic for small enough deviations. Notably, larger spring constants result in larger forces for the same displacement, and the particles tend to stay closer to their equilibrium distance apart. For a fixed time step, the displacement for a single particle due to the spring force $\Delta x_{har} \sim k\Delta t$, moving the particle towards its equilibrium distance. If Δx_{har} is larger than twice the displacement from r_0, the particles will have an even larger potential energy at the next time step and the system will be unstable. Thus, as the spring becomes stiffer (k increases) smaller time steps need to be used to maintain numerical stability. For very stiff springs, the particles are essentially at a fixed distance r_0, yet a very small time step must be used to simulate it. In cases like this, we might want to replace this stiff spring with a "rod" which keeps the particles a fixed distance apart. Then we may be able to choose a larger time step and run simulations over longer times. Care must be taken to do this, as we discuss in the next chapter.

In future chapters, we will introduce other potentials as well. It bears mentioning now that often these potentials are short-ranged (meaning we can accelerate calculation with a cut-off radius) and pair-wise (meaning they involve only two particles at a time). There are interactions which occur between more than two particles. However, the odds of n particles coming close enough to interact via these potentials goes as ϕ^{n-1}, where ϕ is the volume fraction of the system (total volume of the particles divided by the total volume of the system). Thus, except for very high volume fractions, many-body interactions are rare and often neglected in simulations. However, sometimes, many-particle interactions cannot be neglected. This is the case for 2D materials, which are the focus of Chapters 4 and beyond, requiring the inclusion of an additional potential to enforce bending rigidity.

1.2.5 Including other effects

So far, we have only discussed the translation of Brownian particles. However, there are other ways the particles and fluid can influence each other. The fluid can translate, rotate, or deform (strain), applying forces $\mathbf{F}$, torques $\mathbf{T}$, and stresslets $\mathbf{S}$ on the particles. The *grand* resistance (or mobility) problem concerns itself with all of these effects. In the laboratory (stationary) frame of reference[4]:

$$\begin{bmatrix} \mathbf{F} \\ \mathbf{T} \\ \mathbf{S} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\mathcal{R}}_{FU} & \boldsymbol{\mathcal{R}}_{F\Omega} & \boldsymbol{\mathcal{R}}_{FE} \\ \boldsymbol{\mathcal{R}}_{TU} & \boldsymbol{\mathcal{R}}_{T\Omega} & \boldsymbol{\mathcal{R}}_{TE} \\ \boldsymbol{\mathcal{R}}_{SU} & \boldsymbol{\mathcal{R}}_{S\Omega} & \boldsymbol{\mathcal{R}}_{SE} \end{bmatrix} \begin{bmatrix} \mathbf{U}^{\infty} - \mathbf{u} \\ \boldsymbol{\Omega}^{\infty} - \boldsymbol{\omega} \\ \mathbf{E}^{\infty} \end{bmatrix}, \tag{1.28}$$

where $\boldsymbol{\mathcal{R}}_{FU}$ is the resistance tensor coupling the particle velocities to their corresponding forces, $\boldsymbol{\mathcal{R}}_{F\Omega}$ is the resistance tensor coupling the particle rotational velocities to their corresponding forces, and so on. $\mathbf{U}^{\infty}$, $\boldsymbol{\Omega}^{\infty}$, and $\mathbf{E}^{\infty}$ are the bulk fluid's velocity, angular velocity, and rate of strain, respectively, and $\mathbf{u}$ and $\boldsymbol{\omega}$ are the particle velocities and angular velocities, respectively. The resistance tensor here is the grand resistance tensor. The corresponding grand mobility tensor is the inverse of this tensor. The grand resistance tensor is SPD as are each of the diagonal blocks, $\boldsymbol{R}_{FU}$, $\boldsymbol{R}_{T\Omega}$, and $\boldsymbol{R}_{SE}$. In this thesis, the simulations will only include the translational velocities of particles:

$$\mathbf{F} = \boldsymbol{\mathcal{R}}_{FU}(\mathbf{U}^{\infty} - \mathbf{u}). \tag{1.29}$$

Bibliography

[1] GG Stokes. On the effect of the internal friction of fluids on the motion of pendulums. *Mathematical and Physical Papers*, 3:8–106, 1851.

[2] EM Purcell. Life at low reynolds number. In *Physics and our world: reissue of the proceedings of a symposium in honor of Victor F Weisskopf*, pages 47–67. World Scientific, 2014.

[3] G Maruyama. Continuous markov processes and stochastic equations. *Rendiconti del Circolo Matematico di Palermo*, 4:48–90, 1955.

[4] S Kim and SJ Karrila. *Microhydrodynamics: principles and selected applications*. Butterworth-Heinemann, 2013.

[5] DL Ermak and JA McCammon. Brownian dynamics with hydrodynamic interactions. *The Journal of Chemical Physics*, 69(4):1352–1360, 1978.

[6] J Rotne and S Prager. Variational treatment of hydrodynamic interaction in polymers. *The Journal of Chemical Physics*, 50(11):4831–4837, 1969.

[7] PJ Zuk, E Wajnryb, KA Mizerski, and P Szymczak. Rotne–prager–yamakawa approximation for different-sized particles in application to macromolecular bead models. *Journal of Fluid Mechanics*, 741:R5, 2014.

[8] E Wajnryb, KA Mizerski, PJ Zuk, and P Szymczak. Generalization of the rotne–prager–yamakawa mobility and shear disturbance tensors. *Journal of Fluid Mechanics*, 731:R3, 2013.

[9] AM Fiore, F Balboa Usabiaga, A Donev, and JW Swan. Rapid sampling of stochastic displacements in brownian dynamics simulations. *The Journal of Chemical Physics*, 146(12), 2017.

[10] H Hasimoto. On the periodic fundamental solutions of the stokes equations and their application to viscous flow past a cubic array of spheres. *Journal of Fluid Mechanics*, 5(2):317–328, 1959.

[11] EE Keaveny. Fluctuating force-coupling method for simulations of colloidal suspensions. *Journal of Computational Physics*, 269:61–79, 2014.

[12] AM Fiore and JW Swan. Rapid sampling of stochastic displacements in brownian dynamics simulations with stresslet constraints. *The Journal of Chemical Physics*, 148(4), 2018.

[13] KM Reed. *Dynamic simulations of colloidal dispersions: sticky and polarizable particles as building blocks for novel materials*. PhD thesis, Massachusetts Institute of Technology, 2024.

[14] DM Heyes and JR Melrose. Brownian dynamics simulations of model hard-sphere suspensions. *Journal of Non-Newtonian Fluid Mechanics*, 46(1):1–28, 1993.

[15] Z Varga, G Wang, and JW Swan. The hydrodynamics of colloidal gelation. *Soft Matter*, 11(46):9009–9019, 2015.

[16] M Doi and SF Edwards. *The theory of polymer dynamics*, volume 73. Oxford University Press, 1988.

Chapter 2

Methods for constrained Brownian dynamics

In this chapter, we compare two methods, the projected conjugate gradient (PrCG) and the generalized minimum residual (GMRES) methods, for solving the saddle point problem involved in performing constrained Brownian dynamics simulations. Using the example holonomic constraints of rigid bodies, freely jointed chains, and immobile particles, we show that PrCG and GMRES carry the same empirical linear computational complexity. PrCG, however, exhibits some notable advantages over GMRES including lower precomputational and storage burdens, a guaranteed feasible iterate, and trivial extension to new constraint types due to the lack of a preconditioner. Finally, we use PrCG to solve a mixed constraint problem with a rigid body and immobile particles, comparing to the analytical solution at large separations. The content of this chapter was adapted from Funkenbusch, W. T., Silmore, K. S., & Swan, J. W. (2024). "Approaches for fast Brownian dynamics simulation with constraints." Journal of Computational Physics*, 509, 113043.*

2.1 Introduction to constrained systems

Thus far, the particles we have seen have all had 3 degrees of freedom. By this, we mean that we can independently specify the x, y, and z coordinates of each particle. This is not always the case. Consider a particle constrained to the surface of a sphere of radius R (this could represent a charge moving around a particle, for example). The sphere is represented by the equation

$$x^2 + y^2 + z^2 = R^2, \tag{2.1}$$

so if we specify the x and y positions of the particle, the z position can only take 2 values:

$$z = \pm\sqrt{R^2 - x^2 - y^2}. \tag{2.2}$$

Because we can only specify 2 of the 3 components independently, this particle only has 2 degrees of freedom. For N particles constrained to the surface of the sphere, the overall system would only have $2N$ degrees of freedom instead of $3N$. We call conditions which reduce the degrees of freedom of the system *constraints*. Adding constraints complicates our simulations, but before we discuss how, let's discuss why we might want to constraint our system in the first place.

Motivation for constrained systems

The most common reason to constrain a systems is to model more complex colloidal systems which are not just a suspension of floating spheres or rods[a]. For example, large polymers can be modeled with beads connected by inextensible links, where each bead represents one or several monomers[1]. Flexible fibers, such as those present in animal cells, can be modeled with a similar coarse graining scheme that imposes a set of constraints which enforce inextensibility of the whole fiber[2]. While coarse-grained elements are typically treated as spherical beads, they can be arranged and connected rigidly to represent more complicated shapes and even coupled to elastic structures to represent complex bodies like a swimming microscopic organism and its flagellum[3].

We should ask whether constraints such as these can be relaxed and replaced with conservative forces derived from stiff potentials that enforce the desired physical behavior. Indeed, stiff potentials have been used to model rigid bodies[4] as well as polymers. For polymers, this representation is called the bead-spring model[5], contrasting with the bead-rod model which applies explicit rigid constraints to the links between beads[6]. In numerical solutions of the equations of motion, the simulation time step is usually limited by the shortest characteristic relaxation time of the dynamical system to preserve numerical stability[b]. For example, in a polymer chain with stiff spring links, the relaxation time of these springs, and thus the required time step, can be very short relative

[a] Recall that we have equations developed for solutions to our equations for spheres and rods but not arbitrarily-shaped particles.

[b] If our time step was longer, we could not resolve that piece of physics in our system and our simulation might be numerically unstable. See the discussion on harmonic potentials from the previous chapter.

to the relaxation time of the whole polymer. Additionally, when external forces are applied to the polymer, a sufficient stiffness may not be known *a priori*. For example, one may pull the end of a DNA strand through a nanopore for sensing[7], deforming the polymer and causing the links between the beads to stretch or compress. A bond that is stiff enough to give the desired equilibrium behavior of a polymer may not be stiff enough when subjected to a complex nonlinear process of this sort. Perhaps a conservative force that diverges at a finite extension could be employed, but then the bond grows ever stiffer as it is stretched, necessitating shorter time steps still. Rigid constraints which fix the distance between beads, on the other hand, eliminate these fast time scales, allowing for longer time steps to be employed in exchange for increased computational cost per step[8]. Inextensible links of this nature and other constraints which explicitly reduce the number of degrees of freedom of the system cannot be modeled using a simple conservative force field[c]. Thus, practical integration schemes must balance the tradeoff between time step and computational cost per step, generally necessitating linear scaling with the number of beads for constrained systems in order to be competitive with unconstrained methods.

As an example of flow through a complex geometry, Figure 2.1 shows a system of beads, some placed on the surface of a gyroid[d] and others randomly distributed in the interstitial space. A force was then applied to the beads on one side of the gyroid surface. When no constraint is applied to the beads, the beads move as expected. Due to the periodic boundary conditions, the beads with no applied force tend to move in the opposite direction such that there is no net flow through the box. When the beads on the surface of the gyroid are constrained such that they have zero velocity, the velocities of the beads on both sides of the gyroid surface change quite significantly, showing the importance of properly treating constraints in systems.

2.1.1 Current methods for solving of constrained systems

Holonomic constraints, which are the subset of constraints we will focus on in this chapter, are defined as algebraic equations depending only on the positions of the beads[9]. We will give a more formal definition later, but for now the important part is that these constraints define a submanifold

[c] We will see why in a bit.

[d] A porous surface commonly seen in nature due to having a high surface area-to-volume ratio.

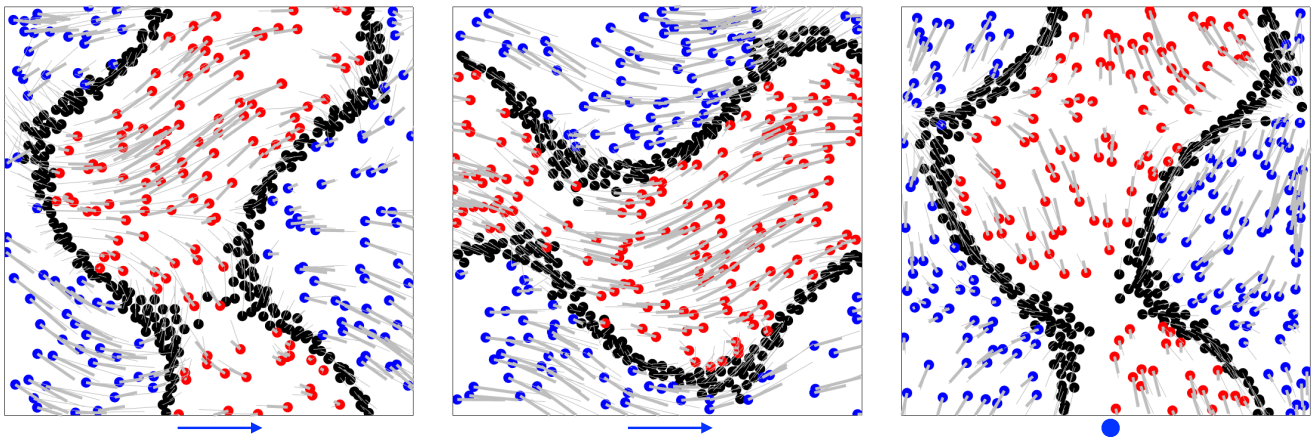

Figure 2.1 Three orthogonal planar cuts with width equal to 4 bead radii through the same cubic, periodic simulation box. The black beads are held fixed by holonomic constraints and have been localized to the surface of a gyroid with unit cell dimension of 80 bead radii, matching that of the periodic simulation cell. The red beads are free to move on one side of the gyroid surface. No external forces are applied to the red beads. The blue beads are free to move on the other side of the gyroid. A uniform force is applied to the blue beads with direction indicated below each planar cut. The thick gray lines show the velocity of the beads projected into the plane when the holonomic constraints are enforced by the PrCG algorithm. The thin gray lines show the velocity of the beads projected into the plane when no constraints are imposed.

of $\mathbb{R}^{3N}$[e], called the *constraint manifold*, which the particles must stay on[9–11]. Therefore, particle velocities must be tangent to the constraint manifold. This is achieved by exerting a constraint force, which we cannot determine *a priori*, on each bead, which in practice involves solving a system of linear equations[12]. Specifically, these equations take the form of a saddle point problem, where the constrained velocities act as Lagrange multipliers. In molecular dynamics simulations, for example, holonomic constraints can be used to enforce fixed bond lengths or angles in molecules. The motion of these constrained systems are often solved using the SHAKE[13], RATTLE[14], or LINCS[15] algorithms, or their variants such as M-SHAKE[16] or P-LINCS[17], which are all based in the method of Lagrange multipliers. Saddle point formulations also appear in mechanical systems with holonomic constraints to solve for particle accelerations[18].

For colloidal suspensions of many particles, Swan and Wang used the Schur complement to eliminate the constraint forces and solved for the constrained velocities in a rigid body system using the preconditioned conjugate gradient method[19]; however this method was superlinear in the number of particles. Previous work by Usabiaga *et al.* used the generalized minimum residual

[e]By submanifold we essentially mean a continuous, lower-dimensional surface in $\mathbb{R}^{3N}$.

(GMRES) method to solve rigid body problems with linear scaling[3]. This method has also been applied with success to other systems, such as those of slender fibers[2].

However, GMRES often requires the construction of a constraint-specific preconditioner, which can be difficult to formulate and may require significant storage and computational costs in precomputation. Furthermore, the iterates produced by GMRES are not guaranteed to be feasible (i.e. to not do work on the system), meaning that one must monitor both the convergence of bead velocities and the feasibility of the solution. To be computationally efficient, one would like to choose the errors accrued by the saddle point solve to match the errors accrued by the integration scheme, which scale with the time step used. However, for a solver like GMRES, the solution tolerance must be chosen keeping in mind feasibility, possibly necessitating more iterations for convergence, even if the bead velocities have converged to the desired tolerance. This is especially relevant given that one advantage of constraining a system is a larger time step, leading to larger errors from the integration scheme.

In this chapter, we use the projected conjugate gradient (PrCG) method without preconditioning[20,21] to solve these constrained dynamical problems common to Brownian dynamics simulations. We find that PrCG exhibits the same linear scaling and similar residual convergence as GMRES, but exhibits several advantages over it:

1. Lower precomputation costs, both computational and storage, depending on the type of constraints used.
2. Feasible iterates, allowing the tolerance to be varied freely according to the desired error tolerance on the bead velocities, rather than being restricted by the satisfaction of the work constraint.
3. A residual which accurately reflects the convergence of the constrained velocities, which are used to calculate the bead velocities.
4. Trivial and robust extension to different types of constraints, including mixed constraints, without the need to formulate a preconditioner.

Because beads are advanced over a finite time step, integration schemes for constrained Brownian dynamics always move the elements tangent to, but potentially off of, the constraint manifold

if the constraints are nonlinear. These "off-manifold" moves must be followed with retractions back onto the constraint manifold[3,22–25]. While retraction schemes are nontrivial to implement, the saddle point solve is often the most computationally intensive part of this process and is thus the focus of this work. In this paper, we solve this saddle point problem for several constraints using PrCG and compare to GMRES.

The remainder of the chapter is organized as follows. In Section 2.2, the saddle point problem as applied to Brownian dynamics simulation is discussed. The relevant forces within the simulation are introduced, and the saddle point problem is formulated in the context of iterative algorithms such as PrCG and GMRES. In Section 2.3, PrCG is compared to GMRES for several example constrained systems: rigid bodies, freely jointed chains, and immobile bead systems. The relative residuals, relative solution errors, and time complexities are examined for each example. Finally, PrCG is applied to an example problem with two distinct types of constraints involving a rigid body moving under a constant force towards a collection of beads fixed in space. The resulting velocity of the rigid body and the force on the fixed beads is compared to analytically derived formulas in the limit of large separation in order to validate the method.

2.2 Saddle point problems for constrained Brownian dynamics simulations

The formulation for constrained systems is similar to unconstrained systems, but with the addition of constraint forces. The other forces in our system can act in any direction and the constraint forces act in response to these other forces such that the particle move tangent to the constraint manifold. Because the constraint forces are coupled to all of the other forces occurring in the system, we do not necessarily know them *a priori*[f] This makes our previous method of finding the particle velocities (of summing the forces and multiplying by the mobility tensor) difficult[g]. There are also often many constraint forces which will satisfy the constraints, so an additional condition is needed to fully specify the system. This makes solving constrained systems more difficult than

[f]That is, we cannot calculate them using just the positions of the particles, as we could for all of the other forces.

[g]It is possible to write down an analytical equation for the constraint forces, as we will show later, but it is very expensive to calculate as it requires inverting the mobility tensor, which requires $\mathcal{O}(N^3)$ operations, where N is the number of particles.

unconstrained systems.

2.2.1 List of forces for constrained systems

We formulate the problem formally now. Consider N interacting beads in 3D space with positions $\mathbf{x} \in \mathbb{R}^{3N}$ immersed in a Stokesian fluid. As discussed in the previous chapter, the particles can experience several forces, with the constraint forces being added to the list:

- Conservative forces, $-\nabla V(\mathbf{x})$, where $V(\mathbf{x})$ is the potential energy of the particles. $V(\mathbf{x})$ is a function of the particle positions and, in this work, is assumed to be smooth with respect to those positions.
- Hydrodynamic forces, $\mathbf{f}_H = -\boldsymbol{\mathcal{M}}^{-1}\mathbf{u}$, where $\boldsymbol{\mathcal{M}} \in \mathbb{R}^{3N\times 3N}$ is the mobility tensor and $\mathbf{u} \in \mathbb{R}^{3N}$ are the particle velocities. The mobility tensor is symmetric positive definite (SPD) and depends on the bead positions. We use the RPY tensor[26] as an approximation for the mobility tensor, as discussed in the previous chapter. We note again that matrix-vector products with this tensor can be performed in log-linear time with the number of particles[27–29].
- Brownian forces, $\mathbf{f}_B$, with $\langle \mathbf{f}_B(t)\rangle = 0$, $\langle \mathbf{f}_B(t)\mathbf{f}_B(0)^T\rangle = 2k_BT\boldsymbol{\mathcal{M}}^{-1}\delta(t)$, where k_BT is the thermal energy, t is the time, $\delta(t)$ is the Dirac delta function, and the angle brackets indicate an average over realizations of this stochastic force.
- Constraint forces, $\mathbf{f}_c$, which restrict the motion of the beads to a manifold defined by the potentially nonlinear equations $\mathbf{q}(\mathbf{x}) = \mathbf{0}$, where $\mathbf{q} : \mathbb{R}^{3N} \to \mathbb{R}^M$. The constraints are assumed to be a function of the particle positions such that they are holonomic. As with the potential energy, we will assume that the constraints are smooth.

2.2.2 Force balance for constrained systems

Constraint forces appear in a variety of Brownian dynamics models. In the immersed boundary method applied to the motion of rigid bodies through a fluid, these constraint forces are the internal forces that must be exerted on the discrete elements of the bodies so that they move rigidly. In the simulation of polymers, the bead-rod model treats a linear polymer as a chain of beads, which are freely jointed but with fixed distance between neighboring beads in the chain.

The over-damped force balance, which we discussed in the previous chapter without constraint

forces, can be rewritten to determine an instantaneous, stochastic velocity of the beads:

$$\mathbf{u} = \boldsymbol{\mathcal{M}}(-\nabla V + \mathbf{f}_c) + \mathbf{u}_B, \tag{2.3}$$

where $\mathbf{u}_B = \boldsymbol{\mathcal{M}}\mathbf{f}_B$ are the Brownian velocities of the beads. The constraint forces, $\mathbf{f}_c$, are still undetermined, but they must act such that the bead positions at each point in time satisfy the constraints, $\mathbf{q}(\mathbf{x}) = 0$. We define the gradient of the constraints: $\mathbf{G} = \nabla\mathbf{q} \in \mathbb{R}^{3N\times M}$. The beads must move tangent to the constraint manifold such that $\mathbf{u} = \boldsymbol{\Sigma}\boldsymbol{\lambda}$, where $\boldsymbol{\Sigma} \in \mathbb{R}^{3N\times P}$ with columns that form a minimal basis for the left null space of $\mathbf{G}$, $\boldsymbol{\lambda} \in \mathbb{R}^P$ are Lagrange multipliers, and P, the degrees of freedom of all the beads, is the dimension of the left null space of $\mathbf{G}$. The columns of $\boldsymbol{\Sigma}$ describe a P-dimensional manifold of $\mathbb{R}^{3N}$ to which the particles must move tangent. We refer to $\boldsymbol{\Sigma}$ in this work as the constraint matrix. Holonomic constraints give rise to constraint forces which are linear combinations of the columns of $\mathbf{G}$[12]. Thus, because $\boldsymbol{\Sigma}$ is a basis for the null space of $\mathbf{G}$, the constraint forces do not dissipate energy and act normal to the constraint manifold (i.e. $\boldsymbol{\Sigma}^T\mathbf{f}_c = \mathbf{0}$). This equation combined with the over-damped momentum balance on the beads yields a system of equations for the constraint forces, $\mathbf{f}_c$, and the multipliers, $\boldsymbol{\lambda}$:

$$\boldsymbol{\mathcal{M}}'\mathbf{x}' = \begin{bmatrix} \boldsymbol{\mathcal{M}} & -\boldsymbol{\Sigma} \\ -\boldsymbol{\Sigma}^T & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{f}_c \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\mathcal{M}}\nabla V - \mathbf{u}_B \\ \mathbf{0} \end{bmatrix} = \mathbf{b}'. \tag{2.4}$$

Equation 2.4 is a so-called saddle point problem[2,30], and solving this equation via the PrCG iterative method will be the focus of this chapter. The bead velocities for integration in time can be calculated from the multipliers: $\mathbf{u} = \boldsymbol{\Sigma}\boldsymbol{\lambda}$. The analytical solution to this problem is simple to write down: $\mathbf{f}_c = \nabla V + \mathbf{f}_b + \boldsymbol{\mathcal{M}}^{-1}\boldsymbol{\Sigma}\boldsymbol{\lambda}$ and $\boldsymbol{\lambda} = (\boldsymbol{\Sigma}^T\boldsymbol{\mathcal{M}}^{-1}\boldsymbol{\Sigma})^{-1}(\nabla V + \mathbf{f}_b)$. However, for large numbers of beads and a dense mobility matrix, directly inverting $\boldsymbol{\mathcal{M}}$ requires $\mathcal{O}(N^3)$ operations, which can be prohibitive. Additionally, calculating $\boldsymbol{\Sigma}$ through QR decomposition[h] requires, at worst, $\mathcal{O}(NM^2)$ operations.

[h]The QR decomposition breaks a matrix, $\mathbf{A}$ down into an orthogonal matrix, $\mathbf{Q}$ and upper triangular matrix, $\mathbf{R}$, such that $\mathbf{A} = \mathbf{QR}$. An orthonormal matrix has $\mathbf{Q}^T\mathbf{Q} = \mathbf{I}$ while an upper triangular matrix only has elements on and above its diagonal. These properties end up being convenient for accelerating calculations compared to working with the raw matrix $\mathbf{Q}$, as we will see later.

2.2.3 Discussion of the constraint matrix

Constraint matrix for nonlinear constraints

The constraint matrix, $\mathbf{\Sigma}$, depends on the positions of the particles and the types of constraint imposed. For Brownian motion of beads restricted to the surface of a sphere with radius R and with center at zero, $q_i(\mathbf{x}) = \|\mathbf{x}\|^2 - R^2$ for $i = 1, \ldots N$. For bead-rod chains, a rigid link between a bead labeled i and a bead labeled j can be assigned a unique element, k, of the constraint equation vector: $q_k(\mathbf{x}) = \|\mathbf{r}_{ij}\|^2 - L_{ij}^2$, where $\mathbf{r}_{ij} = \mathbf{x_i} - \mathbf{x_j}$ is the vector pointing from the center of bead j to the center of bead i and L_{ij} is the length of the bond between those beads. For nonlinear constraints such as these, $\mathbf{G}$ can be calculated analytically and the constraint matrix can subsequently be determined, for example, through a QR decomposition of $\mathbf{G}$ at each time step.

Constraint matrix for linear constraints

In the case of linear constraints, the constraint matrix can be determined without explicitly calculating $\mathbf{G}$. For example, the motion of a rigid body can be described as a combination of its translational and rotational motion. Because both of these are linear transformations of the bead coordinates, the analytical form for the constraint matrix can be written down easily without the need for the constraint equations. For a system containing many rigid bodies, the full constraint matrix is block diagonal, with each block corresponding to an individual rigid body because there are no constraints between beads in different rigid bodies:

$$\mathbf{\Sigma} = \begin{bmatrix} \mathbf{\Sigma}_1 & \mathbf{0} & \ldots & \mathbf{0} \\ \mathbf{0} & \mathbf{\Sigma}_2 & \ldots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \ldots & \mathbf{\Sigma}_{N_{\text{bodies}}} \end{bmatrix}, \tag{2.5}$$

where N_{bodies} is the number of rigid bodies in the simulation.

For a single block corresponding to a rigid body containing B_i beads, $\mathbf{\Sigma}_i$ has at most 6 columns, corresponding to the 3 translational and at most 3 rotational degrees of freedom of the body.

Colinear bodies have only 2 rotational degrees of freedom, and thus their corresponding $\mathbf{\Sigma}_i$ has only 5 columns, and free beads not constrained to a rigid body have a $\mathbf{\Sigma}_i$ with only 3 columns. For a non-colinear rigid body, each of the blocks in $\mathbf{\Sigma}$ has the form:

$$\mathbf{\Sigma}_i = \begin{bmatrix} \mathbf{I}_3 & \mathbf{H}_{i,1} \\ \mathbf{I}_3 & \mathbf{H}_{i,2} \\ \vdots & \vdots \\ \mathbf{I}_3 & \mathbf{H}_{i,B_i} \end{bmatrix}, \tag{2.6}$$

where B_i is the number of beads in the ith body, $\mathbf{I}_3$ is the 3×3 identity tensor, and $\mathbf{H}_{i,\alpha} = \boldsymbol{\epsilon} \cdot (\mathbf{x}_{i,\alpha} - \bar{\mathbf{x}}_i)$, with $\boldsymbol{\epsilon}$ the Levi-Civita tensor, $\mathbf{x}_{i,\alpha}$ the position of the αth bead in body i, and $\bar{\mathbf{x}}_i$ the average position of beads in body i.

The condition number of $\mathbf{\Sigma}_i^T \mathbf{\Sigma}_i$ grows with the squared radius of gyration, so both the long-ranged hydrodynamic interactions as well as the geometry of the rigid body contribute to making iterative solution of the saddle point equations difficult. With $\mathbf{\Sigma}$ constructed this way, the multipliers, $\boldsymbol{\lambda}$, are just the translational and angular velocities of the rigid bodies themselves.

As another example, the constraint matrix for a mixture of immobile beads and free beads (indexed in this order) has a sparse, simple form:

$$\mathbf{\Sigma} = \begin{bmatrix} \mathbf{0} \\ \mathbf{I}_{3N_f} \end{bmatrix} \tag{2.7}$$

where $\mathbf{\Sigma} \in \mathbb{R}^{3N \times 3N_f}$, with N_f the number of free beads and N the total number of beads. Linear constraints such as these or rigid body constraints are convenient in that retraction steps are not needed, unlike nonlinear constraints such as bead-rod constraints. One could conceive of and formulate constraint matrices for other types of linear constraints in a similar manner, such as bodies which are free to rotate but not translate:

$$\mathbf{\Sigma}_i = \begin{bmatrix} \mathbf{H}_{i,1} \\ \mathbf{H}_{i,2} \\ \vdots \\ \mathbf{H}_{i,B_i} \end{bmatrix}, \tag{2.8}$$

with $\mathbf{H}_{i,\alpha}$ as defined above.

2.2.4 Time complexity of iterative solutions to the saddle point problem

Iterative methods for solving the saddle point problem resulting from constrained Brownian dynamics can improve the computational complexity significantly compared to direct inversion of the saddle point matrices. For the modeling of rigid bodies in immersed boundary simulations, this saddle point problem has been solved with the generalized minimum residual (GMRES) method using certain cleverly constructed preconditioners[2,31]. These methods can lead to linear computational complexity with respect to the number of beads in the simulation or the number of rigid bodies when methods like spectrally accurate Ewald summation or fast multipoles are used to compute the product of the mobility matrix with a vector. We find that, in practice, because the condition number of the mobility matrix grows with the number of beads[29], so does the condition number of the saddle point problem, necessitating preconditioning. A complication with this approach is that the preconditioners can be expensive to calculate, but this calculation can be amortised over the duration of a simulation.

One successful approach, discussed in more detail in the Methods Section, uses a Schur decomposition of the saddle point problem to construct a preconditioner that yields the solution one would expect if each of the beads only interacted hydrodynamically with beads in its own rigid body[31]. This can be thought of as precomputing the single body solution for each body if it were isolated. For identical rigid bodies made from B beads, this precalculation requires $O(B^3)$ operations to precompute and $O(B^2)$ memory for storage of the single body solution and can be reused over the course of the simulation for any of the identical bodies by simply rotating it into the appropriate reference frame for that body.

There are cases, however, when precomputation is not as efficient. For example, in the case

above, if the rigid bodies are distinct from one another, the precomputation must be done for each distinct body. There may also be circumstances in which one wants the rigid constraints to change during the simulation such as models of fracture or agglomeration of bodies. Then, the number of rigid bodies as well as their morphology changes dynamically. In that case, one cannot simply amortise the precalculation cost for the preconditioner over the simulation. Wang *et al.* found that high-frequency viscosity calculations of equilibrium configurations of monoclonal antibodies, which can also be cast as a saddle point problem, agreed better with experimental values when nearly-touching antibodies were treated as a single rigid body[32,33]. In this case, it is unlikely that any two rigid bodies in all of the samples are identical, so precomputation must be done for each rigid body in each configuration.

Assuming each body has a distinct morphology, the storage cost for the preconditioner is $\mathcal{O}(N_{\text{bodies}}B^2)$, and the time complexity of calculating the preconditioner for all the particles grows as $\mathcal{O}(N_{\text{bodies}}B^3)$ instead. In general, this precomputation must be done before each saddle point solve. Although the cost for precomputation scales linearly with the number of beads, it has a strong dependence on the number of beads per body and for large bodies may require significant computational effort compared to the saddle point solve itself. Thus, even though iterative methods with specialized preconditioners can reduce the time complexity of the saddle point problem solve to one that is linear in the number of beads for all the examples discussed, there are limitations that are resolved by introduction of an iterative method that requires no preconditioner at all, as we explore below.

2.2.5 Application of projected conjugate gradients

We start by considering a generic saddle point problem, which can be written as

$$\begin{bmatrix} \mathbf{A} & -\boldsymbol{\Sigma} \\ -\boldsymbol{\Sigma}^T & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{y} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \mathbf{b} \\ \mathbf{d} \end{bmatrix}. \tag{2.9}$$

This can be equivalently written as an equality-constrained minimization problem:

$$\begin{aligned} \mathbf{y} = \arg\min_{\mathbf{y}} \quad & \tfrac{1}{2}\mathbf{y}^T\mathbf{A}\mathbf{y} - \mathbf{y}^T\mathbf{b} \\ \text{s.t.} \quad & \boldsymbol{\Sigma}^T\mathbf{y} = -\mathbf{d}. \end{aligned}$$

In the case of Brownian dynamics as formulated above, $\mathbf{y} = \mathbf{f}_c$, $\mathbf{A} = \boldsymbol{\mathcal{M}}$, $\mathbf{b} = (\boldsymbol{\mathcal{M}}\nabla V - \mathbf{u}_B)$, and $\mathbf{d} = \mathbf{0}$. Note that this formulation does not include inequality constraints. Constraints such as these are difficult to implement in general, but can be solved using, for example, interior point methods or by transforming the inequality constraints into equality constraints (as long as the inequality constraints are also smooth)[22,34,35]. However, this implementation is a non-trivial extension of this work, so we do not consider such constraints in this work.

The conjugate gradient (CG) method is a technique for solving linear systems of equations with an SPD matrix (such as the mobility tensor) and is often used to solve unconstrained quadratic programming (QP) problems. Nocedal *et al.*[20] extended the conjugate gradient method to equality-constrained QP problems by projecting the iterative steps taken by the CG method onto the tangent space of the constraint manifold. This algorithm is called the projected conjugate gradient (PrCG) method. When the constraints are affine, solutions starting on the constraint manifold remain on that manifold. In this work, we use a modification of the PrCG method presented by Gould *et al.* (Algorithm 6.2)[21], which improves numerical stability over the original formulation.

The PrCG algorithm requires a linear operator to project changes to the constraint forces tangent to the constraint manifold. Formally, this projection operator can be written as $\mathbf{Z} = \mathbf{I} - \boldsymbol{\Sigma}(\boldsymbol{\Sigma}^T\boldsymbol{\Sigma})^{-1}\boldsymbol{\Sigma}^T$. Directly calculating $\boldsymbol{\Sigma}^T\boldsymbol{\Sigma}$ and its inverse requires $O(NP^2)$ and $O(P^3)$ operations, respectively, which is a significant computational cost given that this projection operator must be calculated at every iteration. However, if $\boldsymbol{\Sigma}$ is replaced with a matrix $\mathbf{Q}$ from a thin QR decomposition, $\boldsymbol{\Sigma} = \mathbf{QR}$, no inversion is necessary as $\mathbf{Q}$ is orthonormal (i.e. $\mathbf{Q}^T\mathbf{Q} = \mathbf{I}$). Using the orthogonal projection drawn from a QR decomposition of $\boldsymbol{\Sigma}$, the saddle point problem (Equation 2.9) can be re-expressed as:

$$\begin{bmatrix} \mathbf{A} & -\mathbf{Q} \\ -\mathbf{Q}^T & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{y} \\ \boldsymbol{\lambda}' \end{bmatrix} = \begin{bmatrix} \mathbf{b} \\ \mathbf{c} \end{bmatrix}, \tag{2.10}$$

where $\mathbf{c} = \mathbf{R}^{-T}\mathbf{d}$ and $\boldsymbol{\lambda}' = \mathbf{R}\boldsymbol{\lambda}$. The bead velocities can be determined with a single matrix

multiplication: $\mathbf{u} = \mathbf{Q}\boldsymbol{\lambda}'$. This reformulation is still a saddle point problem, but now the orthogonal projection operator requires no inversion: $\mathbf{Z} = \mathbf{I} - \mathbf{Q}\mathbf{Q}^T$.

The PrCG algorithm may be initialized with any point on the constraint manifold such as the generic choice $\mathbf{y}_0 = -\mathbf{Q}\mathbf{c}$. Due to the projection step at each iteration of the PrCG algorithm, solutions starting on the manifold remain on the manifold throughout the iterative solution process. This is especially relevant for systems where $\mathbf{c} \neq \mathbf{0}$, where an on-manifold initial guess is not necessarily available for GMRES. In these systems, for GMRES, constraint satisfaction must be satisfied as iteration progresses or additional precomputational effort must be expended to produce an on-manifold initial guess (for example, by re-expressing the saddle point problem as for PrCG through a QR decomposition of $\boldsymbol{\Sigma}$). The PrCG algorithm requires one matrix-vector multiply using the matrix $\mathbf{A}$ and two matrix-vector multiplies with $\mathbf{Q}$ and $\mathbf{Q}^T$ per iteration, which is equivalent to one augmented mobility matrix multiplication in GMRES. In practice, for Brownian dynamics simulations, the multiplication with the mobility tensor will dominate the time complexity because the mobility tensor is dense.

Making the appropriate substitutions for the Brownian dynamics system, Equation 2.10 becomes

$$\begin{bmatrix} \boldsymbol{\mathcal{M}} & -\mathbf{Q} \\ -\mathbf{Q}^T & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{f}_c \\ \boldsymbol{\lambda}' \end{bmatrix} = \begin{bmatrix} \boldsymbol{\mathcal{M}}\nabla V - \mathbf{u}_B \\ \mathbf{0} \end{bmatrix}, \tag{2.11}$$

which is solved using the PrCG algorithm shown in Algorithm 1, adapted from Gould *et al.*[21].

2.3 Results

2.3.1 Methods

To investigate applications of the PrCG approach to solving the saddle point problems relevant to Brownian Dynamics simulations, we consider three types of example systems: rigid bodies, freely jointed chains, and systems with immobile particles. Example rigid bodies, freely jointed chain, and immobile bead scenarios are shown in Figure 2.2, and these systems are described in more detail in their respective subsections. We study the performance as a function of number of beads

Result: $\mathbf{y}$, $\boldsymbol{\lambda}'$
Calculate $\mathbf{Q}$, $\mathbf{R}$ such that $\boldsymbol{\Sigma} = \mathbf{QR}$
Initial guess $\mathbf{y} \leftarrow -\mathbf{Qc}$
Residual $\mathbf{r} \leftarrow \mathbf{Ay} - \mathbf{b}$
Projected residual $\mathbf{g} \leftarrow \mathbf{r} - \mathbf{QQ}^T\mathbf{r}$
Refined projected residual $\mathbf{g} \leftarrow \mathbf{g} - \mathbf{QQ}^T\mathbf{g}$
Initial residual norm $nr_0 \leftarrow \|\mathbf{g}\|$
Step direction $\mathbf{d} \leftarrow -\mathbf{g}$
$\mathbf{r} \leftarrow \mathbf{g}$,
for $i = 1 : N_{iters}$ **do**
  Step distance $\alpha \leftarrow \frac{\mathbf{r}^T\mathbf{g}}{\mathbf{d}^T\mathbf{Ad}}$
  Update guess $\mathbf{y} \leftarrow \mathbf{y} + \alpha\mathbf{d}$,
  Update residuals $\mathbf{r}' \leftarrow \mathbf{r} + \alpha\mathbf{Ad}$, $\mathbf{g}' \leftarrow \mathbf{r}' - \mathbf{QQ}^T\mathbf{r}'$
  $\beta \leftarrow \frac{\mathbf{r}'^T\mathbf{g}'}{\mathbf{r}^T\mathbf{g}}$
  Update step direction $\mathbf{d} \leftarrow -\mathbf{g}' + \beta\mathbf{d}$
  Update residuals $\mathbf{g} \leftarrow \mathbf{g}'$, $\mathbf{r} \leftarrow \mathbf{g}'$
  if $\|\mathbf{g}\|/nr_0 < \epsilon$ **then**
    break
  end
end
Lagrange multipliers $\boldsymbol{\lambda}' \leftarrow -\mathbf{Q}^T(\mathbf{b} - \mathbf{Ay})$

Algorithm 1: PrCG algorithm adapted from Gould *et al.*[21]. N_{iters} is the maximum allowed iterations and ϵ is the desired tolerance.

and number of bodies in each of these cases and compare that performance to solutions using the preconditioned GMRES method. For GMRES, we solve the original saddle point (Equation 2.4) formulation[4] and use the default GMRES method in MATLAB. For PrCG, we solve the saddle point problem transformed by decomposing $\boldsymbol{\Sigma}$ (Equation 2.11).

Both algorithms were run with a fixed residual tolerance of $\epsilon = 10^{-8}$[i]. No conservative inter-bead potentials were applied (i.e. $\nabla V = 0$)[j]. The Brownian velocities should have a covariance proportional to a certain square root of the mobility tensor in general, $\mathbf{u}_B = \sqrt{2k_BT/\Delta t}\boldsymbol{\mathcal{M}} \cdot d\mathbf{W}$, where Δt is the time step, $\mathbf{W}$ is the standard Wiener process, and $\boldsymbol{\mathcal{M}}$ is a matrix such that $\boldsymbol{\mathcal{M}}\boldsymbol{\mathcal{M}}^\dagger = \boldsymbol{\mathcal{M}}$, where the dagger denotes the adjoint operator. As discussed in the previous chapter, calculating $\boldsymbol{\mathcal{M}}$ can be done in $\mathcal{O}(N)$ time using methods such as the spectral Ewald approach[28] or force-coupling method[27,37]. However, in order to isolate the computational burden of

[i]This is a pretty low tolerance to show that the algorithms continue to be stable at these tolerances. The relative performance of the two algorithms doesn't change much with the desired tolerance and results for lower tolerances can be inferred from plots of performance vs. iteration (e.g. Figure 2.3)

[j]Calculation of inter-bead potentials are almost never rate-limiting so we choose to control with the simplest scenario.

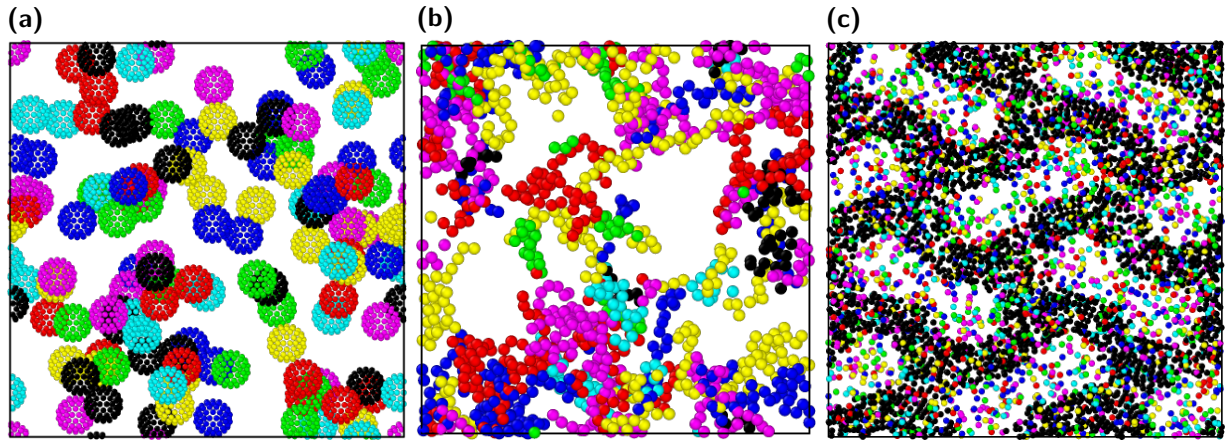

Figure 2.2 (a) Example system of geodesic polyhedra. A slice is shown for clarity. (b) Example system of freely jointed chains. A slice is shown for clarity. (c) Example gyroid immobile system with a volume fraction of $v = 0.075$. Half of the beads are immobile, and these beads are colored black. In all systems, black borders correspond to periodic simulation box boundaries. Images rendered using OVITO[36].

the iterative solves, we instead sample Brownian velocities from a normal distribution (i.e. without hydrodynamic interactions): $\mathbf{u}_B = \sqrt{2k_BT/6\pi\eta a\Delta t}d\mathbf{W}$, with $\Delta t = 10^{-5}$ for all simulations. Beads were moved to provisional positions $\mathbf{x} = \mathbf{x}_0 + \mathbf{u}_B\Delta t/2$, where $\mathbf{x}_0$ were the initial positions of the particles. A constrained solve was then performed at this provisional point with the same Brownian velocities, in a matter similar to what would be used in a predictor-corrector integration scheme. Because the Brownian velocities are stochastic, iterative solutions to the saddle point problem cannot be accelerated through any clever choice of the initial guess, for example, by using the solution at a previous time step in a dynamic simulation.

Restarts are another free parameter in the GMRES algorithm which can be adjusted to improve performance. GMRES increases in computational burden with each iteration, requiring more effort for the Arnoldi process[k]. This can become prohibitive for a large number of iterations. By restarting GMRES every k iterations, the computational burden is reduced at the cost of guaranteed convergence. Thus, k is another parameter in GMRES which must be optimized by

[k]GMRES works on the principal that a certain subspace, called the nth Krylov subspace, where n is the iteration number, gives a good approximation of the exact solution for relatively low values of n. As the iteration number increases, the subspace becomes larger and the cost of each iteration increases with n^2. To circumvent this, the method can be restarted, that is, with a new Krylov subspace, with an updated, better initial guess based on the previous iterations. This can result in some stagnation after the restart, but is generally fine for positive-definite matrices (like the mobility matrix). For more information on GMRES, see the original paper by Saad and Schultz[38]

balancing convergence of the residual with computational cost. For this work, the number of inner iterations per restart was set to a fixed value of $k = 100$ for simulations with $N \geq 100$, and it was set to a value of $k = N$ otherwise.

The mobility matrix multiplication approximation applied in this work is based on the spectral Ewald method[39,40]. This method has linear computational complexity in the number of bead, and the relative tolerance for errors in the calculation was set to a value of 10^{-8}. Thus, if the number of iterations for PrCG or GMRES is constant, the algorithms themselves should also be linear in the number of beads. Therefore, the rate at which the number of iterations for each algorithm increases with systems size is indicative of their computational complexity, and for a given system size, the difference in the number of iterations between algorithms is indicative of their relative performance.

To our knowledge, a preconditioner has only been thoroughly developed for the rigid body problem[31]. However, we translate this same approach, described in detail next, to construct a preconditioner for the freely jointed chain and immobile bead problems and find that it leads to linear computational complexity with respect to the number of beads as with iterative solutions of rigid body problems. In this way, we believe we are using competitive preconditioners for all systems and thus making a fair comparison between methods both on the basis of precomputation effort and the subsequent iterative solve. The preconditioner we used was a slight modification of the preconditioner presented by Balboa Usabiaga *et al.*[31], where the mobility matrix is approximated by only considering interactions between beads in the mobility matrix if they are part of the same constrained body:

$$\widetilde{\boldsymbol{\mathcal{M}}}_{ij} = \begin{cases} \boldsymbol{\mathcal{M}}_{ij} & \text{if } b_i = b_j \\ 0 & \text{otherwise} \end{cases}. \tag{2.12}$$

Here, $\widetilde{\boldsymbol{\mathcal{M}}}_{ij}$ is the ij'th 3×3 block of the preconditioner, b_i is the body of particle i, and $\boldsymbol{\mathcal{M}}_{ij}$ is

the *non-periodic* RPY mobility tensor for particles i and j. The preconditioner is then

$$\mathbf{P} = \begin{bmatrix} \widetilde{\boldsymbol{\mathcal{M}}} & -\boldsymbol{\Sigma} \\ -\boldsymbol{\Sigma}^T & \mathbf{0} \end{bmatrix}. \tag{2.13}$$

The LU decomposition[1] of this preconditioner was calculated as a precomputation step using the default sparse LU decomposition function in MATLAB. This preconditioner is known to be effective for rigid bodies and was used in GMRES for rigid bodies, freely jointed chains, and immobile systems (where each bead is its own body and $\widetilde{\boldsymbol{\mathcal{M}}}$ is a diagonal matrix). Because $\boldsymbol{\Sigma}$ is a shift matrix for immobile systems, $\mathbf{P}$ is very sparse and no LU decomposition was needed. To our knowledge, this preconditioner has not been used for systems of freely jointed chains or immobilized beads before. Application of the *non-preconditioned* GMRES method for these systems can take thousands of iterations to converge even for systems with only a few thousand beads, necessitating the use of a preconditioner.

The other precomputation, the QR decomposition of $\boldsymbol{\Sigma}$, can be calculated using various algorithms to orthonormalize the column vectors such as Householder reflections, Givens rotations, or Gram-Schmidt (GS) methods. QR decomposition on a block-diagonal matrix can be done block-wise to accelerate computation, as shown in Appendix B, with a computational complexity of $\mathcal{O}(nm^2)$ for each $n \times m$ block. All QR decompositions were done block-wise with each body corresponding to a block. For rigid bodies, we used an explicit orthonormalization utilizing a GS algorithm. The details of this orthonormalization can be found in Appendix B. We note here that despite GS algorithms typically being disfavored due to loss of orthogonality for matrices with high condition number, this effect is mitigated in this system due to the block-wise structure. For the freely jointed chain system, QR decomposition was done using the default sparse QR decomposition function in MATLAB. For immobile systems, no QR decomposition was required as $\boldsymbol{\Sigma}$ is orthonormal by construction.

Two different wall clock times are reported for each algorithm: the time to execute the iterative

[1]The LU decomposition is another decomposition of a matrix $\mathbf{A}$ such that $\mathbf{A} = \mathbf{LU}$, where $\mathbf{L}$ is a lower triangular matrix and $\mathbf{U}$ is an upper triangular matrix. This decomposition is convenient because inverting triangular matrices is fast, making inversions of $\mathbf{A}$ fast: $\mathbf{A}^{-1}\mathbf{x} = \mathbf{U}^{-1}\mathbf{L}^{-1}\mathbf{x}$ for an arbitrary vector $\mathbf{x}$. We need to invert the preconditioner often in the algorithm, making this precomputation worthwhile.

solve and the time to perform precomputation of the normal space basis (PrCG for all systems and GMRES for the freely jointed chains) and/or the preconditioner (GMRES) for the iterative scheme. For rigid bodies, PrCG requires the analytical QR decomposition of the constraint matrix and GMRES requires the LU decomposition of the preconditioner. Once these matrices have been determined, subsequent matrix solves are faster. Because of the block diagonal nature of the preconditioner, both decompositions are expected to take $\mathcal{O}(N_{\text{bodies}}B^3)$ time. Even though the mobility matrix is SPD, the augmented matrix has negative eigenvalues corresponding to the introduced Lagrange multipliers, and is thus indefinite. This is a general property of saddle point problems. The preconditioner is similarly indefinite in general, so the Cholesky decomposition cannot be used for precomputation.

For freely jointed chains, both algorithms require finding a basis for the left null space of $\mathbf{G}$. This basis was found using a block-wise QR decomposition of $\mathbf{G}$, which resulted in $2B+1$ columns in $\mathbf{\Sigma}$ per chain. Those columns of $\mathbf{\Sigma}$ are orthonormal by construction so $\mathbf{Q} = \mathbf{\Sigma}$. For GMRES, the LU decomposition of the preconditioner still needs to be calculated. Because GMRES requires one additional precomputation step, it is guaranteed to have a higher precomputation time for freely jointed chains. For the simulations that mix free and immobile beads, no precomputation is required.

The residual plots that we present report the monitored residual for each algorithm normalized by its initial value. Both algorithms terminate when the relative residual is below the designated tolerance. For PrCG, the residual is $\|\mathbf{Z}(\boldsymbol{\mathcal{M}}\mathbf{f_c} - \mathbf{b})\|_2$ for projection operator $\mathbf{Z} = \mathbf{I} - \mathbf{Q}\mathbf{Q}^T$ as discussed previously. For GMRES, the residual is preconditioned and uses the augmented mobility matrix: $\|\mathbf{P^{-1}}(\boldsymbol{\mathcal{M}}'\mathbf{x}' - \mathbf{b}')\|_2$, with $\boldsymbol{\mathcal{M}}'$, $\mathbf{x}'$, and $\mathbf{b}'$ defined analogously to Equation 2.4. These residuals are the values both algorithms use to monitor convergence and determine when to stop iterating.

We also report the norm of the error in the constraint forces and constrained velocities normalized by their initial values. These errors were calculated relative to the "exact" solution, which was approximated by running each algorithm to a tighter tolerance of $\epsilon = 10^{-12}$. These plots monitor the convergence of each algorithm to the exact solution of the saddle point problem. Simulations

ranged in size from $\mathcal{O}(10^2)$ to $\mathcal{O}(10^5)$ beads. We perform all calculations in MATLAB R2020b run on a laptop computer with an Intel Core i9 processor. Many of the calculations in this paper including the mobility matrix multiplications, QR decompositions, and preconditioner formulations can be accelerated using parallel programming and GPU's. In this work, we do not implement these methods as the rate-limiting step is the mobility matrix multiply, for which parallel methods have been explored, for example by Fiore *et al.*[29]. We note here, however, that precomputation in both methods is highly parallelizable due to the block-wise nature of the constraint matrix. Parallel versions of the GMRES algorithm have already been implemented, for example see a paper by Fiore and Swan on fast Stokesian dynamics[40]. Parallel implementation of PrCG would be valuable in future work to test its performance versus GMRES is actual simulations.

2.3.2 Systems of rigid bodies

As an example rigid body constraint, we generated geodesic polyhedra by placing beads on the vertices of a tetrahedron or an $[m, 0]$ geodesic icosahedron, with $m = 1, 2, 3, 4$[41]. Geodesic polyhedra are approximations of a sphere with the number of vertices increasing with m. The polyhedra were formed such that beads connected by a short edge were tangent according to their hydrodynamic radii. These types of bodies have been used previously to model suspensions of spherical particles[42]. We varied the number of bodies ($N_{\text{bodies}} = 2^3, 4^3, 6^3, 8^3, 10^3$) with a fixed number of beads per body ($B = 42$), and varied the number of beads per body ($B = 4, 12, 42, 92, 162$) with a fixed number of bodies ($N_{\text{bodies}} = 6^3$). The bodies were placed uniformly at random in the simulation box. The size of the periodic box was adjusted such that the volume fraction of the bodies if they were treated as solid spheres was approximately constant.

The residuals as a function of iteration number for the geodesic polyhedra are shown in Figure 2.3. As the number of bodies increased from $N_{\text{bodies}} = 2^3$ to $N_{\text{bodies}} = 10^3$ with $B = 42$, the number of required iterations increased from about 60 to 110 for PrCG and from about 40 to 60 for GMRES. At $B = 4$, PrCG and GMRES performed nearly equally, at about 30 iterations. As the size of the bodies increased to $B = 162$, the number of required iterations increased to about 120 for PrCG and to about 70 for GMRES. The algorithms separate quickly in residual during the first few iterations, with GMRES converging faster. However, GMRES then tapers and converges

at about the same rate as PrCG. The major difference in the number of iterations, therefore, is in the first few (about 10) steps each algorithm takes. These results suggest that GMRES is superior to PrCG for rigid bodies in terms of the convergence of the measured residual.

Because GMRES solves the augmented mobility matrix system, it attempts to solve for the constraint forces and constrained velocities simultaneously. These values have uneven impacts on the residual and may not converge at equal rates, especially because a good initial guess for both the constraint forces and constrained velocities is not necessarily available for GMRES. Furthermore, GMRES monitors the preconditioned residual, so its convergence is not necessarily indicative of the convergence of the solution to the original saddle point problem equations. This makes the convergence of the actual forces and velocities uncertain.

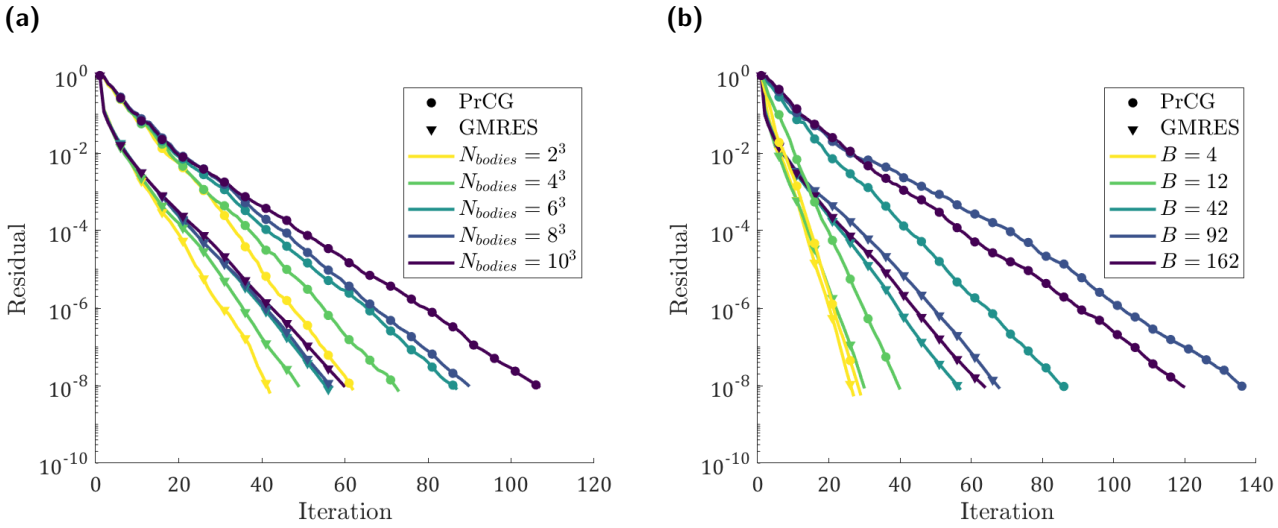

Figure 2.3 Residual ($\|\mathbf{Z}(\boldsymbol{\mathcal{M}}\mathbf{f_c} - \mathbf{b})\|$ for PrCG and $\|\mathbf{P}^{-1}(\boldsymbol{\mathcal{M}}'\mathbf{x}' - \mathbf{b}')\|$ for GMRES) for rigid body geodesic polyhedra systems relative to the residual at the initial guess. (a) Varying the number of bodies, $N_{\text{bodies}} = 2^3, 4^3, 6^3, 8^3, 10^3$, with constant beads per body, $B = 42$. (b) Varying $B = 4, 12, 42, 92, 162$, with constant $N_{\text{bodies}} = 6^3$. Circles correspond to PrCG while triangles correspond to GMRES. Only every 5th iteration is plotted in the scatter for clarity. The system size is denoted by color.

We plot the relative norm of the errors in the constraint forces ($\mathbf{f}_c$) and constrained velocities ($\boldsymbol{\lambda}$) for the rigid bodies in Figure 2.4. For PrCG, the relative error of the constrained velocities from the original formulation (Equation 2.4), $\boldsymbol{\lambda} = \mathbf{R}^{-1}\boldsymbol{\lambda}'$, are calculated for proper comparison with GMRES.

The relative norm of the errors in the constraint forces and constrained velocities are shown in

Figure 2.4. As the number of bodies increases, these errors increase quickly then plateau at about 10^{-7} for GMRES. As B was increased at constant N_{bodies}, these errors also increased to over 10^{-7} for GMRES. For reference, we show the errors at each iteration for each B at constant N_{bodies} in Appendix C.

The residual from the preconditioned GMRES method overestimates the convergence rate of both the constraint forces and constrained velocities. For Brownian dynamics simulations, we want to control the accuracy of the constrained velocities, which are used to calculate the bead velocities: $\mathbf{u} = \boldsymbol{\Sigma}\boldsymbol{\lambda}$. These results mean that, in order to achieve a desired level of accuracy in the constrained velocities, the tolerance of the preconditioned GMRES method must be set to a lower, system-dependent value. This is most likely an artifact of preconditioning, which affects the formulation of the residual but not the values being solved for.

Conversely, PrCG consistently terminated with relative errors for the constraint forces and constrained velocities of about 10^{-8}, which was the desired tolerance, regardless of the number of bodies or the size of the bodies. Because the residual is not preconditioned, only projected, the residual gives an accurate approximation for the true convergence of the constraint forces. The constrained velocities can be determined from these constraint forces after termination through a single mobility matrix multiplication: $\boldsymbol{\lambda} = \mathbf{R}^{-1}\mathbf{Q}^T(\boldsymbol{\mathcal{M}}\mathbf{f_c} - \boldsymbol{\mathcal{M}}\nabla V + \mathbf{u}_B)$.

Precomputation and solve times for both algorithms are shown in Figure 2.5. The mobility matrix multiplications were the dominant calculation in both algorithms, and scale linearly in the number of beads. Therefore, because the number of iterations increased slowly as N_{bodies} increased, the timing for both algorithms is slightly superlinear in N_{bodies}. The number of iterations for both algorithms scaled approximately quadraticly in B. For a given N_{bodies} and B, GMRES required about half the time as PrCG. However, as discussed earlier, GMRES does not converge to the desired tolerance in the constraint forces and constrained velocities, suggesting a smaller difference in the solve time if the same tolerances were reached.

For precomputation, PrCG was linear in N_{bodies}, as expected due to the block-wise nature of the QR decomposition. For these system sizes, precomputation time was nearly constant with B. It is likely that B was not large enough to exhibit clear linear scaling. We find that PrCG

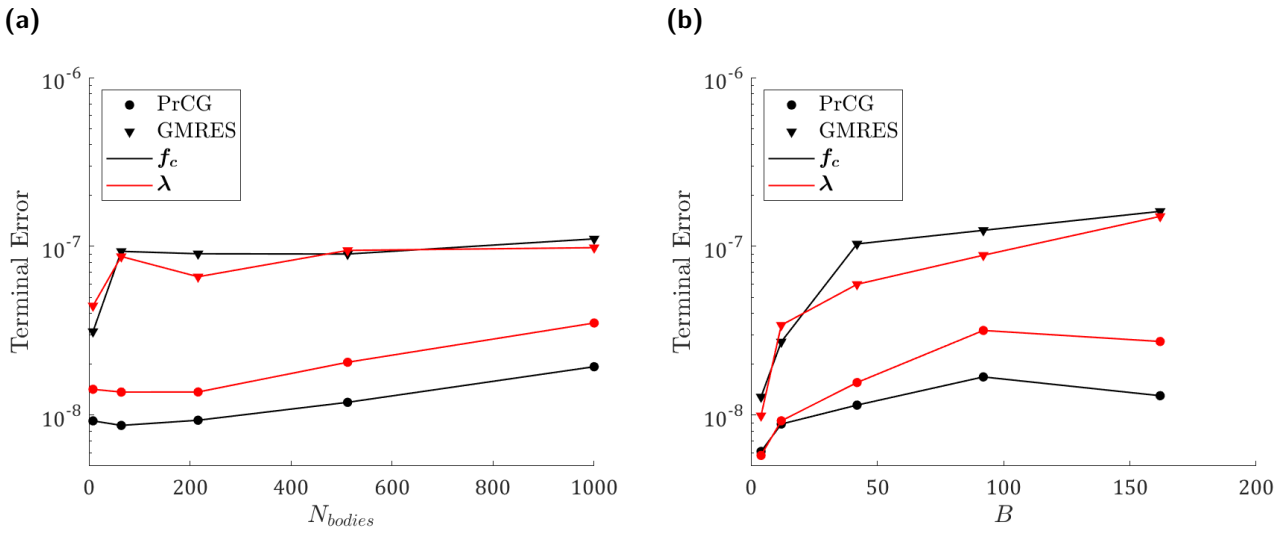

Figure 2.4 Norm of the constraint force ($\mathbf{f}_c$) and constrained velocity ($\boldsymbol{\lambda}$) errors for rigid body geodesic polyhedra systems at algorithm termination relative to the error at the initial guess, (a) varying the number of bodies, $N_{\text{bodies}} = 2^3, 4^3, 6^3, 8^3, 10^3$, with constant beads per body, $B = 42$ and (b) varying $B = 4, 12, 42, 92, 162$, with constant $N_{\text{bodies}} = 6^3$. Circles correspond to PrCG while triangles correspond to GMRES.

performs similarly using the explicit orthonormalization used in this paper when compared to the default sparse QR decomposition function in MATLAB in terms of residuals and errors, but has better precomputation time scaling in B. Precomputation time for GMRES was linear in N_{bodies}, as expected, but worse than quadratic in B. The expected precomputation time for GMRES is $\mathcal{O}(N_{\text{bodies}}B^3)$, so this scaling aligns with expectations, although it is possible that larger B must be sampled to see the cubic scaling more clearly. This poor scaling in B may be rate-limiting for large enough bodies.

2.3.3 Systems of freely jointed chains

For freely jointed chain constraints, we generated chains using a self-avoiding random walk such that no beads in the random walk overlapped with respect to their hydrodynamic radii. This was done to avoid high energy initial configurations of the chains. Neighboring beads were placed such that they were tangent with respect to their hydrodynamic radii. We varied the number of chains ($N_{\text{bodies}} = 2^3, 4^3, 6^3, 8^3, 10^3$) with a fixed number of beads per chain ($B = 50$), and varied the number of beads per chain ($B = 10, 20, 50, 100, 200$) with a fixed number of chains ($N_{\text{bodies}} = 6^3$). Similar to rigid bodies, chains were placed uniformly at random throughout the simulation box.

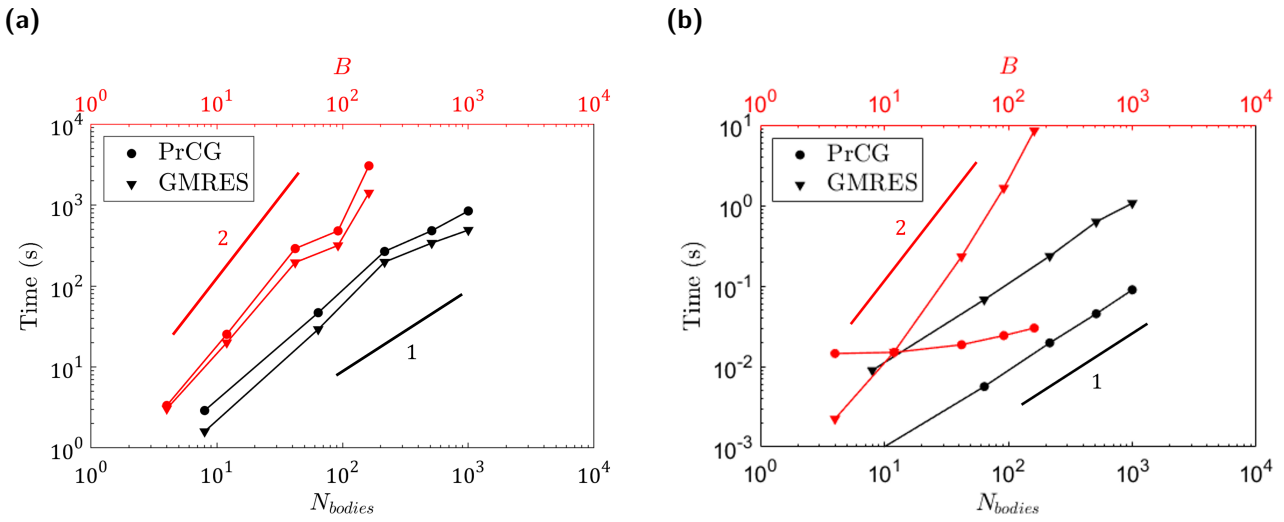

Figure 2.5 (a) Wall clock runtimes and (b) precomputation time for PrCG and GMRES with rigid body geodesic polyhedra systems. For PrCG, precomputation was the QR decomposition of the constraint matrix. For GMRES, it was the LU decomposition of the augmented mobility matrix. For the black points, the number of bodies was varied, $N_{\text{bodies}} = 2^3, 4^3, 6^3, 8^3, 10^3$, with constant beads per body, $B = 42$. For the red points, the beads per body was varied, $B = 4, 12, 42, 92, 162$, with constant number of bodies, $N_{\text{bodies}} = 6^3$.

The periodic simulation box was sized such that the beads had a volume fraction of 20%.

For the freely jointed chains, both GMRES and PrCG exhibited similar performance with respect to the number of iterations and solve times, with PrCG requiring slightly more iterations. The residuals as a function of iteration number are shown in Figure 2.6 and the relative errors in the constraint forces and constrained velocities are shown in Figure 2.7. GMRES and PrCG both achieved relative errors in the constraint forces and constrained velocities of approximately 10^{-8}, matching the convergence of the reported residual. Because the mobility matrix multiplication is the dominant calculation, both algorithms exhibited nearly identical solve times, with both PrCG and GMRES demonstrating roughly linear empirical scaling in both N_{bodies} and B, as shown in Figure 2.8a. For reference, we show the errors for this system at each iteration for each B at constant N_{bodies} in Appendix C.

Relative to the rigid body system, these solves were very fast, showing relatively stable iteration counts with increasing system size and never requiring more than 50 iterations. The precomputation time for GMRES was higher than PrCG in this case because, as discussed earlier, GMRES needs the LU decomposition of its preconditioner to be calculated as well as the same QR decom-

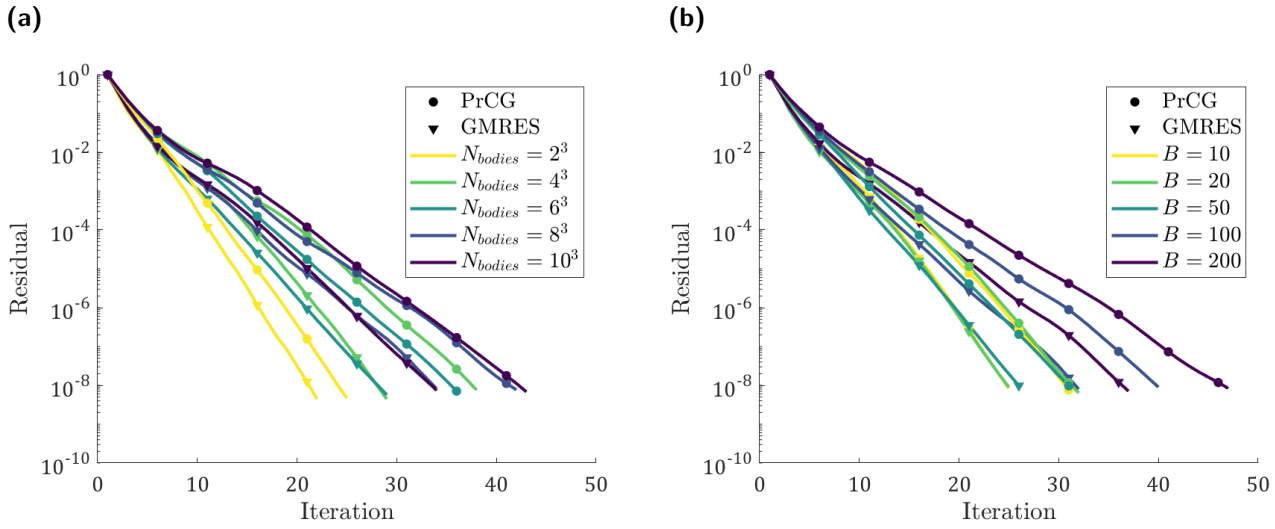

Figure 2.6 Residual ($\|\mathbf{Z}(\boldsymbol{\mathcal{M}}\mathbf{f_c} - \mathbf{b})\|$ for PrCG and $\|\mathbf{P}^{-1}(\boldsymbol{\mathcal{M}}'\mathbf{x}' - \mathbf{b}')\|$ for GMRES) for random walk freely jointed chain systems relative to the residual at the initial guess. (a) Varying the number of bodies, $N_{\text{bodies}} = 2^3, 4^3, 6^3, 8^3, 10^3$, with constant beads per body, $B = 50$. (b) Varying $B = 10, 20, 50, 100, 200$, with constant $N_{\text{bodies}} = 6^3$. Circles correspond to PrCG while triangles correspond to GMRES. Only every 5th iteration is plotted in the scatter for clarity. The system size is denoted by color.

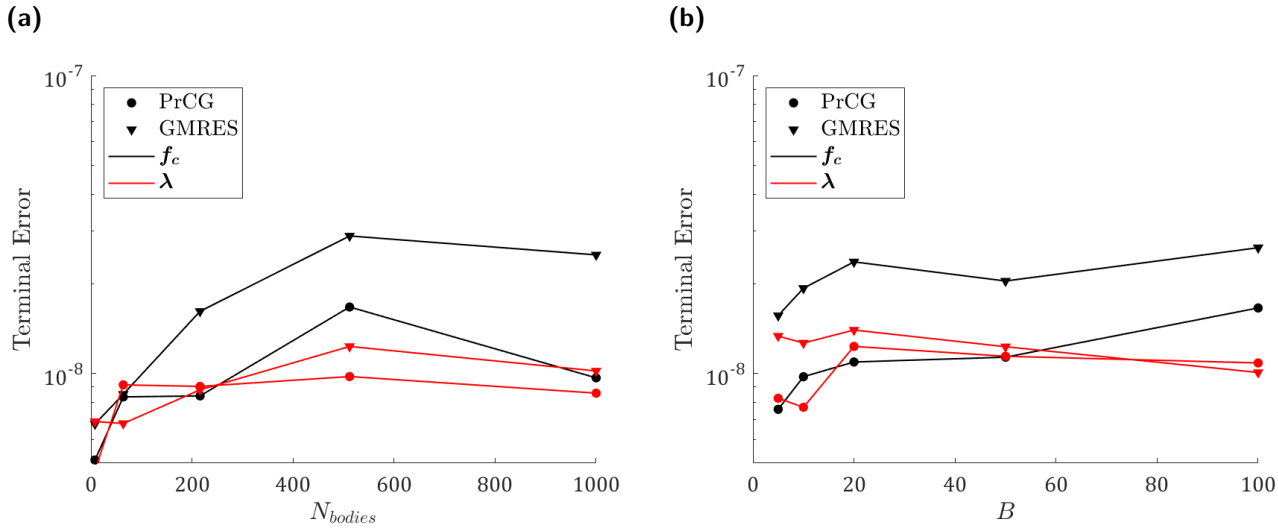

Figure 2.7 Norm of the constraint force ($\mathbf{f}_c$) and constrained velocity ($\boldsymbol{\lambda}$) errors for random walk freely jointed chain systems at algorithm termination relative to the error at the initial guess, (a) varying the number of bodies, $N_{\text{bodies}} = 2^3, 4^3, 6^3, 8^3, 10^3$, with constant beads per body, $B = 50$ and (b) varying $B = 10, 20, 50, 100, 200$, with constant $N_{\text{bodies}} = 6^3$. Circles correspond to PrCG while triangles correspond to GMRES.

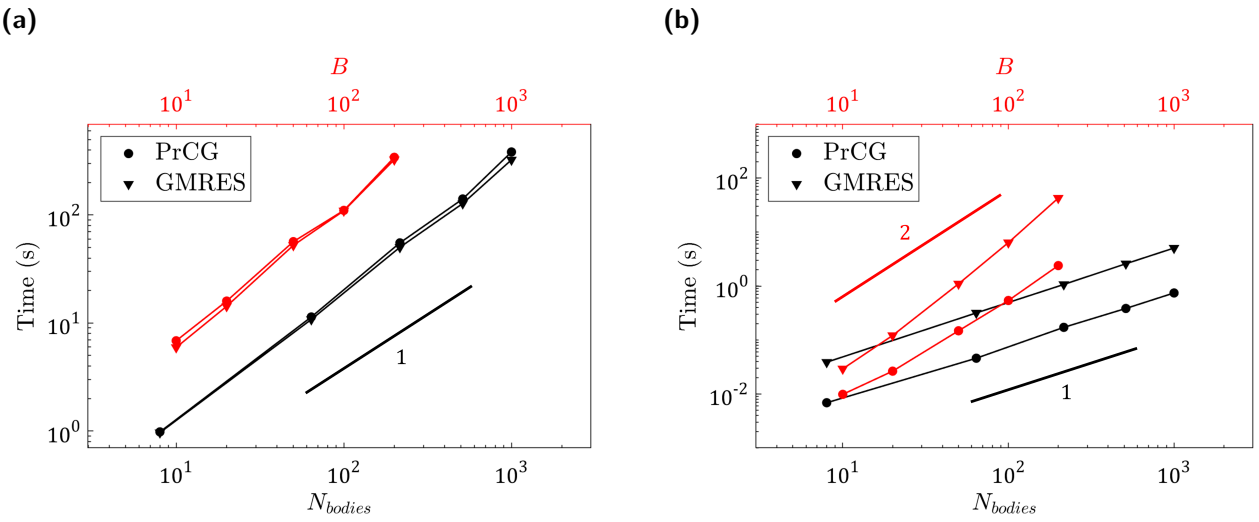

Figure 2.8 (a) Wall clock runtimes and (b) precomputation time for PrCG and GMRES with random walk freely jointed chain systems. For PrCG, precomputation was the QR decomposition of the constraint matrix. For GMRES, it was the QR decomposition of $\mathbf{G}$ and the LU decomposition of the augmented mobility matrix. For the black points, the number of bodies was varied, $N_{\text{bodies}} = 2^3, 4^3, 6^3, 8^3, 10^3$, with constant beads per body, $B = 50$. For the red points, the beads per body was varied, $B = 10, 20, 50, 100, 200$, with constant number of bodies, $N_{\text{bodies}} = 6^3$.

position needed by PrCG. The precomputation time for both PrCG and GMRES were consistent with the expected linear scaling in N_{bodies} and quadratic scaling in B. This quadratic scaling of the precomputation with system size may pose a problem for larger system sizes. Overall, PrCG and GMRES are nearly equivalent in both residual, relative errors, and timing, for the freely jointed chain system, but GMRES requires additional precomputation in calculating the preconditioner.

2.3.4 Systems with immobile particles

Dispersions with a mixture of free and immobile beads were simulated by placing immobile beads in the simulation box either uniformly at random or according to the unit cell of a gyroid with fixed lattice parameters, defining a triply periodic surface. In the case of the gyroid, a random subset of points describing the full gyroid surface were selected and immobile beads were placed at these locations. Then, free beads were placed uniformly at random throughout the interstitial space. For the gyroid immobile system, the total number of beads was selected from the set: $N = 1398, 2797, 5593, 13983, 27966$. Because the lattice parameters were fixed, the volume fraction changed with the total number of beads, up to a maximum volume fraction of 1.5%. In the other

case, termed the "uniform immobile system," all beads were placed uniformly at random. Half of the beads were free and the other half were immobile. The total number of beads was selected from the set: $N = 10^3, 15^3, 20^3, 25^3, 30^3$ with a fixed bead volume fraction of 20%.

The residuals as a function of iteration number for the gyroid immobile systems can be seen in Figure 2.9. PrCG and GMRES required a similar number of iterations at all system sizes, requiring about 150 iterations at the largest systems size ($N = 27966$).

Iteration-to-iteration convergence for GMRES was smooth, but notably stalled at the restart value (100 iterations). Because the number of required iterations increased with system size, this suggests that at large system sizes ($\mathcal{O}(10^5)$ particles and greater), GMRES might slow down relative to PrCG, as either the number of iterations required relative to PrCG will increase due to restarts, or the cost of each iteration relative to PrCG will increase if the iterations between restarts is set too high.

The relative errors in the constraint forces and constrained velocities are shown in Figure 2.10. The relative errors in the constraint forces and constrained velocities converged to about 10^{-8} for all system sizes for both PrCG and GMRES. In this case, however, PrCG seems to do slightly better than GMRES in solving for the constraint forces, but slightly worse than GMRES in solving for the constrained velocities. For reference, we show the errors for this system at each iteration for each N in Appendix C.

For the solve time, both PrCG and GMRES exhibited approximately linear scaling, as shown in Figure 2.11. Because the number of iterations was nearly equal for both algorithms, the algorithm run time was also nearly equal for both algorithms. No precompute time was necessary for either algorithm because the constraint matrix is already orthonormal and the preconditioner for mixtures of free and immobile particles, given by Equation 2.7, is very sparse.

The residuals as a function of iteration number for the uniform immobile systems are shown in Figure 2.12. Compared to other systems, the number of required iterations increased more quickly with system size, to about 450 iterations for both algorithms at the largest system size of $N = 30^3$. This was most likely due to the larger volume fraction of beads (0.2 compared to 0.015 for the gyroid immobile system at $N = 27,966$) and the difference in structure of the immobilized

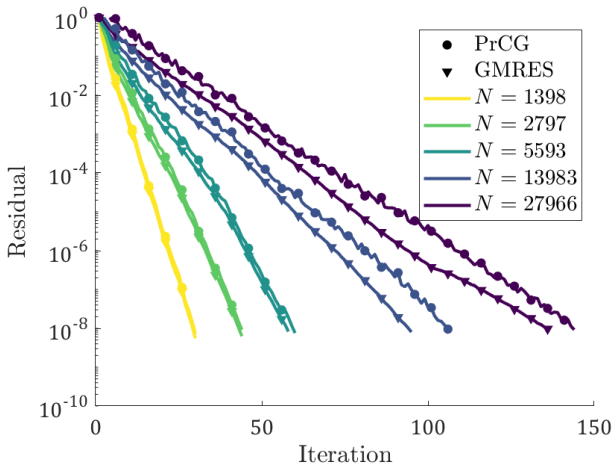

Figure 2.9 Residual ($\|\mathbf{Z}(\boldsymbol{\mathcal{M}}\mathbf{f_c} - \mathbf{b})\|$ for PrCG and $\left\|\mathbf{P}^{-1}(\boldsymbol{\mathcal{M}}'\mathbf{x}' - \mathbf{b}')\right\|$ for GMRES) for gyroid immobile systems relative to the residual at the initial guess. The number of particles was varied from the set $N = 1398, 2797, 5593, 13983, 27966$. Half of the particles were immobile. Circles correspond to PrCG while triangles correspond to GMRES. Only every 5th iteration is plotted in the scatter for clarity. The system size is denoted by color.

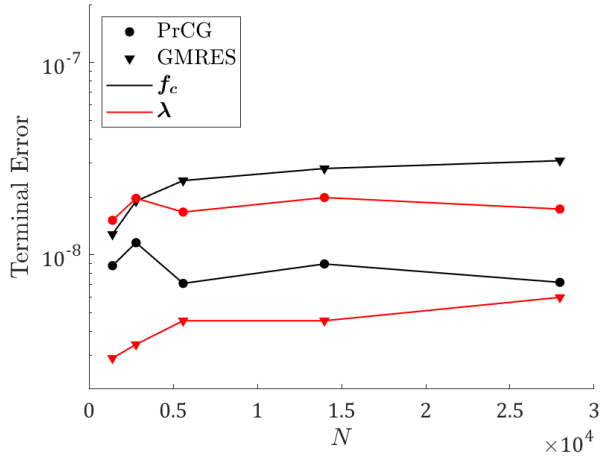

Figure 2.10 Norm of the constraint force ($\mathbf{f}_c$) and constrained velocity ($\boldsymbol{\lambda}$) errors for gyroid immobile systems systems at algorithm termination relative to the error at the initial guess, varying the number of beads from the set $N = 1398, 2797, 5593, 13983, 27966$. Half of the particles were immobile. Circles correspond to PrCG while triangles correspond to GMRES.

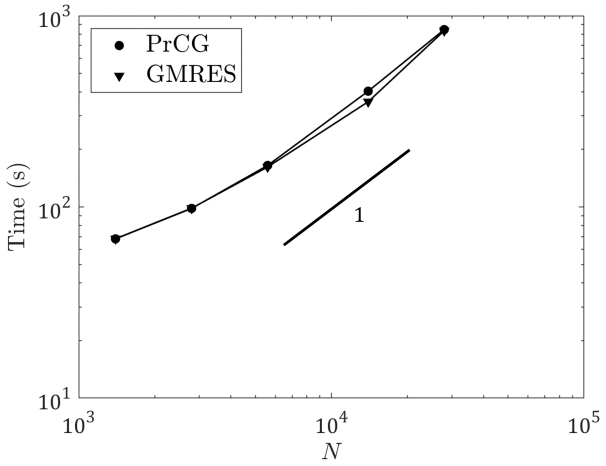

Figure 2.11 Wall clock runtime for PrCG and GMRES with gyroid immobile systems. No precomputation was necessary for either algorithm. The number of particles was varied from the set $N = 1398, 2797, 5593, 13983, 27966$. Half of the particles were immobile.

particles here versus in the gyroid immobile system. Because some of these systems required more than 100 iterations, the effect of restarts, which occur every 100 iterations for GMRES, could be seen clearly. Each restart caused the convergence of GMRES to stall, seen as decreased slopes immediately following the restarts. This is an expected consequence of restarts and increased the number of required iterations for GMRES, worsening as system size increased. At the largest system size of $N = 30^3$, GMRES required nearly 450 iterations while PrCG required less than 400.

GMRES also saw uneven convergence of the constraint forces and constrained velocities, as shown in Figure 2.13, with relative errors in the constrained velocities converging to about 10^{-8} for all system sizes, but with relative errors in the constraint forces increasing from about 10^{-8} to 10^{-7} with increasing system size. Meanwhile, PrCG achieved relative errors of approximately 10^{-8} regardless of system size and immobile particle configuration, suggesting that it is superior to GMRES for these immobile systems. For the solve time, both PrCG and GMRES exhibited superlinear scaling with respect to N due to the increasing number of iterations with system size, as shown in Figure 2.14. For reference, we show the errors for this system at each iteration for each N in Appendix C.

For both the gyroid and uniform immobile systems, the number of required iterations increased with the fraction of immobile particles at a constant total number of particles. Therefore, there might be a trade-off for these systems between algorithm run time and resolution of an immobile

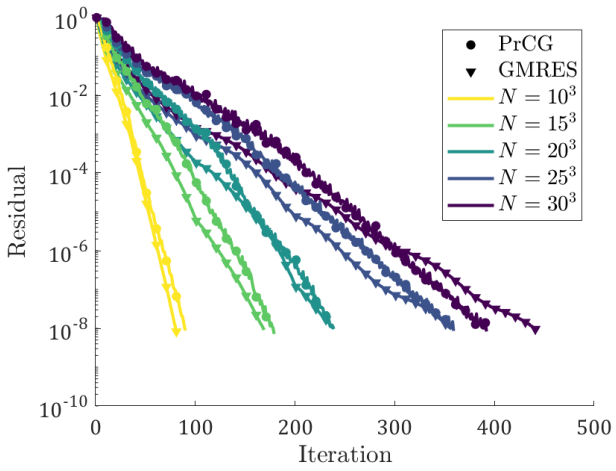

Figure 2.12 Residual ($\|\mathbf{Z}(\boldsymbol{\mathcal{M}}\mathbf{f_c} - \mathbf{b})\|$ for PrCG and $\left\|\mathbf{P^{-1}}(\boldsymbol{\mathcal{M}}'\mathbf{x}' - \mathbf{b}')\right\|$ for GMRES) for uniform immobile systems relative to the residual at the initial guess. The number of particles was varied from the set $N = 10^3, 15^3, 20^3, 25^3, 30^3$. Half of the particles were immobile. Circles correspond to PrCG while triangles correspond to GMRES. Only every 10th iteration is plotted in the scatter for clarity. The system size is denoted by color.

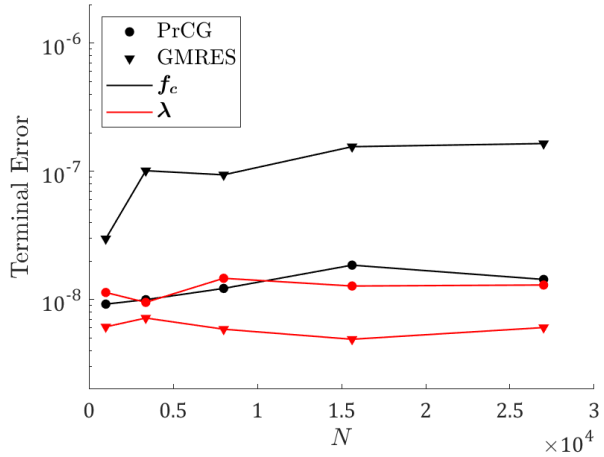

Figure 2.13 Norm of the constraint force ($\mathbf{f}_c$) and constrained velocity ($\boldsymbol{\lambda}$) errors for uniform immobile systems systems at algorithm termination relative to the error at the initial guess, varying the number of beads from the set $N = 10^3, 15^3, 20^3, 25^3, 30^3$. Half of the particles were immobile. Circles correspond to PrCG while triangles correspond to GMRES.

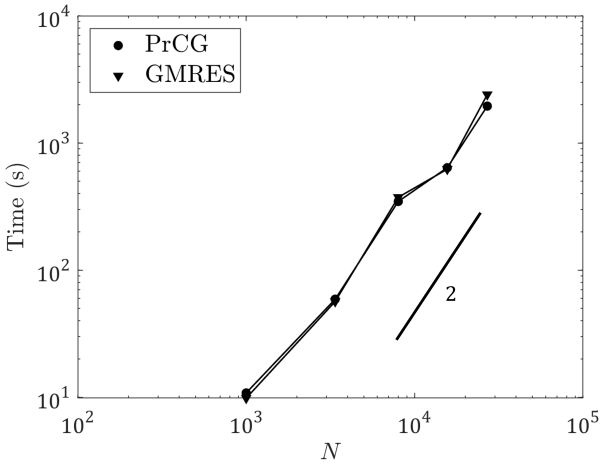

Figure 2.14 Wall clock runtime for PrCG and GMRES with uniform immobile systems. No precomputation was necessary for either algorithm. The number of particles was varied from the set $N = 10^3, 15^3, 20^3, 25^3, 30^3$. Half of the particles were immobile.

boundary (the number of beads used to represent it). Practically, this means that decreasing the resolution of an immobile boundary decreases computational cost more than removing the same number of free particles from the simulation.

2.3.5 Storage costs

We mention here the theoretical storage costs for each algorithm, tabulated in Table 2.1. For precomputation, PrCG requires the storage of $\mathbf{Q}$ and $\mathbf{R}$, which require $\mathcal{O}(3N_{bodies}BP_B)$ and $\mathcal{O}(N_{bodies}P_B^2)$ storage, respectively, where P_B is the characteristic degrees of freedom per body which can range from 1 to $3B$ depending on the system (e.g. $P_B = 6$ for a non-symmetric rigid body). GMRES requires the storage of the preconditioner, which requires $\mathcal{O}(9N_{bodies}B^2)$ storage. This storage is greater than for PrCG because $3B \geq P_B$. This difference comes from the storage of $\widetilde{\boldsymbol{\mathcal{M}}}$ as opposed to $\mathbf{R}$. In the case that the matrix $\widetilde{\boldsymbol{\mathcal{M}}}$ is not stored directly but is instead calculated as a matrix-vector product, only the particle positions are needed and this cost can be removed. Similarly, in cases where $\mathbf{c} = \mathbf{0}$, such as in these simulations, $\mathbf{R}$ is not needed for calculating particle velocities, and its cost can be removed as well. Overall, this means that pre-computational storage costs for PrCG and GMRES are similar, only requiring the storage of $\mathbf{Q}$ and $\boldsymbol{\Sigma}$, respectively, which share the same dimensions and similar sparsity.

For the algorithms themselves, PrCG requires $\mathcal{O}(N)$ storage for various vectors which update

with each iteration. GMRES, on the other hand, requires $\mathcal{O}(\min(k,i)(N+P))$ storage for each of the Arnoldi vectors, where k is the restart value and i is the current iteration number. This means that the relative cost of GMRES increases with the number of iterations.

Overall, both algorithms are similar in both precomputation and algorithm storage costs, with the specifics depending on the system. However, problems which require many iterations for convergence may accrue significant storage costs for GMRES.

	PrCG	GMRES
Precomputation	$\mathcal{O}(3N_{bodies}P_B B)$	$\mathcal{O}(9N_{bodies}B^2)$
Per iteration	$3N$	$\mathcal{O}(\min(k,i)(3N+P))$

Table 2.1 Storage costs for PrCG and GMRES. Precomputation is the cost for storing $\mathbf{Q}$ and $\mathbf{R}$ for PrCG and the $\mathbf{P}$ for GMRES. Per iteration is the additional cost for the algorithm. P_B is the characteristic degrees of freedom per body, k is the restart value of GMRES, and i is the iteration number.

2.3.6 Simulations involving multiple types of constraints

An advantage of using the PrCG method to solve saddle point problems in Brownian dynamics simulations is the ease of incorporating multiple types of constraints. As an additional test, a non-periodic system with both a rigid body and a collection of immobile beads was constructed and solved using PrCG. In this example, a 252mer (a [5, 0] geodesic polyhedron constructed as described in Section 2.3.2) was made a rigid body and centered at the origin. The Stokes radius of the 252mer was determined by applying a force, $\mathbf{F}$, distributed equally amongst the beads, and determining the resultant rigid body velocity, $\mathbf{U}$, with a constrained solve using PrCG. The Stokes radius can then be determined using the Stokes drag formula: $r_s = \frac{\|\mathbf{F}\|}{6\pi\eta\|\mathbf{U}\|}$.

Then, another 252mer was used to specify the coordinates for a set of immobile beads centered a distance b away from the origin along the x-axis. A force was then applied to the rigid body and distributed equally amongst the beads. The velocity of the rigid body is altered by hydrodynamic interactions with the immobile beads, which behave collectively like another sphere held fixed in space. At the same time, a force is induced on the immobile beads due to the motion of the rigid body, and it is the constraint forces that precisely balance these hydrodynamic forces. We sum those constraint forces on the immobile beads to determine the force holding the immobile body in place. When $b/r_s \gg 1$, the asymptotic analytical solution for the resultant velocity on the rigid

body, $\mathbf{U}_1$, and summed force on the collection immobile beads, $\mathbf{F}_2$, are

$$\mathbf{U}_1 = \frac{1}{6\pi\eta r_s}\left[\left(1 - \frac{9}{4}\left(\frac{r_s}{b}\right)^2\right)\hat{\mathbf{r}}\hat{\mathbf{r}} + \left(1 + \frac{9}{16}\left(\frac{r_s}{b}\right)^2\right)(\mathbf{I} - \hat{\mathbf{r}}\hat{\mathbf{r}})\right] \cdot \mathbf{F}_1 + \mathcal{O}\left(\frac{r_s}{b}\right)^3, \tag{2.14}$$

$$\mathbf{F}_2 = -\left[\frac{3}{2}\frac{r_s}{b}\hat{\mathbf{r}}\hat{\mathbf{r}} + \frac{3}{4}\frac{r_s}{b}(\mathbf{I} - \hat{\mathbf{r}}\hat{\mathbf{r}})\right] \cdot \mathbf{F}_1 + \mathcal{O}\left(\frac{r_s}{b}\right)^3, \tag{2.15}$$

where $\hat{\mathbf{r}}$ is the unit vector pointing from the center of one body to the other[43]. Two different forces applied to the rigid body were considered: a force parallel to the x-axis ($\mathbf{F}_1^{\parallel} = F_1[1,0,0]^T$) and a force perpendicular to the x-axis ($\mathbf{F}_1^{\perp} = F_1[0,1,0]^T$).

As the distance b becomes large, the velocity of the rigid body approaches the velocity in the absence of the immobile beads: $\mathbf{U}_1 = \frac{\mathbf{F}_1}{6\pi\eta r_s}$. In the same circumstance, the force on the immobile beads approaches zero. From the analytical solutions, we expect the difference in the rigid body velocity from its velocity in absence of the immobile beads to increase with $(r_s/b)^2$ and the force on the immobile beads to increase with (r_s/b) for large separations.

The rigid body velocity and immobile particle forces were calculated using PrCG for various dimensionless separations, (r_s/b). The results are shown in Figure 2.15. As expected, for both the parallel and perpendicular forces, the rigid body velocity scales with $(r_s/b)^2$ and the immobile particle force scales with (r_s/b). Furthermore, when the rigid body and immobile beads are in contact, the rigid body should be unable to move and the immobile beads should experience a constraint force equal in magnitude and opposite in direction to the force applied to the rigid body, as seen in the results at the largest (r_s/b), for which the rigid body and immobile bead groups were tangent, and for which the asymptotic analytical model breaks down. Because the immobile beads in the system with the perpendiular force are not directly obstructing the rigid body, the velocity and force for this system are lower for a given (r_s/b), but both drop to zero once the constrained elements are in contact. Importantly, once the constraint matrix has been identified, no additional work is necessary for precomputaiton in this mixed constraint system. The construction of $\mathbf{\Sigma}$ is straightforward and $\mathbf{Q}$ can be calculated efficiently using a block-wise QR decomposition. This contrasts with GMRES, for which a novel preconditioner may need to be developed for such a

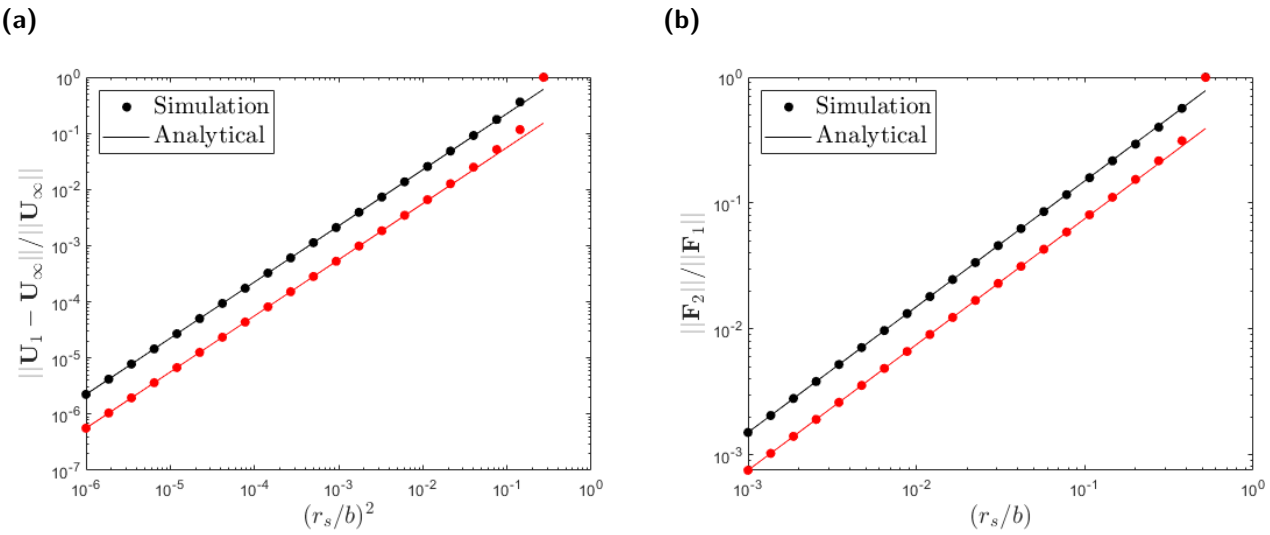

Figure 2.15 (a) Normalized difference between rigid body velocity with the immobile beads and its velocity at infinite separation, $\mathbf{U}_\infty = \frac{\mathbf{F}_1}{6\pi\eta r_s}$. (b) Total force on the immobile beads. Results from the simulation are shown as points, and the analytical solution is shown as a solid line. The rightmost point corresponds to the rigid body and immobile bead groups being in contact. Black points correspond to a force parallel to the x-axis ($\mathbf{F}_1^{\parallel} = F_1[1,0,0]^T$). Red points correspond to a force perpendicular to the x-axis ($\mathbf{F}_1^{\perp} = F_1[0,1,0]^T$).

system which combines multiple types of constraints.

2.4 Conclusions

In this chapter, we show that saddle point problems in Brownian dynamics simulations can be solved efficiently using the projected conjugate gradient (PrCG) method with certain advantages over the typically utilized GMRES method. Specifically, much of the complexity of the PrCG algorithm can be front-loaded into the QR decomposition of the constraint matrix, which can be performed efficiently for block diagonal constraint matrices. For GMRES, a preconditioner must be calculated, which is often the most storage-intensive step and can be expensive computationally as well. That said, the preconditioner presented by Balboa Usabiaga *et al.*[31] is found to be effective for freely-jointed chains and immobile systems as well as for its original application of rigid bodies. GMRES also often requires the same QR decomposition as PrCG to generate the constraint matrix. In all systems studied in this work, precomputation was cheaper for PrCG compared to GMRES.

PrCG has been shown to exhibit the same empirical linear computational complexity as GMRES for simple Brownian rigid bodies, freely jointed chains, and systems of immobile particles.

Furthermore, its residual reliably predicts the relative error in both the constraint forces and constrained velocities. For GMRES, the reported residual was not always indicative of the relative errors in the constraint forces and constrained velocities. The offset between the reported residual and true convergence was a function of the constraint type, system size, and particle configuration, making it difficult to determine the offset *a priori*. These issues with GMRES suggest an additional (often overlooked) aspect of developing preconditioners for these systems: the reported residual may not always directly correlate with the error in the solutions themselves.

The iterates produced by PrCG are guaranteed to be feasible, which is not true for GMRES. For GMRES, initial guesses which are infeasible should converge on this constraint in a similar manner to residual convergence. In these cases, setting a tolerance for GMRES simultaneously sets the tolerance for feasibility, even though this should ideally be satisfied exactly. This means that extra computational effort needs to be expended to make the solution feasible. For example, one could converge the residual past where the velocities have converged to the desired tolerance such that feasibility is satisfied to the desired tolerance. Alternatively, one could generate an initial guess which is feasible, for example by using a QR decomposition of the constraint matrix to transform the saddle point problem to the form used for PrCG and using the same initial guess as PrCG (although this does not guarantee that future iterates continue to be feasible). These methods both require extra effort in either solve or precomputation time. PrCG guarantees feasibility at each iteration, requiring only the QR decomposition.

PrCG trivially extends to simulations with rigid bodies of different shapes and sizes and/or different types of constraints without the need for careful formulation or storage of a preconditioner. In many constrained systems, the formulation of a preconditioner comprises much of the work in developing the model, and PrCG obviates the need to develop one for new constrained systems.

In summary, we find that PrCG is an effective algorithm for conducting constrained Brownian dynamics simulations. It exhibits the same empirical computational complexity as GMRES and is more easily extended and robust to different types of constrained systems.

Bibliography

[1] D Fritz, VA Harmandaris, K Kremer, and NFA Van Der Vegt. Coarse-grained polymer melts based on isolated atomistic chains: Simulation of polystyrene of different tacticities. *Macromolecules*, 42(19):7579–7588, 2009.

[2] O Maxian, A Mogilner, and A Donev. Integral-based spectral method for inextensible slender fibers in stokes flow. *Physical Review Fluids*, 6(1):014102, 2021.

[3] F Balboa Usabiaga and B Delmotte. A numerical method for suspensions of articulated bodies in viscous flows. *Journal of Computational Physics*, 464:111365, 2022.

[4] A Vazquez-Quesada, F Balboa Usabiaga, and R Delgado-Buscalioni. A multiblob approach to colloidal hydrodynamics with inherent lubrication. *The Journal of Chemical Physics*, 141(20), 2014.

[5] M Doi and SF Edwards. *The theory of polymer dynamics*, volume 73. Oxford University Press, 1988.

[6] W Bruns and R Bansal. Molecular dynamics study of a single polymer chain in solution. ii. bead–spring model. *The Journal of Chemical Physics*, 75(10):5149–5152, 1981.

[7] R Kumar Sharma, I Agrawal, L Dai, PS Doyle, and S Garaj. Complex dna knots detected with a nanopore sensor. *Nature Communications*, 10(1):4473, 2019.

[8] HC Öttinger. Brownian dynamics of rigid polymer chains with hydrodynamic interactions. *Physical Review E*, 50(4):2696, 1994.

[9] H Goldstein. *Classical mechanics*. Pearson Education India, 2011.

[10] R Froese and I Herbst. Realizing holonomic constraints in classical and quantum mechanics. *Communications in Mathematical Physics*, 220:489–535, 2001.

[11] E Hairer. Solving differential equations on manifolds. *Lecture Notes*, 2011.

[12] P Betsch. The discrete null space method for the energy consistent integration of constrained mechanical systems: Part i: Holonomic constraints. *Computer Methods in Applied Mechanics and Engineering*, 194(50-52):5159–5190, 2005.

[13] JP Ryckaert, G Ciccotti, and HJC Berendsen. Numerical integration of the cartesian equations of motion of a system with constraints: molecular dynamics of n-alkanes. *Journal of Computational Physics*, 23(3):327–341, 1977.

[14] HC Andersen. Rattle: A "velocity" version of the shake algorithm for molecular dynamics calculations. *Journal of Computational Physics*, 52(1):24–34, 1983.

[15] B Hess, H Bekker, HJC Berendsen, and JGEM Fraaije. Lincs: a linear constraint solver for molecular simulations. *Journal of Computational Chemistry*, 18(12):1463–1472, 1997.

[16] V Kräutler, WF Van Gunsteren, and PH Hünenberger. A fast shake algorithm to solve distance constraint equations for small molecules in molecular dynamics simulations. *Journal of Computational Chemistry*, 22(5):501–508, 2001.

[17] B Hess. P-lincs: A parallel linear constraint solver for molecular simulation. *Journal of Chemical Theory and Computation*, 4(1):116–122, 2008.

[18] J Yen. Constrained equations of motion in multibody dynamics as odes on manifolds. *SIAM Journal on Numerical Analysis*, 30(2):553–568, 1993.

[19] JW Swan and G Wang. Rapid calculation of hydrodynamic and transport properties in concentrated solutions of colloidal particles and macromolecules. *Physics of Fluids*, 28(1), 2016.

[20] J Nocedal and S Wright. *Numerical optimization*. Springer Science & Business Media, 2006.

[21] NIM Gould, ME Hribar, and J Nocedal. On the solution of equality constrained quadratic programming problems arising in optimization. *SIAM Journal on Scientific Computing*, 23(4):1376–1395, 2001.

[22] KS Silmore and JW Swan. Locally feasibly projected sequential quadratic programming for nonlinear programming on arbitrary smooth constraint manifolds. *arXiv preprint arXiv:2111.03236*, 2021.

[23] TW Liu. Flexible polymer chain dynamics and rheological properties in steady flows. *The Journal of Chemical Physics*, 90(10):5826–5842, 1989.

[24] JE Butler and ESG Shaqfeh. Brownian dynamics simulations of a flexible polymer chain which includes continuous resistance and multibody hydrodynamic interactions. *The Journal of Chemical Physics*, 122(1), 2005.

[25] G Ciccotti, T Lelievre, and E Vanden-Eijnden. Projection of diffusions on submanifolds: Application to mean force computation. *Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences*, 61(3):371–408, 2008.

[26] J Rotne and S Prager. Variational treatment of hydrodynamic interaction in polymers. *The Journal of Chemical Physics*, 50(11):4831–4837, 1969.

[27] EE Keaveny. Fluctuating force-coupling method for simulations of colloidal suspensions. *Journal of Computational Physics*, 269:61–79, 2014.

[28] AM Fiore, F Balboa Usabiaga, A Donev, and JW Swan. Rapid sampling of stochastic displacements in brownian dynamics simulations. *The Journal of Chemical Physics*, 146(12), 2017.

[29] AM Fiore and JW Swan. Rapid sampling of stochastic displacements in brownian dynamics simulations with stresslet constraints. *The Journal of Chemical Physics*, 148(4), 2018.

[30] B Sprinkle, F Balboa Usabiaga, NA Patankar, and A Donev. Large scale brownian dynamics of confined suspensions of rigid particles. *The Journal of Chemical Physics*, 147(24), 2017.

[31] F Balboa Usabiaga, B Kallemov, B Delmotte, et al. Hydrodynamics of suspensions of passive and active rigid particles: a rigid multiblob approach. *Communications in Applied Mathematics and Computational Science*, 11(2):217–296, 2017.

[32] G Wang, Z Varga, J Hofmann, IE Zarraga, and JW Swan. Structure and relaxation in solutions of monoclonal antibodies. *The Journal of Physical Chemistry B*, 122(11):2867–2880, 2018.

[33] PK Lai, JW Swan, and BL Trout. Calculation of therapeutic antibody viscosity with coarse-grained models, hydrodynamic calculations and machine learning-based parameters. *mAbs*, 13(1):1907882, 2021.

[34] D Gabay. Minimizing a differentiable function over a differential manifold. *Journal of Optimization Theory and Applications*, 37:177–219, 1982.

[35] GP McCormick. *Nonlinear programming: Theory, algorithms, and applications*. John Wiley & Sons, Inc., 1983.

[36] A Stukowski. Visualization and analysis of atomistic simulation data with ovito–the open visualization tool. *Modelling and Simulation in Materials Science and Engineering*, 18(1):015012, 2009.

[37] K Yeo and MR Maxey. Simulation of concentrated suspensions using the force-coupling method. *Journal of Computational Physics*, 229(6):2401–2421, 2010.

[38] Y Saad and MH Schultz. Gmres: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing*, 7(3):856–869, 1986.

[39] M Wang and JF Brady. Spectral ewald acceleration of stokesian dynamics for polydisperse suspensions. *Journal of Computational Physics*, 306:443–477, 2016.

[40] AM Fiore and JW Swan. Fast stokesian dynamics. *Journal of Fluid Mechanics*, 878:544–597, 2019.

[41] MJ Wenninger. *Spherical models*, volume 3. Courier Corporation, 1999.

[42] G Wang and JW Swan. Surface heterogeneity affects percolation and gelation of colloids: dynamic simulations with random patchy spheres. *Soft Matter*, 15(25):5094–5108, 2019.

[43] S Kim and SJ Karrila. *Microhydrodynamics: principles and selected applications*. Butterworth-Heinemann, 2013.

Chapter 3

Primer on rheology and 2D materials

In this chapter, we give a background on rheology and 2D materials, which will be the focus of the remainder of this thesis. We begin by discussing non-Newtonian fluids, specifically their applications and affect on the momentum balance. Then, we give a brief introduction on simple shear, a common rheological technique, and discuss how it is applied to our simulations. Finally, we give a brief introduction to 2D materials, their applications, as well as previous experimental, theoretical, and computational work.

3.1 Introduction to rheology

Rheology is the study of flow. Specifically, much of rheology is interested in the study of non-Newtonian fluids[a]. Recall that Newtonian fluids have a constant viscosity with respect to the strain rate, so a non-Newtonian fluid is any fluid with a non-constant viscosity with respect to the strain rate. We will explore the consequences of this in the following sections. We start by presenting some common non-Newtonian fluids and their basic properties. Then, we give a common rheological experiment which is used to find the fluid viscosity and explaining how it is implemented in simulation. Finally, we introduce the basic math connecting these experiments to the fluid viscosity.

[a] As they are richer and more complicated than Newtonian fluids

3.1.1 Types of non-Newtonian fluids

We note here that the study of non-Newtonian fluids is not purely academic, as such fluids are ubiquitous in daily life. While many common fluids (such as air and water) are Newtonian[b], there are many non-Newtonian fluids as well which exhibit more complex and interesting behavior than Newtonian fluids. Ketchup, for example, will not come out of its bottle until the bottle is squeezed (i.e. a strain is applied). Ketchup thus behaves solid-like at low strain rates but like a fluid at higher strain rates. This is called a *Bingham plastic* or yield stress fluid. Many other food products as well as care products (often kept in bottles) are Bingham plastics. For example, ketchup, mustard, other condiments, shampoos and lotions, and blood all fall under this category. Often, this has to do with some solid-like structure in the fluid which is broken down by large enough strain rates.

Hand sanitizer sits pretty nicely in the hands, but once it is rubbed (i.e. sheared), it flows more easily. Its viscosity thus decreases with strain rate. This is called a shear-thinning fluid or pseudoplastic. Many of the examples for Bingham plastics, such as shampoo and blood, are also shear-thinning. Again, this often has to do with some structure in the fluid which is broken down with increasing strain rates, causing the fluid to flow more easily.

Oobleck, a mixture of cornstarch and water, flows like a liquid under low strain rates. However, under high strain rates, its viscosity increases drastically[c]. Such a fluid is called a shear-thickening fluid or dilatant. One way shear-thickening might arise is some "jamming" mechanism, where large enough strain rates lead to particles being too tightly packed to flow properly.

These (Bingham plastics, pseudoplastics, and dilatants) make up the "basic" non-Newtonian fluids and are depicted in Figure 3.1. Many of these fluids may also exhibit thixotropy, which is, roughly, time-dependent shear-thinning behavior[d], causing the entire flow history to affect the viscosity (and making modeling much more difficult). There are also fluids which can exhibit both shear-thinning and shear-thickening depending on the shear rate. Such a fluid will be the topic of the next chapter. In order to design these materials to behave in the ways we wish, it is important

[b] At least in the range of conditions we typically care about.
[c] This has led to fun videos where people can run across pools of oobleck.
[d] There are often debates on the exact definition of thixotropy, so I will avoid defining it with certainty.

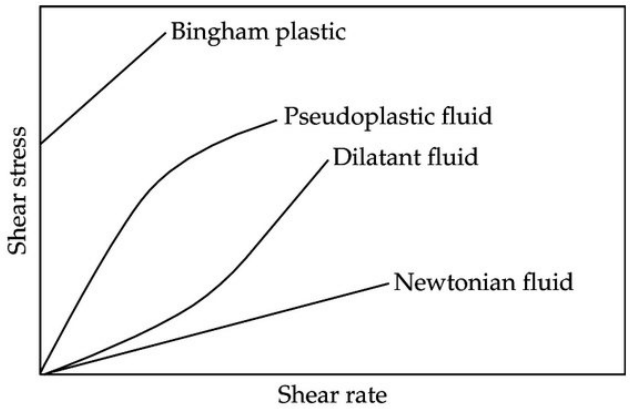

Figure 3.1 Basic non-Newtonian fluids. Recall that the viscosity is proportional to the slope of the lines. Reproduced from Mehrabi and Setayeshi[1].

to understand non-Newtonian fluids, both their macroscopic properties such as viscosity and how their composition causes these macroscopic properties. As discussed previously, a common method of studying the microscopic origins of macroscopic fluid behaviors is simulation.

3.1.2 Simple shear

Often, we wish to run simulations which mimic experiments to compare results. This equates to applying the same velocity gradient to our simulations as those applied in experiments. One common rheological experiment (performed in "rheometers") is simple shear (we will frequently refer to simple shear as just "shear"). In a simple shear experiment, a fluid is placed in a small gap between two parallel plates[e]. One of these plates moves while the other remains stationary. Because the fluid grips the plates, this creates a velocity gradient, as depicted in Figure 3.2. For small enough gaps, the profile will be linear, so the velocity gradient only has one non-zero component given by the plate velocity, u_x, divided by the gap between the plates, H:

$$L_{xy} \equiv \frac{\partial u_x}{\partial y} = u_x/H \equiv \dot{\gamma}, \tag{3.1}$$

[e]There are many slight variations, such as a cone and a plate (called a cone-plate rheometer) instead of two plates (called a parallel plate rheometer), but the basic concept is the same.

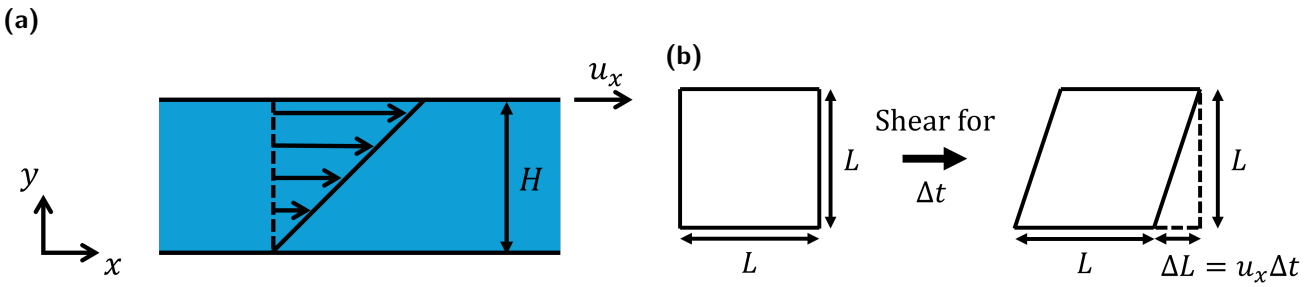

Figure 3.2 (a) Depiction of simple shear and (b) deformation of a fluid element in simple shear.

where $\mathbf{L}$ is the velocity gradient tensor[f] and $\dot{\gamma}$ is the *shear rate*[g], which has units of inverse time. A fluid element in simple shear is deformed at a rate determined by the shear rate. Specifically, a differential cube in the fluid becomes a parallelpiped after shearing. This process is depicted in Figure 3.2. The *strain* which has been applied via shear is the total length of the fluid element relative to its original length: $(L + \Delta L)/L = \dot{\gamma} t$. The *strain rate* is the rate at which this changes over time: $\dot{\gamma}$. We will use these terms in the future.

Similar profiles are common in real applications. For example, toothpaste near its container's walls experiences shear. This is why it's so difficult to get all the toothpaste out of its container; the container wall is a stationary surface so there is stationary fluid next to it, requiring very large shear rates to remove most of it. Many lubrication processes, which involve thin fluid layers by definition, also involve simple shear.

Other common imposed velocity gradients are extensional flows, where fluid is stretched in some directions and compressed in others. Such fluid kinematics might model an extrusion process or chewing of certain foods. Applying different flow deformation protocols can give different information about a fluid. In this thesis, our results focus on applied simple shear.

3.1.3 Applying simple shear to simulations

One convenient aspect of simple shear (and many other flow types) is that it is self-similar in the sense that the shear rate is the same no matter where we are in the fluid or how close we zoom in.

[f] L_{ij} is the ij'th component of $\mathbf{L}$.

[g] Conceptually, simple shear consists of a deformation (parallel planes of fluid slide relative to each other) and a rotation (in Figure 3.2, this rotation is clockwise). This rotation is about the plane of flow and velocity gradient. Thus, we can refer to simple shear with respect to three axes: the flow axis in the direction of flow, the shear axis in the direction of the velocity gradient, and the vorticity axis about which the fluid rotates. Pure shear refers to shear with no rotation.

Thus, when we simulate simple shear, we need not simulate an entire rheometer. Rather, we can simulate a smaller, characteristic piece of the fluid and apply the overall shear rate to it (by giving each particle an x velocity proportional to its y coordinate)[h]. The behavior of this piece of fluid should be the same for the rest of the simulation box due to self-similarity. Thus, the equations of motion (Equation 1.20) become

$$\Delta\mathbf{x} = \Delta t(-\boldsymbol{\mathcal{M}}\boldsymbol{\nabla}V + \mathbf{u}_B + \mathbf{L}\mathbf{x}). \tag{3.2}$$

3.1.4 The new momentum balance

We now discuss how the fluid viscosity can be extracted from a simple shear experiment or simulation. A more general version of the Navier-Stokes equation, discussed in Chapter 1, which does not assume constant viscosity[i] is the Cauchy momentum equation:

$$\rho\frac{D\mathbf{u}}{Dt} = -\boldsymbol{\nabla}p + \boldsymbol{\nabla}\cdot\boldsymbol{\tau} + \mathbf{f}_b, \tag{3.3}$$

where ρ is the fluid density, $\frac{D\mathbf{u}}{Dt}$ is the material derivative of the fluid velocity, $\mathbf{u}$, as described previously, p is the pressure, $\mathbf{f}_b$ are the body forces (per unit volume) on the fluid, and $\boldsymbol{\tau}$ is the *deviatoric stress tensor*[j]. This stress tensor term replaces the viscous term from the Navier-Stokes equation (Equation 1.4). We can relate the stress tensor to the fluid viscosity, which we do in the next section.

3.1.5 Viscosity and the stress tensor

Simple shear of an isotropic, incompressible fluid produces a stress with 6 non-zero components:

[h] We also use different boundary conditions, called Lees-Edwards boundary conditions[2] to accommodate periodicity and shear.

[i] As with the Navier-Stokes equation from earlier, we will still assume that the fluid is incompressible, that is, constant density, here. Dealing with compressible fluids is possible, we would just need to take into account the effects of the expansion and compression of the fluid due to the forces it experiences.

[j] The tensor is "deviatoric" because we have removed the pressure component and put it as its own term. This tensor thus gives how much the fluid "deviates" from an isotropic one with just the pressure from the surrounding fluid. We will refer to this tensor as just the stress tensor for the rest of this thesis.

$$\boldsymbol{\tau} = \begin{bmatrix} \tau_{xx} & \tau_{yx} & 0 \\ \tau_{xy} & \tau_{yy} & 0 \\ 0 & 0 & \tau_{zz} \end{bmatrix} \tag{3.4}$$

where x is the flow direction and y is the shear direction. Only 2 of the diagonal components are independent and experimentally, we cannot separate the pressure and normal stresses, so we instead calculate differences in normal stresses, which remove the pressure contribution:

$$N_1 \equiv \tau_{xx} - \tau_{yy} \tag{3.5}$$

and

$$N_2 \equiv \tau_{yy} - \tau_{zz}. \tag{3.6}$$

N_1 and N_2 are called the first and second normal stress differences, respectively. Normal stress differences are important in many non-Newtonian fluids and can produce interesting effects such as a fluid climbing a stirring rod as opposed to creating a depression. However, in this thesis we focus on the off-diagonal components of the stress tensor. For a more detailed and rigorous description of these concepts and non-Newtonian fluids in general, see Volume 1 of The Dynamics of Polymeric Liquids[3].

For an isotropic fluid, the stress tensor is symmetric, so $\tau_{xy} = \tau_{yx}$. τ_{yx} is called the shear stress. The shear stress can be related to the viscosity and shear rate with

$$\tau_{yx} = \eta\dot{\gamma}. \tag{3.7}$$

The viscosity thus acts as the proportionality constant between the shear stress and the shear rate[k]. For a Newtonian fluid, η is constant and $\boldsymbol{\nabla} \cdot \boldsymbol{\tau} = \eta\nabla^2\mathbf{u}$, recovering the Navier-Stokes equation. But Equation 3.7 is valid for non-Newtonian fluids as well, in which case the viscosity is a function

[k]Note that the viscosity here can depend not only on the shear rate but also other parameters. For example, a fluid might have a delay in its response to changes in shear stress, causing time-dependence. In this case, the viscosity must take the entire deformation history of the fluid as input, which drastically increases the complexity of the problem.

instead of a constant. Modeling the fluid viscosity is thus an important step in modeling the stress tensor, which in turn allows us to model the fluid itself with the Cauchy momentum equation (Equation 3.3). This makes a major focus of rheology the development of constitutive models for fluid viscosity and stress tensor.

3.2 Introduction to 2D materials

The remainder of this thesis will focus on 2D materials (also referred to as sheets), specifically how they behave while suspended in a fluid undergoing simple shear. By 2D material, we mean a 2D polymer. That is, it consists of a repeat unit, called a monomer, which is repeated along 2 dimensions. This is in contrast to 1D polymers, which are repeated only along a single dimension[1] and 3D polymers (more commonly referred to as crystals), whose monomers (more commonly referred to as unit cells) are repeated along all 3 dimensions. All three are depicted in Figure 3.3a. 2D polymers lie in an interesting in-between space between 1D polymers and crystals, combining aspects of the flexibility of 1D polymer chains with the rigidity of 3D structures. For a nice introduction on how and why 2D polymers are fundamentally different from their 1D counterpart, I recommend the paper "How the world changes by going from one- to two-dimensional polymers in solution" by Schlüter *et al.*[4]. For this thesis, we will move on to applications and theory.

3.2.1 Applications of 2D materials

Certainly the most ubiquitous 2D material is graphene, which consists of carbon atoms arranged in a hexagonal lattice (i.e. each carbon atom in the sheet's interior has 6 neighbors) depicted in Figure 3.3b. Graphene possesses large surface areas, tunability (e.g. through functionalization to materials such as graphene oxide, another common 2D material), and favorable electrical, optical, and catalytic properties while being lightweight, flexible, and mechanically stable, allowing them to see success as membranes for separations[5], in biomedical applications such as drug delivery[6,7],

[1] 1D polymers can still bend and take up a 3D space. Their classification is based purely on their topology. To identify the dimension of a polymer, we might think about selecting two random monomers in it. On average, the number of steps along the polymer we would have to take for a 1D polymer would go as the number of monomers, N, to the first power. For a 2D polymer, the number of steps would go as $N^{1/2}$. For a 3D polymer, it would go as $N^{1/3}$. For a polymer with a complicated connectivity, such a measure would give us a way of measuring its dimension, which may not be an integer value!

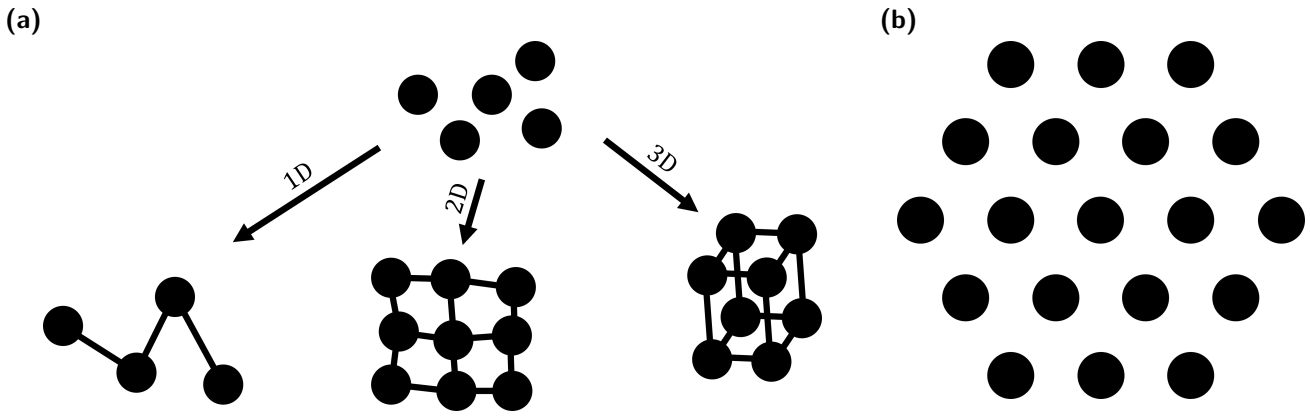

Figure 3.3 (a) Sample 1D, 2D, and 3D polymer constructions from repeat units (circles). (b) Hexagonal lattice where each bead has 6 neighbors.

in flexible electronic devices[8], and more[9–11]. 2D transition metal dichalcogenides (2D TMDs), which, as their name suggests, consist of a transition metal and a chalcogen, possess favorable electronic and optical properties. These properties can be orthogonal to those of graphene. For example, 2D TMDs may possess a band gap, which graphene lacks. These properties lead to 2D TMDs being promising in applications such as electronics[12,13] and catalysis[14,15]. Finally, synthetic 2D polymers, a field still in its infancy, represents a massive expansion of the design space for 2D materials, opening up the doors to many more applications[16]. For the time being, we will use graphene as a representative example.

In order to apply graphene and its derivatives to industrial and commercial use, it is important to be able to produce them at high quality and with relatively low cost. Many of the methods of producing graphene such as mechanical exfoliation[17] and various bottom-up synthesis strategies[18–20], have been restricted by high costs and/or limited scalability[21–24]. Liquid exfoliation, which typically involves using either sonication or shear of a material in solution to separate its stacked layers, represents a cheap, facile, and easily scalable method for producing 2D materials for certain applications[24–29]. Furthermore, many of the applications of graphene involve the use a suspension precursor regardless of the synthesis method (e.g. graphene inks for the printing of flexible electronics[30]). Therefore, in order to optimize the production and application of graphene as well as other 2D materials, it is important to understand how suspensions of 2D materials behave in flow.

3.2.2 Current theoretical and computational understanding of 2D materials

While there has been significant progress studying 2D materials in the past several decades, our understanding, especially compared to their 1D (polymers) and 3D (crystals) counterparts is still lacking. Much of the early theoretical, computational, and experimental work on sheets[m] focused on their equilibrium properties in the absence of flow. Specifically, there was much interest in finding the critical exponent, η, which quantifies how the radius of gyration of the sheet, R_g, scales with the characteristic size of the flat sheet, L: $R_g \sim L^{\eta}$. Computational studies of self-avoiding sheets[n] (in contrast with simulations of phantom sheets[31,32]) with and without bending rigidity (which resists local deformations of a sheet) suggest the lack of a finite-temperature transition from a rough but flat state ($\eta = 1$) to a crumpled state ($\eta \approx 0.8$)[33–41]. Meanwhile, theoretical predictions and experiments show both phases[42–48]. The flat and crumpled phases may have very different functionality such as different mechanical properties, rheological properties, solvent-accessible surface areas, or functionalities, making it important to be able to realize them in simulations.

One way to potentially induce a crumpled phase in simulation is the addition of defects which decrease the effective bending rigidity of a sheet and relax the self-avoiding constraint[49–51]. Another method is the inclusion of attractive interactions which can overcome the bending rigidity of the sheets, which may exist in a real system in the form of depletion or electrostatic interactions, for example. Indeed, the addition of such attractive interactions to simulations results in the appearance of crumpled and/or collapsed/compact ($\eta = 2/3$) phases not observed in self-avoiding simulations without attractive interactions[37,48,52–54]. This highlights the importance of considering and including attractive interactions when studying 2D sheets.

There have also been advances on the dynamics of 2D sheets when subjected to flows. Xu and Green studied phantom 2D sheets under shear and biaxial extensional flow and calculated their semiaxial lengths as a function of shear strength among other parameters[55,56]. Notably, they observed shear-thinning behavior but no extensional thinning. Yu and Graham studied

[m] At the time, they were often referred to as "tethered membranes" to distinguish them from fluid membranes without fixed connectivity.

[n] Self-avoiding means that the sheet cannot intersect itself. A sheet which can intersect itself is called "phantom."

phantom 2D sheets under planar, biaxial, and uniaxial extensional flows and found compact and stretched conformations with bistable behaviors[57,58]. They also classified different stable and unstable deformation modes. Salussolia and Botto studied 1D sheets in 2 dimensions under shear with full hydrodynamics[o], finding that pairs of sheets could separate and reassemble under certain conditions[59]. Perrin, Li, and Botto also studied 1D bilayer sheets in 2 dimensions and found that, due to a lateral hydrodynamic force at moderate sheet separations, the buckling transition occurred much earlier than for single-layer sheets[60], despite the bending rigidity of multi-layer sheets being higher than single layer sheets[61]. At low enough separations, lubrication forces dominated and prevented buckling[60]. Silmore, Strano, and Swan, whose model we extend in this thesis, showed that the rotation of 2D sheets in shear closely matched the predictions given by Jeffery for the rotation of rigid ellipsoids[62] under certain conditions of shear rate (relative to bending rigidity) and initial condition, beyond which orientations were shown to be chaotic[63]. They also predicted and observed the buckling modes for the initial buckling of the sheets. They later added thermal energy to the system as well and observed shear-thinning followed by shear-thickening rheological behavior[64].

Experimentally, Zhang *et al.* found shear-thinning into shear-thickening rheological behavior in graphene suspensions, which they explained using the breakdown and buildup of multisheet structures[65]. Shim *et al.* found similar behavior in graphene oxide suspensions and used X-ray scattering to quantify the relative frequencies of different sheet orientations[66]. Notably, they found that the behavior was different for dilute enough suspensions, suggesting different mechanisms in each regime.

In many real systems, such as graphene, bending rigidity is several orders of magnitude greater than thermal energy[67,68] (graphene oxide being an exception, with bending rigidity about the same as thermal energy at room temperature[69]). This raises the question as to whether the non-monotonic rheological behavior observed experimentally can be generated in the athermal limit. As discussed earlier, self-interactions are a potential method of producing conformational changes in sheets, which can affect their rheological properties, as we explore in the next chapter.

[o]That is, including short-range hydrodynamics, called lubrication forces, as well as long-range hydrodynamic interactions.

Bibliography

[1] M Mehrabi and S Setayeshi. Computational fluid dynamics analysis of pulsatile blood flow behavior in modelled stenosed vessels with different severities. *Mathematical Problems in Engineering*, 2012(1):804765, 2012.

[2] AW Lees and SF Edwards. The computer study of transport processes under extreme conditions. *Journal of Physics C: Solid State Physics*, 5(15):1921, 1972.

[3] RB Bird, RC Armstrong, and O Hassager. *Dynamics of polymeric liquids, Vol. 1: fluid mechanics*, volume 1. John Wiley & Sons, Inc., 2 edition, 1987.

[4] AD Schlüter, P Payamyar, and HC Öttinger. How the world changes by going from one-to two-dimensional polymers in solution. *Macromolecular Rapid Communications*, 37(20):1638–1650, 2016.

[5] N Nidamanuri, Y Li, Q Li, and M Dong. Graphene and graphene oxide-based membranes for gas separation. *Engineered Science*, 9(9):3–16, 2020.

[6] P Kumar, P Huo, R Zhang, and B Liu. Antibacterial properties of graphene-based nanomaterials. *Nanomaterials*, 9(5):737, 2019.

[7] S Goenka, V Sant, and S Sant. Graphene-based nanomaterials for drug delivery and tissue engineering. *Journal of Controlled Release*, 173:75–88, 2014.

[8] R You, YQ Liu, YL Hao, et al. Laser fabrication of graphene-based flexible electronics. *Advanced Materials*, 32(15):1901981, 2020.

[9] J Mu, F Gao, G Cui, et al. A comprehensive review of anticorrosive graphene-composite coatings. *Progress in Organic Coatings*, 157:106321, 2021.

[10] SM Majhi, A Mirzaei, HW Kim, SS Kim, and TW Kim. Recent advances in energy-saving chemiresistive gas sensors: A review. *Nano Energy*, 79:105369, 2021.

[11] B Wang, T Ruan, Y Chen, et al. Graphene-based composites for electrochemical energy storage. *Energy Storage Materials*, 24:22–51, 2020.

[12] B Radisavljevic, A Radenovic, J Brivio, V Giacometti, and A Kis. Single-layer mos2 transistors. *Nature Nanotechnology*, 6(3):147–150, 2011.

[13] M Acerce, D Voiry, and M Chhowalla. Metallic 1t phase mos2 nanosheets as supercapacitor electrode materials. *Nature Nanotechnology*, 10(4):313–318, 2015.

[14] D Voiry, M Salehi, R Silva, et al. Conducting mos2 nanosheets as catalysts for hydrogen evolution reaction. *Nano Letters*, 13(12):6222–6227, 2013.

[15] Z Li, J He, H Wang, B Wang, and X Ma. Enhanced methanation stability of nano-sized mos 2 catalysts by adding al 2 o 3. *Frontiers of Chemical Science and Engineering*, 9:33–39, 2015.

[16] W Wang and AD Schlüter. Synthetic 2d polymers: a critical perspective and a look into the future. *Macromolecular Rapid Communications*, 40(1):1800719, 2019.

[17] KS Novoselov, AK Geim, SV Morozov, et al. Electric field effect in atomically thin carbon films. *Science*, 306(5696):666–669, 2004.

[18] I Shtepliuk, V Khranovskyy, and R Yakimova. Combining graphene with silicon carbide: synthesis and properties–a review. *Semiconductor Science and Technology*, 31(11):113004, 2016.

[19] V Agarwal and PB Zetterlund. Strategies for reduction of graphene oxide–a comprehensive review. *Chemical Engineering Journal*, 405:127018, 2021.

[20] F Liu, P Li, H An, et al. Achievements and challenges of graphene chemical vapor deposition growth. *Advanced Functional Materials*, 32(42):2203191, 2022.

[21] R Raccichini, A Varzi, S Passerini, and B Scrosati. The role of graphene for electrochemical energy storage. *Nature Materials*, 14(3):271–279, 2015.

[22] KS Novoselov, VI Fal' ko, L Colombo, et al. A roadmap for graphene. *Nature*, 490(7419):192–200, 2012.

[23] SH Choi, SJ Yun, YS Won, et al. Large-scale synthesis of graphene and other 2d materials towards industrialization. *Nature Communications*, 13(1):1484, 2022.

[24] J Phiri, P Gane, and TC Maloney. High-concentration shear-exfoliated colloidal dispersion of surfactant–polymer-stabilized few-layer graphene sheets. *Journal of Materials Science*, 52(13):8321–8337, 2017.

[25] U Khan, A O'Neill, M Lotya, S De, and JN Coleman. High-concentration solvent exfoliation of graphene. *Small*, 6(7):864–871, 2010.

[26] JN Coleman, M Lotya, A O'Neill, et al. Two-dimensional nanosheets produced by liquid exfoliation of layered materials. *Science*, 331(6017):568–571, 2011.

[27] Y Hernandez, V Nicolosi, M Lotya, et al. High-yield production of graphene by liquid-phase exfoliation of graphite. *Nature Nanotechnology*, 3(9):563–568, 2008.

[28] A Ciesielski and P Samorì. Graphene via sonication assisted liquid-phase exfoliation. *Chemical Society Reviews*, 43(1):381–398, 2014.

[29] KR Paton, E Varrla, C Backes, et al. Scalable production of large quantities of defect-free few-layer graphene by shear exfoliation in liquids. *Nature Materials*, 13(6):624–630, 2014.

[30] TS Tran, NK Dutta, and NR Choudhury. Graphene inks for printed flexible electronics: Graphene dispersions, ink formulations, printing techniques and applications. *Advances in Colloid and Interface Science*, 261:41–61, 2018.

[31] Y Kantor, M Kardar, and DR Nelson. Tethered surfaces: Statics and dynamics. *Physical Review A*, 35(7):3056, 1987.

[32] Y Kantor and DR Nelson. Crumpling transition in polymerized membranes. *Physical Review Letters*, 58(26):2774, 1987.

[33] M Plischke and D Boal. Absence of a crumpling transition in strongly self-avoiding tethered membranes. *Physical Review A*, 38(9):4943, 1988.

[34] FF Abraham, WE Rudge, and M Plischke. Molecular dynamics of tethered membranes. *Physical Review Letters*, 62(15):1757, 1989.

[35] JS Ho and A Baumgärtner. Self-avoiding tethered membranes. *Physical Review Letters*, 63(12):1324, 1989.

[36] FF Abraham and DR Nelson. Diffraction from polymerized membranes. *Science*, 249(4967):393–397, 1990.

[37] FF Abraham and DR Nelson. Fluctuations in the flat and collapsed phases of polymerized membranes. *Journal de Physique*, 51(23):2653–2672, 1990.

[38] E Frey and DR Nelson. Dynamics of flat membranes and flickering in red blood cells. *Journal de Physique I*, 1(12):1715–1757, 1991.

[39] MJ Bowick and A Travesset. The statistical mechanics of membranes. *Physics Reports*, 344(4-6):255–308, 2001.

[40] MJ Bowick, A Cacciuto, G Thorleifsson, and A Travesset. Universality classes of self-avoiding fixed-connectivity membranes. *The European Physical Journal E*, 5:149–160, 2001.

[41] SB Babu and H Stark. Dynamics of semi-flexible tethered sheets: A simulation study using stochastic rotation dynamics. *The European Physical Journal E*, 34:1–7, 2011.

[42] RR Chianelli, EB Prestridge, TA Pecoraro, and JP DeNeufville. Molybdenum disulfide in the poorly crystalline" rag" structure. *Science*, 203(4385):1105–1107, 1979.

[43] T Hwa, E Kokufuta, and T Tanaka. Conformation of graphite oxide membranes in solution. *Physical Review A*, 44(4):R2235, 1991.

[44] X Wen, CW Garland, T Hwa, et al. Crumpled and collapsed conformation in graphite oxide membranes. *Nature*, 355(6359):426–428, 1992.

[45] MS Spector, E Naranjo, S Chiruvolu, and JA Zasadzinski. Conformations of a tethered membrane: Crumpling in graphitic oxide? *Physical Review Letters*, 73(21):2867, 1994.

[46] B Tang, Z Xiong, X Yun, and X Wang. Rolling up graphene oxide sheets through solvent-induced self-assembly in dispersions. *Nanoscale*, 10(8):4113–4122, 2018.

[47] B Tang, E Gao, Z Xiong, et al. Transition of graphene oxide from nanomembrane to nanoscroll mediated by organic solvent in dispersion. *Chemistry of Materials*, 30(17):5951–5960, 2018.

[48] P Li, S Wang, F Meng, et al. Conformational scaling relations of two-dimensional macromolecular graphene oxide in solution. *Macromolecules*, 53(23):10421–10430, 2020.

[49] SW Cranford and MJ Buehler. Packing efficiency and accessible surface area of crumpled graphene. *Physical Review B*, 84(20):205451, 2011.

[50] MK Blees, AW Barnard, PA Rose, et al. Graphene kirigami. *Nature*, 524(7564):204–207, 2015.

[51] D Yllanes, SS Bhabesh, DR Nelson, and MJ Bowick. Thermal crumpling of perforated two-dimensional sheets. *Nature Communications*, 8(1):1381, 2017.

[52] FF Abraham and M Kardar. Folding and unbinding transitions in tethered membranes. *Science*, 252(5004):419–422, 1991.

[53] D Liu and M Plischke. Monte carlo studies of tethered membranes with attractive interactions. *Physical Review A*, 45(10):7139, 1992.

[54] GS Grest and IB Petsche. Molecular dynamics simulations of self-avoiding tethered membranes with attractive interactions: Search for a crumpled phase. *Physical Review E*, 50(3):R1737, 1994.

[55] Y Xu and MJ Green. Brownian dynamics simulations of nanosheet solutions under shear. *The Journal of Chemical Physics*, 141(2):024905, 2014.

[56] Y Xu and MJ Green. Brownian dynamics simulation of two-dimensional nanosheets under biaxial extensional flow. *Journal of Polymer Science Part B: Polymer Physics*, 53(17):1247–1253, 2015.

[57] Y Yu and MD Graham. Coil–stretch-like transition of elastic sheets in extensional flows. *Soft Matter*, 17(3):543–553, 2021.

[58] Y Yu and MD Graham. Wrinkling and multiplicity in the dynamics of deformable sheets in uniaxial extensional flow. *Physical Review Fluids*, 7(2):023601, 2022.

[59] G Salussolia, C Kamal, J Stafford, N Pugno, and L Botto. Simulation of interacting elastic sheets in shear flow: Insights into buckling, sliding, and reassembly of graphene nanosheets in sheared liquids. *Physics of Fluids*, 34(5):053311, 2022.

[60] H Perrin, H Li, and L Botto. Hydrodynamic interactions change the buckling threshold of parallel flexible sheets in shear flow. *Physical Review Fluids*, 8(12):124103, 2023.

[61] N Lindahl, D Midtvedt, J Svensson, et al. Determination of the bending rigidity of graphene via electrostatic actuation of buckled membranes. *Nano Letters*, 12(7):3526–3531, 2012.

[62] GB Jeffery. The motion of ellipsoidal particles immersed in a viscous fluid. *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character*, 102(715):161–179, 1922.

[63] KS Silmore, MS Strano, and JW Swan. Buckling, crumpling, and tumbling of semiflexible sheets in simple shear flow. *Soft Matter*, 17(18):4707–4718, 2021.

[64] KS Silmore, MS Strano, and JW Swan. Thermally fluctuating, semiflexible sheets in simple shear flow. *Soft Matter*, 18(4):768–782, 2022.

[65] X Zhang, Y Tang, X Wang, et al. Dispersion and rheological properties of aqueous graphene suspensions in presence of nanocrystalline cellulose. *Journal of Polymers and the Environment*, 26:3502–3510, 2018.

[66] YH Shim, H Ahn, S Lee, SO Kim, and SY Kim. Universal alignment of graphene oxide in suspensions and fibers. *ACS Nano*, 15(8):13453–13462, 2021.

[67] P Liu and YW Zhang. Temperature-dependent bending rigidity of graphene. *Applied Physics Letters*, 94(23):231912, 2009.

[68] JW Jiang, Z Qi, HS Park, and T Rabczuk. Elastic bending modulus of single-layer molybdenum disulfide (mos2): finite thickness effect. *Nanotechnology*, 24(43):435705, 2013.

[69] P Poulin, R Jalili, W Neri, et al. Superflexibility of graphene oxide. *Proceedings of the National Academy of Sciences*, 113(40):11088–11093, 2016.

Chapter 4

Dynamics of a self-interacting sheet in shear flow

Solution processing of 2D materials such as graphene is important for applications thereof, yet a complete fundamental understanding of how 2D materials behave dynamically in solution is lacking. In this chapter, we extend the model developed by Silmore et. al.[1] *by adding short-ranged Lennard-Jones interactions to 2D sheets in shear flow. We find that the addition of these interactions allows for a rich landscape of conformations which depend on the balance between shear strength, bending rigidity, and interaction strength as well as the initial configuration of the sheet. We explore this conformational space and classify sheets as flat, tumbling, 1D folded, or 2D folded based on their conformational properties. We use kinetic and energetic arguments to explain why sheets adopt certain conformations within the folded regime. We calculate the stresslet and find that, even in the absence of thermal fluctuations and multiple sheet interactions, shear-thinning followed by shear-thickening behavior can appear. Finally, we extend this model to two sheet simulations to model shear exfoliation. We find that, even with just two sheets, there are new conformational behaviors — flipping and waltzing — which had significant effects on the suspension properties. As with single sheets, we explain why sheets adopt these conformations as well as explain how they result in the observed rheological behavior. The content of this chapter regarding single sheet simulations was adapted from Funkenbusch, W. T., Silmore, K. S., & Doyle, P. S. (2024). "Dynamics of a self-interacting sheet in shear flow."* Soft Matter.

4.1 Introduction

In the previous chapter, we discussed the current understanding of 2D material suspensions, both with and without flows. A missing piece of this understanding was a sheet model in 3 dimensions with both self-interactions and flow. In this chapter, we develop such a model, using the model developed by Silmore *et. al.*[1] as a starting point. By examining this model, we hope to explain the interesting, non-monotonic rheological properties seen in real suspensions of 2D materials.

4.2 Model for self-avoiding, self-interacting 2D materials in shear

In order to understand more dense suspensions of 2D materials, it is important to first understand how a single sheet behaves. Such a model also informs understanding of dilute suspensions, in which sheets are unlikely to interact with one another[a]. As discussed in the previous chapter, self-avoiding interactions are important and thermal energy tends to be small. This motivates our model of an athermal, self-avoiding, self-interaction sheet in shear flow.

We construct hexagonal sheets with circumradius[b] $L = 39a$ with beads of radius a such that each interior bead has 6 tangent neighbors, totalling $N = 1141$ beads, as depicted in Figure 4.2a. These sheets are initially flat and lie in the flow-vorticity plane[c]. They are then rotated by a fixed angle $\theta = 5°$ about the vorticity axis to induce an initial flipping of the sheet due to shear. It is then rotated by a varying angle ϕ about the flow axis to sample many initial configurations. These angles are depicted visually in Figure 4.1. We sample from $\phi = 0°$ to $\phi = 90°$ with samples at every $5°$. In averages over ϕ, we give relative weights to each ϕ proportional to $\sin\phi$, corresponding to initially randomly oriented sheets[d]. We use a fixed time step of $\dot{\gamma}t = 2 \times 10^{-4}$ for all simulations, where $\dot{\gamma}$ is the shear rate. Simulations were run for 2000 strain cycles ($\dot{\gamma} = 2000$) and results were calculated using the last 200 strain cycles using data from every $100\dot{\gamma}t$, as analysis of the

[a] That is, the average distance between them is much larger than the range of the interactions between them.

[b] The circumradius is the distance from the center to one of the corners. It is called the circumradius because it is the radius of the smallest circle which completely encloses the shape.

[c] Recall that the vorticity axis is the axis perpendicular to both the velocity and velocity gradient. The shear axis points in the direction of increasing x velocity.

[d] Due to the fixed rotation angle θ, we won't be sampling every possible initial condition. This is because adding an extra parameter would require many more simulations to analyze. However, due to the results of this chapter, such an analysis may be worthwhile, as we discuss later.

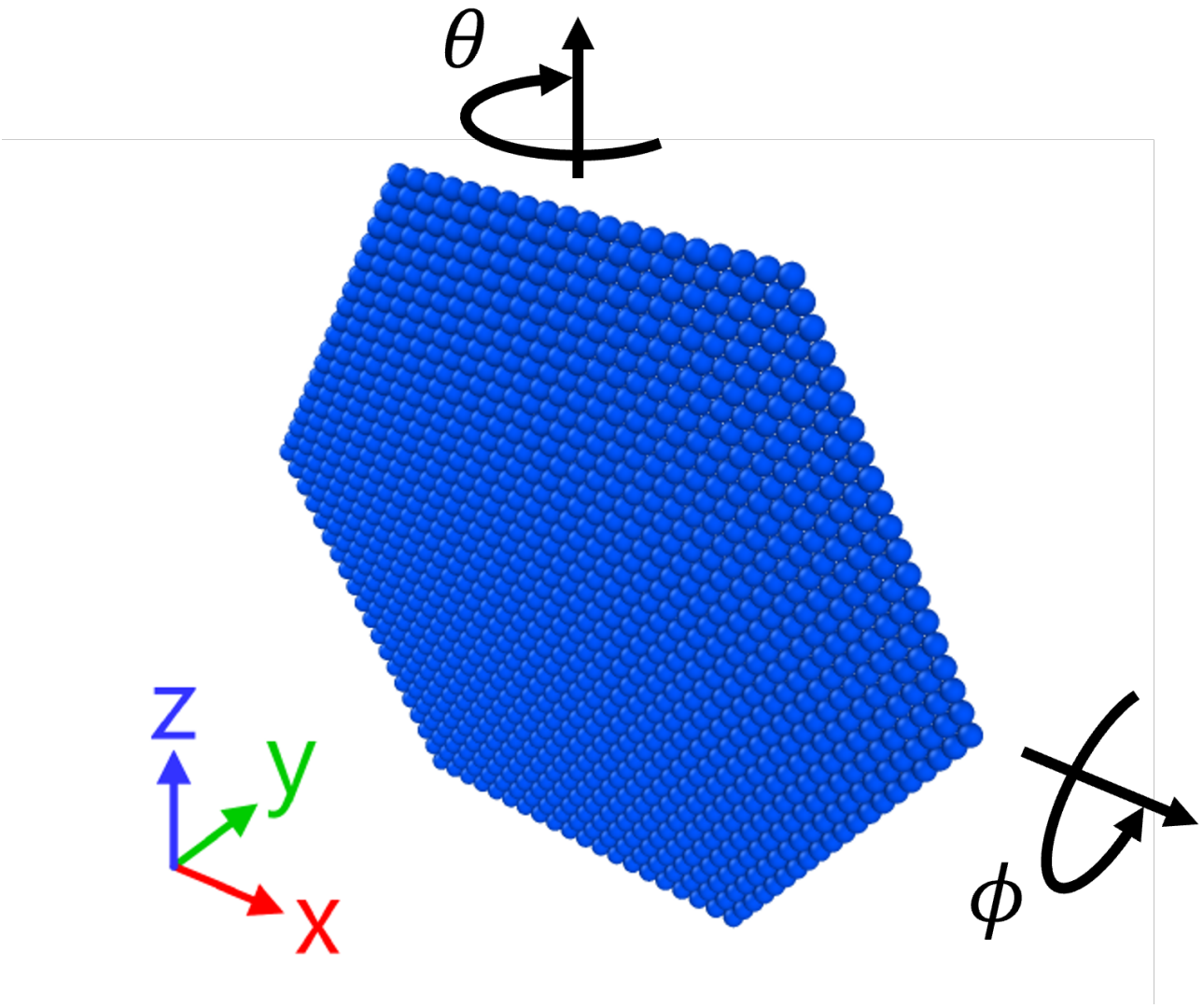

Figure 4.1 Depiction of initial angle conditions. θ is the angle about the vorticity (z) axis. ϕ is the angle about the flow (x) axis. y is the shear axis.

autocorrelation[e] of sheet properties such as the radius of gyration at small interaction strengths (i.e. for tumbling sheets, as described later, who have the slowest decaying autocorrelations) showed that approximately every $100\dot{\gamma}t$ are independent.

The system is integrated forward in time using an Euler-Maruyama integrator, as discussed in Chapter 1, with the following equations of motion for Brownian dynamics with hydrodynamics:

[e]If the autocorrelation between two points is large, then they are correlated with each other and we don't really have two independent points, because knowing one we could predict the other. So, we don't want to use both points in our analyses as they would result in smaller error bars than desired, as we would think we have more independent points than we do. So, one strategy is to examine the autocorrelation function and see when it becomes small, and only sample points which are that distance apart in time.

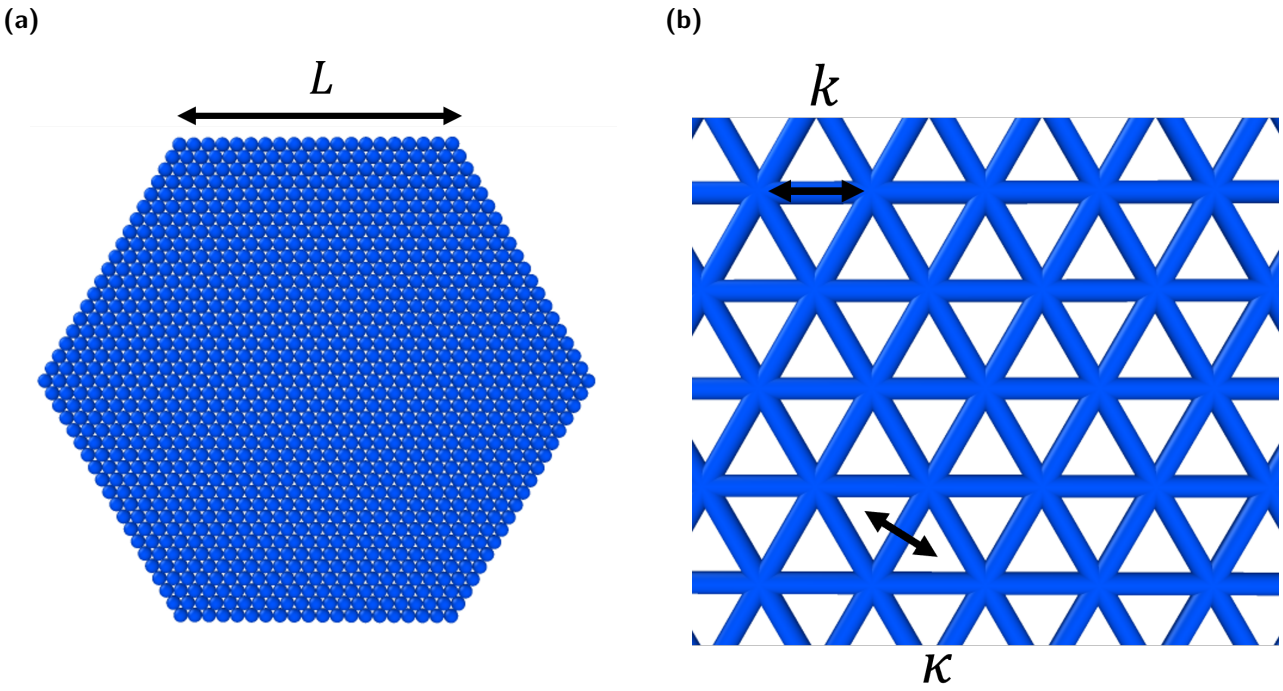

Figure 4.2 (a) Example sheet with beads of radius a drawn to scale. $L = 39a$ is the circumradius. (b) Connectivity of the sheets, where the intersection between lines represents the center of beads. Between neighboring beads (along lines), a harmonic potential with spring constant k is applied. Between neighboring triangles, a dihedral potential with bending rigidity κ is applied.

$$d\mathbf{x}_i = \left(-\sum_j \mathcal{M}_{ij} \frac{\partial U_i}{\partial \mathbf{x}_j} + \mathbf{L}\mathbf{x}_i \right) dt, \tag{4.1}$$

where $(\mathbf{L})_{mn} = \dot{\gamma}\delta_{m1}\delta_{n2}$ is the velocity gradient tensor for simple shear and U_i is the sum of the applied potentials on bead i. To achieve long-range hydrodynamics, we use the Rotne-Prager-Yamakawa (RPY) tensor[2] for $\mathcal{M}_{ij}$, which is given analytically by

$$\mathcal{M}_{ij} = \frac{1}{6\pi\eta a} \begin{cases} \left(\frac{3a}{4r} + \frac{a^3}{2r^3} \right) \mathbf{I} + \left(\frac{3a}{4r} - \frac{3a^3}{2r^3} \right) \hat{\mathbf{r}}\hat{\mathbf{r}}^T & r > 2a \\ \left(1 - \frac{9}{32a} \right) \mathbf{I} + \frac{3}{32a} \hat{\mathbf{r}}\hat{\mathbf{r}}^T & r \leq 2a, \end{cases} \tag{4.2}$$

where η is the fluid viscosity, r is the distance between two beads, and $\hat{\mathbf{r}}$ is the unit vector pointing from particle i to particle j. Note that, because we wish to study the behavior of a single, isolated sheet, we use the non-periodic RPY tensor here. Our boundary conditions for the system are still periodic such that the sheet will return to the simulation box if it leaves, but it will not

interact hydrodynamically with its periodic images. This also has the added benefit of saving on computation time as the number of particles is small ($\mathcal{O}(10^3)$), although the RPY tensor product used here is N^2 in the number of beads.

The Rotne-Prager-Yamakawa tensor models long-range, pairwise hydrodynamic interactions between beads, with each bead acting as a regularized Stokeslet[1]. Thus, finite discretizations of these beads as sheets have some degree of permeability. However, Yu and Graham recently showed that, for similarly discretized sheets, the permeation velocity (i.e. the fluid velocity normal to the sheet surface) in extensional flows tends to be small (on the order of 1% relative to the sheet size and strain rate)[3]. We expect shear flows to carry similar permeation velocities. Second, for small enough sheets, lubrication forces are relatively small compared to the forces from self-interaction. We discuss this assumption and the sheet size limit in Appendix D. Conceptually, because the applied self-interaction includes a repulsive component and equilibrium distance, small enough sheets do not approach closely enough for lubrication forces to be significant.

Simulations were run using HOOMD-blue[4] on NVIDIA GeForce GTX 1080 Ti's with a custom package from Silmore *et. al.*[1] which was adapted from Fiore *et. al.*[5]. All images of sheets were rendered using Ovito[6].

4.2.1 Applied potentials

We now describe the potential applied to beads. First, to enforce the connectivity of the sheet, we apply a harmonic potential between neighboring beads:

$$U_{\text{bond}} = \frac{k}{2}(r - r_0)^2 \tag{4.3}$$

where k is the spring constant and $r_0 = 2a$ is the equilibrium distance between beads. This is visualized in Figure 4.2b. We use a stiffness of $k = 1000 \times 6\pi\eta\dot{\gamma}a$. The continuum 2D Young's modulus, Y, which characterizes the in-plane stiffness (the "stretchiness") of the sheet, can be connected to the discrete spring constant, k, with $Y = 2k/\sqrt{3}$[7].

We also apply a dihedral potential between neighboring triangles to give the sheet bending rigidity:

$$U_{\text{dhdl}} = \kappa(1 - \hat{\mathbf{n}}_i \cdot \hat{\mathbf{n}}_j), \tag{4.4}$$

where κ is the bending rigidity (out-of-plane stiffness) and $\hat{\mathbf{n}}_i$ is the normal vector to triangle i[f]. This discrete bending rigidity can be mapped to a continuum bending rigidity, $\tilde{\kappa}$, with $\tilde{\kappa} = \kappa/\sqrt{3}$ for this hexagonal triangulation[8].

The values of k and κ were chosen such that the in-plane stiffness of sheets is much larger than their out-of-plane stiffness, as is true in many real systems. This can be quantified using the Föppl-von Kármán number, FvK $\sim kL^2/\kappa$, which was between 10^3 and 10^7 for the simulations in this paper. Thus, the sheets are inextensible relative to bending.

We also apply hard-sphere interactions between non-neighboring beads with a pair-potential which places overlapping beads tangent to each other under Rotne-Prager-Yamakawa dynamics (as described in Chapter 1):

$$U_{\text{HS}} = \begin{cases} \frac{16\pi\eta a^2}{\Delta t}\left(2a \ln \frac{2a}{r} + (r - 2a)\right), & 0 \leq r \leq 2a \\ 0, & r > 2a. \end{cases} \tag{4.5}$$

Finally, we apply a short-ranged interaction in the form of a truncated Lennard-Jones potential between non-neighboring beads:

$$U_{LJ}(r) = 4\epsilon\left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6}\right], \tag{4.6}$$

where ϵ is the interaction strength and σ is the interaction range. We use a turn-on distance of $r_{on} = 2a$ and cut-off radius of $r_{cut} = 2.5\sigma$. For all simulations, we set $\sigma = 4\sqrt{6}a/3$[g].

Beads can interact via this potential along the sheet surface, which may change the in-plane behavior of the sheets. However, we find empirically that, even for the largest interaction strengths used, the harmonic bonds between neighboring beads in a flat sheet extend by less than 0.1% of

[f]Note that these normal vectors must be defined such that they are oriented along the same direction and the potential is always positive (that is, there's always a penalty for bending).

[g]Note that it is not important for a given real system to have an interaction potential with the same form as we apply in our simulations, only that it has the same essential features (that is, that the potential is short-ranged). The repulsive component to our applied potential might have its role filled by lubrication forces in real systems, as we discuss in Appendix D.

their equilibrium distance. Therefore, we expect that self-interactions do not significantly affect the in-plane interactions of the sheet.

4.2.2 Relevant dimensionless groups

The addition of self-interactions adds two new parameters to the system compared to a system without self-interactions: the interaction strength, ϵ, and interaction range, σ. To give a roughly constant interaction energy for a bead moving parallel to a sheet, we require $\sigma \gg a$. As mentioned earlier, for all sheets in this thesis, we choose $\sigma = 4\sqrt{6}a/3 \approx 3.27a$. While ϵ is the energy scale for the interaction between two beads, the energy scale for the interaction of a bead with a plane of beads separated by the equilibrium distance of σ is

$$\tilde{\epsilon}\sigma^2 = \epsilon\frac{\sigma^2}{2\sqrt{3}a^2}, \tag{4.7}$$

where $\tilde{\epsilon} = \epsilon/2\sqrt{3}a^2$ is the interaction energy density of the sheet. For two parallel sheets of characteristic size L aligned with the shear-vorticity plane and separated by a distance σ, shear acts as a force trying to separate the two sheets by sliding them along each other, while their interaction resists this sliding. Taking the ratio of these two forces gives a dimensionless group characterizing the ability of attractive interaction to resist shear:

$$\chi \equiv \frac{\tilde{\epsilon}\sigma^2}{6\pi\eta\dot{\gamma}L^2\sigma}\left(\frac{L}{a}\right), \tag{4.8}$$

where η is the fluid viscosity and $\dot{\gamma}$ is the shear rate. This dimensionless group characterizes the ability of nearby sections of the sheet to slide along each other. A derivation of this dimensionless quantity can be found in Appendix E.

We note the existence of the bead radius, a, in this dimensionless number. In work by Silmore *et. al.*[1,9], a was the smallest resolvable length scale and did not play a role in the dynamics. Here, because interactions happen between individual beads, a is relevant. The quantity (L/a) is proportional to the number of interactions along the edge of the sheet. In a physical system, a now corresponds to the interacting elements of a sheet, for example, individual carbon atoms in a graphene sheet interacting via van der Waals forces. Typical solution-processed sheets have

sizes ranging from nanometers to micrometers[10] and may interact at a wide range of distances[11]. Further control over the shear rate in experiments means that this dimensionless number can span several orders of magnitudes, and we thus expect this dimensionless number to reveal interesting transitions in real systems.

While the ratio of bending rigidity to shear, S, with

$$S = \frac{\kappa}{\pi\eta\dot{\gamma}L^3}, \tag{4.9}$$

is a relevant dimensionless group for the dynamics of this system, as we discuss in the results, we find that another illuminating group is the ratio between bending rigidity and interaction strength:

$$K \equiv \frac{\kappa}{\tilde{\epsilon}\sigma^2}. \tag{4.10}$$

Interaction tries to move sections of the sheet together by folding, while bending rigidity resists this folding. Thus, when $K \gg 1$, attraction interactions cannot overcome bending rigidity, and we expect the sheet to behave as in work by Silmore *et. al.*[1,9]. This dimensionless group is also convenient because it is a function of material properties, while χ can be tuned experimentally by adjusting shear rate.

Plots varying K and χ^{-1} thus have a convenient physical interpretation: moving along the first axis adjusts the material property of bending rigidity to interaction strength, while moving along the second axis changes the experimental property of shear force relative to interaction strength. The value of S can be determined from the values of χ, K, σ and a with $S = 6\chi K(a/L)(\sigma/L)$.

4.3 Conformational properties

4.3.1 Identifying sheet conformations

The conformational properties of sheets are highly sensitive to experimental conditions (χ), material properties (K), and initial conditions (e.g. the initial orientation of the sheets relative to the flow axis, ϕ). We identify four different conformations that sheets can exhibit: flat, tumbling, 1D folding, and 2D folding. An example of each is shown in Figure 4.3. Videos of simulations

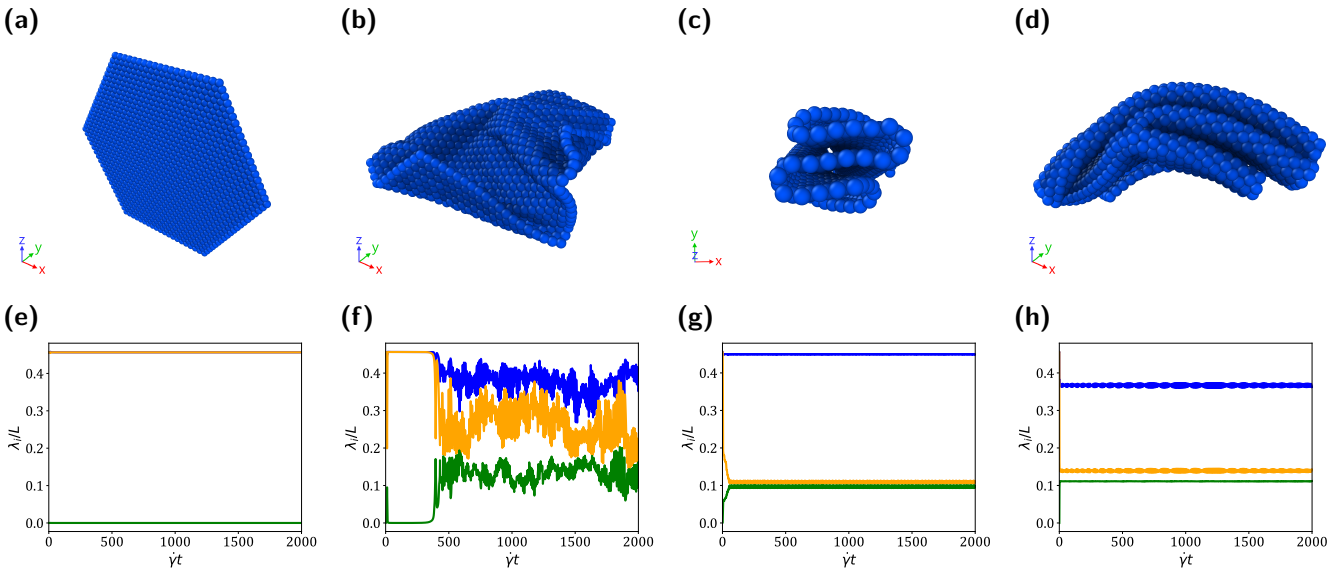

Figure 4.3 (a-d) Example conformations for (a) flat, (b) tumbling, (c) 1D folded, and (d) 2D folded sheets. x is the flow direction and y is the shear direction. (e-h) Square root of the eigenvalues of the gyration tensor, λ_i, over time for the (e) flat, (f) tumbling, (g) 1D folded, and (h) 2D folded sheets. Green is the smallest, orange is the second largest, and blue is the largest characteristic length.

for each conformation along with their corresponding average signed local mean curvature (see Section 4.3.3) can be found in the ESI of the paper published on this chapter[12]. We examine how the values of χ, K, and ϕ lead to each of these conformations in future sections. In this section, we show how each conformation can be characterized.

We identify these conformations by looking at the square root of the eigenvalues of the gyration tensor, λ_i ($i = 1, 2, 3$ with $\lambda_1 > \lambda_2 > \lambda_3$), which correspond to the three characteristic lengths of the conformation. We take averages of each λ_i over the last $250\dot{\gamma}t$ of the simulation every $0.25\dot{\gamma}t$, denoted $\bar{\lambda}_i$.

Flat sheets have almost no bends and therefore $\bar{\lambda}_1$ and $\bar{\lambda}_2$ are near their maximum value of $0.456L$ at all times. We categorize sheets as flat if $\bar{\lambda}_2 > 0.4L$. They are the only sheet conformation we observe which are not necessarily continuously rotating about the vorticity axis.

Tumbling sheets are characterized by impermanent folds which cause their λ_i to fluctuate significantly throughout the simulation. We categorize sheets as tumbling if, regardless of $\bar{\lambda}_i$, the smallest standard deviation in λ_i (over the last $200\dot{\gamma}t$) is greater than $7 \times 10^{-3}L$ or the largest standard deviation in λ_i is greater than $3 \times 10^{-2}L$.

1D folded sheets are characterized by a series of parallel folds, resulting in a λ_1 close to the maximum value, but a much smaller λ_2. We categorize sheets as 1D folded when $\bar{\lambda}_1 > 0.4L$ and $\bar{\lambda}_2 < 0.4L$.

2D folded sheets are characterized by the appearance of non-parallel folds, causing λ_1 to deviate significantly from its maximum value. We categorize sheets as 2D folded when $\bar{\lambda}_1 < 0.4L$. 1D folded and 2D folded sheets are referred to together as folded sheets. Folded sheets are distinct from tumbling sheets, which also have folds, because their folds are persistent over time.

The boundaries between different sheet conformations were chosen by looking at histograms of $\bar{\lambda}_i$ and the standard deviation in λ_i. These histograms can be found in Appendix F. There is a clear gap in the histogram of $\bar{\lambda}_2$ at $0.4L$, making $\bar{\lambda}_2 > 0.4L$ a natural choice for characterizing a sheet as flat. The transition between 1D and 2D folded sheets appears continuous in that sheets can have both parallel and non-parallel folds. The cutoff between these conformations was therefore chosen by eye by finding a value of $\bar{\lambda}_1$ which seems to correspond to the beginning of the appearance of non-parallel folds. For tumbling sheets, the cutoffs were chosen again by eye such that each distribution of standard deviations for tumbling sheets appears normal-like.

4.3.2 Phase map of sheet conformations

We performed a parameter sweep across χ, K, and ϕ and categorized each conformational state as flat, tumbling, 1D folded, or 2D folded. We plot using χ^{-1} instead of χ as increasing χ^{-1} corresponds to increasing shear rate, as one might see in a rheological experiment. As this is a large, 3D phase space, we present 2 slices at $\phi = 0°$ and $\phi = 45°$, shown in Figure 4.4. First looking at the phase plot for $\phi = 0°$, we see clear divisions between each conformation. For large K (> 1), attractive interactions cannot overcome bending rigidity, and sheets tend to be flat. At large χ^{-1} (> 100) and low K ($\lesssim 1$), shear is able to break local attractive interactions, and the sheets tumble. At low χ^{-1} ($\lesssim 100$) and low K ($\lesssim 1$), shear is unable to break local attractive interactions, so folds are permanent and the sheets are 1D or 2D folded. Lower values of χ^{-1} at low K (< 1) correspond to less well-aligned folds and thus more 2D folded sheets.

At $\phi = 45°$, the features of the plot remain the same with one notable exception. The tumbling region expands to occupy the high χ^{-1} (> 100), high K (> 1) regime, similar to Figure 6 in work by

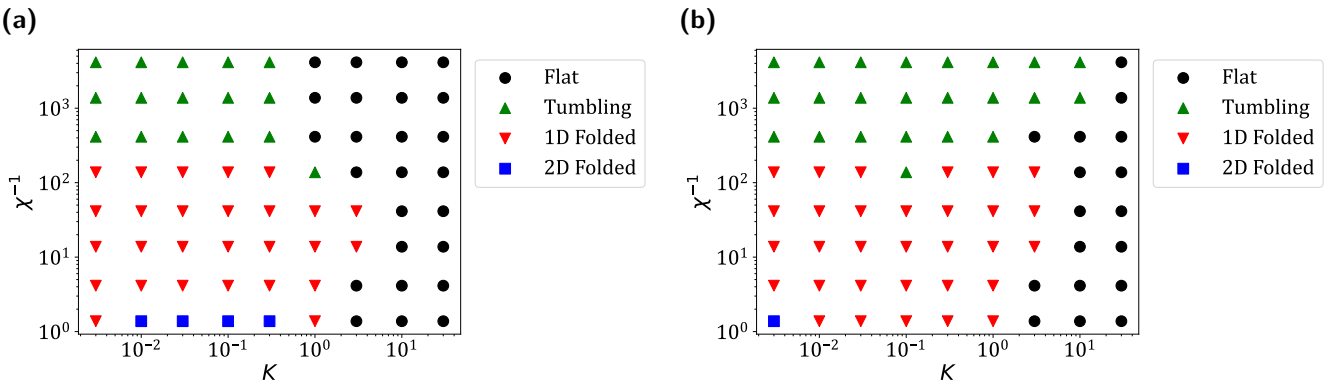

Figure 4.4 Conformational phase map of sheets with initial conditions (a) $\phi = 0°$ and (b) $\phi = 45°$.

Silmore *et. al.*[1], where the continuously tumbling regime was larger for larger ϕ. The specific sheets which tumble for $K \gtrsim 1$ are unpredictable, but on average tumbling behavior is more common at larger ϕ and χ^{-1}. This makes sense, as larger ϕ mean larger initial deformations from the flat conformation and larger χ^{-1} at constant K mean stronger initial buckling due to a reduced value of S, the dimensionless ratio of bending rigidity to shear strength. However, the tumbling behaviors in the sheets in this thesis are notably different than those observed by Silmore *et. al.*[1] due to the presence of self-interaction. For example, in Figure 4.5, the sheet forms several, slowly sliding folds which continuously flip in sequence. We term this "teacup" behavior. This behavior was observed for many initial orientations near the tumbling/folded boundary ($1.4 \times 10^2 \leq \chi^{-1} \leq 1.4 \times 10^3$). Teacup behavior is usually transient in the sense that sheets alternate between it and more typical tumbling behavior over long time scales (several hundred $\dot{\gamma}t$). It is classified as tumbling due to the existence of non-persistent folds and is difficult to distinguish from typical tumbling behavior using the values of $\bar{\lambda}_i$ and the standard deviations of λ_i. A video of a simulation with this behavior can be found in the ESI of the paper published on this chapter[12].

Within a given region (flat, tumbling, and folded), sheet behavior can be predicted reliably. Near the boundaries between regions, sheet behavior may be highly sensitive to initial orientation.

4.3.3 Bending modes of 1D folded sheets

We calculate the signed local mean curvature of the sheets and calculate the average over 101 equally spaced snapshots during the last $200\dot{\gamma}t$, giving an average signed local mean curvature

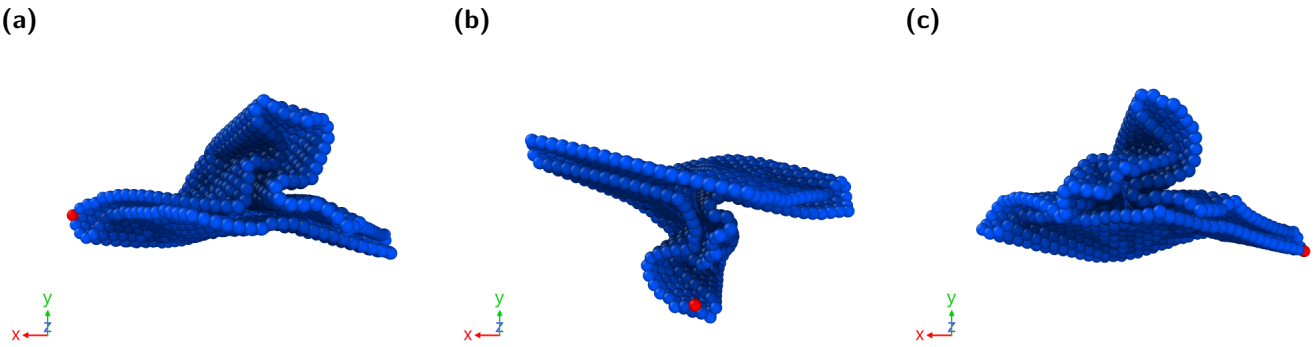

Figure 4.5 Sequential snapshots showing a 180° rotation of a sheet exhibiting teacup behavior, rotating in the counterclockwise direction with $\chi = 2.42 \times 10^{-3}$, $K = 0.01$, and $\phi = 85°$. x is the flow direction and y is the shear direction. A single bead is colored red to guide the eye.

.

(ASLMC). We plot these for the $\phi = 0°$ and $\phi = 45°$ conditions in Figure 4.6. These plots correspond well to the characterizations given in the previous section. Flat sheets have no local mean curvature, and thus have an ASLMC of about 0 at all points. Tumbling sheets have significant but non-persistent folds, and their ASLMC's thus appear noisy. Folded sheets have persistent folds, and thus are characterized by sharp bands of high magnitude ASLMC. 1D folded sheets have folds which are aligned along a single axis, while 2D folded sheets have non-parallel folds which can branch off into more folds. Some sheets, especially for $\phi = 45°$ have characteristics of both 1D and 2D folded sheets.

1D folded sheets at the boundary with tumbling sheets (e.g. $\chi^{-1} = 1.4 \times 10^2$) typically have two close parallel folds and moderate curvature throughout the rest of the sheet. These 1D folded sheets take on a "rolled-up" conformation, as seen in Figure 4.7. A video of a simulation of a rolled-up sheet can be found in the ESI of the paper published on this chapter[12]. Interaction is strong enough to cause an initial folding of the sheet, but shear is strong enough to anneal all but the last two folds. This results in rolled-up sheets being an even more energetically favorable state due to the high degree of contact for self-interactions and gentle folding throughout. Similar conformations have been observed in graphene oxide suspensions without flow (called "nanoscrolls")[13]. 1D folded sheets have a number of folds ranging from 2-folds in rolled up sheets, to 6, evenly spaced folds. There appears to be no clear pattern to the exact number of folds that will appear, although roughly the number appears to increase with decreasing χ^{-1} unless the sheet 2D folds.

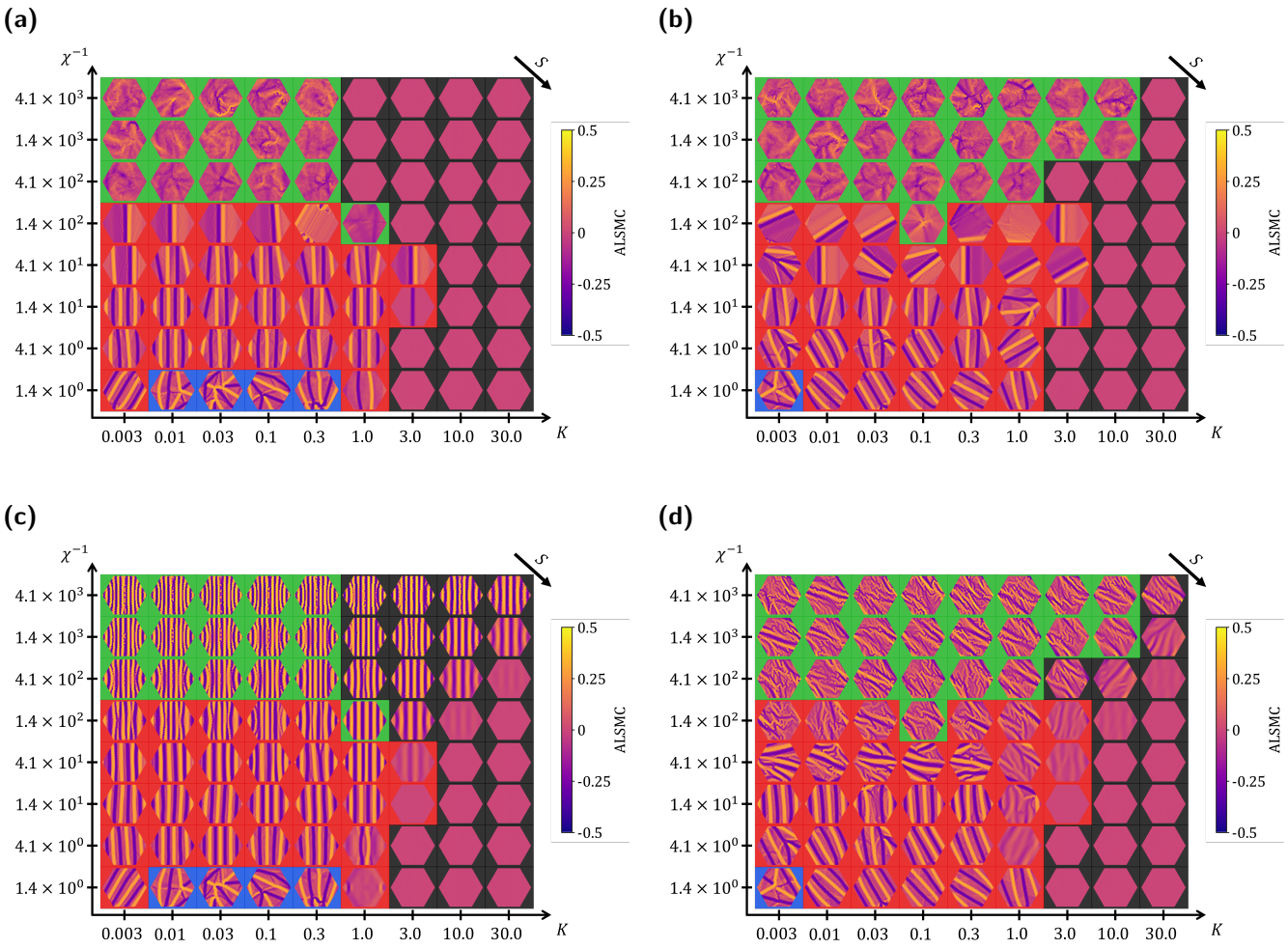

Figure 4.6 Average signed local mean curvatures for sheets during the last $200\dot{\gamma}t$ with (a) $\phi = 0°$ and (b) $\phi = 45°$. Signed local mean curvatures for sheets at their initial flip with (c) $\phi = 0°$ and (d) $\phi = 45°$. Backgrounds correspond to different conformational behaviors (black: flat, green: tumbling, red: 1D folded, blue: 2D folded). Diagonal lines running up and to the right correspond to sheets with constant S. Moving right or down corresponds to increasing S by about a factor of 3. The arrow in the top right corner of each plot indicates the direction of maximally increasing S. For reference, the sheet at $K = 1.0$, $\chi^{-1} = 1.4$ has $S \times 10^5 = 920$.

We can estimate the expected number of folds using a simple energetic argument balancing the bending and interaction energies of a 1D folded sheet, which is detailed in Appendix G. Doing so, we obtain the following estimate for the optimal number of folds, n^*_{fold}, in a rectangular 1D folded sheet:

$$n^*_{fold} \approx \sqrt{\frac{2L/w_{fold} + 1}{1 - \beta + \alpha K}} - 1, \tag{4.11}$$

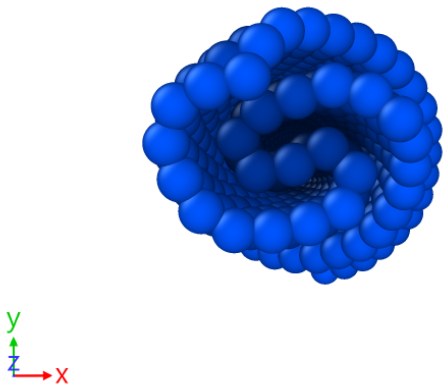

Figure 4.7 Example of a 1D folded sheet that adopts a rolled-up conformation. x is the flow direction and y is the shear direction.

where α and β are fit parameters. α is a measure of the relative importance of bending to interaction strength and β is a measure of the interaction energy of beads within a fold. We find $\alpha = 0.0618 \pm 0.0010$ and $\beta = 0.528 \pm 0.003$ ($\pm$ one standard deviation) for rectangular sheets. w_{fold} is a correlation length characterizing the width of a fold, which is an unknown function of the system parameters, including perhaps the Föppl-von Kármán number. We observe that all folded sheets in our simulations have $w_{fold} \approx 6a$, which corresponds to the smallest fold a sheet can have. This means that folds under the conditions in this thesis are controlled by the smallest length scale, a.

For small χ^{-1} ($\lesssim 41$), shear is not strong enough to break up all the folds, and the sheet can obtain an energetically stable folded configuration. In the small K limit for sheets with $L = 39a$, Equation 4.11 approaches about 4.46, which is close to 4, the most common number of folds observed. For sheets with $L = 79a$, Equation 4.11 approaches about 6.62. A series of simulations run with $L = 79a$ and $\phi = 0°$ in the 1D folded regime showed 6 folds at higher χ^{-1} (shear rates) and 8 folds at lower χ^{-1}.

For slightly higher χ^{-1} (≈ 141), shear is strong enough to break up all but 2 folds and the sheet adopts a more energetically favorable rolled-up conformation, which is not modelled by Equation 4.11. At even higher χ^{-1} ($\gtrsim 410$), shear is strong enough to continuously break new folds which form as the sheet tumbles.

Because $\alpha \approx 0.05$, the number of folds predicted by Equation 4.11 does not decrease significantly until $K \approx 20$. The sheets flattening at about $K = 3.0$, a full order of magnitude lower, is therefore likely a kinetic phenomenon, which we explore in the next section.

4.3.4 Creation of persistent folds in 1D folded sheets

To explain why sheets in shear adopt configurations with parallel folds as opposed to other configurations (e.g. the "double folded" configuration observed by Abraham and Kardar[14]), we return to work by Silmore *et. al.*[1], which calculated the buckling modes of sheets in shear as a function of the ratio of bending rigidity to shear, $S = \kappa/\pi\eta\dot{\gamma}L^3$, at $\phi = 0°$ and at the maximal in-plane stress for a flipping sheet, $\theta = 45°$. This analysis did not consider attractive interactions, which could change the buckling behavior, however we believe it to still be useful. Lines of constant S correspond to lines of slope 1 in plots of K versus χ^{-1}. In our simulations, $S \times 10^5$ ranges from 9.4×10^{-4} to 2.8×10^4, which covers more than 10 of the first buckling transitions. Notably, the buckling modes calculated by Silmore *et. al.*[1] have alternating signs in curvature, just like what is observed in the 1D folded sheets here.

Figure 4.6 shows the local mean curvature for sheets averaged over the last $200\dot{\gamma}t$ and during their initial flip (determined as when the average normal vector of the triangles in the sheet is closest to $\theta = 45°$). As χ^{-1} decreases for low K, the number of folds at the initial buckle decreases as expected because the effective bending rigidity increases relative to shear strength (given by S). In the 1D folded regime, sheets have folds which which are sharper in the initial buckle than the buckles that appear in the absence of interactions, showing that interaction is strong enough to affect the initial buckling. In this regime, the number of folds often decreases from the initial buckle to the final conformation, showing annealing to a more energetically stable configuration. As K increases, the magnitude of the curvature in the initial flip decreases, again due to an increasing S, until the sheet eventually appears flat. While the analysis done by Silmore *et. al.*[1] was only done for sheets with $\phi = 0°$, the same trends are seen in sheets with any value of ϕ, as seen in Figure 4.6d.

We propose that persistent folds in 1D folded sheets are formed during the initial buckling of the sheet during the first flip followed by annealing caused by shear towards an energetically stable

state. If shear is too strong ($\chi^{-1} \gtrsim 410$), the folds formed during buckling are not persistent, and the sheet tumbles or flattens. At an intermediate value of shear ($\chi^{-1} \approx 141$), folds form during buckling but most are broken up by shear, resulting in an energetically favorable rolled-up sheet. At lower values of shear ($\chi^{-1} \lesssim 41$), shear anneals the sheets towards their energetically preferred number of folds as predicted by Equation 4.11. If shear is too weak, it can only anneal partially (thus why some 1D folded sheets have more than 4 folds). Annealing can only decrease the number of folds. So, if the number of bends in the initial buckle is less than the energetically ideal number of folds, it is kinetically trapped with that number of folds. Because the buckling modes of sheets in shear have parallel folds, the resulting conformations have parallel folds.

At low K, interactions are stronger than buckling, so buckling will propagate into folds. For $K \gtrsim 1$, interactions are not strong enough to overcome bending, so the extent of buckling is determined by the competition between bending rigidity and shear (i.e. the value of S). This explains how sheets can transition from flat to 1D folded and back to flat with increasing shear at $K = 3.0$. At low shear, the sheets do not buckle strongly, and interactions are insufficient to fold the sheet. As shear increases, S decreases, and the sheets buckle more strongly. Only if the buckling is strong enough can it initiate folding through interactions. Once shear becomes too strong, however, interaction is not strong enough to create persistent folds, so the sheet is flat. The critical value of $S \times 10^5$ above which sheets will not buckle if $K \sim 3$ is about 943 (which is indeed the lowest value of S in these simulations above the first buckling transition determined by Silmore *et. al.*[1]). The critical value of $\chi \times 10^5$ below which shear is too strong for sheets to fold is approximately 710.

$S = 6\chi K(a/L)(\sigma/L)$, so depending on the size of a given sheet, the 1D folded region between two flat regions may be inaccessible. Specifically, given a critical S^* and χ^*, we require $S < S^*$, $\chi > \chi^*$, and $K \sim 1$. As L increases, S decreases relative to χ, and we expect the 1D folded region to increase in width. Similarly, the region decreases in width with increasing σ. Therefore, this region increases in size as the size of the sheet increases relative to interaction range. Because $\chi^* \approx S^*$, $K \sim 1$, and $(L/a) \gg 1$, this region should exist if σ is not much larger than L (if $(\sigma/L) \lesssim (1/K)(S^*/6\chi^*)(L/a)$), that is, if sheets are not much smaller than the range of their

interactions. In such a system, χ is no longer the relevant dimensionless parameter as each bead would interact with every other bead in the sheet and therefore $\tilde{\epsilon}\sigma^2$ is no longer the relevant energy scale.

If both χ^{-1} and K are small, small deviations from the alignment of the folds caused by finite-size, edge, and/or initial orientation effects will cause the spontaneous formation of more folds, and the sheet can 2D fold. This is supported by the 2D folded sheets in Figure 4.6c, which are already in their 2D folded configuration at the first flip.

It is important to consider how sheet behavior would change with sheet shape. The equation for the optimal number of folds was derived for a rectangular sheet but the overall arguments are valid as long as the number of beads vertically along the folds does not change quickly across the width of the sheet (relative to the width of folded regions). This, along with the equation matching with the hexagonal sheet simulations in this thesis, suggest that Equation 4.11 is applicable for a broad range of sheet shapes with slowly changing widths.

Changing the sheet shape will affect the bending modes of the sheet and therefore the critical values of S corresponding to different buckling modes. This might affect the shape of the right edge of the folded/flat boundary due to the earlier discussion of the importance of S. Additionally, videos of 1D folded sheets (which can be found in the ESI of the paper published on this chapter[12]) show that the process of a 1D folded sheet losing folds involve these folds "sliding" along the width of the sheet until it hits the edge of the sheet, where the fold disappears. This process may be affected by the shape of the sheet. Simulations with different sheet shapes (e.g. rectangular or circular) would be valuable in illuminating the effects of sheet shape on the formation and breaking of 1D folds.

4.4 Rheological properties

4.4.1 Sheet viscosity calculations

The stress of a dilute suspension of force-free rigid particles is

$$\boldsymbol{\Sigma} = -p\langle \mathbf{I} \rangle + 2\eta \mathbf{E}^{\infty} + n\langle \tilde{\boldsymbol{\Sigma}} \rangle, \tag{4.12}$$

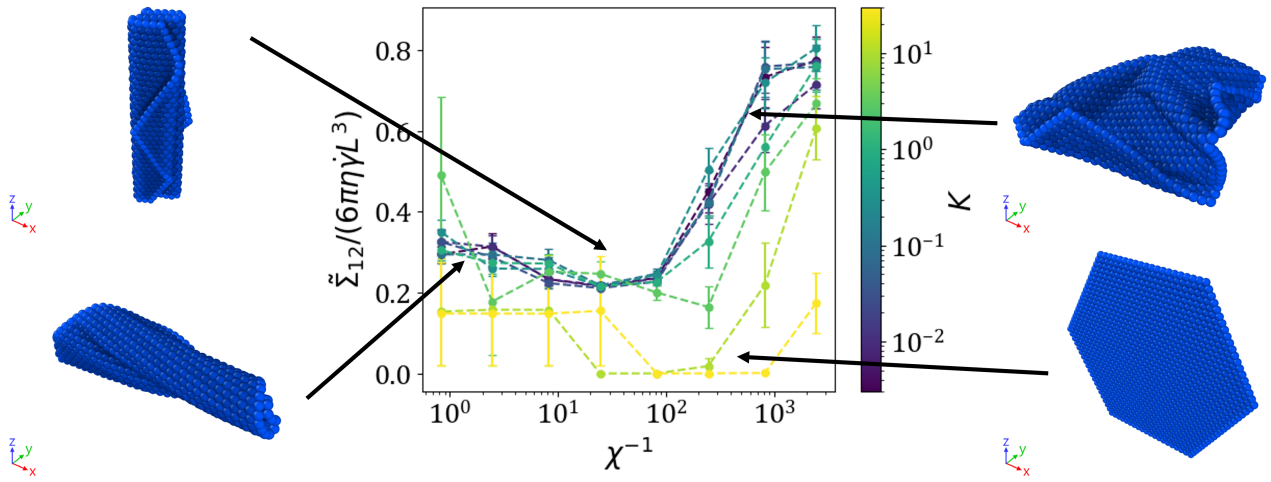

Figure 4.8 Stresslet ("sheet viscosity") averaged over the last $200\dot{\gamma}t$ and all ϕ. Error bars are 95% confidence intervals. Dotted lines drawn to guide the eye. On the borders are characteristic behaviors for different regions. x is the flow direction and y is the shear direction. Top left: high alignment folded. Bottom left: low alignment folded. Top right: tumbling. Bottom right: flat.

where p is the pressure, $\mathbf{E}^\infty$ is the rate-of-strain tensor, n is the number density of particles, and $\tilde{\mathbf{\Sigma}}$ is the stresslet (the first moment of the stress on a particle)[15]. As in previous work[9], we calculate an approximate upper bound on the stresslet using the minimum-bounding ellipsoid of the sheet, its Jeffrey orbits[16], and the stress for an ellipsoidal particle as found in Kim and Karilla[17]. The resulting average off-diagonal (flow-gradient) contribution to the stress over the last $200\dot{\gamma}t$, which is expected to grow linearly with the viscosity of a dilute suspension of these sheets, is shown in Figure 4.8.

The sheet viscosity shows 2 different behaviors based on K. For $K \gtrsim 10.0$, sheets have a small sheet viscosity at low χ^{-1} (i.e. low shear rates). Flat sheets which rotated about the vorticity axis contributed a higher stress than flat sheets in the flow-vorticity plane, with $\tilde{\Sigma}_{12}/(6\pi\eta\dot{\gamma}L^3) > 1.5$. These sheets were ones initially oriented with their normal close to the voriticty axis ($\phi = 85°$ or $90°$). Finally, once shear is strong enough to induce tumbling, shear-thickening behavior begins.

For $K \lesssim 1.0$, there is steady shear-thinning followed by shear-thickening behavior which appears near the same critical χ^{-1} as for the high K behavior. We discuss the origin of these behaviors in the next section.

4.4.2 Explanation of non-monotonic rheological properties

The nature of this shear-thinning into shear-thickening behavior is different than that with thermal energy but no interactions, which resulted from the "u-turn" radius of a flipping sheet[9]. In our current work, sheets in the shear-thinning regime are 1D or 2D folded.

We calculate several summary statistics for our simulations: the radius of gyration, R_g/L, the relative shape anisotropy, ζ^2, and the orientation of the largest axis of the minimum bounding ellipsoid of the sheet dotted with the vorticity axis, $|\mathbf{v}_1 \cdot \hat{z}|$, which we term the "alignment" of the sheet. The radius of gyration is a measure of the size of a sheet. Flat sheets have the largest possible radius of gyration for an inextensible sheet ($R_g \approx 0.65L$). 1D folded sheets have moderate $R_g \approx 0.5L$ and 2D folded sheets have smaller $R_g \lesssim 0.4$. Tumbling sheets have widely varying values of R_g but tend to average around the value for 1D folded sheets, $R_g \approx 0.5$. The relative shape anisotropy is a measure of the shape of a sheet. It ranges from 0 to 1, with 0 corresponding to a spherically symmetric sheet and 1 corresponding to a linear sheet. 1D folded sheets, who have a single large axis and are therefore "line-like," have large values of $\eta^2 \approx 0.7$. Flat sheets have low values of $\eta^2 \approx 0.25$. Different 2D folded sheets can have very different values of η^2 corresponding to different fold patterns. Tumbling sheets again have widely varying values of η^2, but tend to average with very low values of $\eta^2 \lesssim 0.2$, corresponding to a more "sphere-like" sheet. Finally, the alignment also varies from 0 to 1 and is a measure of the orientation of a sheet with respect to the vorticity axis.

Scatter plots of the sheet viscosity versus each of these quantities can be found in Appendix H. We note several notable features of these plots in the following discussion.

In Figure 4.9, we plot these three summary statistics as a function of χ^{-1} and K to explain the observed rheological behavior.

Folded: low $\chi^{-1} \lesssim 1.4 \times 10^2$, low $K \lesssim 3.0$ behavior

In this regime, sheets are 1D or 2D folded. Below $\chi^{-1} \sim 1.4 \times 10^1$, R_g/L and ζ^2 increase with shear rate. This corresponds to fewer sheets adopting the 2D folded conformation at higher shear rates. However, at even higher χ^{-1}, these values plateau despite shear-thinning continuing. Peculiarly,

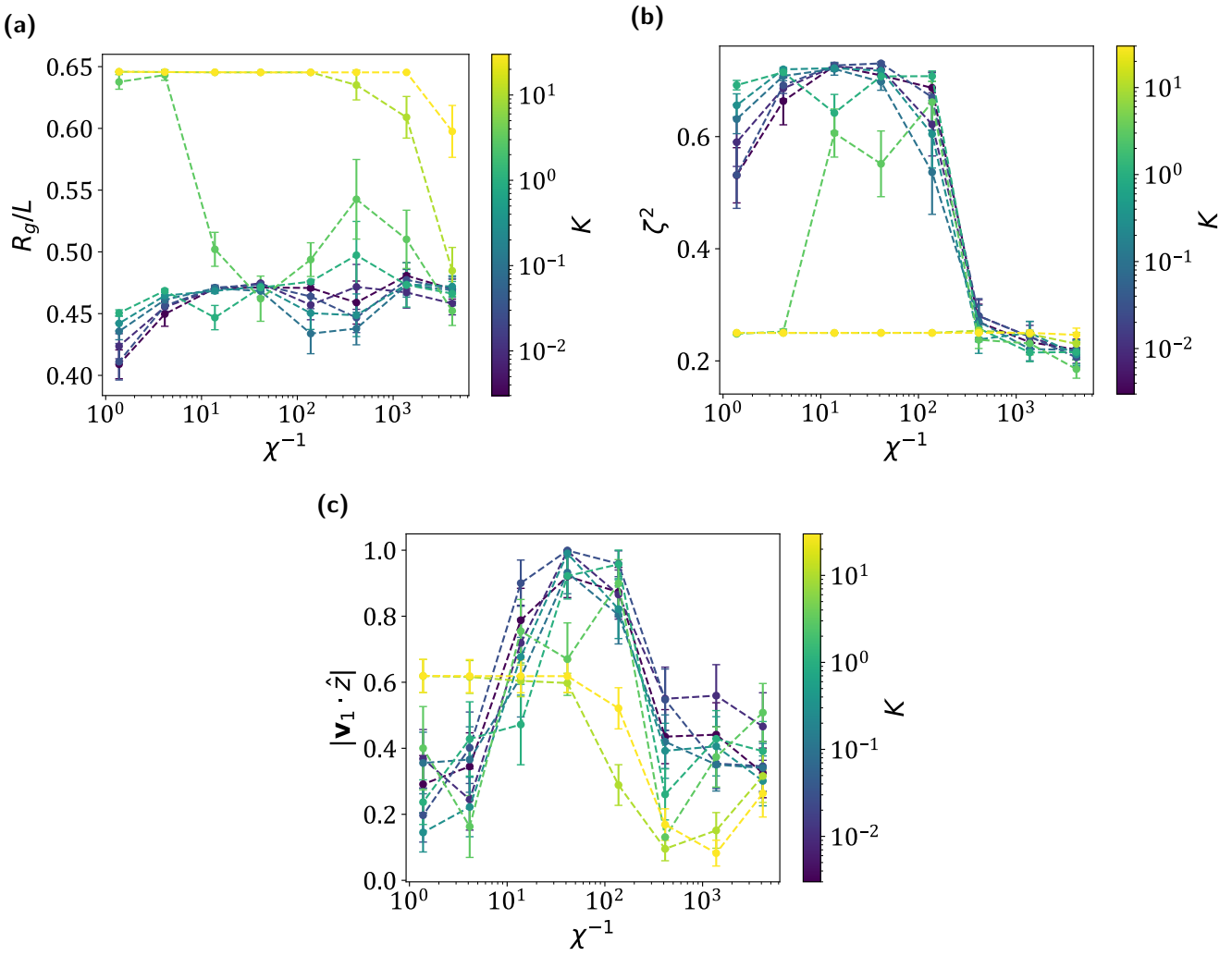

Figure 4.9 (a) Radius of gyration, (b) relative shape anisotropy, and (c) alignment averaged over the last $200\dot{\gamma}t$ and all ϕ. Error bars are 95% confidence intervals. Dotted lines drawn to guide the eye.

there appears to be no strong correlation between radius of gyration and sheet viscosity for folded sheets. Instead, the 1D folded sheets form two distinct clusters, one with a higher sheet viscosity ($\sim 0.35 \times 6\pi\eta\dot{\gamma}L^3$) and one with a lower sheet viscosity ($\sim 0.2 \times 6\pi\eta\dot{\gamma}L^3$).

This can be explained by looking at the average alignment of the sheets, which increases with shear rate. For prolate spheroids (ellipsoids with one large axis and two small axes) such as the minimum bounding ellipsoid of 1D folded sheets, large average alignment ("log-rolling") behavior is favored, which is why large average alignments result in lower sheet viscosities. Conceptually, this is because the distance the sheet "sticks out" into the shear axis is lower. These two types of motions (high and low average alignment) were observed in the past as the long-time behavior for

ellipsoidal particles in shear[18].

Indeed, for 1D folded sheets, most sheets have an average alignment close to either 1 or 0, indicating that they reach one of these terminal behaviors. The mechanism for these sheets deviating from their Jeffrey orbits to reach this high or low average alignment state is possibly the deformability of the sheets, which has been shown to influence the orbits sheets take in this manner[19,20]. 1D folded sheets with high average alignments have lower stresses than 1D folded sheets with low average alignments, explaining the two clusters observed in a scatter plot of sheet viscosity versus the R_g/L. The behavior of individual sheets is erratic, necessitating an average over initial condition. Thus, shear-thinning is due to a statistical average over many initial conditions. At higher shear rates, sheets on average adopt more log-rolling behavior, causing shear-thinning. This makes sense, as stronger shears cause greater perturbations in sheets, and thus allow for them to be more likely to be able to access the more favorable, lower stress, rotational behavior. Note that the decrease in average alignment at $\chi^{-1} \sim 1.4 \times 10^2$ is due to the rare appearance of tumbling sheets at this value of χ^{-1}.

Tumbling: high $\chi^{-1} \gtrsim 1.4 \times 10^2$, low $K \lesssim 3.0$ behavior

Once shear increases enough to cross the tumbling/folded boundary, R_g/L decreases slightly before recovering to close to the 1D folded value. ζ^2 and the average alignment, however, both decrease drastically. While the trends for R_g/L and the average alignment upon increasing χ^{-1} further are noisy and depend on the specific value of K, ζ^2 continuously decreases with shear rate. Thus, shear-thickening is caused by sheets with lower values of ζ^2 sticking out further into the shear axis at larger shear rates. This makes sense, as lower values of ζ^2 correspond to more spherically symmetric sheets, where the effect of increasing average alignment on sheet viscosity is lower. Thus, the sheet viscosity is higher for the same values of R_g/L and average alignment. Note that an average alignment value of approximately 0.52 corresponds to random orientation, which is around where the average alignments hover past the tumbling transition, suggesting that there is less preference for a particular rotational behavior. This is explained by the small values of ζ^2.

Sheets exhibiting teacup behavior have similar sheet viscosities to folded sheets. This behavior is seen as a low sheet viscosity cluster of low average alignment tumbling sheets at a range of R_g/L.

This behavior decreases in frequency at higher shear rates as it is a self-interaction-dependent behavior, which could contribute to the shear-thickening behavior. However, because this behavior is relatively rare, we expect that its effect is small. A detailed study of teacup behavior, its frequency as a function of χ^{-1} and K, and its effects on the sheet viscosity would be enlightening.

Flat: high $K \gtrsim 3.0$ behavior

For large enough values of K, bending rigidity overcomes interaction strength, and sheets are flat. The deviation from zero sheet viscosity, which would be the case for an infinitely thin sheet in the flow-vorticity plane, comes from a fraction of the flat sheets which lie in the flow-shear plane and rotate like a discus about the vorticity axis. Because these sheets stick out into the shear axis by the maximum amount possible for inextensible sheets, these sheets produce the highest sheet viscosities of any other conformational or rotational behavior. As shear increases, this behavior becomes less likely, again resulting in overall shear-thinning behavior. In Figure 4.8, the sheet viscosity curves for low χ^{-1} and $K \geq 10.0$ are flat. To observe shear-thinning behavior here, we suspect that more initial conditions need to be sampled. Past the tumbling transition, some sheets begin to tumble, resulting in shear-thickening as before.

The value of $K \sim 3.0$ appears to be right at the boundary between low K and high K behaviors and exhibits a mix of both behaviors, resulting in complex trends in R_g/L, ζ^2, and average alignment.

4.5 Exfoliation simulations — two sheets

A natural extension of single sheet simulations is multiple sheet simulations. As discussed in Chapter 3, sheet-sheet interactions are important for their macroscopic properties. Furthermore, one important process for the production of and subsequent application of 2D materials is exfoliation, where flow such as shear is applied to stacked sheets to separate them. Such as system has been studied in the past, using molecular dynamics simulations[21,22], which are limited by computational cost and can therefore only simulate relatively small sheets[h] and in two dimensions[23], which lacks what could be a very relevant third dimension. The simulations in this thesis are a middle-ground

[h] That said, these simulations do provide great insight into the specific physical mechanisms occurring here.

of these in terms of computational complexity and system detail, so, when thinking about adding more sheets to a simulation, the separation of two stacked, interacting sheets is a logical place to start. We present some preliminary results here before discussing further extensions.

4.5.1 Exfoliation model

We apply the same model as for single sheets, but with two sheets which begin stacked laterally a distance σ away from each other. These sheets interact with each other as well as with themselves. For simplicity and applicability to real systems, we consider identical sheets with the same parameters: bending rigidity, in-plane stiffness, interaction parameters (both internally and with each other), time step, etc.. This is a likely scenario in a real system[i]. We again rotate the sheets about the vorticity axis by a fixed angle θ to give an initial deviation from the flow axis and then about the flow axis by a fixed angle ϕ to simulate different initial conditions. We simulate the system for a variety of χ^{-1} and K, as in Chapter 4. We again render all sheet images using Ovito[6].

4.5.2 Results: types of sheet behaviors

We find that sheets can behave in three different ways depending on the values of χ^{-1}, K, and ϕ, which we depict in Figure 4.10. First, sheets can separate. In this case, the sheets eventually separate and move in separate directions due to shear and behave as earlier for single sheets. We call these separated sheets and this was the expected result if shear is strong relative to interactions, that is, $\chi^{-1} \gtrsim 1$. Second, sheets can flip. In this case, the sheets do not slide along each other and instead flip about the vorticity axis. Recall that in these sheets, the in-plane stiffness of the sheets is relatively large. For two sheets (or any object with non-zero thickness) to bend without any sliding, one must stretch while the other compresses. This is because, after a bend, the sheet on the outer part of the sheet covers a wider arc than the inner sheet. So, because the in-plane stiffness of these sheets is large, if the sheets cannot slide along each other, their bending is greatly restricted, causing the sheets to flip while remaining mostly flat. We call these sheets flipping and

[i]Modeling non-identical sheets also has applications. For example, multi-layer sheets have higher effective bending rigidities than individual sheets, as discussed in Chapter 3[24,25]. Therefore, a two sheet simulation where one sheet has a higher bending rigidity might approximate, for example, the peeling of a single sheet off a larger stack without needing to simulate the entire stack. Another common scenario might be a smaller sheet (in terms of L) separating from a larger sheet or a stack of larger sheets.

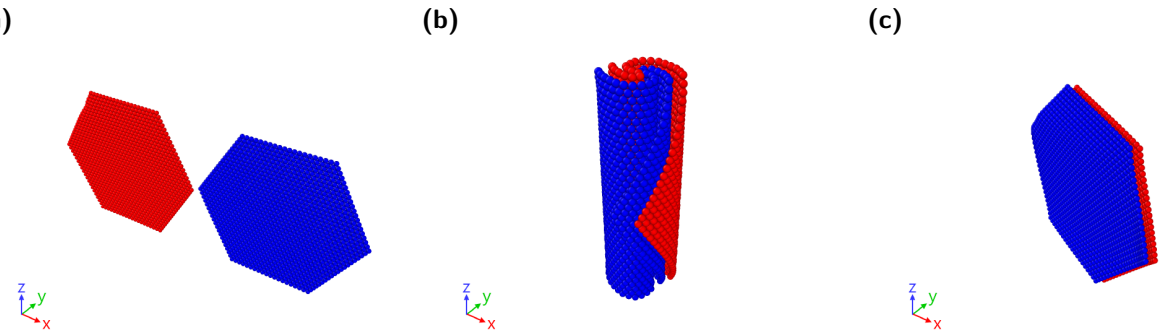

Figure 4.10 Sheet behaviors for two initially stacked sheets. (a) Two separating sheets ($\chi^{-1} = 410$, $K = 0.1$, $\phi = 0°$). (b) Waltzing sheets ($\chi^{-1} = 41$, $K = 0.1$, $\phi = 0°$). (c) Flipping sheets ($\chi^{-1} = 4.1$, $K = 0.1$, $\phi = 0°$). x is the flow direction and y is the shear direction.

this was the expected behavior is shear is weak relative to interactions, that is, $\chi^{-1} \lesssim 1$. Note that, unlike for single sheets, these sheets will periodically flip as their finite thickness (recall that single sheets are hydrodynamically thin in this model) causes them to always protrude into the shear axis. Note also that these sheets, when compared to single sheets, also much more frequently exhibit discus-like rotational behavior due to these flips. These rotations can exhibit a wide variety of alignments, just like discus behavior for single sheets.

Finally, sheets can initially slide along each other but wrap around each other and eventually rotate together. We call these sheets waltzing. In Figure 4.11, we depict the typical process for sheets to begin waltzing. The process involves an initial sliding of the sheets along each other. The edges of the two sheets which have separated then curl back towards the other sheet due to interactions, before eventually wrapping around each other fully. Notice that, because the sheets are sliding along each other, they are able to bend even though they are relatively inextensible. This behavior resembles rolled-up sheets due to its large relative shape anisotropy, suggesting that this conformation again has to do with minimizing stresses on the sheet due to shear.

In Figure 4.12, we show phase plots for the sheet behavior at two values of ϕ. We begin by interpreting the behavior at $\phi = 0°$. For low $K \lesssim 1$ and at large $\chi^{-1} \gtrsim \mathcal{O}(10^2)$, shear overcomes interaction and sheets separate. For low $K \lesssim 1$ and low $\chi^{-1} \lesssim \mathcal{O}(10^0)$, sheets cannot slide along each other and they are flipping. For moderate χ^{-1}, sheets can slide along each other, but interaction causes the sheets to wrap around each other. Shear is not strong enough to break this wrapping, so the sheets waltz. As K increases, the wrapping of sheets after their initial slide

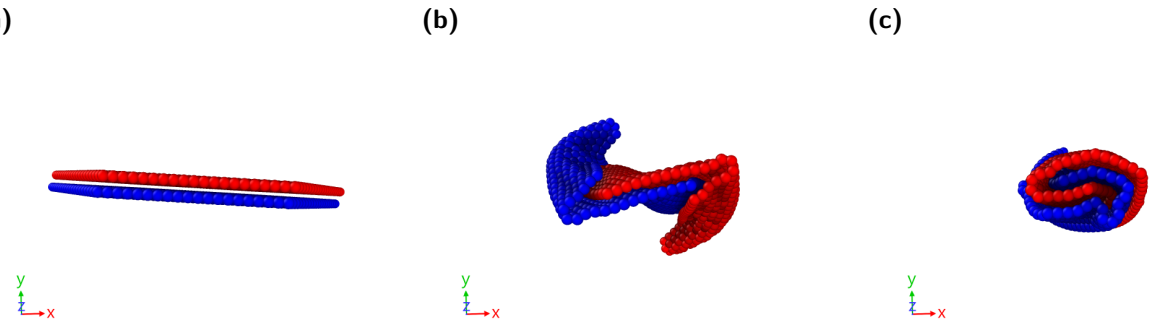

Figure 4.11 Formation of waltzing behavior for $\chi^{-1} = 141$, $K = 0.1$, $\phi = 0°$. (a) Initial condition. (b) Initial sliding followed by sheet ends wrapping due to interactions. (c) Completion of waltzing. x is the flow direction and y is the shear direction.

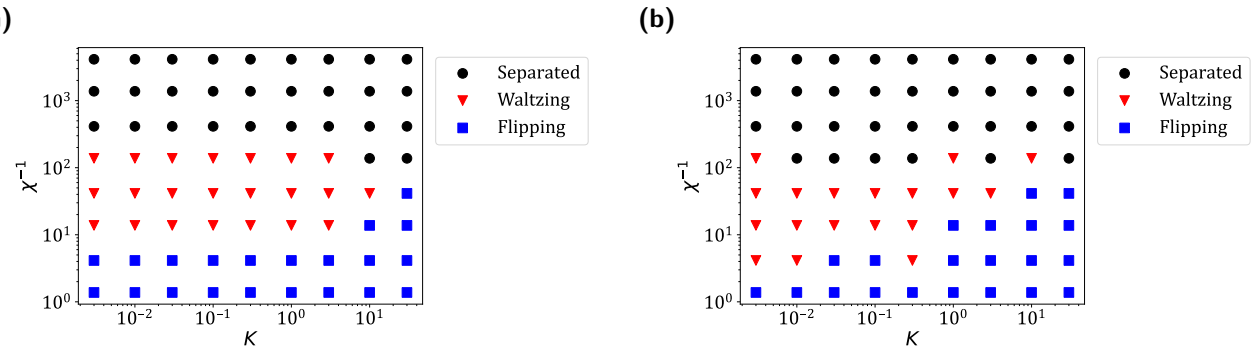

Figure 4.12 Phase plots for two sheet behaviors for (a) $\phi = 0°$ and (b) $\phi = 45°$.

is restricted. For χ^{-1} on the low end of waltzing for low K, the time for sheets to fully slide along each other and separate is longer than their flipping time, causing them to exhibit flipping behavior. For χ^{-1} on the high end of waltzing for low K, the opposite is true, causing them to separate. For other values of ϕ, the behavior is more complicated due to the initial orientation of the sheets, but the trends are the same. For $K = 30$, no waltzing behavior was observed, which, notably, is near the predicted transition in K to no folds given by Equation 4.11, where we predict wrapping (which forms a fold) to be energetically unfavorable.

Notably, the separated/waltzing boundary for low K here is the same as the tumbling/folded boundary for single sheets, suggesting that a similar wrapping mechanism occurs in the formation of a 1D folded sheet in its initial buckling. Interestingly, compared to single sheets, sheets waltz at much larger K from an initially flat configuration. This makes sense, as the extra thickness provided by the second sheet provides a force in the shear direction which enhances wrapping.

4.5.3 Results: rheological properties

As a first pass, we can estimate the rheological properties of a two sheet suspension by utilizing the rheological properties of a single sheet suspension. Specifically, if the sheets are flipping or waltzing, we can calculate the sheet viscosity as for single sheets. If the sheets separate, we can estimate the sheet viscosity as twice the single sheet viscosity at the given parameters (K, χ^{-1}, and ϕ). This is just an estimate as sheets do not necessarily behave as they did in the single sheet simulations after separating due to the initial condition dependence of the behaviors. Note that we cannot just calculate the sheet viscosity for each individual sheet after separation, as due to the periodic boundary conditions, sheets can continue to collide and interact after separating. One solution would be to run simulations for each sheet using the post-separation initial condition, although this was not done here.

The estimated two sheet simulation sheet viscosities are shown in Figure 4.13. Past the separated/waltzing boundary ($\chi^{-1} \gtrsim 1.4 \times 10^2$), the viscosity is the same as for single sheets, but double due to the additional sheet. For low K before the separated/waltzing boundary, we see sharp shear-thinning. This is because flipping behavior, which is more likely at low χ^{-1}, produces higher stresses. This is because the "u-turn" radius of flipping sheets, as described by Silmore *et al.*[1], is very large due to their restricted bending. Waltzing sheets, on the other hand, are similar to 1D folded sheets and thus represent a low stress conformation. Again, we see shear-thinning due to averaging over many initial conditions. For low enough K, all sheets are waltzing at $\chi^{-1} = 1.4 \times 10^2$, causing a plateau in the sheet viscosity.

For large $K > 10$, we see more complicated behavior, where there is shear-thickening before passing the separated/waltzing boundary. This shear-thickening is caused by an increase in the frequency of discus behavior, which produces much higher stresses than sheets periodically flipping about the vorticity axis while sitting in the flow-vorticity plane. Higher χ^{-1} correspond to larger shear rates, which in turn cause more initial configurations to be "knocked" into this higher stress behavior. Note that, because these sheets stay relatively flat (i.e. they are not very deformable), we lose the mechanism for sheets to transition between Jeffrey orbits, and thus they do not transition to a high or low alignment state, unlike with single sheets. Comparing the orientation of these

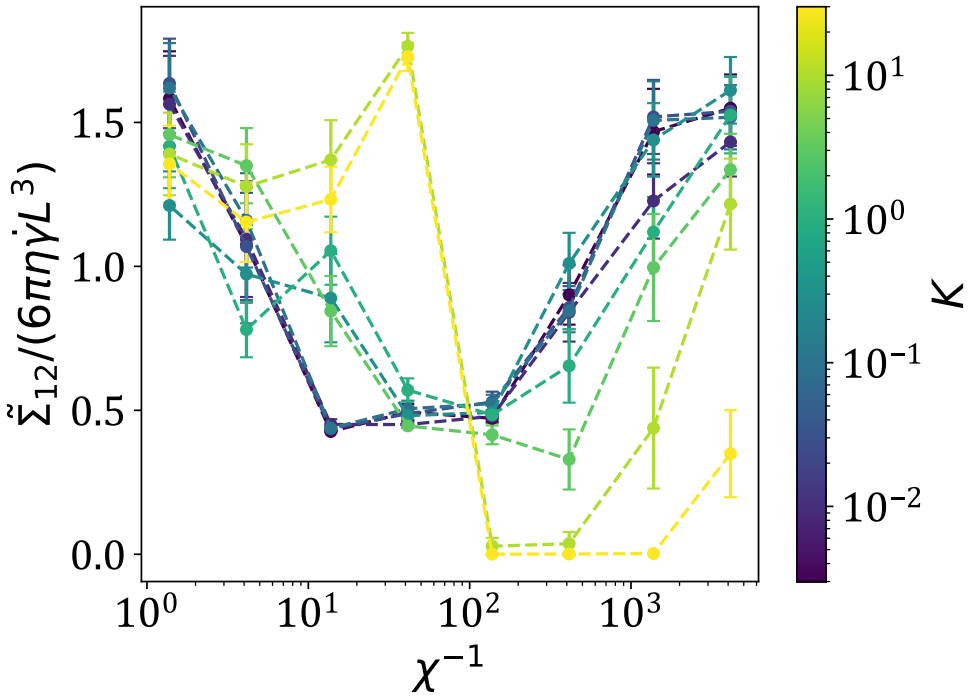

Figure 4.13 Sheet viscosity for two sheet simulations averaged over the last $200\dot{\gamma}t$ and all ϕ. Error bars are 95% confidence intervals. Dotted lines drawn to guide the eye.

sheets over time with the predicted orientations from Jeffrey orbits would help confirm this. For $\chi^{-1} = 4.1 \times 10^1$, all sheets with $K = 10$ and $K = 30$ are exhibiting discus behavior. Once χ^{-1} is large enough to cause sheets to separate, they behave as flat, single sheets with very low stress contributions, explaining the sharp drop off in sheet viscosity. Recall, however, that we predicted that flat behavior for $K \lesssim 20$ is a kinetic phenomenon. Especially for sheets with larger initial rotations, θ, about the vorticity axis, sheet-sheet interactions might cause sheets to be perturbed from the flat state and tumble instead. This might make the sheet viscosity drop off we observe much less sharp.

4.6 Conclusions

In this chapter, we examined the role of short-ranged attractive interactions in semi-flexible, athermal sheets in shear flow. We found a rich set of conformations which depend on two dimensionless groups: the material properties of the sheet (K) and experimental conditions (χ). We characterized these sheets as flat, tumbling, 1D folded, or 2D folded based on the eigenvalues of the gyration tensor. We found roughly that sheets folded when $K \lesssim \mathcal{O}(1)$ and $\chi^{-1} \lesssim \mathcal{O}(10^2)$, tumbled when

$K \lesssim \mathcal{O}(1)$ and $\chi^{-1} > \mathcal{O}(10^2)$, and were flat otherwise, although the exact behavior depended highly on initial condition near the boundaries.

We used the average signed local mean curvature of the sheets to show the nature of each type of sheet. Specifically, we identified parallel folds in 1D folded sheets and non-parallel folds in 2D folded sheets. We used a simple energetic argument to estimate the number of folds in a 1D folded sheet and showed that our equation matched well with the number of folds for these sheets in the low K limit. We also showed the relevance of the bending rigidity to shear (S) in inducing folding when $K \sim 3$. From this, we proposed a mechanism of an initial buckling followed by shear-induced annealing towards the most energetically favorable number of folds. The strength of shear determines the degree of annealing which is possible. We also discussed the expected effect of changing L on the rheological properties of the sheet based on the predicted number of folds in a 1D folded sheet.

We also calculated an approximate upper-bound on the stresslet, which is expected to grow linearly with the viscosity of a dilute suspensions of these sheets. We found shear-thinning followed by shear-thickening behavior with increasing shear rate, with different behavior depending on whether K is greater than or less than 1. This shear-thinning is present in the absence of sheet-sheet interactions and thermal fluctuations. Instead, it is a result of the average conformational and rotational behaviors of folded sheets. The changes in the conformation of a sheet with changing initial conditions follow trends, but are chaotic in the sense that small changes in the initial conformation can cause unpredicted changes in the final conformation near the boundaries between different conformations, similar to observations by Silmore *et. al.*[1]. We have yet to study the effect of changes in θ, the initial orientation of the sheet about the vorticity axis.

While the rheological behavior we observed in our exfoliation simulations appears similar to single sheets (although with much more pronounced shear-thinning behavior), just as when comparing athermal, self-interacing sheets to thermal, non-self-interacting sheets, the mechanisms which produced this behavior were different. Even with just two sheets, we found new conformational behaviors (flipping and waltzing) which had significant effects on the suspension properties. There are still many questions to be answered for two sheet simulations. What are the properties

of separated sheets from exfoliation compared to those observed in single sheet simulations? If two separate sheets collide, what are the resulting conformations as a function of χ^{-1} and K? What do the rotational properties of the different observed behaviors look like? And so on.

We note here that thermal fluctuations cause out-of-plane stiffening for sheets[26–29], which can change their bending rigidity quite significantly. Translating these simulations from the athermal limit to real systems, therefore, requires careful consideration of the effective bending rigidity of the system.

This type of shear-thinning into shear-thickening behavior is often attributed to the buildup and breakdown of agglomerates or other multi-sheet structures. However, we show that even in the dilute limit, this behavior can still emerge.

Bibliography

[1] KS Silmore, MS Strano, and JW Swan. Buckling, crumpling, and tumbling of semiflexible sheets in simple shear flow. *Soft Matter*, 17(18):4707–4718, 2021.

[2] J Rotne and S Prager. Variational treatment of hydrodynamic interaction in polymers. *The Journal of Chemical Physics*, 50(11):4831–4837, 1969.

[3] Y Yu and MD Graham. Wrinkling and multiplicity in the dynamics of deformable sheets in uniaxial extensional flow. *Physical Review Fluids*, 7(2):023601, 2022.

[4] JA Anderson, J Glaser, and SC Glotzer. Hoomd-blue: A python package for high-performance molecular dynamics and hard particle monte carlo simulations. *Computational Materials Science*, 173:109363, 2020.

[5] AM Fiore, F Balboa Usabiaga, A Donev, and JW Swan. Rapid sampling of stochastic displacements in brownian dynamics simulations. *The Journal of Chemical Physics*, 146(12):124116, 2017.

[6] A Stukowski. Visualization and analysis of atomistic simulation data with ovito–the open visualization tool. *Modelling and Simulation in Materials Science and Engineering*, 18(1):015012, 2009.

[7] MJ Bowick, A Košmrlj, DR Nelson, and R Sknepnek. Non-hookean statistical mechanics of clamped graphene ribbons. *Physical Review B*, 95(10):104109, 2017.

[8] G Gompper and DM Kroll. Random surface discretizations and the renormalization of the bending rigidity. *Journal de Physique I*, 6(10):1305–1320, 1996.

[9] KS Silmore, MS Strano, and JW Swan. Thermally fluctuating, semiflexible sheets in simple shear flow. *Soft Matter*, 18(4):768–782, 2022.

[10] L Liu, Z Shen, M Yi, X Zhang, and S Ma. A green, rapid and size-controlled production of high-quality graphene sheets by hydrodynamic forces. *RSC Advances*, 4(69):36464–36470, 2014.

[11] GL Klimchitskaya and VM Mostepanenko. Van der waals and casimir interactions between two graphene sheets. *Physical Review B*, 87(7):075439, 2013.

[12] WT Funkenbusch, KS Silmore, and PS Doyle. Dynamics of a self-interacting sheet in shear flow. *Soft Matter*, 20(22):4474–4487, 2024.

[13] B Tang, E Gao, Z Xiong, et al. Transition of graphene oxide from nanomembrane to nanoscroll mediated by organic solvent in dispersion. *Chemistry of Materials*, 30(17):5951–5960, 2018.

[14] FF Abraham and M Kardar. Folding and unbinding transitions in tethered membranes. *Science*, 252(5004):419–422, 1991.

[15] GK Batchelor. The stress system in a suspension of force-free particles. *Journal of Fluid Mechanics*, 41(3):545–570, 1970.

[16] GB Jeffery. The motion of ellipsoidal particles immersed in a viscous fluid. *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character*, 102(715):161–179, 1922.

[17] S Kim and SJ Karrila. *Microhydrodynamics: principles and selected applications*. Butterworth-Heinemann, 2013.

[18] GI Taylor. The motion of ellipsoidal particles in a viscous fluid. *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character*, 103(720):58–61, 1923.

[19] T Omori, Y Imai, T Yamaguchi, and T Ishikawa. Reorientation of a nonspherical capsule in creeping shear flow. *Physical Review Letters*, 108(13):138102, 2012.

[20] H Zhao and ESG Shaqfeh. The dynamics of a non-dilute vesicle suspension in a simple shear flow. *Journal of Fluid Mechanics*, 725:709–731, 2013.

[21] C Fu and X Yang. Molecular simulation of interfacial mechanics for solvent exfoliation of graphene from graphite. *Carbon*, 55:350–360, 2013.

[22] RC Sinclair, JL Suter, and PV Coveney. Graphene–graphene interactions: friction, superlubricity, and exfoliation. *Advanced Materials*, 30(13):1705791, 2018.

[23] G Salussolia, C Kamal, J Stafford, N Pugno, and L Botto. Simulation of interacting elastic sheets in shear flow: Insights into buckling, sliding, and reassembly of graphene nanosheets in sheared liquids. *Physics of Fluids*, 34(5):053311, 2022.

[24] H Perrin, H Li, and L Botto. Hydrodynamic interactions change the buckling threshold of parallel flexible sheets in shear flow. *Physical Review Fluids*, 8(12):124103, 2023.

[25] N Lindahl, D Midtvedt, J Svensson, et al. Determination of the bending rigidity of graphene via electrostatic actuation of buckled membranes. *Nano Letters*, 12(7):3526–3531, 2012.

[26] DR Nelson and L Peliti. Fluctuations in membranes with crystalline and hexatic order. *Journal de Physique*, 48(7):1085–1092, 1987.

[27] DR Nelson, T Piran, and S Weinberg. *Statistical mechanics of membranes and surfaces*. World Scientific, 2004.

[28] A Fasolino, JH Los, and MI Katsnelson. Intrinsic ripples in graphene. *Nature Materials*, 6(11):858–861, 2007.

[29] P Liu and YW Zhang. Temperature-dependent bending rigidity of graphene. *Applied Physics Letters*, 94(23):231912, 2009.

Chapter 5

Shear annealing of self-interacting sheets

Solution processing is an important step in producing and applying many 2D materials. The previous chapter demonstrated that self-interacting sheets in shear exhibit initial condition-dependent behavior, suggesting that time-dependent shear protocols can be utilized to tune their properties. In this chapter, we use two simple protocols, linear shear annealing and step changes in shear, to show how knowledge of the phase map for 2D materials can be exploited to generate sheets with the desired conformational and rotational properties. The content of this chapter was adapted from Funkenbusch, W. T., Silmore, K. S., & Doyle, P. S. (2024). "Shear annealing of a self-interacting sheet." Soft Matter *(submitted).*

5.1 Motivation for time-dependent flow protocols in 2D materials

One significant feature of the single sheet behavior observed in Chapter 4 was that the conformational behavior observed was highly dependent on initial condition. This presents a potential problem for application in real systems. Especially considering that the initial condition in our simulations were flat sheets and no flat sheets were observed for $K < 1.0$, suggesting such a state may not be accessible at any shear rate. Of particular importance, the no shear configuration of sheets might also look very different from a flat sheet[1–4]. Thus, it is important to examine what conformations can be obtained from many initial conditions. More broadly, we can ask questions about how the conformations of these systems can be controlled over time by applying non-steady flows or, more generally, by varying the parameters of the system over time.

This idea is not without precedent. Suspensions of colloidal particles exhibit different microstructures depending on factors such as the shear rate and volume fraction, analogous to sheet conformations. These microstructures, in turn, result in unique macroscopic material properties (liquids, gels, glasses, and crystals) which can be tuned for a particular application. Notably, colloidal suspensions with attractive interactions, even at volume fractions much lower than the glass transition, exhibit the same characteristic shear-thinning into shear-thickening rheological behavior seen in sheet suspensions due to the breakdown and formation of particle clusters[5–9]. The time scales for these microstructural changes cause the material to have memory of the flow history[10]. This memory allows for tuning of the final microstructure of a material through control of the applied flow protocol, for example by varying the "quenching" time from a high-shear "rejuvenation" (memoryless) regime[11,12]. Similarly, the tumbling regime for 2D sheets, which appears at high shear rates, represents a region where the conformation of sheets is continuously changing (i.e., they have access to many conformations, similar to a rejuvenation regime)[13]. The folded regime similarly represents a regime where sheet conformation and rotation do not change significantly with time and where the final conformational and rotational properties of sheets possibly depends highly on the initial configuration and flow history. Thus, it makes sense to examine the 2D sheet system from a similar framework as that used for colloidal suspensions.

We apply two different protocols to examine this: annealing protocols and step protocols.

5.2 Annealing simulations

Similarly to shear protocols applied to attractive colloidal suspensions[12], we apply a linear annealing shear protocol to sheets. First, we run a constant shear simulation at $\chi^{-1} = 4.1 \times 10^2$ for $2000\dot{\gamma}_0 t$, where $\dot{\gamma}_0$ is the initial shear rate. Then, the shear rate is decreased linearly to zero over a variable quench time $\dot{\gamma}_0 t_q$, relative to the initial shear rate: $\dot{\gamma}(T) = \dot{\gamma}_0(1 - T/t_q)$, where T is the time since beginning the annealing process. Finally, the sheet is allowed to equilibrate at no shear for another $2000\dot{\gamma}_0 t$. We choose equilibration times of $2000\dot{\gamma}_0 t$ to correspond to the longest time to observe conformational changes in Chapter 4, allowing for the collection of statistics. The largest quench time, $\dot{\gamma}_0 t_q = 10000$ is chosen to be much larger than this time. The smallest quench time,

$\dot{\gamma}_0 t_q = 10$, is chosen to be around than the typical time for a half rotation of a 1D folded sheet, which we believe to be the smallest relevant time scale for conformational changes in the system. Averages are taken over the last $500\dot{\gamma}_0 t$ of the constant shear and no shear portions. We expect that, by varying the quench time, the final conformation of the sheets will vary. Specifically, as quench time increases, sheets will have more time at moderate shear rates to rearrange themselves into more energetically favorable conformations.

Sample configurations along with their corresponding signed local mean curvature (SLMC, as described in previous work Silmore *et al.*[13]) at the start of annealing ($\dot{\gamma}_0 t = 2000$) and at the end of the simulation ($\dot{\gamma}_0 t = 4000 + \dot{\gamma}_0 t_q$) for $K = 0.03$ are shown in Figure 5.1. A video of an annealing simulation can be found in the ESI†. The images on the left of each subfigure, which correspond to tumbling sheets, can display a variety of folding patterns. After annealing, the sheets appear to become more compact due to the presence of attractive interactions (for $K \lesssim 1$). However, the behavior of an individual sheet depends on its configuration at the onset of annealing. Therefore, it is necessary to look at averages over many initial conditions.

To do this, we measure several properties of the final conformation: the radius of gyration, R_g, and the relative shape anisotropy, ζ^2. These properties are described in Chapter 4. As shown in that chapter, these properties give a more detailed description of sheets than just their conformations. Flat sheets have large R_g and low ζ^2, 1D folded sheets have moderate R_g and high ζ^2, 2D folded sheets have lower R_g and ζ^2 values than 1D folded sheets, and tumbling sheets typically have moderate R_g and low ζ^2, with larger variances in these values.

We report these summary statistics for the final sheet configurations in Figure 5.2. For $K \lesssim 1$, the radius of gyration does not change much with quench time. Its approximate value of $0.45L$ is smaller than tumbling sheets, showing that sheets become more compact due to attractive interactions to fairly consistent value, which makes sense for $K \lesssim 1$, as attractive interaction can overcome bending rigidity. For $K \gtrsim 1$, there are flat sheets as well as folded sheets, causing a higher average radius of gyration.

For $K \lesssim 1$, the relative shape anisotropy of the sheets increases on average with quench time. This indicates that sheets on average exhibit more 1D folded conformations. Indeed, Figure 5.2c

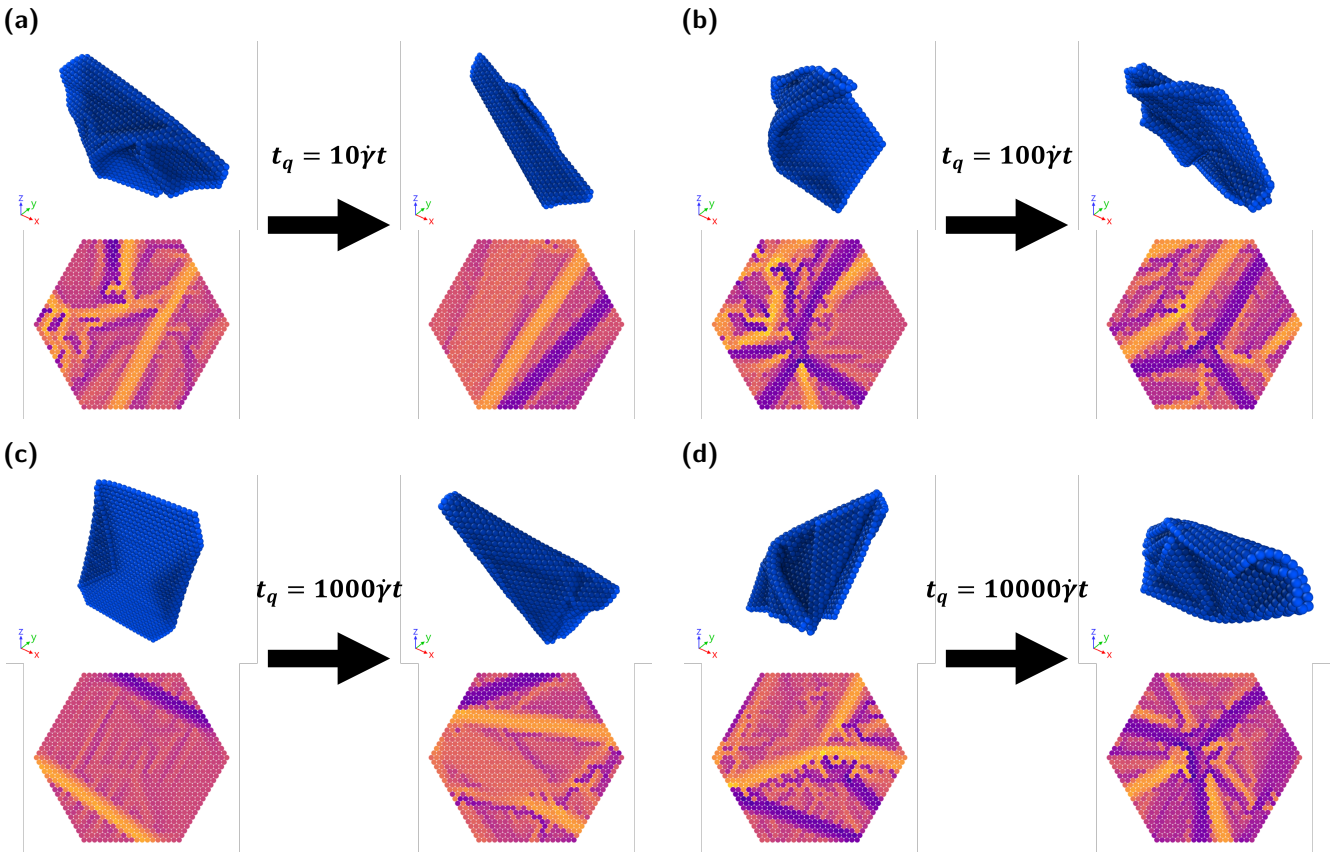

Figure 5.1 Sample configurations (top) and their corresponding signed local mean curvatures (bottom), shown before (left) and after (right) annealing, for $K = 0.03$ and (a) $t_q = 10\dot{\gamma}t$, (b) $t_q = 100\dot{\gamma}t$, (c) $t_q = 1000\dot{\gamma}t$, and (d) $t_q = 10000\dot{\gamma}t$. Before annealing, sheets are sheared with a constant $\chi^{-1} = 4.1 \times 10^2$, which is in the tumbling regime. In configurations, x is the flow direction and y is the shear direction. Signed local mean curvature plots are all drawn to the same scale.

calculates the fraction of sheets which are 1D folded, which matches with the relative shape anisotropy. This makes sense, as 1D folded sheets are slightly more energetically favorable than 2D folded sheets (which we discuss in more detail in Chapter 6). Longer quench times gives more time for sheets which were more 2D folded-like to anneal towards this more energetically favorable conformation. The fraction of 1D folded sheets not increasing monotonically and instead only trending upwards is due to two reasons. First, it is possibly a sampling error due to the relatively few number of initial conditions sampled (19). Second, because this is a classification scheme where all sheets are classified as either flat, 1D folded, or 2D folded, subtle changes in the average behavior of the sheets (such as ζ^2) are not picked up on. This highlights the importance of considering these summary statistics in our analyses. For moderate $K \sim 1$, the relative shape anisotropy and

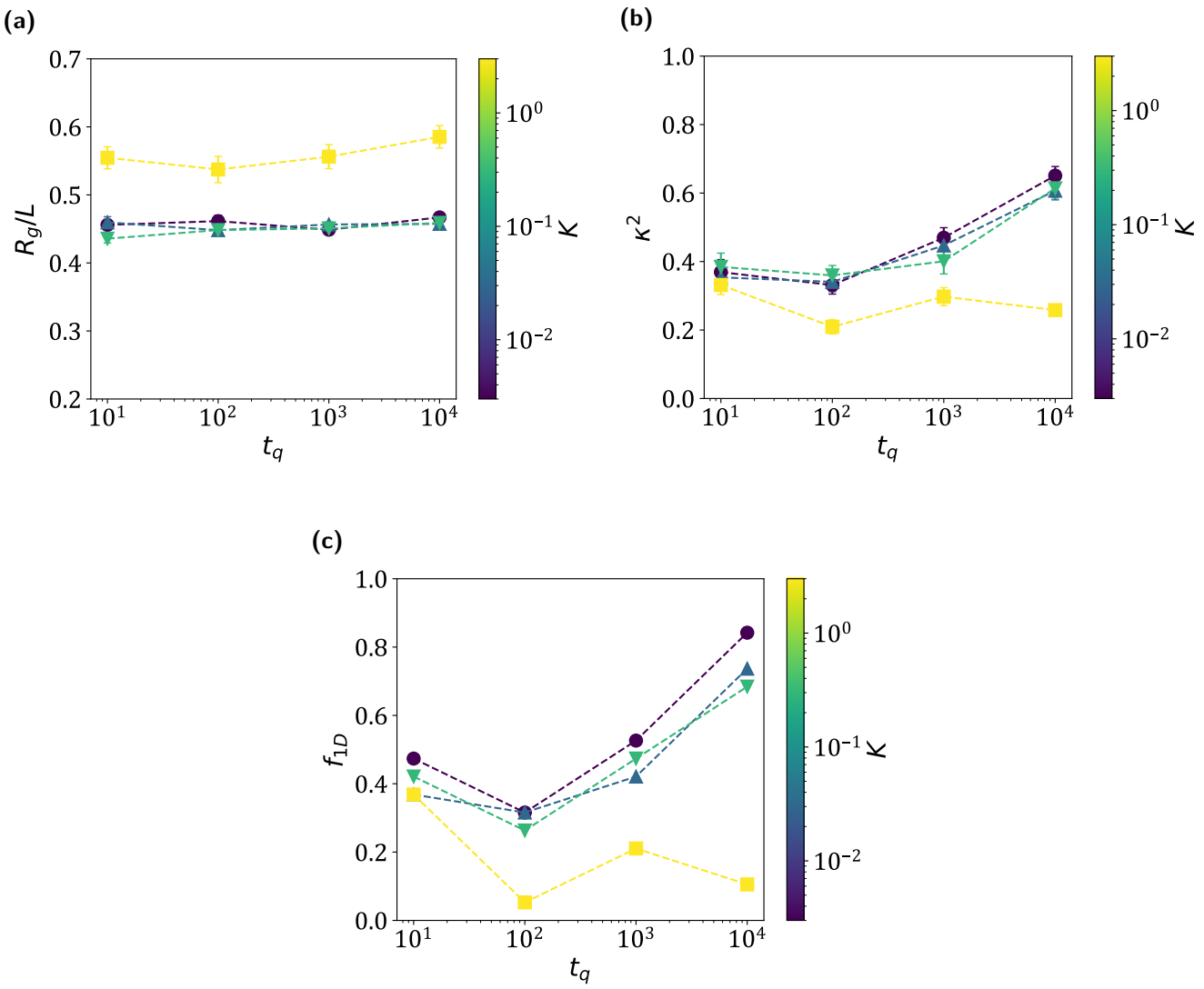

Figure 5.2 Sheet (a) radius of gyration and (b) relative shape anisotropy as a function of quench time, averaged over the last $500\dot{\gamma}_0 t$ of the no shear portion of annealing simulations and all ϕ. Error bars indicate 95% confidence intervals and dotted lines are drawn to guide the eye. (c) The fraction of sheets which were 1D folded after annealing.

fraction of 1D folded sheets also increased with quench time due to increased annealing, although the trend is suppressed because most sheets were flat in this regime. We suspect the initial decrease to be due to the relatively small number of annealing strain cycles causing larger variance in the final sheet conformations. For $K \gtrsim 10$, the relative shape anisotropy is relatively constant, corresponding to nearly all sheets being flat.

These results show that controlling quench time can indeed influence the final conformations of sheets. For large K, sheets tend to be flat in the athermal limit. For moderate to low $K \lesssim 1.0$, the

average relative shape anisotropy of the sheets increases significantly with quench time. For the shortest quench times, about 40% of sheets were 1D folded. For the longest quench times studied here, most sheets (about 50% to 85%) were 1D folded. It is unknown if long enough quench times would produce nearly 100% 1D folded sheets or if there is an asymptote with this shear protocol. For thermal energies which are small relative to interaction strength, these no shear conformations are likely stable, showing that the final properties of a suspension can be tuned by varying flow protocol. That said, in the no shear limit, the effects of thermal energy become more important, making an examination of the effects of thermal energy, as discussed in Chapter 6, even more important.

We note here that in these simulations and in the step protocol simulations, 1D folded sheets which appear at the end of protocols are usually rolled-up sheets. That is, they have either 0 or 2 folds along their center and a fairly constant curvature throughout the rest of the sheet (for example, see Step 2 in Figure 5.4). These sheets have slightly larger relative shape anisotropy values than sheets with many folds as their characteristic sizes perpendicular to the largest axis tend to be closer in size. Their appearance after protocols suggests that these sheets are more kinetically accessible than sheets with many folds, as sheets with many folds require alternating curvatures, which are generated due to the initial buckling from a flat initial condition. This, combined with rolled-up sheets' favorable energetic properties (see Chapter 6), explains why they are the more likely form of 1D folded sheet from a variety of initial conditions.

5.3 Step protocols

Next, we show that the features of the phase plot can be exploited to tune the final properties of a dilute suspension of sheets. As a target, we choose 1D folded, high alignment sheets at relatively low shear rates. This value was shown in the previous chapter to correspond to the viscosity of a dilute suspension of sheets, with high alignments corresponding to lower viscosities. In this chapter, we use the eigenvector of the gyration tensor corresponding to its largest eigenvalue as opposed to the largest semi-axis of the minimum bounding ellipsoid used in Chapter 4, as it points in a similar direction and is much cheaper to calculate.

A suspension with high alignment might have favorable macroscopic properties. For example, this suspension would have the minimal viscosity for a dilute suspension for a particular value of K as seen in the previous chapter. In addition, this suspension would be highly ordered and thus might carry favorable electronic properties, packing properties when concentrated, etc.. We would like to achieve this state from both the high-shear tumbling regime and from folded sheets with lower alignments, which would suggest that this desired state could be achieved from many initial conditions. In particular, achieving this state from tumbling would be desirable as tumbling accesses many different configurations.

Note the typical orientations of different conformations. Flat sheets have alignments which depend on their orientation. Flat sheets which rotate about the voriticity axis have alignments close to 0 while flat sheets in the flow-vorticity plane can have a range of alignments[a]. 2D folded sheets similarly have a range of alignments corresponding to the particular orbit they occupy. Tumbling sheets, on the other hand, tend to have alignments close to what would be achieved with a random sheet orientation, $|\mathbf{v}_1 \cdot \hat{z}| \approx 0.52$. This makes sense, as these sheets tend to have low values of ζ^2 and therefore little preference for different alignments in shear. 1D folded sheets tend to have alignments close to 0 or 1 depending on the initial condition, with 1 being more likely for higher shear rates.

To achieve this 1D folded, high alignment state, we note the features of Figure 4.8[b]. In particular, this figure shows that high alignment for sheets with relatively low values of K is achieved at values of χ^{-1} just before the tumbling transition ($1.4 \times 10^1 \lesssim \chi^{-1} \lesssim 1.4 \times 10^2$). While these results were achieved from flat initial configurations, we suspect that these shear rates have the effect of creating 1D folded, high alignment sheets for many initial configurations. The mechanism for changing average alignments was discussed in Chapter 4, having to do with the deformability of these sheets allowing sheets to shift Jeffrey orbits[14–16]. As χ^{-1} increases, the low alignment state becomes less favorable due to its higher stress, causing an increase in the high alignment state. However, it remains to be seen if this high alignment can be achieved from 1D folded sheets which

[a]This is because flat sheets have two large axes and thus their largest axis can point in many directions in the plane perpendicular to their short axis. This makes the alignment for flat sheets a less meaningful value to calculate than for other conformations.

[b]Recall that we use a slightly different, nearly equivalent definition of the alignment in this chapter compared to the previous one.

are already in low alignment orbits. Because sheet sections can slide along each other, this region likely has the additional effect of rearranging 2D folded sheets into lower stress, more energetically favorable 1D folded sheets[c]. At large enough values of χ^{-1}, the sheets tumble and alignment is broken down due to the low relative shape anisotropy of tumbling sheets, leading to this Goldilocks zone for alignment. Therefore, targeting this region requires knowledge of the tumbling transition.

We design two step protocols to target this region. Protocol 1 starts in the 1D folded regime at moderate alignments, $\chi^{-1} = 1.4 \times 10^1$ (Step 0), and steps into the high alignment zone, $\chi^{-1} = 1.4 \times 10^2$ (Step 1), before stepping back down to the original $\chi^{-1} = 1.4 \times 10^1$ (Step 2). Each step is run for equal strains of $2000\dot{\gamma}t$, where $\dot{\gamma}$ is the shear rate at each step. This protocol is run at a variety of K. We visualize this protocol on an abstraction of the phase plot from Chapter 4, seen in Figure 5.3.

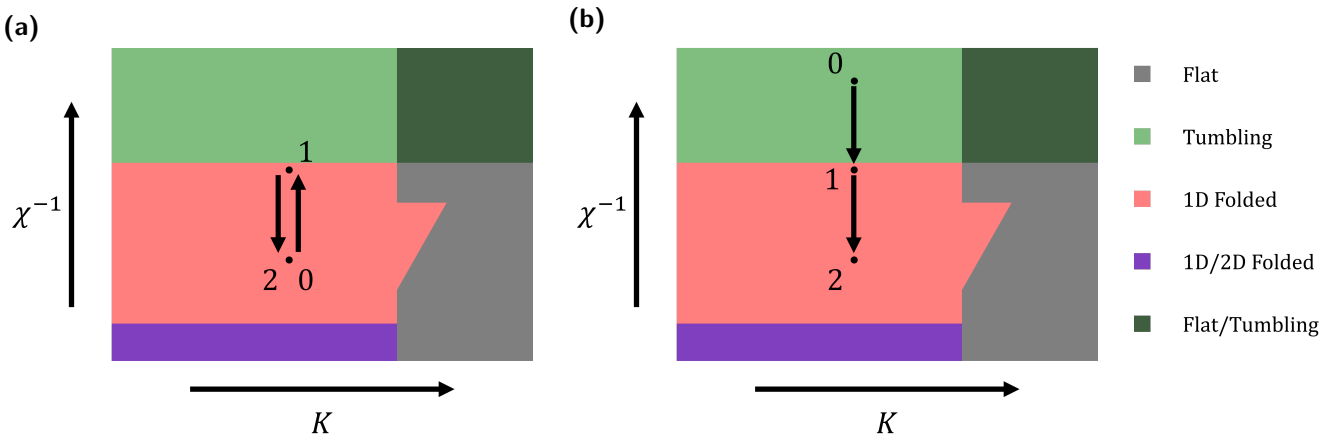

Figure 5.3 Abstractions of different protocols. Dots with numbers indicate constant χ^{-1} simulations run for $2000\dot{\gamma}t$. Arrows represent a factor of 10 change in χ^{-1}. (a) Protocol 1, which starts at $\chi^{-1} = 1.4 \times 10^1$. (b) Protocol 2, which starts at $\chi^{-1} = 1.4 \times 10^3$. The boundary between the flat/tumbling and tumbling regions lies at about $K = 1$. The boundary between the tumbling and 1D folded regions lies at about $\chi^{-1} = 1.4 \times 10^2$. The boundary between the 1D/2D folded and 1D folded regions lies at about $\chi^{-1} = 1.4$. For detailed values, see Figure 4.6.

For tunability, it is important that high alignment is achieved upon increasing χ^{-1} *and* is maintained upon returning to the lower χ^{-1}. Once sheets are in a high alignment state, we don't

[c]Again, we discuss energetics in more detail in Chapter 6.

expect reducing χ^{-1} to produce a mechanism for alignment to decrease[d], as the low alignment state still produces higher stresses than the high alignment state, making it likely that the high alignment state can be retained.

Protocol 2 starts in the tumbling regime, $\chi^{-1} = 1.4 \times 10^3$ (Step 0), steps down into the high alignment zone, $\chi^{-1} = 1.4 \times 10^2$ (Step 1), then steps down to $\chi^{-1} = 1.4 \times 10^1$. Each step is again run for $2000\dot{\gamma}t$ and this protocol is also run at a variety of K. This protocol tests if high alignment, 1D folded sheets can be generated from the tumbling regime, which accesses many more configurations than the 1D folded sheets. We visualize this protocol in Figure 5.3.

We calculate the radius of gyration, R_g, and relative shape anisotropy, ζ^2 as before as well as the alignment, $|\mathbf{v}_1 \cdot \hat{z}|$ of these simulations. Again, the behavior of an individual sheet is erratic, necessitating averages over many initial conditions, ϕ.

5.3.1 Protocol 1: Staying within the 1D folded regime

In Protocol 1, sheets are in the 1D folded regime for the whole simulation. Sample configurations at the end of each constant χ^{-1} step are shown in Figure 5.4.

We report the summary statistics for the sheets averaged over the last $500\dot{\gamma}t$ of each step in Figure 5.5. These plots for all initial conditions for the example value of $K = 0.03$ are shown in Figure 5.6 and these plots for all individual values K are provided in the ESI of the paper submitted on this work. For all K, the radius of gyration and relative shape anisotropy do not change significantly, showing that most sheets remained 1D folded throughout the simulations. Furthermore, the average alignment of these sheets shows that sheets are indeed aligned upon moving to the high alignment region at Step 1. This is despite many sheets having near-zero average alignments during Step 0. This confirms that this region can induce alignment in sheets from a variety of initial conditions, even a low alignment condition. Significantly, this high alignment is maintained upon returning to the original χ^{-1} in Step 2, as expected and desired. Even though sheets stayed 1D folded, their folds could change. In particular, higher shear rates allowed for fold

[d]Thermal energy can cause sheets to rotate. However, the rotational diffusivity of a sheet goes as L^{-3}, meaning that as sheet size increases, this rotation quickly becomes slow and is overcome by the effects of the stress of a particular alignment. We don't do a specific analysis of this here, although this again shows the importance of analyzing the effects of thermal fluctuations.

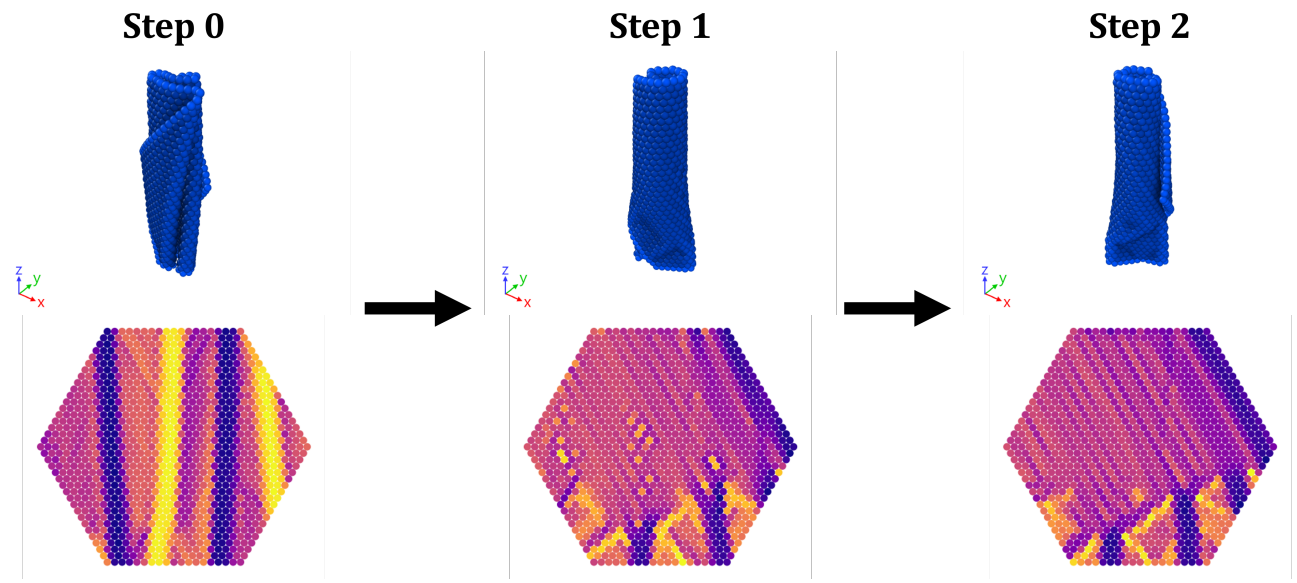

Figure 5.4 Sample configurations (top) and their corresponding signed local mean curvatures (bottom), shown at the end of each constant χ^{-1} step for Protocol 1. In configurations, x is the flow direction and y is the shear direction. Signed local mean curvature plots are all drawn to the same scale.

annealing and the formation of rolled-up sheets, for example as seen in Figure 5.4.

Note that the one sheet with $K = 0.03$ whose relative shape anisotropy decreased significantly in Step 1 exhibited teacup behavior, which is a classification of tumbling discussed in the previous chapter. Returning to a lower χ^{-1} in Step 2 caused this sheet to return to a high alignment, 1D folded conformation. However, this was not always the case for sheets exhibiting teacup behavior in Step 1. Some sheets became 2D folded in Step 2[e] while others became low alignment, 1D folded sheets. This is why the achieved alignment for many $K \leq 1.0$ was not perfect and indicates that avoiding tumbling is potentially important for achieving consistently high alignments and 1D folding, although the behavior was rare for all K.

For large enough $K \gtrsim 3$, the average radius of gyration is higher, the average relative shape anisotropy is lower, and the average alignment is lower. Examination of individual simulations reveals that this is due to the presence of a fraction of flat sheets. Some of these flat sheets become 1D folded at Step 1 due to the higher χ^{-1}, as indicated by the decrease in average radius of gyration and increase in relative shape anisotropy. Because flat sheets in the flow-vorticity plane can have a variety of alignments due to having two large axes[f], this makes the average alignment for large

[e]Recall that 2D folded sheets can have a wide variety of alignments.
[f]Unfortunately, this makes it so that the specific value of alignment we find for $K \geq 3.0$ is less meaningful.

K lower, even though most sheets are either high alignment 1D folded sheets or flat sheets in the flow-vorticity plane. Together, these data show that the final properties of the suspension can indeed be changed by modifying flow protocol.

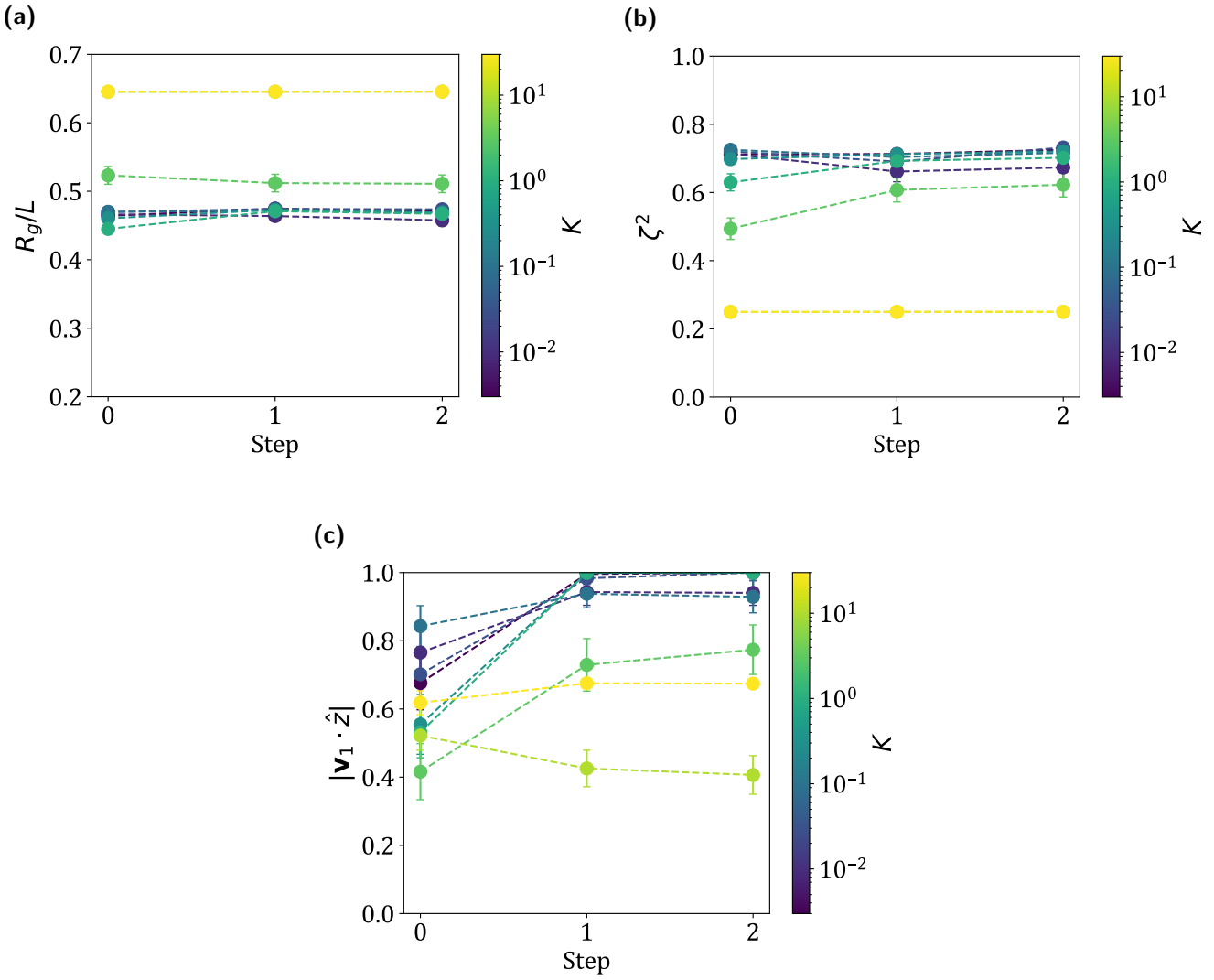

Figure 5.5 Sheet (a) radius of gyration, (b) relative shape anisotropy, and (c) alignment averaged over the last $500\dot{\gamma}t$ of each step for Protocol 1. Error bars indicate 95% confidence intervals and dotted lines are drawn to guide the eye.

5.3.2 Protocol 2: Tumbling to 1D folding

In Protocol 2, sheets start in the tumbling regime, are moved to the high alignment region by lowering χ^{-1}, then are moved outside of the high alignment region by lowering χ^{-1} further. Sample configurations at the end of each constant χ^{-1} step are shown in Figure 5.7.

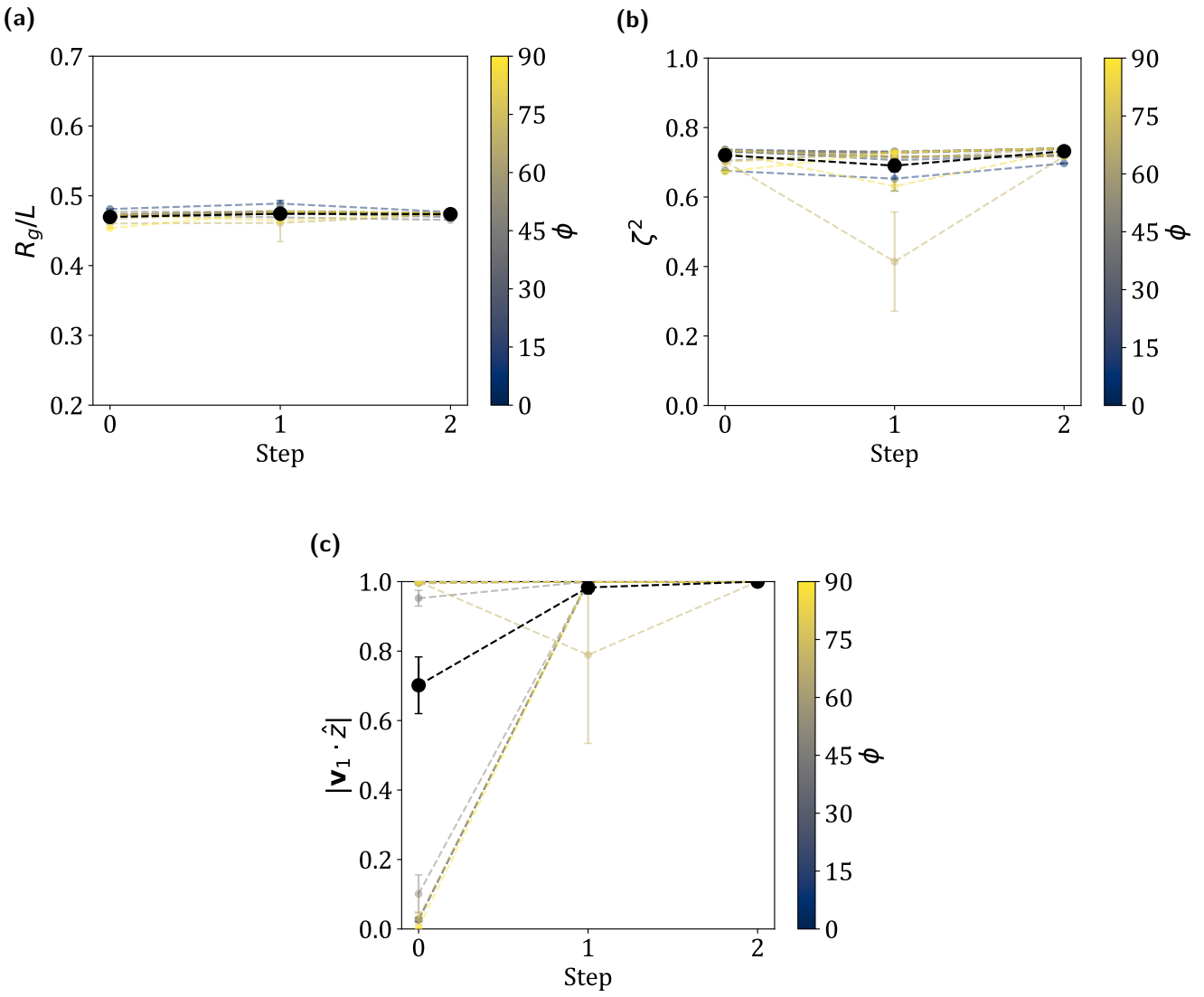

Figure 5.6 Sheet (a) radius of gyration, (b) relative shape anisotropy, and (c) alignment for each initial condition with $K = 0.03$, averaged over the last $500\dot{\gamma}t$ of each step for Protocol 1. Black lines indicate the weighted average over all initial conditions. Error bars indicate 95% confidence intervals and dotted lines are drawn to guide the eye.

We report the summary statistics for a range of K in Figure 5.8. These plots for all initial conditions for the example value of $K = 0.03$ are shown in Figure 5.9 and these plots for all individual values K are provided in the ESI of the paper submitted on this work. For $K \lesssim 1$ and from Step 0 to Step 1, the radius of gyration of the sheets stays approximately the same while the relative shape anisotropy increases. This shows that sheets are indeed transitioning from tumbling to 1D folded from Step 0 to Step 1. The alignment also increases from Step 0 to Step 1, as expected,

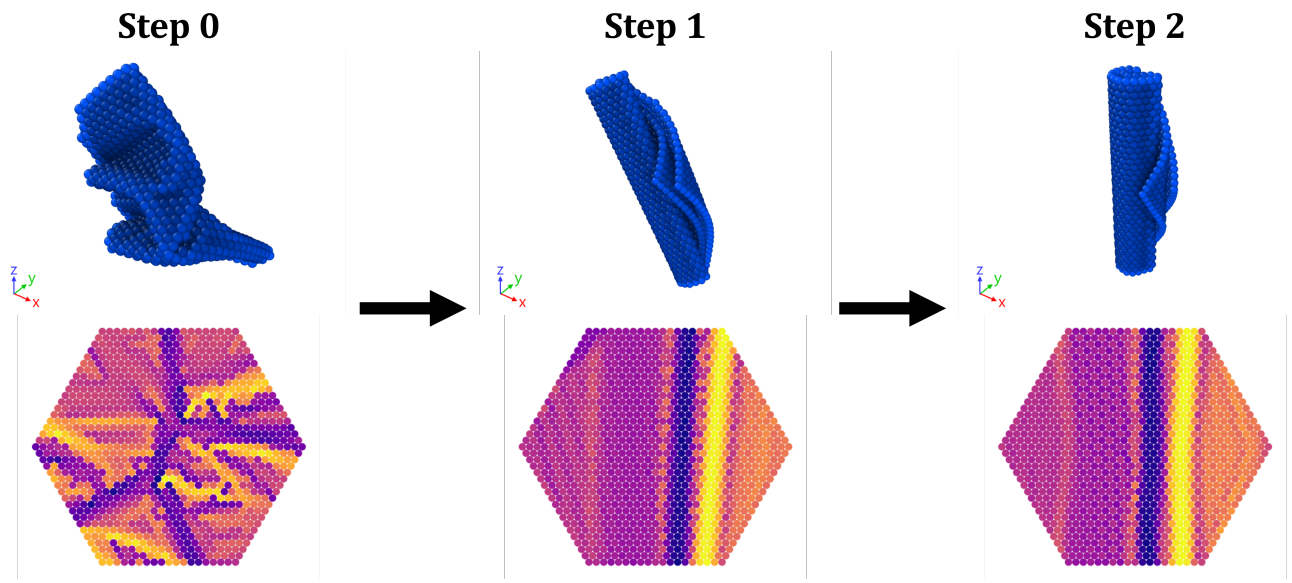

Figure 5.7 Sample configurations (top) and their corresponding signed local mean curvatures (bottom), shown at the end of each constant χ^{-1} step for Protocol 2. In configurations, x is the flow direction and y is the shear direction. Signed local mean curvature plots are all drawn to the same scale.

with sheets transitioning from a roughly random orientation with tumbling to high alignment 1D folded conformations. Interestingly, this increase also occurs from Step 1 to Step 2. This means that not only does entering the high alignment regime increase alignment, but changing χ^{-1} within the folding regime also can increase the average alignment of sheets. Lower values of χ^{-1}, even outside the high alignment regime, can change the rotational behavior of sheets, evidently allowing sheets to adopt new, lower stress orbits. Notably, the final alignment decreases with increasing K, suggesting that sheet deformability is key for sheet alignment.

Analysis of individual simulations also shows several sheets adopt high average alignment 2D folded behavior, depicted in Figure 5.9, which originates from teacup tumbling behavior described previously. Protocol 2 observes more of this behavior than Protocol 1 due to tumbling sampling more behaviors which are "closer" conformationally to teacup behavior. This results in a lower average relative shape anisotropy for this protocol. However, the average alignment which is achieved is similar to Protocol 1. The resulting sheet viscosity from such a suspension, however, would be higher than in Protocol 1 due to the more frequent presence of 2D folded sheets.

For $K \gtrsim 3.0$, the behavior is similar to $K \lesssim 1.0$, but with the presence of flat sheets causing higher radii of gyration, lower relative shape anisotropies, and lower average alignments. For

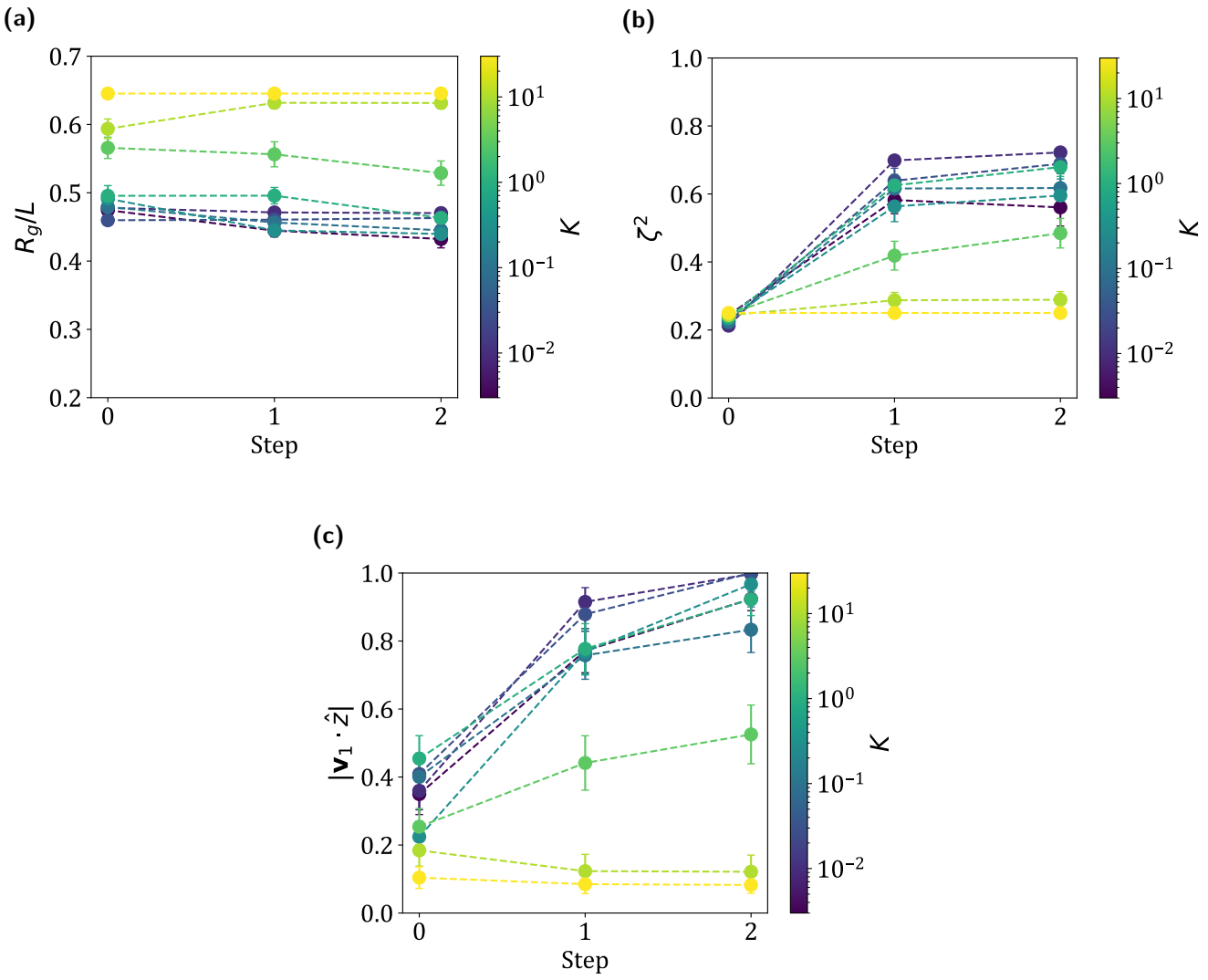

Figure 5.8 Sheet (a) radius of gyration, (b) relative shape anisotropy, and (c) alignment averaged over the last $500\dot{\gamma}t$ of each step for Protocol 2. Error bars indicate 95% confidence intervals and dotted lines are drawn to guide the eye.

$K = 30.0$, every sheet is flat throughout the simulation, causing no significant change in any summary statistic.

Together, these data show that, even from the tumbling regime where sheets sample many different configurations, high alignment, 1D folded behavior can be achieved using fairly simple shear protocols.

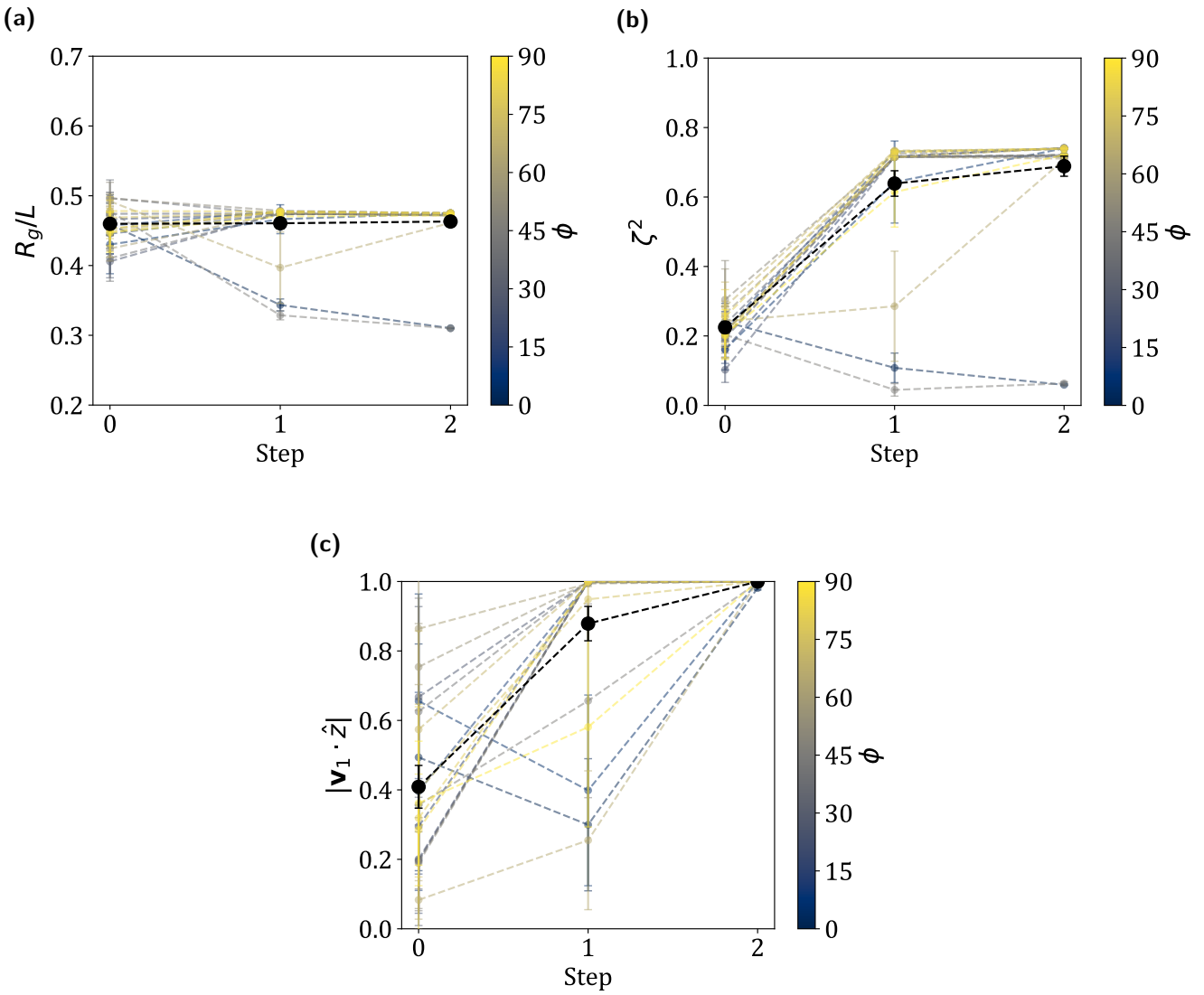

Figure 5.9 Sheet (a) radius of gyration, (b) relative shape anisotropy, and (c) alignment for each initial condition with $K = 0.03$, averaged over the last $500\dot{\gamma}t$ of each step for Protocol 2. Black lines indicate the weighted average over all initial conditions. Error bars indicate 95% confidence intervals and dotted lines are drawn to guide the eye.

5.4 Conclusions for time-dependent protocols

In this chapter, we examine several time-dependent protocols for self-interacting, semi-flexible sheets in simple shear. Through linear annealing from tumbling to zero shear, we show that the average final properties of the sheets (in the athermal limit) can be tuned through varying the quench time. In particular, while the radius of gyration of sheets remains about the same regardless of quench time, the relative shape anisotropy increases due to the increased presence of 1D folded

sheets. For large enough K, sheets are flat instead and these trends disappear.

Next, by implementing step changes in χ^{-1}, we show that sheets can also be aligned. Specifically, we show that sheets in the 1D folded regime can be aligned by increasing χ^{-1} to a value in a high alignment regime, which lies just below the tumbling transition. This alignment is maintained upon returning to a lower value of χ^{-1}, showing hysteresis in the rotational behavior of these sheets. This protocol could be used to tune bulk properties. For example, a suspension of high alignment sheets has a lower viscosity than sheets at the same shear rate but lower average alignments.

We also show that alignment can be achieved from the tumbling regime, again by stepping into the high alignment 1D folding regime before decreasing the value of χ^{-1} lower. Interestingly, the alignment of these sheets increases upon lowering χ^{-1} below the high alignment regime, showing that changing χ^{-1} allows sheets to adopt new, lower stress orbits. The final conformations of these sheets are 1D folded or 2D folded, depending on the initial condition. The final alignment decreasing with K suggests that deformability is vital to achieving high alignment. For high enough values of K, sheets are flat, and the trends disappear.

It is possible that further design is possible through varying the time at each step or values of χ^{-1} at each step, time between steps or applied rate of change of χ^{-1} between steps, or by implementing more complex protocols with more steps. For example, due to the increase in average alignment from Step 1 to Step 2 in Protocol 2, a cyclic shear protocol which avoids the tumbling regime might achieve even better alignments after several cycles.

Furthermore, it might be possible to tune sheet properties even more specifically by introducing changes in K as well as χ^{-1}. This might be achieved experimentally, for example, by changing solvent to induce more or less favorable self-interactions and/or bending rigidities (see, for instance, work by Tang *et al.*[1,2]) or by varying temperature to induce changes in bending rigidity[17–20]. Studies on the effect of protocols varying K would be valuable.

We note here that our simulations do not explore the entire space of potential sheet conformations. For example, kinetoplast sheets exhibit "puckered" equilibrium behavior[21] and nanoplatelets can be induced to have a helical structure through the addition of ligands[3]. Examination of the responses of these sorts of initial conditions in shear would be valuable, although care must be

taken as these materials might exhibit inhomogeneities (e.g. kinetoplast densities at the rim versus the interior of the sheet).

Using simple protocols to traverse the χ^{-1}, K phase space, we show that the conformational and rotational properties of sheets can be tuned with varying degrees of consistency. We hope that this work can help inform the design of such protocols to produce sheets with the desired properties.

Bibliography

[1] B Tang, Z Xiong, X Yun, and X Wang. Rolling up graphene oxide sheets through solvent-induced self-assembly in dispersions. *Nanoscale*, 10(8):4113–4122, 2018.

[2] B Tang, E Gao, Z Xiong, et al. Transition of graphene oxide from nanomembrane to nanoscroll mediated by organic solvent in dispersion. *Chemistry of Materials*, 30(17):5951–5960, 2018.

[3] H Po, C Dabard, B Roman, et al. Chiral helices formation by self-assembled molecules on semiconductor flexible substrates. *ACS Nano*, 16(2):2901–2909, 2022.

[4] D Monego, S Dutta, D Grossman, et al. Ligand-induced incompatible curvatures control ultrathin nanoplatelet polymorphism and chirality. *Proceedings of the National Academy of Sciences*, 121(9):e2316299121, 2024.

[5] JF Brady and G Bossis. Stokesian dynamics. *Annual Review of Fluid Mechanics*, 20(1):111–157, 1988.

[6] G Bossis and JF Brady. The rheology of brownian suspensions. *The Journal of Chemical Physics*, 91(3):1866–1874, 1989.

[7] J Bender and NJ Wagner. Reversible shear thickening in monodisperse and bidisperse colloidal dispersions. *Journal of Rheology*, 40(5):899–916, 1996.

[8] J Bergenholtz, JF Brady, and M Vicic. The non-newtonian rheology of dilute colloidal suspensions. *Journal of Fluid Mechanics*, 456:239–275, 2002.

[9] J Mewis and NJ Wagner. *Colloidal suspension rheology*. Cambridge university press, 2012.

[10] S Jamali, GH McKinley, and RC Armstrong. Microstructural rearrangements and their rheological implications in a model thixotropic elastoviscoplastic fluid. *Physical Review Letters*, 118(4):048003, 2017.

[11] N Koumakis, E Moghimi, R Besseling, et al. Tuning colloidal gels by shear. *Soft Matter*, 11(23):4640–4648, 2015.

[12] S Jamali, RC Armstrong, and GH McKinley. Time-rate-transformation framework for targeted assembly of short-range attractive colloidal suspensions. *Materials Today Advances*, 5:100026, 2020.

[13] KS Silmore, MS Strano, and JW Swan. Buckling, crumpling, and tumbling of semiflexible sheets in simple shear flow. *Soft Matter*, 17(18):4707–4718, 2021.

[14] GI Taylor. The motion of ellipsoidal particles in a viscous fluid. *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character*, 103(720):58–61, 1923.

[15] T Omori, Y Imai, T Yamaguchi, and T Ishikawa. Reorientation of a nonspherical capsule in creeping shear flow. *Physical Review Letters*, 108(13):138102, 2012.

[16] H Zhao and ESG Shaqfeh. The dynamics of a non-dilute vesicle suspension in a simple shear flow. *Journal of Fluid Mechanics*, 725:709–731, 2013.

[17] DR Nelson, T Piran, and S Weinberg. *Statistical mechanics of membranes and surfaces*. World Scientific, 2004.

[18] DR Nelson and L Peliti. Fluctuations in membranes with crystalline and hexatic order. *Journal de Physique*, 48(7):1085–1092, 1987.

[19] P Liu and YW Zhang. Temperature-dependent bending rigidity of graphene. *Applied Physics Letters*, 94(23):231912, 2009.

[20] A Fasolino, JH Los, and MI Katsnelson. Intrinsic ripples in graphene. *Nature Materials*, 6(11):858–861, 2007.

[21] AR Klotz, BW Soh, and PS Doyle. Equilibrium structure and deformation response of 2d kinetoplast sheets. *Proceedings of the National Academy of Sciences*, 117(1):121–127, 2020.

Chapter 6

Future considerations for self-interacting 2D materials in shear

There are many directions the work from the previous two chapters could go. In this chapter, we discuss some of these directions, primarily focusing on work related to the model discussed in the previous chapter.

6.1 Introduction

The primary focus of the preceding work on sheets was to study how self-interaction affects the conformational properties (and thereby the bulk properties) of a sheet suspension and how these properties can be controlled with flow. Due to this and time constraints, there are several directions this work could go which would constitute either a more in-depth analysis of existing simulations or an interesting extension of the work requiring more simulations. We present some of these directions here, focusing primarily on directions which use the model presented in this work.

6.2 Initial condition effects

In this thesis, we examined the effect of initial sheet orientation by varying the initial angle of a flat sheet about the flow axis, ϕ. We used a fixed angle about the vorticity axis, $\theta = 5°$, as this gives a deviation from the flow-shear plane which causes the sheet to initially flip. One primary result of Chapter 4 was the importance of initial condition in the final sheet conformation which is

obtained, making it important to study the effect of varying θ as well. The specific conformation (number of folds) for 1D folded sheets in particular is highly dependent on the initial buckling of the sheet, which is affected by θ. Sheets at high χ^{-1} and K, which were tumbling or flat depending on the initial conformation, might also be affected by the value of θ. Finding these effects would be valuable for improving the results of Chapter 4, but would likely not significantly affect the results of Chapter 5, which were largely achieved from the tumbling regime, which we expect to not be significantly impacted by θ.

6.3 Confirmation of scalings

Several new dimensionless groups were presented in Chapter 4, but their scalings were not confirmed. Specifically, the scaling of S with L was confirmed by Silmore *et al.*[1] and the scaling for n^*_{fold} was confirmed numerically and empirically with simulations of sheets with $L = 79a$, but scalings for χ and K have yet to be confirmed.

The ratio of shear strength to interaction strength, χ, includes predictions for how the system should behave as L and σ are varied. Because all simulations done in Chapter 4 used the same sheet dimensions and interaction range, this scaling was not tested. To confirm this scaling, it should be shown that the critical χ for tumbling is constant. This should be done in a parameter sweep over χ, K, as seen in Chapter 4. While a sweep over ϕ would be ideal, this would drastically expand the parameter space and the tumbling/folded boundary was not a function of ϕ, so it is likely unnecessary.

There are several limitations to scaling confirmation. First, while finite-size effects for self-avoiding, thermally-fluctuating sheets without flow seem to disappear for very small sheets (for example, Figure 3 in work by Babu and Stark[2]), the finite-size limit may be higher for self-interacting sheets in shear. For example, Yllanes *et al.* studied thermally-fluctuating perforated sheets and found finite-size effects persisted to $L/a \approx 50$ due to the addition of a new length scale corresponding to the size of a perforation[3]. Similarly, the existence of σ in our simulations presents a new length scale which may introduce finite-size effects at smaller sheet sizes[a]. Our

[a]Due to our introduced length scale being similar in size to that introduced by Yllanes *et al.*, we expect sheets significantly smaller than those implemented in our simulations to start exhibiting finite-size effects.

relevant dimensionless group goes as L^{-1}, necessitating large changes in L to confirm scaling predictions. The relatively weak L dependence of χ, the possibility of finite-size effects at smaller L, along with the simulations being quadratic in the number of beads (and thus quadratic in L), make it more difficult, but not prohibitively difficult, to confirm the L scaling of χ.

Testing χ's scaling with σ is more difficult. Due to self-interactions occurring along the length of the sheet, σ cannot be increased too much before they begin to significantly affect the in-plane stiffness of the sheets (more on this later), necessitating stiffer bead-bead spring constants and therefore shorter time steps, making simulations more expensive. Due to self-avoiding interactions, σ has a minimum value of $2a$ before hard-sphere interactions come into play before the potential energy minimum of self-interactions. However, because of the repulsive nature of the self-interactions and hydrodynamically thin nature of the sheets in these simulations, hard-sphere interactions can likely be removed while maintaining the self-avoiding nature of the sheets, allowing for much smaller σ. However, there is the additional restriction that $\sigma \gg a$ so that the interaction energy of a bead moving parallel to a sheet is approximately constant. This gives a fairly small range of σ which can be tested. It is difficult to devise a scheme for which interactions along the sheet are neglected, but such a thing might be necessary to test an appropriate range of σ without drastically decreasing the time step or moving to a constrained system (more on this later as well).

Similarly, K's scaling with L (no predicted dependence) and σ ($K \sim \sigma^{-2}$) could be tested by finding the folded-flat transition with varying L and σ, using similar precautions as those discussed earlier.

6.4 Tumbling time for tumbling and folded sheets

Another interesting aspect of tumbling sheets is their tumbling time, which is roughly the time for a sheet to make half a revolution about the vorticity axis. For 1D polymers, the tumbling time is a complex function of polymer parameters (such as the persistence length, bending rigidity equivalent for 1D polymers) and shear strength[4,5]. For 2D sheets, Silmore *et al.* proposed and confirmed scaling arguments for the flipping time of relatively inflexible sheets[1]. The flipping time in simulations was calculated using peaks in the total bending energy of a sheet. Such a method

might work for folded or flat sheets. Because folded sheets appear to behave like deformable ellipsoids, one potential result is sheets adopting different trajectories according to an effective orbit constant and aspect ratio, as seen by Zhang, Lam, and Graham[6] in deformable "capsules." However, due to their constant deformation, the tumbling time for tumbling sheets is non-trivial to find in simulation or develop scaling arguments for. The former is possible, but difficult. This is, in part, because sheet deformations can cause discontinuous changes in the "orientation" of a sheet, as characterized by, for example, the minimum bounding ellipsoid semi-axes or the eigenvectors of the gyration tensor. It is often much easier to identify a rotation by eye than to identify one in general by looking at sheet properties.

Still, resolving this question would be valuable in answering several questions. For example, how does the tumbling time scale with shear rate in the folded and tumbling regimes? Are these scalings different? Does the tumbling transition represent a continuous or discontinuous change in the tumbling time of sheets? And so on. A study on the flipping dynamics of tumbling and folded sheets was not conducted here, but would be valuable in the future.

6.5 Adding thermal fluctuations

In Chapters 4 and 5, we neglected thermal fluctuations, arguing that thermal energy is often small relative to bending rigidity. This is true, however there are cases, notably graphene oxide, where thermal energy may be comparable to bending rigidity and interactions. In this case, their role might be quite important. For example, how stable are folded conformations when thermal energy is comparable to interaction strength? Thermal fluctuations might, for example, provide the necessary "push" to anneal sheets to their energetically most favorable number of folds. They could also allow the folds in the 2D folded sheets to rearrange to a more favorable, 1D folded conformation.

While we haven't performed such simulations yet due to thermal simulations being significantly more computationally expensive than athermal ones, we can make predictions based on the relative strengths of thermal energy, k_BT, to interaction strength, $\tilde{\epsilon}\sigma^2$, thermal energy to bending rigidity, κ, and bending rigidity to interaction strength. If $k_BT/\tilde{\epsilon}\sigma^2 \ll 1$, then sheets behave as in Chapter

4. This is true because, in the small $K \equiv \kappa/\tilde{\epsilon}\sigma^2$ limit, interactions dominate both thermal energy and bending rigidity. In the large K limit, $k_BT/\kappa \ll 1$ necessarily and thermal energy is negligible compared to both interactions and bending rigidity. In both cases, the sheets behave as in Chapter 4 neglecting some of the potential effects noted above. If $k_BT/\tilde{\epsilon}\sigma^2 \gg 1$, then interactions are negligible and sheets behave as in work by Silmore *et al.*[7] due to similar analysis of the relevant limits. When thermal energy is comparable to bending rigidity and/or interactions, we expect more complex phenomena to arise. We believe all of these limits to be worth exploring.

Even if thermal energy does not significantly affect the overall conformations of sheets, it still may affect the viscosity by causing random fluctuations along the shear axis. These fluctuations would become more strongly depressed with increasing shear rate, likely resulting in the shear-thinning behavior seen by Silmore *et al.*[7].

Finally, in Chapter 5, we annealed sheets to zero shear. In the absense of shear, thermal energy becomes a more relevant parameter which should be considered. Thermal energy much larger than interaction strength would likely break apart folded conformations in the absence of shear[8]. There may be an intermediate level of thermal energy which is insufficient to break apart the folded conformations but strong enough to cause rearrangement of folds, perhaps causing sheets to adopt 1D folded conformations. A study on the effect of this thermal energy would be valuable in the future.

6.6 Sheet size scaling of sheet viscosity

In Figure 4.8, we scale the sheet viscosity to L^3. However, given two additional length scales, σ and a, this is not necessarily the proper scale. Furthermore, the proper scale for the sheet viscosity might be different for flat vs. tumbling vs. folded sheets. We have not confirmed any such scalings here, but we believe it warrants investigation and discuss predicted scalings here.

Because $w_{fold} \sim a$, where a is the equivalent of a molecular or atomistic length scale in a 2D sheet, the number of folds is expected to increase for larger sheets, with $n^*_{fold} \sim (L/a)^{1/2}$, as derived in Appendix G. For a 1D folded sheet, its largest characteristic size scales as L because this dimension is parallel to the folds. Another characteristic size is the width of each folds times

the number of folds: $\sigma n_{fold} \sim \sigma(L/a)^{1/2}$. The last characteristic size is the width of each flat region: $w_{flat} \sim L/n_{fold} \sim a(L/a)^{1/2}$. The sheet viscosity will grow roughly as the cross-sectional area of the sheet in the shear-vorticity plane times the length of the sheet along the shear axis. If the sheet is log-rolling, for example, its cross-sectional area in the shear-vorticity plane goes as $L^{3/2}$ and its length along the shear axis goes as $L^{1/2}$. The sheet viscosity would therefore go as L^2. Orientations of sheets with their largest dimension rotating about the vorticity plane have sheet viscosities which go as $L^{5/2}$. Rolled-up sheets occupy a roughly circular area perpendicular to their largest axis which goes as $L\sigma$, so its other characteristic sizes go as $\sqrt{L\sigma} \sim L^{1/2}$, so its sheet viscosity will have the same scaling as sheets with many folds. Assuming each characteristic size of a tumbling sheet scales with L, the stress scales as L^3, which is stronger than in the 1D folded regime.

If we were to treat a as simply the smallest resolvable length scale in the system, we would expect w_{fold} to grow proportionally to L. In this case, n^*_{fold} is not a function of L, and the three characteristic sizes of the sheet grow as, from largest to smallest, L, L, and σ. In the limit of $L \gg \sigma$, the sheet will preferentially align itself with the flow-vorticity plane, and behave effectively as a flat sheet with a small width. The sheet viscosity, assuming the largest axis is aligned with the vorticity axis, would thus grow as $L\sigma^2$, which is weaker than when a is the atomistic or molecular length scale in the system. This highlights the importance of treating a as an explicit length scale in these systems.

As L increases, χ^{-1} increases, and sheets are pushed toward the tumbling regime. Sheet suspensions are often poly-disperse[9] and the total contribution to the viscosity in a dilute suspension is the sum of the individual sheet viscosities. So, the total contribution to the viscosity is a weighted sum of the sheet viscosity from each particle size's sheet viscosity, each of which is non-monotonic with shear rate.

6.7 Sheets with finite in-plane stiffness

In the sheets simulated in this thesis, the Föppl-von Kármán number was large such that sheets were relatively inextensible relative to bending. However, finite extensibility may be important

for the specific behaviors certain sheets exhibit. For example, a bilayer[b] can bend only if the two layers slide along each other (which, depending on the strength of the interactions keeping them stacked, may be difficult) or if the sheets extend or compress[c]. As another example, finite extensibility allows for sheets to bend in ways which would not otherwise be possible. Therefore, an examination of the effects of finite extensibility on these sheets would be valuable. For example, the freely-jointed constraint used in Chapter 2 could be used to examine a sheet with zero extensibility. Then, the spring constant could be varied to see how sheet behavior changes with extensibility. Such an examination was not performed in this thesis as it represents a large increase in the size of the phase space and, again, in these systems the Föppl-von Kármán number is typically large.

6.8 Energetic map for sheet conformations

We have calculated the energies of each type of conformation and find that folded sheets are $\mathcal{O}(1\tilde{\epsilon}\sigma^2)$ per bead more energetically favorable than flat sheets, with 1D folded sheets being slightly more energetically favorable than 2D folded sheets (by $\mathcal{O}(10^{-1}\tilde{\epsilon}\sigma^2)$ per bead), and rolled-up sheets being slightly more energetically favorable than 1D folded sheets with many folds (by $\mathcal{O}(10^{-2} - 10^{-1}\tilde{\epsilon}\sigma^2)$ per bead). Meanwhile, tumbling sheets have widely varying energies, which can be significantly more or less energetically favorable than flat sheets (by $\mathcal{O}(1\tilde{\epsilon}\sigma^2)$ per bead in either direction).

That said, we have yet to examine nearby conformations to examine the transition between these conformations. By perturbing different conformations, we could generate an energetic map which would give an idea of the energetic barrier required to transition between conformations. This would be useful, for example, in determining the required thermal energy needed to cause phase transitions. Such an analysis would also not require running simulations, only performing, for example, a Monte Carlo simulation of nearby states and doing energetic analysis of the generated states.

[b] That is, two sheets stacked on top of one another.

[c] This is common in mechanics. In a bend in a material with finite thickness, the outer part of the bend must extend while the inner part must compress. Thus, the bending of these materials depends on their extensibility.

6.9 Sheet fracture

One aspect of flow processing of sheets that is typically neglected in models is fracture. By varying the time and strength of processing protocols such as shear and sonication, the size distribution of the resulting sheet suspension can be changed[10]. This is important because the properties of 2D materials depend on their size (for example, note how the dimensionless parameters S and χ depend on L).

In theory, to model sheet fracture, we simply need to add a condition for when the bonds between beads should break. For example, we might break a bond between beads when they achieve a certain separation or separating force. Then, we could run simulations at various values of K, χ, and dimensionless fracture threshold and monitor the number of sheets and their sizes over time, comparing with the distributions observed in real systems.

In practice, sheet fracture causes sheets to become smaller and thus finite-size effects can quickly become relevant. Implementing a sheet fracture model thus requires careful consideration and potentially moderation of these effects. For example, in a model where a represents the smallest resolvable length scale in the system, the bead mesh of fractured sheet sections might need to be refined to avoid finite-size effects, although this might increase the computational cost of the simulation significantly over time.

6.10 Adding more sheets

Another question we could ask is what happens when we add more sheets. We might consider exfoliation of many stacked sheets, for example. In this case, waltzing is unlikely to appear as the larger stack which the exfoliating sheet is peeling from is both thicker and stiffer than the exfoliating sheet. However, there will likely be wrapping of some sort, resulting in potentially interesting structures and behavior. The wrapping sheets might, for example, restrict the exfoliation of future sheets, or exfoliation of future sheets might allow wrapped sheets to separate fully. It is difficult to make any concrete predictions.

Staying within the dilute regime, we could also see how aggregation of sheets could occur. For example, we might consecutively add more sheets to a simulation, observing what sorts of

structures are formed and under what conditions sheets aggregate. This might model a dilute regime where collisions between sheets are rare.

We might also consider more dense suspension of sheets. We could initialize a suspension of sheets, for example, by doing a thermal simulation in the absence of shear and self-interactions, then shear the system and observe the results. We could ask interesting questions about the resulting suspension. For example, do clusters of sheets form? Does the system percolate? How do these higher-order structures affect the rheological properties? How do the answers to these questions change with varying χ^{-1}, K, sheet volume fraction, k_BT/κ, etc.? A logical place to start might be to do no shear simulations to see how the system behaves without it before introducing the extra complexity of shear into the system.

Because the algorithm used in this thesis goes as the number of beads in the simulation squared, we quickly become limited in the number of sheets we can add to a simulation. To remedy this, we would need to use an algorithm which is linear in the number of beads, such as the PSE method discussed in Chapter 1[11,12]. This makes sense for more dense suspensions, where the periodic nature of the RPY tensor used for these methods more accurately captures the desired physics. In our single and two sheet simulations, we were interested in the dilute limit, where the sheets do not interact hydrodynamically with other sheets in the suspension, motivating the non-periodic (unbounded) RPY tensor, which is faster for a relatively small number of beads.

6.11 Conclusions

In this chapter, we presented some potential directions for future work on self-interacting sheets in shear using the model presented in this work. Our priority in exploring them depends on our desired application.

Exploring the effects of varying θ is desirable for improving the results from Chapter 4. However, it adds an additional parameter to the system which drastically expands the design space for simulations and we expect the effects to only be minor.

Confirming the length scalings of the new dimensionless parameters would be useful not only in translating this work to sheet simulations of different sizes but also in predicting their values

(for example, the tumbling/folded transition in χ^{-1}) for real materials. This was not done in this thesis due to the limitations discussed in that section, however these can be overcome given time for longer simulations.

Analysis of the tumbling time for tumbling sheets would be useful first from a fundamental level. Better understanding of how sheets tumble would help us better understand their conformational and rotational properties, again ultimately leading to better understanding of their rheological properties. This was not done in this thesis due to the computational difficulty of identifying a flip, although we believe this can be overcome.

The addition of thermal fluctuations would be useful not only in understanding the effects of small thermal fluctuations to the system, but also very practically for certain real materials such as graphene oxide, with bending rigidities which are comparable to thermal energy at room temperature. However, this was also not implemented in this thesis due to time limitations, as calculating thermal fluctuations are the most expensive part of these simulations.

Proving the sheet size scaling of sheet viscosity would help translate our results to real systems as well as allow us to better understand the role of, for example, polydispersity in the observed rheological behavior of sheets. This was not done in this thesis due to the same considerations as discussed for the confirmation of size scalings for the dimensionless groups.

While many 2D materials have large Föppl-von Kármán numbers, and are thus relatively inextensible, understanding the effects of finite in-plane thickness would be useful for extending this work to new materials with finite extensibilities as well as for stacks of sheets whose behaviors are dependent on small but finite extensions. However, this also adds a new parameter to the design space which makes it tricky to study in practice.

Creating an energetic map of sheet conformations would help our fundamental understanding of how sheets transition between different conformations. Studying sheet fracture would be important for production applications, where the sheet size distribution changes most quickly over time. Learning how this process occurs could help design flow protocols for producing sheets with the desired size distribution. Furthermore, such as study would not require simulating dynamics, making it relatively cheap to study compared to the other considerations listed here.

Finally, adding more sheets would be helpful for studying non-dilute suspensions, which are common in many real applications. The main limitation here is, again, time, as each additional sheet adds to the computational burden of a single simulation. In this thesis, we focused on the behavior of single sheet and two sheet systems to create the building blocks for these more complex systems.

Overall, there is still much to learn about self-interacting sheets which can be illuminated using this model. For fundamental understanding, the effects of varying θ, confirmation of size scalings for dimensionless

Bibliography

[1] KS Silmore, MS Strano, and JW Swan. Buckling, crumpling, and tumbling of semiflexible sheets in simple shear flow. *Soft Matter*, 17(18):4707–4718, 2021.

[2] SB Babu and H Stark. Dynamics of semi-flexible tethered sheets: A simulation study using stochastic rotation dynamics. *The European Physical Journal E*, 34:1–7, 2011.

[3] D Yllanes, SS Bhabesh, DR Nelson, and MJ Bowick. Thermal crumpling of perforated two-dimensional sheets. *Nature Communications*, 8(1):1–8, 2017.

[4] M Harasim, B Wunderlich, O Peleg, M Kröger, and AR Bausch. Direct observation of the dynamics of semiflexible polymers in shear flow. *Physical Review Letters*, 110(10):108302, 2013.

[5] PS Lang, B Obermayer, and E Frey. Dynamics of a semiflexible polymer or polymer ring in shear flow. *Physical Review E*, 89(2):022606, 2014.

[6] X Zhang, WA Lam, and MD Graham. Dynamics of deformable straight and curved prolate capsules in simple shear flow. *Physical Review Fluids*, 4(4):043103, 2019.

[7] KS Silmore, MS Strano, and JW Swan. Thermally fluctuating, semiflexible sheets in simple shear flow. *Soft Matter*, 18(4):768–782, 2022.

[8] FF Abraham and M Kardar. Folding and unbinding transitions in tethered membranes. *Science*, 252(5004):419–422, 1991.

[9] J Amaro-Gahete, A Benítez, R Otero, et al. A comparative study of particle size distribution of graphene nanosheets synthesized by an ultrasound-assisted method. *Nanomaterials*, 9(2):152, 2019.

[10] C Cai, N Sang, Z Shen, and X Zhao. Facile and size-controllable preparation of graphene oxide nanosheets using high shear method and ultrasonic method. *Journal of Experimental Nanoscience*, 12(1):247–262, 2017.

[11] AM Fiore, F Balboa Usabiaga, A Donev, and JW Swan. Rapid sampling of stochastic displacements in brownian dynamics simulations. *The Journal of Chemical Physics*, 146(12), 2017.

[12] AM Fiore and JW Swan. Rapid sampling of stochastic displacements in brownian dynamics simulations with stresslet constraints. *The Journal of Chemical Physics*, 148(4), 2018.

Chapter 7

Concluding remarks

In this thesis, we examined examined methods for soft matter simulations and applied simulation methods to further the fundamental understanding of 2D materials. In Chapter 1, we gave a brief introduction to Brownian dynamics simulations with hydrodynamics. Due to the size of real soft matter systems, which cause these systems to have low Reynolds numbers and large Schmidt numbers, we are able to make simplifying assumptions which allow these simulations to be run quickly, despite long-range hydrodynamics which couple the motion of all particles in the system. We derived the equations of motion for the system, identifying complications and simplifications as they arose. Finally, we discussed how simulations are done in log-linear time, common conservative potentials in soft matter models, and how more degrees of freedom could be included in our particles.

In Chapter 2, we introduced constrained Brownian dynamics simulations, explaining how the equations of motion from Chapter 1 must be modified, discussing why constrained systems are more difficult to implement than unconstrained systems, introducing the current state-of-the-art methods for simulating these systems (GMRES), and deriving the time-complexity for implementing different pieces of these simulations. We then introduced PrCG, a different method for solving constrained systems which is simpler to implement than GMRES for new or mixed constraints and carries certain advantages such as guaranteed feasibility of iterates. We showed in our results that PrCG and GMRES carry the same computational complexity and similar asymptotic storage costs on a variety of constraint types. Finally, we compared PrCG to the analytical solution of

a mixed constraint system, showing that the numerical and analytical solutions agreed. Overall, we showed in this chapter that PrCG is a competitive method which might be advantageous over GMRES in constrained Brownian dynamics simulations for certain systems.

In Chapter 3, we introduced the necessary background on rheology necessary to understand the remainder of the thesis. We discussed non-Newtonian fluids and their applications, simple shear, the necessary modifications to the equations of motion, as well as how these changes are implemented in simulations. As a part of this, we introduced viscosity and the stress tensor, how they are connected, and why we care about developing models for them. We then moved our focus to 2D materials, which were the focus of the remainder of the thesis. We introduced 2D materials, their applications, our current theoretical understanding of their behavior, and their rheological properties.

In Chapter 4, we connected the pieces from the previous chapter, presenting a model for athermal, semi-flexible, self-interacting sheets in simple shear flow. Using this model, we discovered a rich conformational landscape, showing flat, tumbling, 1D folded, and 2D folded behavior. We derived a new dimensionless parameter balancing the strength of shear to interaction strength, using this parameter as well as the ratio between bending rigidity and interaction strength to create a phase map of these conformations. We calculated a proxy for the sheet viscosity and showed interesting non-monotonic rheological behavior that matches with experimental observations. We used the conformational and rotational behavior of the sheets to explain these rheological properties, demonstrating that the shear-thinning behavior we observed was a result of averaging over many initial conditions. This chapter gave a new explanation for how these rheological behaviors can arise even in the dilute limit and in the absence of thermal energy. Then, we used the same model to study the shear exfoliation of two sheets. We showed that sheets exhibit separating, waltzing, or flipping depending on the previously determined dimensionless groups. We connected these behaviors again to the rheological behavior of the dilute suspension, and explained the behavior using the observed conformations and the insights from our single sheet simulations.

In Chapter 5, we examined the initial condition dependence of our sheet systems by implementing different shear protocols to our system in an attempt to control the final properties of the

suspension. Inspired by colloidal suspensions, we implemented a linear shear annealing protocol and showed that the relative shape anisotropy of sheets increases with quench time. Next, we utilized the features of the phase map from Chapter 4 to design protocols with step changes in shear, showing that sheets 1D folded sheets with high alignments can be produced quite reliably from a variety of initial conditions. Using only simple flow protocols, we showed that the conformational and rotational behaviors of a dilute suspension of sheets can be tuned. The results of this chapter are promising for showing how the properties of suspensions of 2D materials can be controlled.

Finally, in Chapter 6, we discussed a few potential future considerations for self-interacting 2D materials in shear. Specifically, we discussed loose ends from Chapter 4: analyzing the effect of varying θ, confirming the scaling of χ^{-1}, calculating the tumbling time for different sheet behaviors, adding thermal fluctuations, finding the scalings of sheet viscosity for different sheet behaviors, examining the effects of finite in-plane stiffness, and generating an energetic map for sheet conformations. Then, we discussed potential extensions to the model: sheet fracture and dense suspensions of sheets. We discussed potential complications for studying these, usually involving larger or more numerous simulations, which caused us to prioritize the work presented in this thesis. For applicability to real systems, we believe confirmation of the size scalings of χ^{-1} and the sheet viscosity are particularly important. Sheet fracture and dense sheet suspensions, on the other hand, are two interesting new directions this work could go after these scalings are confirmed.

This model, just like 2D materials as a whole, is still in its infancy. As time goes on and methods for synthesizing new 2D materials develop, I believe that this and similar works as well as future work based on these models will serve as important reference for understanding, explaining, and ultimately controlling the properties of these materials to be optimal for a specific application. I look forward to future scientific discoveries in this field, which I hope that the work I have done here will aid.

Appendices

Appendix A

Notes on notation

In this thesis, non-bold symbols refer to scalar values, while bold symbols will refer to vector, matrix, or tensor values. Non-bold symbols for otherwise vector values indicate either the norm of the vector (e.g. $u = \|\mathbf{u}\|$) or a component of the vector, depending on the context (unless otherwise specified, such as the particle acceleration, $\mathbf{a}$, and particle radius a). Vectors are column vectors. The transpose of a matrix, $\mathbf{x}$, is $\mathbf{x}^T$. The inverse of a matrix, $\mathbf{x}$, is $\mathbf{x}^{-1}$. $\boldsymbol{\nabla}$ is the gradient operator and ∇^2 is the Laplacian operator. $\cdot$ is the dot product.

A.1 Nomenclature

Variables will be defined when first introduced but recurring ones will be included here as well for reference.

a - particle radius (hydrodynamic or hard-sphere)

D - diffusivity

$\mathbf{E}$ - strain rate tensor

ϵ - error, error tolerance, or interaction strength

$\tilde{\epsilon}$ - per-area interaction energy density

η - fluid viscosity or critical exponent

$\mathbf{F}_B$ - Brownian force

$\mathbf{F}_H$ - hydrodynamic force

$\mathbf{F}_P$ - potential force

$\mathbf{f}_c$ - constraint force

ϕ - particle volume fraction or sheet rotation about the vorticity axis

$\mathbf{G}$ - gradient of constraint equations

$\dot{\gamma}$ - shear rate

$\mathbf{I}$ - identity matrix

K - dimensionless parameter characterizing the strength of bending rigidity to interactions

k - spring constant or magnitude of reciprocal lattice vector

κ - bending rigidity

k_BT - thermal energy

$\mathbf{L}$ - velocity gradient tensor

L - characteristic length or circumradius of a sheet

λ - Lagrange multipliers ("constrained velocities") or the square root of an eigenvalue of the gyration tensor

$\bar{\lambda}$ - average of the square root of an eigenvalue of the gyration tensor

$\boldsymbol{\mathcal{M}}$ - mobility tensor

N - number of particles in a simulation, for example the number of beads in a sheet

$\mathcal{O}$ - "order of"

$\mathbf{P}$ - preconditioner matrix

p - pressure

$\mathbf{q}$ - constraint equations

ρ - fluid density

$\mathbf{Q}$ - orthonormal matrix in QR decomposition

$\mathbf{R}$ - upper-triangular matrix in QR decomposition

R_g - radius of gyration

$\hat{\mathbf{r}}$ - unit vector pointing between particles

r - distance between particles

$\boldsymbol{\mathcal{R}}$ - resistance tensor

Re - Reynolds number

S - dimensionless parameter characterizing the strength of bending rigidity to shear

Sc - Schmidt number

$\mathbf{\Sigma}$ - constraint matrix

$\tilde{\mathbf{\Sigma}}$ - stresslet

σ - interaction range

t - time

Δt - time step

θ - sheet rotation about the flow axis

U - potential energy

$\mathbf{u}$ - particle velocity

$\mathbf{u}_B$ - Brownian velocity

V - potential energy

$\mathbf{v}_1$ - the unit vector pointing in the direction of the largest semi-axis of the minimum bounding ellipsoid of a sheet or the unit vector pointing along the largest eigenvector of the gyration tensor of a sheet

$\mathbf{x}$ - particle position (or arbitrary vector)

Δx - particle displacement

χ - dimensionless parameter characterizing the strength of interactions to shear

$\mathbf{Z}$ - projection operator

$\hat{z}$ - unit vector pointing in the z-direction (the vorticity axis in shear simulations)

ζ^2 - relative shape anisotropy

Appendix B

Analytical orthonormalization of the constraints on rigid bodies

We can calculate $\mathbf{Q}$ for rigid bodies analytically using Gram-Schmidt methods. Gram Schmidt (GS) and Modified Gram Schmidt (MGS) algorithms for calculating the QR decomposition of a matrix are typically disfavored due to the increasing loss of the orthogonality of the columns at each iteration from machine precision rounding errors, which increase with the condition number of the matrix being factored[1,2]. However, this disadvantage can be mitigated for the rigid body system. Specifically, the loss in orthogonality in the GS and MGS algorithms generally only occurs after many iterations, even for a matrix with a large condition number[3]. Thus, the block diagonal structure of the constraint matrix can be exploited to decrease the loss in orthogonality of the factorization. Each diagonal block has at most six columns, and the three columns associated with translational motion are orthogonal by construction. The block QR decomposition of $\mathbf{\Sigma}$ can be expressed as:

$$\mathbf{\Sigma} = \begin{bmatrix} \mathbf{\Sigma}_1 & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{\Sigma}_2 & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{\Sigma}_{N_{\text{bodies}}} \end{bmatrix} = \mathbf{QR} = \begin{bmatrix} \mathbf{Q}_1 & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}_2 & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{Q}_{N_{\text{bodies}}} \end{bmatrix} \begin{bmatrix} \mathbf{R}_1 & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_2 & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{R}_{N_{\text{bodies}}} \end{bmatrix}, \quad \text{(B.1)}$$

where N_{bodies} is the number of bodies with $\mathbf{\Sigma}_i = \mathbf{Q}_i \mathbf{R}_i$.

By performing the QR decomposition of each block individually, at most 6 columns are orthogonalized at a time by the GS algorithm, and thus loss of orthogonality due to numerical errors is small. The computational cost of this algorithm is also greatly reduced compared to algorithms which do not exploit the block diagonal structure of the constraint matrix. Specifically, each block of the constraint matrix requires $\mathcal{O}(B)$ operations to decompose. This means the overall decomposition for all of the N_{bodies} bodies requires $\mathcal{O}(N)$ time. Because this decomposition is linear in the number of beads, it presents no special burdens computationally. This decomposition is also easy to parallelize, as each block associated with a different rigid body can be orthonormalized independently. However, we do not implement such a parallelization in this work.

Furthermore, the columns associated with translational motion in each block are already orthogonal to all of the other columns, and the columns associated with angular motion are relatively simple. This combination of properties allows for a simple, explicit expression for the GS QR decomposed constraint matrix. This orthonormal matrix depends conveniently on the gyration tensor for the positions of the beads within a body:

$$\boldsymbol{S}_i = \frac{1}{B_i} \sum_{\alpha=1}^{B_i} \left(\mathbf{x}_{i,\alpha} - \bar{\mathbf{x}}_i\right)^T \left(\mathbf{x}_{i,\alpha} - \bar{\mathbf{x}}_i\right), \tag{B.2}$$

where the sum is done across the B_i beads in the body labeled i. The block of the constraint matrix associated with this body can be written as a set of column vectors: $\mathbf{\Sigma}_i = \left[\mathbf{e}^t_{i,1}\, \mathbf{e}^t_{i,2}\, \mathbf{e}^t_{i,3}\, \mathbf{e}^r_{i,1}\, \mathbf{e}^r_{i,2}\, \mathbf{e}^r_{i,3}\right]$, with the superscript t corresponding to the translational and the superscript r corresponding to the rotational components. We start by considering the case of a non-colinear body (with 6 degrees of freedom). Then, the orthonormalization of columns in $\mathbf{\Sigma}_i$ from a GS QR decomposition of that block can be written as: $\mathbf{Q}_i = B_i^{-1/2} \left[\mathbf{e}^t_{i,1}\, \mathbf{e}^t_{i,2}\, \mathbf{e}^t_{i,3}\, \hat{\mathbf{e}}^r_{i,1}\, \hat{\mathbf{e}}^r_{i,2}\, \hat{\mathbf{e}}^r_{i,3}\right]$. The orthonormalized columns associated with rotational motion of the rigid body are

$$\hat{\mathbf{e}}^r_{i,1} = \frac{\mathbf{e}^r_{i,1}}{\sqrt{w_1(\mathbf{S}_i)}} \tag{B.3a}$$

$$\hat{\mathbf{e}}_{i,2}^{r} = \frac{\mathbf{e}_{i,2}^{r} + f_1(\mathbf{S}_i)\mathbf{e}_{i,1}^{r}}{\sqrt{w_2(\mathbf{S}_i)}} \tag{B.3b}$$

$$\hat{\mathbf{e}}_{\mathbf{i,3}}^{\mathbf{r}} = \frac{\mathbf{e}_{i,3}^{r} + f_2(\mathbf{S}_i)\mathbf{e}_{i,1}^{r} + f_3(\mathbf{S}_i)\mathbf{e}_{i,2}^{r}}{\sqrt{w_3(\mathbf{S}_i)}}, \tag{B.3c}$$

with

$$\mathbf{f}(\mathbf{S}) = \begin{bmatrix} \frac{S_{12}}{w_1(\mathbf{S})} \\ \frac{S_{13}}{w_1(\mathbf{S})} + \frac{S_{12}S_{23}}{w_1(\mathbf{S})w_2(\mathbf{S})} + \frac{S_{12}^2 S_{13}}{w_1(\mathbf{S})^2 w_2(\mathbf{S})} \\ \frac{S_{23}}{w_2(\mathbf{S})} + \frac{S_{12}S_{13}}{w_1(\mathbf{S})w_2(\mathbf{S})} \end{bmatrix} \tag{B.4}$$

and

$$\mathbf{w}(\boldsymbol{S}) = \begin{bmatrix} S_{22} + S_{33} \\ S_{11} + f_1(\mathbf{S})^2 S_{22} + (1 + f_1(\mathbf{S})^2) S_{33} - 2c f_1(\mathbf{S}) S_{12} \\ (1 + f_3(\mathbf{S})^2) S_{11} + (1 + f_2(\mathbf{S})^2) S_{22} + (f_2(\mathbf{S})^2 + f_3(\mathbf{S})^2) S_{33} \\ \qquad - 2(f_2(\mathbf{S}) f_3(\mathbf{S}) S_{12} + f_3(\mathbf{S}) S_{23} + f_2(\mathbf{S}) S_{13}) \end{bmatrix}. \tag{B.5}$$

This explicit formulation is easily parallelizable as each row in a body's orthonormal block, $\mathbf{Q}_i$, depends only on the gyration tensor of that body and the corresponding row in $\mathbf{\Sigma}_i$. Computation and storage are both $\mathcal{O}(N)$. The upper-triangular matrix may also be calculated easily. Each 6×6 block along the diagonal corresponds to a single body, with

$$\mathbf{R}_i(\mathbf{S}) = B_i^{1/2} \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{S_{22}+S_{33}}{\sqrt{w_1(\mathbf{S})}} & -\frac{S_{12}}{\sqrt{w_1(\mathbf{S})}} & -\frac{S_{13}}{\sqrt{w_1(\mathbf{S})}} \\ 0 & 0 & 0 & 0 & \frac{S_{11}+S_{33}-f_1(\mathbf{S})S_{12}}{\sqrt{w_2(\mathbf{S})}} & -\frac{S_{23}+f_1(\mathbf{S})S_{13}}{\sqrt{w_2(\mathbf{S})}} \\ 0 & 0 & 0 & 0 & 0 & \frac{S_{11}+S_{22}-f_2(\mathbf{S})S_{13}-f_3(\mathbf{S})S_{23}}{\sqrt{w_3(\mathbf{S})}} \end{bmatrix}. \tag{B.6}$$

The overall matrix is $6N_{\text{bodies}} \times 6N_{\text{bodies}}$. The inverse of this matrix may be calculated quickly using back substitution because each rigid body corresponds to only a 6×6 block, the first 3 columns of which are trivial. Applying this matrix inverse recovers the translational and angular velocities for rigid bodies, requiring only $\mathcal{O}(N_{\text{bodies}})$ operations.

For beads arranged on a line, such as dimers, there are only 2 rotational degrees of freedom

associated with rigid body motions that change the bead positions. In this case, the last three columns of $\mathbf{\Sigma}$ are linearly dependent. More care must be taken to reduce the dimensionality of the rigid body motions by projecting out rotations along the lineal axis of each body. This will happen naturally during the GS process. One of the columns of $\mathbf{Q}$ will be zero and the QR decomposition can be treated as rank revealing by retaining only those columns and elements of $\mathbf{R}$ needed to build a full rank $\mathbf{\Sigma}$. This can also be done prior to the decomposition by calculating the relative shape anisotropy from the gyration tensor. Unconstrained beads can also be considered rigid bodies which have no rotational degrees of freedom. In this other degenerate case, the $\mathbf{H}$ matrices composing $\mathbf{\Sigma}$ are all zero, as is the gyration tensor. Simply neglecting those columns of $\mathbf{\Sigma}$ corresponding to rotation and including only the translational components of $\mathbf{Q}$ and $\mathbf{R}$ resolves the degeneracy. In this case, the solution for the constraint forces acting on individual beads is trivial, as those forces are necessarily zero. However, these beads still contribute to hydrodynamic interactions, so their translational components in $\mathbf{Q}$ must be included when constraints are present in the system.

Appendix C

Constraint force and constrained velocity errors at each iteration

Here we include constraint force and constrained velocity errors for each algorithm as they iterate for rigid bodies (Figure C.1), freely jointed chains (Figure C.2), gyroid immobile (Figure C.3), and uniform immobile (Figure C.4) systems.

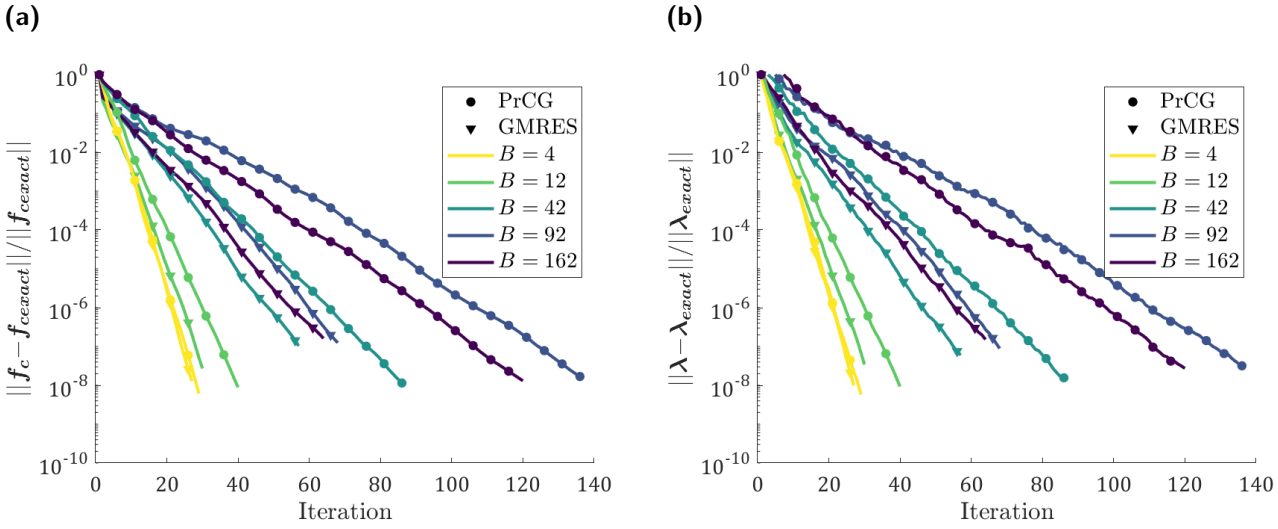

Figure C.1 Norm of (a) constraint force and (b) constrained velocity errors relative to the error at the initial guess, for geodesic polyhedra with varying beads per body, $B = 4, 12, 42, 92, 162$, with constant number of bodies, $N_{\text{bodies}} = 6^3$. Circles correspond to PrCG while triangles correspond to GMRES. Only every 5th iteration is plotted in the scatter for clarity. The system size is denoted by color.

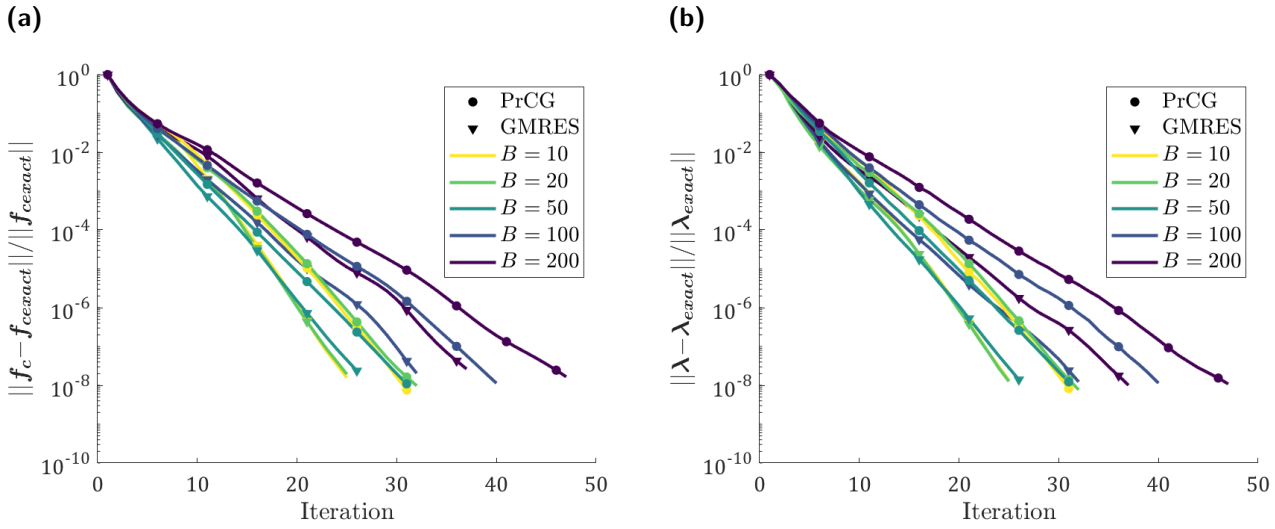

Figure C.2 Norm of (a) constraint force and (b) constrained velocity errors relative to the error at an initial guess of all zeros, for freely jointed chains with varying beads per body, $B = 10, 20, 50, 100, 200$, with constant number of bodies, $N_{\text{bodies}} = 6^3$. Circles correspond to PrCG while triangles correspond to GMRES. Only every 5th iteration is plotted in the scatter for clarity. The system size is denoted by color.

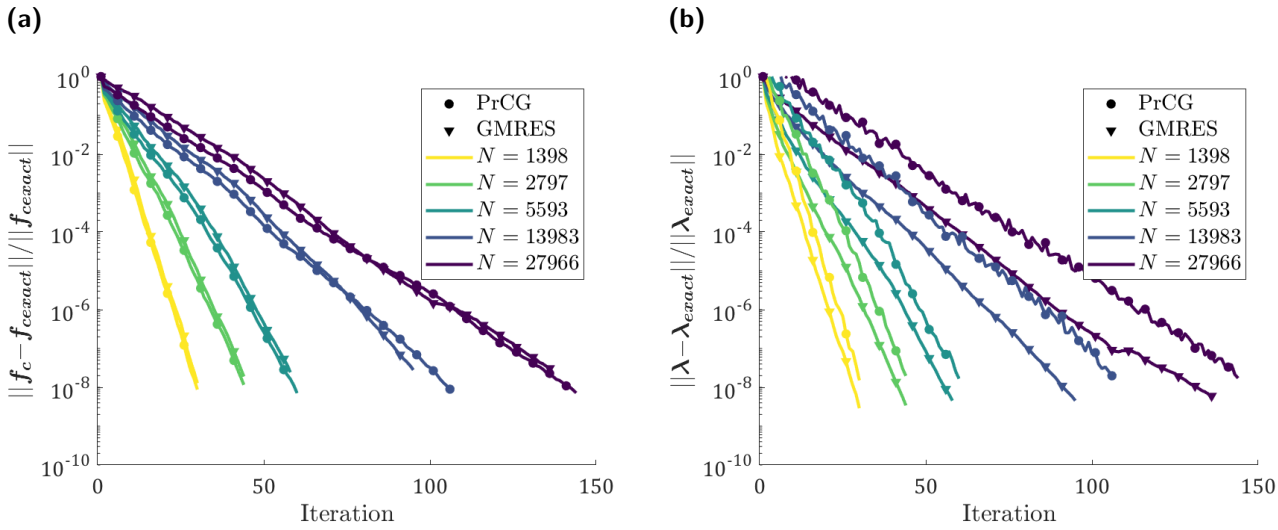

Figure C.3 Norm of (a) constraint force and (b) constrained velocity errors relative to the error at an initial guess of all zeros, for gyroid immobile systems with varying number of beads, $N = 1398, 2797, 5593, 13983, 27966$, with half of the beads being immobile. Circles correspond to PrCG while triangles correspond to GMRES. Only every 5th iteration is plotted in the scatter for clarity. The system size is denoted by color.

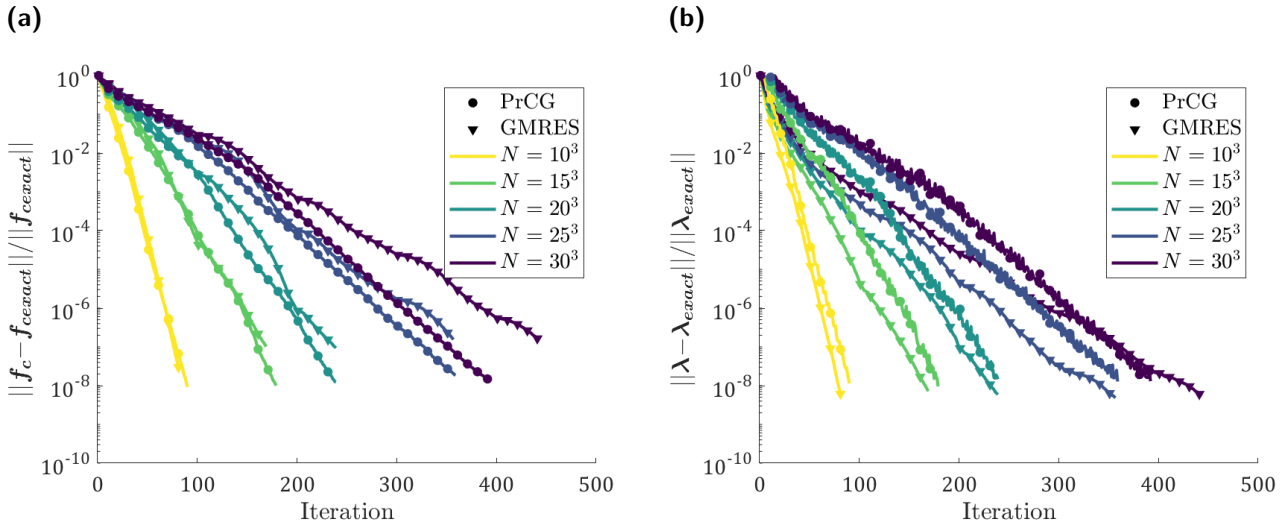

Figure C.4 Norm of (a) constraint force and (b) constrained velocity errors relative to the error at an initial guess of all zeros, for uniform immobile systems with varying number of beads, $N = 10^3, 15^3, 20^3, 25^3, 30^3$, with half of the beads being immobile. Circles correspond to PrCG while triangles correspond to GMRES. Only every 10th iteration is plotted in the scatter for clarity. The system size is denoted by color.

Appendix D

Neglecting lubrication forces

As parallel sections of sheet approach each other, a lubrication force is generated. Consider two parallel sections of sheet of area, A, approaching due to shear and initially separated by a distance, 2σ. The strongest lubrication forces will be generated when the characteristic length of these sheet sections is L. The lubrication force goes as the lubrication pressure, p_{lub}, times the area of the sheet sections:

$$F_{\text{lub}} \sim p_{\text{lub}} A. \tag{D.1}$$

The scaling for p_{lub} can be obtained from the lubrication equation:

$$\frac{\partial p_{\text{lub}}}{\partial x} = \eta \frac{\partial^2 u}{\partial z^2}. \tag{D.2}$$

The x direction goes laterally along the sheets and the z direction goes perpendicularly from the sheets. Therefore, $dx \sim L$ and $dz \sim \sigma$. u is the lateral velocity. The maximum relative velocity of the two sheet sections due to shear is $\sigma\dot{\gamma}$. The lateral velocity, to satisfy the continuity equation, thus goes as $u \sim L\dot{\gamma}$. Together,

$$F_{\text{lub}} \sim \frac{A\eta\dot{\gamma}L^2}{\sigma^2}. \tag{D.3}$$

The energetic benefit of bringing a bead from non-interacting to interacting with a neighboring

sheet section goes as $\tilde{\epsilon}\sigma^2$. The distance it must travel to go from non-interacting to interacting goes as σ. In bringing two parallel sheet sections together, the number of beads which become interacting with the neighboring sheet section goes as A/a^2. Therefore, the force of interaction goes as

$$F_{\text{int}} \sim \tilde{\epsilon}\sigma^2 \frac{1}{\sigma} \frac{A}{a^2}. \tag{D.4}$$

Taking the ratio between the lubrication and interaction forces gives their relative strength:

$$\frac{F_{\text{lub}}}{F_{\text{int}}} \sim \frac{\eta\dot{\gamma}L^2a^2}{\tilde{\epsilon}\sigma^3}. \tag{D.5}$$

Our simulations, which neglect lubrication forces, are valid when this ratio is much less than unity. For example, graphene has inter-atom separation, $a \sim \mathcal{O}\left(1 \text{ Å}\right)$, and athermal Lennard Jones interaction strength, $\tilde{\epsilon} \sim \mathcal{O}\left(0.1 \text{ eVÅ}\right)$ with interaction range $\sigma \sim \mathcal{O}(1 \text{ nm})$[4]. We use the above approximation for the lubrication force even though the continuum approximation breaks down at these length scales. This means that even at high shear rates, $\dot{\gamma} \sim \mathcal{O}(10^3 \text{ s}^{-1})$, lubrication forces are small for graphene sheets in water as long as the sheet size, $L \ll 10$ μm. This is reasonable for the $\mathcal{O}(\mu\text{m})$ flakes produced by exfoliation techniques[5–7] and becomes better for smaller shear rates or larger interaction ranges. Furthermore, the repulsive nature of the Lennard-Jones interaction at small distances restricts sheet segments from getting too close, similarly to lubrication forces. For purely attractive potentials, lubrication forces could fill a similar role in restricting the distance between neighboring sheet segments. Thus, we believe our decision to neglect lubrication forces is valid for real systems.

Appendix E

Derivation of χ

Consider 2 parallel sheets of characteristic size L interacting *via* a short-ranged potential of range σ and separated by their equilibrium distance, σ. If the sheets are sheared such that the flow-vorticity plane cuts between them, the shear force trying to separate the sheets can be approximated as

$$F_{\text{shear}} = 6\pi\eta a\sigma\dot{\gamma}N_{\text{beads}}, \tag{E.1}$$

where $\sigma\dot{\gamma}$ is the relative velocity induced on the sheets due to shear ($6\pi\eta a\sigma\dot{\gamma}$ is the Stokes' drag on a sphere) and $N_{\text{beads}} \sim (L/a)^2$ is the number of beads in each sheet.

We now consider the force required to slide the sheets and break the short-ranged interactions between them. Assuming the sheet is large ($L \gg a$) and the separation is large ($\sigma \gg a$), the energy of the beads in the bulk of the sheet does not change as the sheets slide relative to each other – only beads which are at the leading and trailing edges of the slide matter. The number of beads which separate as a result of a slide of distance σ can be approximated as

$$\Delta N_{\text{slide}} \sim \frac{L\sigma}{a^2}. \tag{E.2}$$

The change in energy for each of these beads is approximately

$$\Delta E_{\text{bead}} \sim \tilde{\epsilon}\sigma^2. \tag{E.3}$$

Thus, the force required to separate the beads is approximately

$$F_{\text{slide}} \sim \frac{\Delta N_{\text{beads}} \Delta E_{\text{bead}}}{\sigma} = \tilde{\epsilon}\sigma^2 \left(\frac{L\sigma}{a^2}\right)\left(\frac{1}{\sigma}\right). \tag{E.4}$$

Taking the ratio of these two forces gives the dimensionless parameter we desire to an order 1 geometry-, orientation-, and packing-specific constant:

$$\chi \equiv \frac{F_{\text{slide}}}{F_{\text{shear}}} \sim \frac{\tilde{\epsilon}\sigma^2}{6\pi\eta\dot{\gamma}L^2\sigma}\left(\frac{L}{a}\right). \tag{E.5}$$

Appendix F

Gyration tensor eigenvalue histograms

The ij'th component of the gyration tensor, $\mathbf{S}$, is given by

$$S_{ij} = \frac{1}{N}\sum_{k=1}^{N}(r_i^{(k)} - \bar{r}_i)(r_j^{(k)} - \bar{r}_j), \tag{F.1}$$

where N is the number of particles in the sheet, $r_i^{(k)}$ is the i'th component of the position of particle k, and $\bar{r}_i$ is the i'th component of the center of mass of the particles. The square root of the eigenvalues of $\mathbf{S}$, λ_i, give a measure of the characteristic sizes of the sheet, while their standard deviation gives a measure of how much these sizes fluctuate, and thus how much the sheet deforms over time. We can use these values to distinguish between different sheet conformations. Please see the main article for details on the criteria used.

For a given simulation snapshot, we define $\lambda_1 > \lambda_2 > \lambda_3$. For each simulation, we define $\bar{\lambda}_i$ to be the average of λ_i and std$[\lambda_i]$ to be the standard deviation of λ_i, both over the last $250\dot{\gamma}t$ of each simulation, with samples taken every $0.25\dot{\gamma}t$. These quantities were used to identify sheet conformations. We include histograms of λ_i and std$[\lambda_i]$ for all simulations in Chapter 4 below.

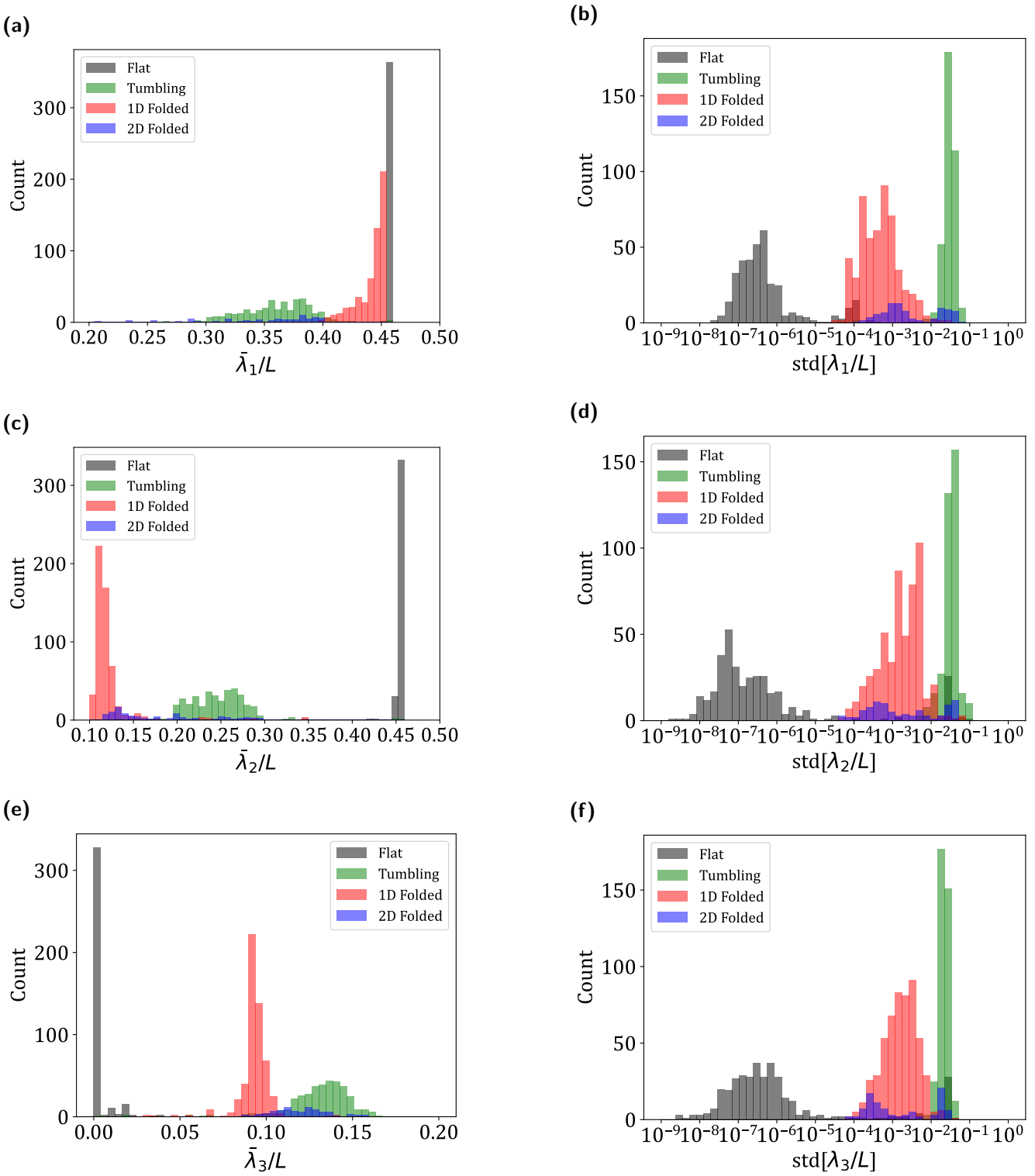

Figure F.1 Histograms of (a, c, e) the average square root of the eigenvalues of the gyration tensor and (b, d, f) the standard deviation of the square root of the eigenvalues of the gyration tensor, both over the last $250\dot{\gamma}t$ of each simulation. (a-b) The largest eigenvalue; (c-d) the second largest eigenvalue; and (e-f) the smallest eigenvalue.

Appendix G

Derivation and confirmation of scaling of optimal number of folds

We wish to obtain an approximation for the optimal number of folds in a 1D folded sheet. We make the following simplifying assumptions. First, the folds are parallel and equally spaced across the length of the sheet, with regions of flat sheet between them. Second, only neighboring parallel sections of sheet will interact with each other (i.e. the interaction is short-ranged). Third, the sheet is rectangular with characteristic length $2L$ and characteristic width $2W$. Fourth, that there is no stretching or compression of the sheet (i.e. FvK $\gg 1$). We define n_{fold} to be the number of folds in the sheet, w_{fold} to be the width of a fold, and w_{flat} to be the width of a flat region. With these assumptions, we obtain the following balance for the length of the sheet:

$$2L = n_{fold} w_{fold} + (n_{fold} + 1) w_{flat}. \tag{G.1}$$

The energy of bending roughly goes as the number of triangles which are within folds times the bending rigidity,

$$E_{bending} \sim \kappa n_{fold} \left(\frac{W w_{fold}}{2a^2} \right). \tag{G.2}$$

The energy of interaction goes as

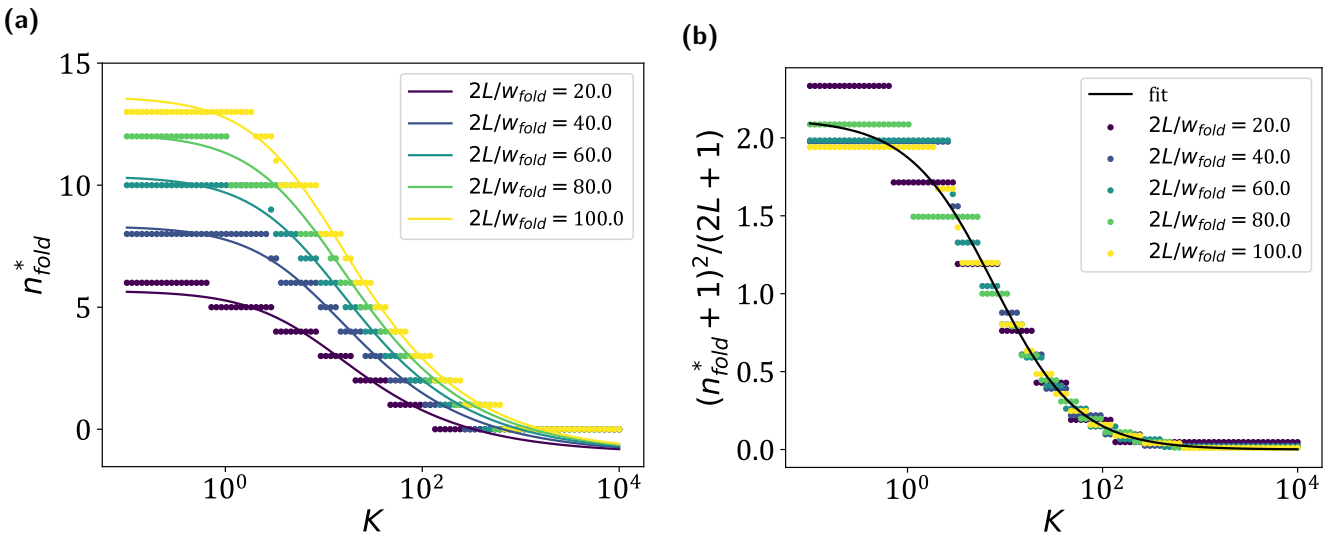

Figure G.1 (a) Optimal number of folds for varying K and L/w_{fold}. Points are data points and solid lines are Equation G.5. Optimal values of α and β were found to be $\alpha = 0.0618 \pm 0.0010$ and $\beta = 0.528 \pm 0.003$ ($\pm$ one standard deviation) using least-squares regression with $R^2 = 0.989$. (b) Master curve using Equation G.5 to collapse all plots of n^*_{fold}.

$$E_{interaction} \sim \tilde{\epsilon}\sigma^2 n_{fold}\left(\frac{W w_{flat}}{a^2} + \beta' \frac{W w_{fold}}{2a^2}\right), \tag{G.3}$$

where the first term sums interactions for beads between parallel regions and the second term sums interactions for beads within folds. The parameter β' is a fold geometry-specific parameter added as a measure of the strength of interactions for beads within folds. We wish to minimize the quantity

$$E = \alpha' E_{bending} - E_{interaction} \tag{G.4}$$

with respect to n_{fold} to obtain the optimal number of folds, n^*_{fold}. The term α' is a fold geometry-specific parameter added as a measure of the relative strength of bending rigidity to interaction strength. Doing so, we obtain

$$n^*_{fold} \approx \sqrt{\frac{2L/w_{fold} + 1}{1 - \beta + \alpha K}} - 1, \tag{G.5}$$

where $\alpha = \alpha'/2$ and $\beta = \beta'/2$ have been redefined for convenience. We note that the the charac-

teristic width of the sheet, $2W$, disappears from the final equation as all interactions are linear in this term. For non-rectangular sheets which are wide enough such that edge effects are irrelevant (i.e. most beads are interior beads), the changing width affects the energy because folds and flat regions can be in locations with different widths. This effect is small for slowly-changing widths and thin folds and/or thin flat regions.

We confirm this scaling by generating rectangular sheets (such that each interior bead of these sheets has 6 neighbors) and finding their optimal number of folds. All neighboring beads in these sheets, even the ones in folds, are a constant distance, $2a$, apart, and neighboring parallel regions of sheet are a constant distance, σ, apart. We choose $w_{fold} = 6a$ and $\sigma = 4\sqrt{6}a/3$ for consistency with our simulations. A folded region consists of 1 row of beads within a flat region, 1 row at the "crease" of the fold, and 1 row within the neighboring flat region. We find the optimal number of folds for a given K and L/w_{fold} with constant $W/w_{fold} = 50.0$. We then fit these data to Equation G.5 using least-squares regression, and find $\alpha = 0.0618 \pm 0.0010$ and $\beta = 0.528 \pm 0.003$ ($\pm$ one standard deviation). The fit can be seen in Figure G.1. This equation appears to fit the data quite well, with $R^2 = 0.989$. The data also collapse fairly well onto a master curve, with the largest deviation coming from the smallest sheet, where edge effects are the most relevant. This suggests that $n^*_{fold} \sim \sqrt{L/w_{fold}}$, as predicted.

Appendix H

Scatter plots of sheet viscosity versus sheet summary statistics

The radius of gyration is given by the square root of the sum of the eigenvalues of the gyration tensor:

$$R_g = \sqrt{\sum \lambda_i^2}. \tag{H.1}$$

The radius of gyration is a measure of the overall **size** of a sheet.

The relative shape anisotropy can similarly be defined in terms of the eigenvalues of the gyration tensor:

$$\zeta^2 = \frac{3}{2}\frac{\sum \lambda_i^4}{(\sum \lambda_i^2)^2} - \frac{1}{2} \tag{H.2}$$

The relative shape anisotropy is bounded between 0 and 1. $\zeta^2 = 0$ only if the beads are spherically symmetric and $\zeta^2 = 1$ only if the beads are all colinear. The relative shape anisotropy is therefore a measure of the **shape** of a sheet.

The alignment is given by the magnitude of the dot product of the unit vector pointing in the same direction as the largest axis of the minimum bounding ellipsoid, $\mathbf{v}_1$, with the vorticity axis, $\hat{z}$: $|\mathbf{v}_1 \cdot \hat{z}|$. The alignment is therefore a measure of the **orientation** of a sheet.

Below, we include scatter plots for the sheet viscosity versus the average of each of these

quantities over the last $200\dot{\gamma}t$ with samples taken every $100\dot{\gamma}t$ for all simulations from Chapter 4.

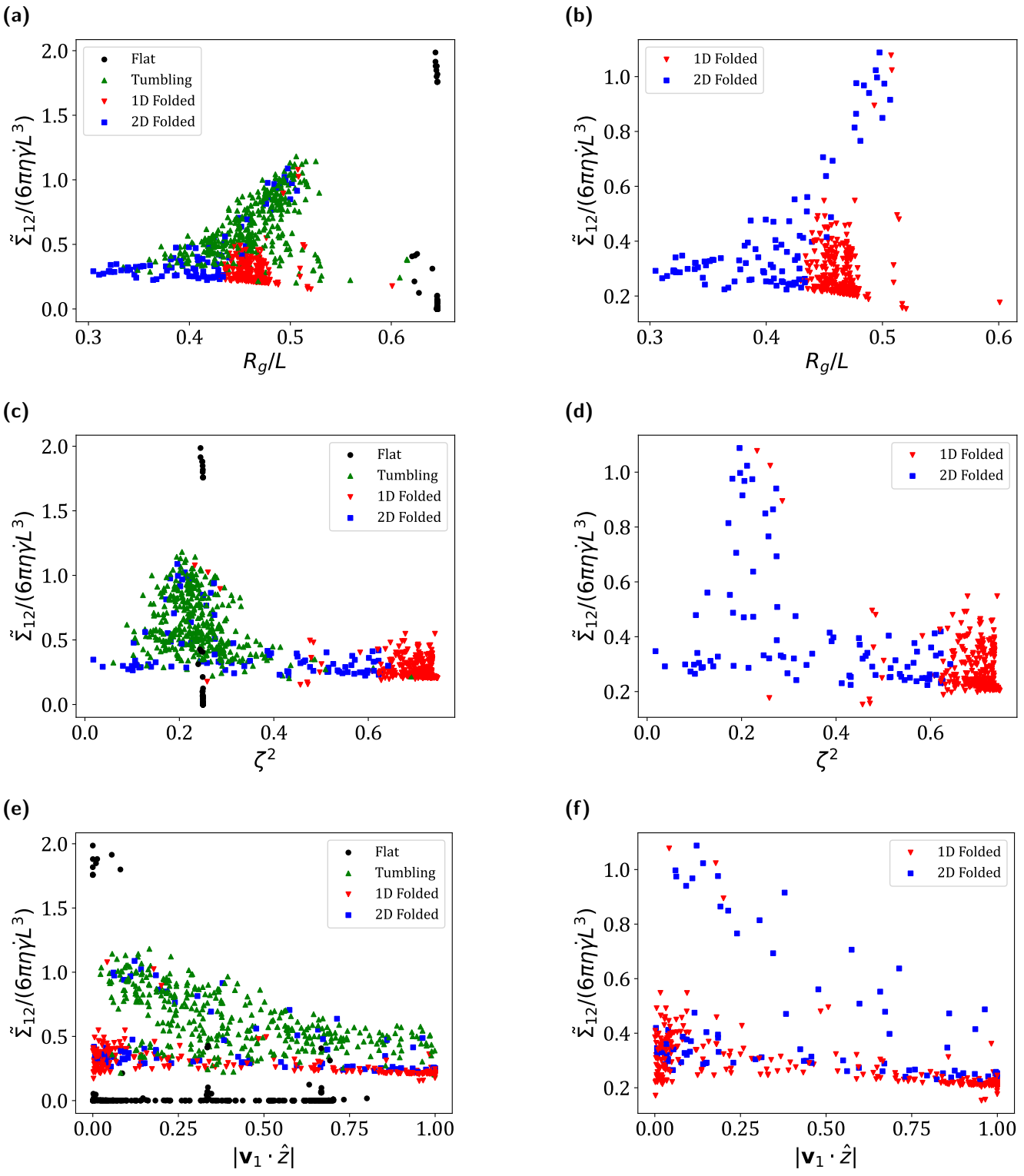

Figure H.1 Scatter plots of the sheet viscosity for all simulations from Chapter 4 versus the (a-b) radius of gyration, (c-d) relative shape anisotropy, and (e-f) alignment of sheets averaged over the last $200\dot{\gamma}t$ of each simulation. (a, c, e) Scatter plots for all simulations; and (b, d, f) scatter plots for only folded sheets.

Bibliography

[1] GH Golub and CF Van Loan. *Matrix computations*. JHU Press, 2013.

[2] Å Björck. Solving linear least squares problems by gram-schmidt orthogonalization. *BIT Numerical Mathematics*, 7(1):1–21, 1967.

[3] L Giraud, J Langou, and M Rozloznik. The loss of orthogonality in the gram-schmidt orthogonalization process. *Computers & Mathematics with Applications*, 50(7):1069–1075, 2005.

[4] GL Klimchitskaya and VM Mostepanenko. Van der waals and casimir interactions between two graphene sheets. *Physical Review B*, 87(7):075439, 2013.

[5] KR Paton, E Varrla, C Backes, et al. Scalable production of large quantities of defect-free few-layer graphene by shear exfoliation in liquids. *Nature Materials*, 13(6):624–630, 2014.

[6] U Khan, A O'Neill, H Porwal, et al. Size selection of dispersed, exfoliated graphene flakes by controlled centrifugation. *Carbon*, 50(2):470–475, 2012.

[7] JY Moon, M Kim, SI Kim, et al. Layer-engineered large-area exfoliation of graphene. *Science Advances*, 6(44):eabc6601, 2020.