

Abstract of “Data-Calibrated Modeling of Biological Soft Matter with Dissipative Particle Dynamics and High-Performance Bayesian Uncertainty Quantification” by Clark M. R. Bowman, Ph.D., Brown University, May 2018.

Fluids play a key role in the mechanics of cells and the soft-matter structures composing them; from the movement of molecules such as amino acids via diffusion to the viscous forces propelling flagellating organisms, accounting for the role of the fluid environment is critical to understanding the mechanisms at play. One of the major challenges in mathematical modeling of such systems is developing an accurate, yet computationally feasible representation of the fluid. Continuum approaches such as Navier-Stokes neglect the thermal fluctuations and boundary effects that can dominate flow behavior at the nanoscale, while explicit models of the fluid via, e.g., molecular dynamics become computationally infeasible for systems on the order of biological cells which may have millions or billions of molecules. In this thesis, we use dissipative particle dynamics, a particle method which uses coarse-grained particles interacting with artificial forces chosen to generate accurate fluid behavior, to model a number of biofluidic systems of experimental interest, including the diffusion of DNA in constrained environments, the mechanics of poration in phospholipid membranes, and the mechanical force profile of polymerizing actin networks in the cytoskeleton. The simulation results are used to examine at the nanoscale the dynamics underlying macroscopic behaviors observed in experiment, answering a number of questions about the forces, energies, motions, and mechanisms of biological soft matter with measurements from the explicitly modeled fluid environment. We also introduce a framework for high-performance Bayesian uncertainty quantification and demonstrate an application to inferring structural properties of lipids in a bilayer membrane, illustrating the feasibility of data-driven model calibration for our complex dissipative particle simulations using parallel computing. Together, these methods allow for a uniquely fine-scale look at the mechanics underpinning a number of biologically relevant phenomena.

Data-Calibrated Modeling of Biological Soft Matter with Dissipative Particle Dynamics and
High-Performance Bayesian Uncertainty Quantification

by

Clark M. R. Bowman

B. S., University of Rochester, 2012

Sc. M., Brown University, 2013

A dissertation submitted in partial fulfillment of the
requirements for the Degree of Doctor of Philosophy
in the Division of Applied Mathematics at Brown University

Providence, Rhode Island

May 2018

© Copyright 2018 by Clark M. R. Bowman

This dissertation by Clark M. R. Bowman is accepted in its present form by
the Division of Applied Mathematics as satisfying the dissertation requirement
for the degree of Doctor of Philosophy.

Date _____

Anastasios Matzavinos, Director

Recommended to the Graduate Council

Date _____

Derek Stein, Reader
Departments of Physics and Engineering

Date _____

Mark Chaplain, Reader
School of Mathematics and Statistics,
University of St Andrews

Approved by the Graduate Council

Date _____

Andrew G. Campbell
Dean of the Graduate School

Vita

EDUCATION

Ph.D. Applied Mathematics

2013 - 2018

Brown University

Advisor: Anastasios Matzavinos (matzavinos@brown.edu)

M.Sc. Applied Mathematics

2012 - 2013

Brown University

B.Sc. Mathematics, B.Sc. Physics

2008 - 2012

University of Rochester

PUBLICATIONS

5. **C. Bowman**, M. Chaplain, and A. Matzavinos. Dissipative particle dynamics simulation of critical pore size in a lipid bilayer membrane. Submitted. Preprint available at <https://arxiv.org/abs/1710.09268>.
4. Z. Chen, K. Larson, **C. Bowman**, P. Hadjidoukas, C. Papadimitriou, P. Koumoutsakos, and A. Matzavinos. Data-driven prediction and origin identification of epidemics in population networks. Submitted. Preprint at <https://arxiv.org/abs/1710.07880>.

3. **C. Bowman**, K. Larson, A. Roitershtein, D. Stein, and A. Matzavinos. Bayesian uncertainty quantification for particle-based simulation of lipid bilayer membranes. In *Cell Movement: Modeling and Applications*. Springer, accepted.
2. D. Kim, **C. Bowman**, J. T. Del Bonis-O'Donnell, A. Matzavinos, and D. Stein. Giant acceleration of DNA diffusion in an array of entropic barriers. *Physical Review Letters*, 118 (4): 048002, 2017.
1. **C. Bowman**, J. K. Hodge, and A. Yu. The potential of iterative voting to solve the separability problem in referendum elections. *Theory and Decision*, 77 (1): 111-124, 2014.

TALKS AND POSTERS

7. Bayesian uncertainty quantification for complex models. Invited talk. Math Slam, Brown University, 2017.
6. Bayesian uncertainty quantification for complex models using transitional Markov chain Monte Carlo. Invited talk. Pattern theory seminar, Brown University, 2017.
5. Using dissipative particle dynamics to model DNA flow across periodic nanopits. Contributed talk. SMB Annual Meeting, Atlanta, GA, 2015.
4. Modeling biological polymer networks and the flow of DNA in entropic wells. Invited talk. DPD group seminar, Brown University, 2015.
3. Information theoretically secure computation protocols in the quantum noisy storage model. Research presentation. UCLA/IPAM RIPS, Los Angeles, CA, 2012.
2. Can iterative voting solve the separability problem in referendum elections? Poster. The 2012 JMM Annual Meeting, Boston, MA, 2012.

1. Can iterative voting solve the separability problem in referendum elections? Contributed talk.
MAA MathFest, Lexington, KY, 2011.

TEACHING

Brown University

9. Statistical Inference I

Head Teaching Assistant (incl. course planning and organization)	Spring 2017
	Fall 2016
Teaching Assistant (incl. guest lectures)	Spring 2016
	Fall 2015

8. Brown-ICERM-Kobe Simulation Summer School

Instructor	Summer 2015
------------	-------------

7. Introduction to Computational Linear Algebra

Guest Lecturer	Fall 2014
----------------	-----------

6. Recent Applications of Probability and Statistics (Graduate)

Teaching Assistant (incl. guest lectures)	Spring 2014
---	-------------

5. Computational Probability and Statistics

Teaching Assistant	Fall 2013
--------------------	-----------

University of Rochester

4. Waves and Modern Physics

Teaching Assistant

Spring 2012

3. Applied Fourier Series & Boundary Value Problems

Teaching Assistant

Fall 2011

2. Electricity and Magnetism

Teaching Assistant

Fall 2011

1. Introduction to Mechanics

Teaching Assistant

Spring 2011

SERVICE

Brown Applied Math Graduate Mentor

2015 - 2018

Undergraduate mentees: Kenneth Peluso, Thabo Samakhoana (Class of 2018)

Brown Math Resource Center Staff

2012 - 2014

TRAVEL AWARDS

Society for Mathematical Biology Landahl Travel Grant

2016

SMB Annual Meeting, Nottingham, UK

Brown University Graduate Conference Funding

2016

SMB Annual Meeting, Nottingham, UK

Brown University Graduate Conference Funding

2015

SMB Annual Meeting, Atlanta, GA

Preface and Acknowledgments

If I have seen further, it is by standing on the shoulders of giants.

– *Isaac Newton, Letter to Robert Hooke, 1676*

The immortal words of Newton are a reminder that all advancements in human knowledge rely on the achievements of those who came before. Personally, I see them equally well as a reminder of the importance of recognizing the positive effects others have on your life; humans are by nature social creatures, and we rely on others to support us in myriad ways as we endeavor to fulfill our ambitions. The work before you is the latest entry in a long list of examples.

I owe the greatest debt to my advisor, Anastasios Matzavinos, who has spent a great deal of time over the past four years barraging me with interesting projects to pursue and ideas to consider. The interpersonal element of a work collaboration is an important one, and at this point I cannot imagine having completed my dissertation work with anyone else. For your patience, enthusiasm, and one too many last-second meetings, thank you.

I extend another thank you to reader Derek Stein, who has been an ideal collaborator and mentor in the three years we have known each other. Your contagious enthusiasm and brilliant intuition made our meetings a continual source of enlightenment (and occasionally amusement). I would also like to sincerely thank reader Mark Chaplain, who has for the second time agreed to dedicate his time and effort to my pursuits on short notice.

I owe a special thank you to Daniel Kim, whose months of experimental work and incredible

organizational skills made possible the publishing of a wonderful paper, and to Karen Larson and Zhizhong Chen, my comrades-in-arms under the Matzavinos Group flag and future co-authors of multiple manuscripts in submission. I sincerely hope I will have the opportunity to work with all of you again.

To all of my other friends in the Division, especially recent Ph.D. Wilson McKerrow, soon-to-be Ph.D. Leroy Jia, and probably next-year Ph.D. Michael Burkhart, thank you for occasionally functioning as my last bastion of sanity. Being in the presence of so many talented colleagues has been a constant source of inspiration and encouragement.

I would be remiss without ending with a truly heartfelt statement of gratitude to my family. To my parents Alan and Amy, who have done everything in their power and more to see me happy, and to my siblings Graham, Emma, and Darcy, whose collective empathy and diligence have been equal parts reassuring and motivational: thank you. It is no exaggeration to call your support fundamental to my success.

I will leave you here with the results. I hope they help you to see just a bit further.

Clark M. R. Bowman

Brown University

Providence, RI

April 19, 2018

Contents

List of Tables	xiii
List of Figures	xiv
1 Introduction	1
2 Particle Simulation for Fluid Dynamics	4
2.1 Molecular Dynamics (MD)	5
2.2 Brownian Dynamics (BD)	7
2.3 Dissipative Particle Dynamics (DPD)	9
2.4 Numerical Solutions	14
3 Dissipative Particle Dynamics Models of Biological Soft Matter	16
3.1 Polymer Models	17
3.1.1 Wormlike Chain	17
3.1.2 Freely Jointed Chain	18
3.1.3 Freely Rotating Chain	19
3.1.4 Bead-Spring Chain	20
3.2 DNA in an Array of Entropic Barriers	20
3.2.1 Experimental Setup and Results	23

3.2.2	Giant Acceleration of Diffusion (GAD)	26
3.2.3	DPD Simulation of the Diffusion Curve	29
3.2.4	The DPD Force Landscape	32
3.2.5	Discovery of Design Principles Through DPD	39
3.2.6	Brownian Dynamics and Future Directions	53
3.3	Membrane Models	55
3.3.1	Energy Minimization for Surfaces	56
3.3.2	Three-dimensional DPD Membrane	57
3.3.3	Two-dimensional DPD Membrane	60
3.4	Critical Pore Size in Films and Membranes	63
3.4.1	Existence of a Critical Radius	65
3.4.2	Membranes of Varied Thickness	67
3.4.3	Pores in Finite Membranes	69
3.5	Polymerization of Actin Filaments	72
3.6	Vesicles Propelled by Actin Networks	81
4	High-Performance Bayesian Uncertainty Quantification	85
4.1	Framework	87
4.2	Markov Chain Monte Carlo (MCMC)	89
4.3	Transitional Markov Chain Monte Carlo (TMCMC)	93
5	Uncertainty Quantification for Complex Models	97
5.1	DPD Lipid Membrane	97
5.1.1	Membrane Curvature	98
5.1.2	Parameter Estimation	100
5.1.3	Results	102
5.2	Other Applications	108

6	Conclusions and Directions	109
A	Computational Details	114
	Bibliography	115

List of Tables

3.1	Pairwise Coefficients for 3D DPD Membrane	59
3.2	Pairwise Coefficients for 2D DPD Membrane	61

List of Figures

2.1	Schematic of particle simulations for a polymer (red) in water (blue). (A) Molecular dynamics approach. Forces on the polymer are exerted by individual water molecules which repel each other according to a basic model for neutral volume exclusion, e.g., a Lennard-Jones potential. (B) Brownian dynamics approach. The water is not modeled explicitly, with the polymer subject to random forces corresponding to Brownian motion of water molecules and drag forces corresponding to friction against the background flow. (C) Coarse-grained approach (e.g., dissipative particle dynamics). Instead of modeling individual water molecules, coarse-grained “clumps” of water are the discrete unit which interact with artificial potentials chosen to yield target physical properties. The polymer (red) may also be coarse-grained.	6
2.2	Schematic of DPD particles and their degree of interaction. Moving from (a) to (c), the dissipative particles move together until completely overlapping. Despite occupying the same location in space, their component atoms need not overlap, and so the repulsive potential should not diverge.	11

3.1	Polymer models with example point $\mathbf{r}(s)$ at arc length s . (A) Wormlike chain. (B) Freely jointed chain with Kuhn length ℓ_K . Joints between rods are idealized ball bearings. (C) Freely rotating chain with fixed colatitudinal angle θ . Links are free to rotate only on the given cone, i.e., out of the page. (D) Bead-spring chain. Unlike in (B) and (C), the links do not have fixed length ℓ_K , but are instead Hookean springs with resting length ℓ_K	18
3.2	(A) Sketch of a tilted periodic energy landscape. (B) Illustration of a nanopit array embedded in a nanoslit, with DNA hopping from pit-to-pit. (C) Side-view of the nanofluidic geometry indicating h , H , and a . A simulated polymer hops between pits under an applied drive of $v = 0.05$. (D) Time-sliced fluorescence images of λ DNA climbing a column of high-barrier pits ($h = 75$ nm, $H = 124$ nm) for $\Delta p = 400$ mbar. (E) DNA trajectories from the same experiment. The dashed line indicates $\Delta y = v_p t$ ($v_p = 0.7 \mu\text{m s}^{-1}$).	22
3.3	Dependence of MSD on τ in (A) the device with low barriers for $\Delta p =$ (a) 250, (b) 140, (c) 90, and (d) 60 mbar; and in (B) the device with high barriers for $\Delta p =$ (i) 750, (ii) 650, (iii) 555, and (iv) 400 mbar. The corresponding D/D_0 and v_s are indicated in (C). The dashed line has a slope of one. (C) Dependence of D/D_0 on v_s for low (blue dots) and high (red squares) barriers with $\tau = 2$ s. Error bars are the standard error from a bootstrap analysis of 200 resamplings of the full dataset.	25
3.4	Dependence of D/D_0 on $v_s/v_c - 1$ in devices with high (red squares) and low (blue dots) barriers, and for simulation data (diamonds). The lines are fits of Eq. 3.13 to the corresponding data, with fit parameters μ and F_c	29

3.5	Histogram of computed $\langle r_g \rangle$ for 5 million steps of simulation with example bending amplitude $K = 0.25$ in pit-free channel (left) and for Apr. 2, 2015 dataset of experiment with low barriers (right). The mean $\langle r_g \rangle$ from simulation was matched to the low barrier $\langle r_g \rangle$ by adjusting K and using the spatial scaling of 40 LJ units to 1 μm which matched the device geometry.	31
3.6	Normalized log velocity autocorrelation functions for the simulated polymer with $v = 0.04, 0.05$, and 0.06 plotted on (a) short and (b) long timescales in simulation time units. The viscous drag coefficients were obtained from slopes of log-linear fits of the data in the range $0.2 \leq \tau \leq 1.0$ simulation units, shown shaded in grey. . . .	36
3.7	(A) Side view of a simulated polymer inside a nanfluidic device, near the critical point, with $v = 0.05$. (B) Effective force landscapes $-d\Psi(y)/dy$ for $v = 0.04, 0.05$, and 0.06 (labeled curves), computed from Eq. 3.18. The position axis is aligned with the nanofluidic geometry in (A). The inset shows an expanded view of the force landscape around the critical point. The dashed line indicates zero force.	38
3.8	Dependence of v_p/v_0 on F/F_c in devices with high (red squares) and low (blue dots) barriers, and simulation data (black diamonds). The corresponding lines plot Eq. 3.32 using the fit values of μ and F_c . The dot-dashed line shows $v_p/v_0 = 1$	39
3.9	Long-time evolution of simulated DNA radius of gyration and occupancy function at the critical velocity $v = 0.2$. The radius of gyration fluctuates as the polymer alternates between extension during jumps and compression during entropic trapping.	41
3.10	Evolution of simulated DNA position and occupancy function in the region surrounding the double jump occurring around time step 40 million in Fig. 3.9.	41
3.11	Top and side view of a simulated DNA “double jump” at $v = 0.2$ featuring the polymer (red) juxtaposed with a subset of dissipative fluid beads (blue). The DNA at least partially explores the depth of all three pits.	42

3.12	Top and side view of simulated DNA in a control channel without nanopits at $v = 0.2$. The DNA moves together with the fluid and diffuses freely.	43
3.13	Evolution of simulated DNA position and radius of gyration in the control channel without nanopits. After relaxing from its initial configuration, the polymer reaches neutral diffusion in the channel as it is driven forward at the mean flow velocity. . .	43
3.14	Top and side view of simulated DNA in a grooved control channel at $v = 0.2$ in the groove (top) and during a jump (bottom). The “rubber-band” flow profile drives the polymer to oscillate between periods of extension and compression.	44
3.15	Evolution of simulated DNA radius of gyration and occupancy function in the grooved control channel. Peaks in $\langle r_g \rangle$ correspond to jumps between grooves.	45
3.16	Top and side view of simulated DNA in a device with deeper nanopits at $v = 0.2$. The deeper pits provide a stronger entropic trapping effect and reduce the mean forward velocity of the DNA.	47
3.17	Evolution of simulated DNA position and occupancy in a device with deeper nanopits at $v = 0.2$. Single jumps between nanopits occur at stochastic intervals and at a slower rate than the base experiment shown in Fig. 3.11.	47
3.18	Top and side view of simulated DNA in the device with shallow nanopits at $v = 0.2$. The shallow pits weaken the entropic trapping effect and allow the DNA to stream more freely, similarly to observed behavior in the control channel without nanopits. .	48
3.19	Evolution of simulated DNA position and occupancy function in the device with shallow pits at $v = 0.2$. A reduced entropic barrier allows the forward end of the polymer to easily escape nanopits, resulting in a number of extensions across three pits simultaneously.	48

3.20	Top and side view of simulated DNA in device with doubled pit separation. As the forward end of the DNA must travel farther before entering and exploring the weaker driving flow in the following nanopit, DNA extension during jumps increases significantly.	49
3.21	Evolution of simulated DNA position and occupancy (top) and radius of gyration (bottom) in device with doubled pit separation. Peaks in $\langle r_g \rangle$ are significantly taller than the base simulation.	50
3.22	Top and side view of simulated DNA in a device with reduced flow velocity $v = 0.1$. At this driving flow rate, the polymer cannot overcome the entropic barrier to forward motion. The illustrated subset of fluid particles (blue), which began in a rough vertical line, show the flow profile wrapping around the nanopits.	51
3.23	Evolution of simulated DNA position and occupation function for a device with reduced flow velocity $v = 0.1$. The rear-most section of the polymer explores backward into the pit, but the forward-most section is unable to overcome the trapping effect.	51
3.24	Top and side view of simulated DNA in a device with increased flow velocity $v = 0.3$. At this flow rate, the polymer streams nearly freely across pits, with extension resulting from the flow profile of fluid in the device.	52
3.25	Evolution of simulated DNA position and occupation function in a device with increased flow velocity $v = 0.3$. The mean flow rate of the DNA increases sharply as the polymer is pulled into the high-speed flow sections between the nanopit arrays.	52
3.26	Advancing Fig. 3.24 by 800,000 time steps. Accelerated flows on either side of the nanopit array result in massive extension of the DNA.	53

3.27	(a) Lipid with three heads and two tails comprising six tail particles each. Bonds are shown as thin rods. Three-body potentials are imposed among triplets with consecutive horizontal bonds. (b) 3D view of simulated membrane. Fluid particles (not shown) fill the space on both sides of the membrane. (c) Side view of simulated membrane showing the lipid heads oriented to separate the membrane interior from the surrounding fluid.	59
3.28	Schematic of a lipid with one head particle (red) and two tails comprising six tail particles each (gray). Bonds are shown as thin rods. The angle between tails, labeled ϕ , minimizes energy at a specified neutral angle ϕ_0	61
3.29	(a) Two-dimensional membrane initialization. Head particles (red) line the outside; each has two lipid tails of six tail particles (gray) directed inward. (b) Fluid initialization. Water particles are placed on the vertices of a square lattice outside the membrane. (c) Membrane after 2500 steps.	62
3.30	(a) 3D view of simulated membrane with pore. Fluid particles (not shown) fill the space on both sides of the membrane and in the pore. (b) Side and front view of initial membrane configuration. Blue lines at boundary show the simulation box. (c) Binary matrix (value 1 shown in black) resulting from smoothing and thresholding the front view in (b).	64
3.31	Rendered images of pores of three different initial sizes (rows). Four images throughout the simulation are shown, progressing from left to right. Pores shrink or grow exponentially depending on the initial pore radius.	66
3.32	Membrane pore size over time as calculated by Eq. (3.41). Size (y axis) is shown in units of the average membrane width, $h = 7.98$. Several initial sizes are shown, ranging from $r_0/h = 1.21$ (bottom) to $r_0/h = 2.56$ (top). The shaded region denotes possible values of the critical ratio r_c/h below which pores seal – $1.34 < r_c/h < 1.47$	66

3.33	Pore size over time via Eq. (3.41) for a membrane with increased lipid density. Three initial sizes are shown: top: $r_0/h = 3.19$, middle: $r_0/h = 2.52$, and bottom: $r_0/h = 1.86$. The shaded region corresponds to the possible range of r_c/h derived in Figure 3.32.	67
3.34	Membrane pore size over time from Eq. (3.41). Size (y axis) uses the new average membrane width $h = 8.63$. The shaded region corresponds to the new range for the critical radius, $1.48 < r_c/h < 1.59$	68
3.35	Pore size over time via Eq. (3.41) for a membrane with 12 particles per lipid tail. Only the smallest pore simulated was not stable for the duration of the simulation. This membrane had a width of $h = 13.49$, or 69.05% thicker than the membrane of Figures 3.32-3.34.	69
3.36	Rendered images of simulation results for truncated membranes; time increases left to right. Front and internal side views for the original membrane with 6 particles per lipid tail are shown in (a), while the modified membrane with 12 particles per lipid tail is shown in (b). The original membrane seals its pore but simultaneously transitions into a layered cluster. Conversely, the thicker membrane seals its pore but remains in a stable finite disc indefinitely. The interior of the membrane in (b) comprises only tail particles and is not shown.	70
3.37	(a) Side view of flexed actin filaments (green) and 3D membrane (red lipid heads, silver lipid tails) at initialization. The purple particles on the far left end of each filament are fixed, forcing the filaments to relax into the membrane. (b) 3D view of same system. Fluid particles (not shown) fill the space on both sides of the membrane.	73
3.38	(a) Side view of flexed actin filaments and 3D membrane at $t = 240$. (b) 3D view of same system. The actin filaments unfurl and exert forward pressure on the membrane, creating a flow of fluid with a feedback loop as the membrane is driven ahead. . . .	74

3.39	Membrane mean x position over time for the simulation shown in Figs. 3.37 and 3.38.	
	After an initial period of acceleration, the forward motion of the membrane achieves a terminal velocity of roughly 12.34 units per 100,000 steps ($v = 0.0617$).	74
3.40	(a) Actin filament (blue) with marked terminal end (orange) in the presence of freely diffusing monomers (green). Free monomers within a specified distance of the free end, shown with the dotted circle, result in polymerization with a fixed step-by-step probability. (b) Filament and monomer after polymerization. The newly bonded monomer is the new terminal end, which will then repeat the process by forming new bonds with other free monomers.	75
3.41	(a) Side view of polymerizing actin filaments (green with blue terminal monomers) and 3D membrane (red lipid heads, silver lipid tails) at initialization. Faded particles are freely diffusing monomers. (b) 3D view of same system. Fluid particles not shown.	78
3.42	(a) Side view of polymerizing actin filaments and 3D membrane at $t = 240$. Diffusing monomers (blue) have been made opaque to show the effects of polymerization. (b) 3D view of same system. The polymerizing filaments exert a forward pressure on the membrane, driving net forward motion which creates a flow drawing additional actin monomers to the polymerization interface.	78
3.43	(a) Average membrane x position as a function of timestep for the polymerizing actin model in the presence of a retarding force. Colored curves correspond to force magnitudes specified in the legend applied to each dissipative lipid bead. Shaded region from 150000 – 250000 steps was used to estimate the membrane velocity. (b) Closer view of (a) for small times.	79
3.44	Mean membrane velocity v from $t = 150$ to $t = 250$ as a function of the retarding force f . The dotted line shows the least-squares linear fit $v = 0.0408 - 12.315f$	80

3.45	(a) 2D lipid vesicle (red lipid heads, silver lipid tails) at circular initialization. Water with numerical density $\rho = 2.1$ inside and outside the vesicle is not shown. The membrane is divided into four regions which experience different applied forces; sections of the top and bottom are driven forward by an applied force $f = 0.15$ in the x -direction, while a section on the left is pulled backward by an applied force $f = -0.25$. As these sections are equal in length, the net force on the vesicle is forward. (b) Lipid vesicle after equilibration under applied forces.	82
3.46	Time evolution of the vesicle of Fig. 3.45 rightward in the fluid channel. Water beads (not shown) fill the shaded region of the channel, i.e., both inside and outside the vesicle.	84
4.1	The Metropolis algorithm applied to the density $p(x) \propto 0.3e^{-0.2(x+5)^2} + 0.7e^{-0.2(x-15)^2}$ (solid curve). The proposal distribution used is a Gaussian with $\sigma = 4$. Shown are the most recent 10% of samples after the specific number of steps. (a) For short times, the chain gets “stuck” in the smaller peak, unable to cross the bottleneck to explore the full distribution. (b) For intermediate times, although the distribution is explored fully, the chain does not spend the correct proportion of time in each peak. (c) For large times, convergence to the invariant distribution is guaranteed via irreducibility and aperiodicity of the chain.	92
4.2	Simulated annealing with cooling schedule $T(k) = 5(1 - 10^{-7}k)$ for the toy problem of Figure 4.1. Again, the proposal distribution used is a zero-mean Gaussian with $\sigma = 4$. Shown are the most recent 10% of samples after the specific number of steps. (a, b) Even for short times, the high temperature smooths the target distribution, allowing sampling from both peaks. (c) As the temperature falls to zero, the distribution clusters around the global maximum.	93

5.1	Schematic of tracked lipids and method for determination of curvature. Angles are with respect to a fixed reference angle (here shown as vertical lines) and can thus be negative. The derivation for approximate curvature κ relies only on the absolute difference $ \theta_2 - \theta_1 $ and tail-to-tail distance d	99
5.2	5000 samples from approximate posterior with $\phi_t^* = 100$, $\phi_b^* = 20$ ($\kappa^* = 0.03834$). Histograms on the major diagonal are the empirical marginal distributions of ϕ_t , ϕ_b , s , from left to right. Plots in the upper right are the empirical joint distributions of pairs of parameters; the specific parameters and values appear in the margin. Shaded plots in the lower left are smoothed representations of the joint posterior, color-coded by probability – blue (low) to yellow (high).	103
5.3	5000 samples from approximate posterior with $\phi_t^* = 80$, $\phi_b^* = 20$ ($\kappa^* = 0.02087$). See Fig. 5.2 for description of subfigures.	104
5.4	5000 samples from approximate posterior with $\phi_t^* = 20$, $\phi_b^* = 20$ ($\kappa^* = 0.001480$). See Fig. 5.2 for description of subfigures.	105
5.5	5000 samples from approximate posterior with $\kappa^* = 0$. See Fig. 5.2 for description of subfigures.	106

Chapter 1

Introduction

Fluids play a critical role in cellular biology, where the mechanics of biological molecules and structures associated with a wide range of life processes are governed by interaction with their fluid environment. Biological matter within the cell interacts in the context of the cell cytoplasm, which is mostly water; many molecules, such as amino acids, move through the cytoplasm by self-diffusion [108]. In plant and fungal cells, materials are transported throughout the cell by flows which develop in the cytoplasm as a result of the movement of myosin-coated organelles in the cytoskeleton [117]. Fluids are also relevant to cell motility; organisms with flagella such as *E. Coli* rely on viscous resistance to their rotating or oscillating tails to move forward [8]. Motility can also be driven by fluidic mechanisms within the cell, such as actin polymerization and membrane blebbing [134, 14]. Fluids are thus an essential component of modeling many biologically relevant systems at the cellular scale.

Studying mechanics at this scale can be challenging for a number of reasons. Experimentally, these biofluidic systems can be difficult to observe directly, as their short spatial and temporal scales can require measurements to be taken at unfeasibly high resolution and frequency to directly observe the mechanics of interest. Experiments thus often rely on macroscale observables, drawing inference about the underlying mechanics via hypothesis or fitting to a physical or analytical model. Yet there is an additional challenge inherent to analyzing nanofluids: at extremely small scales, the

usual methods for continuum fluid modeling, most notably partial differential equations such as the Navier-Stokes and Euler equations, become unphysical given the vacuum between discrete water molecules at the molecular scale. In particular, fluids at this scale are affected heavily by thermal fluctuations and by the physical conditions defining fluid-surface interfaces [58].

We present in this thesis a collection of nanofluidic systems involving biological soft matter, including polymers such as DNA and filamentous actin and the phospholipid membranes which separate cells from their surroundings. These systems are subject to the challenges described above in that their size makes direct observation of the mechanics impossible: conclusions about the dynamics on small time scales must be drawn from mathematical and physical models calibrated with measurable data from experiment or hypothesized based on physical constraints. A micro-scale approach must be taken to model the fluid: the principal among these is molecular dynamics, which models the fluid molecules individually and tracks their motion via Newton's second law, which relates forces to motion, under the assumption of specific forces of interaction between molecules based on their distance and physical characteristics. Yet the basic model of molecular dynamics is subject to its own challenges: biofluidic systems such as lipid vesicles and their fluid environment may contain millions or even billions of molecules, making such simulations intractable even with modern high-performance computing facilities.

The core approach of this thesis is dissipative particle dynamics (DPD), a framework proposed in 1992 by Hoogerbrugge and Koelman for modeling fluids at scales in between the nanoscale realm of molecular dynamics and the macroscale realm of continuum approaches [49]. DPD uses a molecular dynamics-like approach in which system components are explicitly modeled with discrete particles; however, the DPD particles need not correspond directly to single molecules, but may instead represent coarse-grained clumps of many molecules. Analysis of the DPD approach by Español and Warren in 1995 [31] showed that such a system can reproduce accurate behavior of fluids in a variety of situations given a careful choice of interacting forces between particles, opening the door to applications of particle-like methods to bigger systems and longer times than were possible with

naïve molecular dynamics. With modern parallel computing architectures, DPD simulations are now capable of modeling systems as large as hundreds of lipid vesicles simultaneously [129]. DPD models of the collection of nanofluidic systems studied in this thesis will prove to be practical ways of bridging the explicit fluid models necessary to reproduce nanofluidic mechanics to the scale of the full systems, allowing for a uniquely fine-scale look at the mechanics underpinning a number of biologically relevant phenomena.

This thesis is divided into four main chapters. Chapter 2 will provide the necessary background on molecular dynamics, dissipative particle dynamics, and another common mesoscale particle method known as Brownian dynamics, which is based on the Langevin equation; the latter will be useful in analyzing a particular nanofluidic system described in Section 3.2, which has a number of parallels to Brownian particles in a potential energy landscape which diffuse via the overdamped one-dimensional Langevin equation.

Finally, Chapter 4 introduces a high-performance framework for Bayesian uncertainty quantification using recent advances in the Bayesian statistics literature (in particular, the Transitional Markov chain Monte Carlo or TMCMC algorithm). Specifically, we describe a framework for Bayesian parameter estimation, which has a natural application to mathematical models such as the DPD simulations presented in this thesis: as the parameters for DPD models are chosen based on physical arguments or to match observed measurements of target physical quantities, a probabilistic framework for parameter estimation allows these parameters and their corresponding uncertainties to be fit automatically from real-world or targeted measurements. Chapter 5 describes an application of this framework to the type of DPD membrane model studied in Chapter 3, providing an explicit example of successful parameter estimation for a DPD simulation and illustrating the feasibility of the framework for future applications to expensive particle simulations.

Chapter 2

Particle Simulation for Fluid Dynamics

We begin with a basic introduction to common particle simulation methods, i.e., the broad set of simulation techniques which rely on discretizing a fluidic system into fundamental particles which evolve jointly forward in time. Particle simulations are physically motivated; at the length and time scales characteristic of behaviors of interest in biological cells, for example, particles might correspond directly to molecules (or even atoms) interacting via the centuries-old Newton formulation of classical mechanics. In that sense, particle simulations would seem to be bound inextricably to the microscale, where they are a fundamental description of the underlying physics – yet there are numerous applications of particle methods to mesoscale (mid-range) and even macroscale systems using heavily coarse-grained particles (see, e.g., smoothed-particle hydrodynamics (SPH), not discussed here, which is used for modeling flows in oceanography, ballistics, and even astrophysics [91, 83, 127, 121]).

At the macroscale, the field of fluid dynamics is often synonymous with analytical and numerical solutions to partial differential equations (PDEs) derived from basic physical assumptions such as

conservation of momentum and conservation of mass. Perhaps the most famous such PDE is the incompressible Navier-Stokes equation

$$\partial_t \mathbf{v} + (\mathbf{v} \cdot \nabla) \mathbf{v} = -\frac{1}{\rho} \nabla p + \nu \nabla^2 \mathbf{v}, \quad (2.1)$$

which describes the flow velocity $\mathbf{v}$ as a function of time t in terms of the pressure p , density ρ , and kinematic viscosity ν [35]. Numerical solutions can be obtained using a variety of approaches from the PDE literature, including finite-difference (FD), finite-element (FE), and spectral methods, which largely rely on discretization of the problem to approximate the continuum solution [124, 37, 81]. Yet in the regime of microscopic fluids, where the simulation region may be only a few orders of magnitude larger than individual water molecules, the continuum assumptions inherent to PDE fluid models break down: the differential volume element of the fluid relative to the scale of the system is no longer large enough to neglect the underlying microscopic structure. There also exist systems which are discontinuous even at the macroscale, such as pedestrian flow on a walkway or rarefied gas dynamics [112, 115]. These systems require a physical treatment of the individual components of the system and provide a natural setting for modeling with particle methods.

2.1 Molecular Dynamics (MD)

The simplest and most widely used class of particle simulations is *molecular dynamics* (MD), i.e., the direct simulation of atoms or molecules via Newton's laws [59, 73]. Since all particles are modeled explicitly and all pairwise forces (e.g., the Lennard-Jones potential between neutral particles or Coulomb potential between charged particles) are included, molecular dynamics is often considered a ground truth for systems where such simulations are computationally feasible. A schematic of a molecular dynamics simulation of a polymer chain appears in Figure 2.1(A).

The fundamental governing equation of molecular dynamics simulations is Newton's second law

$$m_i \frac{d\mathbf{v}_i}{dt} = \mathbf{f}_i, \quad (2.2)$$

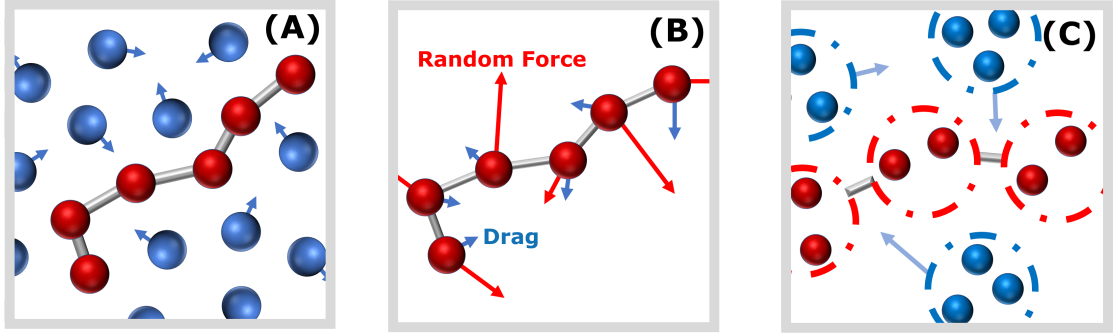


Figure 2.1: Schematic of particle simulations for a polymer (red) in water (blue). (A) Molecular dynamics approach. Forces on the polymer are exerted by individual water molecules which repel each other according to a basic model for neutral volume exclusion, e.g., a Lennard-Jones potential. (B) Brownian dynamics approach. The water is not modeled explicitly, with the polymer subject to random forces corresponding to Brownian motion of water molecules and drag forces corresponding to friction against the background flow. (C) Coarse-grained approach (e.g., dissipative particle dynamics). Instead of modeling individual water molecules, coarse-grained “clumps” of water are the discrete unit which interact with artificial potentials chosen to yield target physical properties. The polymer (red) may also be coarse-grained.

which relates the acceleration of each particle i (the derivative of its velocity $\mathbf{v}_i$) with its mass m_i and the applied force $\mathbf{f}_i$ exerted on the particle. In practice, the force is often the gradient of a conservative potential U :

$$\mathbf{f}_i = -\frac{\partial}{\partial \mathbf{r}_i} U(\mathbf{r}), \quad (2.3)$$

where $\mathbf{r} = (\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_N)$ is the tuple of three-dimensional positions of all N particles in the system. Together, Eqs. 2.2 and 2.3 form a coupled system of ordinary differential equations which is the core of the molecular dynamics approach.

The potential U is the mechanism by which particles interact. It is often expressed as the sum of pairwise conservative potentials which describe the interactions between particles as a function of their distance and physical properties such as electric charge, but can also include potentials describing the structure of several particles jointly, such as spring potentials between particles which are considered “bonded” or dihedral angle potentials on consecutive bonds in a chain of particles. More complex potentials such as AMBER [138, 20], CHARMM [10], and OPLS [57], which are designed to model specific interactions between larger molecules such as proteins and polymers, may

be necessary for the simulation behavior to match experimental data of more intricate observables such as phase coexistence.

The main difficulty in modeling systems via molecular dynamics is computational complexity: since each individual atom or molecule is tracked separately, the cost of the simulation grows massively with the scale of the system. For many systems on the order of biological cells, where the number of molecules may be in the billions, current computing power is not sufficient for practical simulation via molecular dynamics [59].

2.2 Brownian Dynamics (BD)

There are a number of approaches for bridging the basic idea of particle simulation to the so-called “mesoscale,” the spatial scale between the realms of molecular dynamics and continuum methods. One common approach, referred to as Brownian dynamics, does not explicitly model fluid particles, instead modeling only the solute, which is usually the material of interest. Forces which would be exerted on the solute by the solvent are incorporated via Gaussian force terms. As fluid usually composes the bulk of such simulations, Brownian dynamics simulations are orders of magnitude cheaper than comparable simulations using molecular dynamics.

Like molecular dynamics, the basic physical motivation for Brownian dynamics is Newton’s second law applied to each particle i (refer to Eq. 2.2). Unlike molecular dynamics, where the force $\mathbf{f}_i$ acting on particle i is a function of the positions $\mathbf{r}$ of all particles in the system, the force used in Brownian dynamics does not depend on the position of individual fluid particles, as the fluid is not modeled explicitly. Instead, it is written as the sum of two forces $\mathbf{F}_i^D$ and $\mathbf{F}_i^R$, defined as

$$\mathbf{F}_i^D = -\gamma_i \mathbf{v}_i, \quad (2.4)$$

$$\mathbf{F}_i^R = \boldsymbol{\eta}_i(t), \quad (2.5)$$

where γ_i is a drag coefficient and $\boldsymbol{\eta}_i(t)$ is a stochastic force which obeys the white noise assumption,

i.e.,

$$\langle \boldsymbol{\eta}_i(t) \rangle = \mathbf{0}, \quad (2.6)$$

$$\langle \boldsymbol{\eta}_i^\mu(t) \boldsymbol{\eta}_i^\nu(t') \rangle = \Gamma \delta_{\mu\nu} \delta(t - t'), \quad (2.7)$$

where μ and ν are Cartesian components (e.g., x, y, z), Γ is the strength of the noise, $\delta_{\mu\nu}$ is the Kronecker delta (1 if $\mu = \nu$, else 0), and $\delta(t - t')$ is the Dirac delta. It should be emphasized that all forces, positions, and velocities are implicitly functions of the time t , which is made explicit for $\boldsymbol{\eta}_i(t)$ only to emphasize its time-varying stochastic nature.

The D in $\mathbf{F}_i^D$ denotes *dissipative* – it is a frictional force which scales with the velocity of the particle with respect to a stationary background fluid. The R in $\mathbf{F}_i^R$ denotes *random* – the effect of the fluid particles which would have been explicitly modeled in the molecular dynamics approach is approximated as an artificial stochastic force. The conditions in Eq. 2.6 and 2.7 necessitate that the net random force $\mathbf{F}_i^R$ exerted on particle i over a small time interval Δt is the Cartesian product of three one-dimensional Gaussians with variance Δt , which is the same force behind classical three-dimensional Brownian motion (thus *Brownian dynamics*). In a sense, the assumptions of Brownian dynamics are that the fluid particles are largely obeying classical Brownian motion, and so their modeling can be neglected entirely, instead applying a Brownian-like force directly to the particle i of interest. When the explicit form of the forces is substituted into Eq. 2.2, the result is the original Langevin equation

$$m_i \frac{d\mathbf{v}_i}{dt} = -\gamma_i \mathbf{v}_i + \boldsymbol{\eta}_i(t). \quad (2.8)$$

As Brownian dynamics operates on the scale of ensembles of particles, the noise strength Γ is not a free parameter, but must depend in some sense on the temperature T of the system; a statistical argument invoking the equipartition theorem shows that $\Gamma = 2k_B T \gamma$ in terms of the Boltzmann constant k_B [19], which is an example of a more general result known as the fluctuation-dissipation theorem. The free parameters of the basic Brownian dynamics approach are thus only the drag coefficients γ_i and mass m_i of each particle together with the system temperature T .

It is important to note that Eq. 2.8 does not depend on the location of any other particle j , but only models the effect of the now-absent fluid; for this reason, interactions between Brownian particles are supplemented with the same types of potentials used in molecular dynamics simulations, especially the Lennard-Jones force which is an approximation of the interaction between neutral molecules. The effect of these forces, along with forces which model multi-particle interactions such as bonds and angles, is summed to the right-hand side of (2.8). A schematic of a BD-style polymer, which couples Brownian forces with particle-to-particle forces and bonds, appears in Fig. 2.1(B).

The main strength of the Brownian approach is also, on occasion, a drawback: as the surrounding fluid is assumed to be at equilibrium, non-equilibrium phenomena in the fluid, e.g., microflows, are completely neglected. In systems where such flows may dictate important properties of the mechanics, including several of the biological systems studied in this thesis, Brownian dynamics simulations may not be appropriate, though the Langevin equation and its relation to Brownian motion will prove important to the analytical understanding of the DNA device described in Section 3.2. In the interim, we instead develop an alternative approach which achieves its computational efficiency by different means.

2.3 Dissipative Particle Dynamics (DPD)

The bulk of this thesis will focus on the method of dissipative particle dynamics (DPD), a mesoscale approach which continues to explicitly model the fluid, instead deriving an improvement in computational efficiency from a rescaling of the particles themselves [59, 74]. Rather than modeling individual atoms or molecules, the DPD method tracks the movement of “dissipative particles,” which correspond to clumps of many atoms or molecules. This coarse-graining introduces a number of physical problems. Most notably, clumps of particles, which comprise mostly empty space, should be able to pass through each other, but should experience varying forces depending on the degree to which they overlap. To address these difficulties, the dissipative particles interact via artificial

forces, chosen to achieve target physical properties and to obey the correct hydrodynamic behavior on aggregate [113, 31]. In spite of its nonphysical nature, comparisons with Navier-Stokes and molecular dynamics simulations have shown the dissipative particle dynamics method to produce accurate results for a wide variety of simple- and complex-geometry flows [61].

In general, since DPD forces are defined on a pairwise basis between particles, they can be used in conjunction with other sets of forces by selectively applying each force only to the desired interactions. Here, for the sake of notational simplicity, we will analyze a system consisting of dissipative particles alone. As before, let the mass and velocity of particle i be given by m_i and $\mathbf{v}_i$, respectively. The equation of motion for particle i can be written as

$$m_i \frac{d\mathbf{v}_i}{dt} = \sum_{j \neq i} \mathbf{F}_{ij}^C + \sum_{j \neq i} \mathbf{F}_{ij}^D + \sum_{j \neq i} \mathbf{F}_{ij}^R, \quad (2.9)$$

where $\mathbf{F}_{ij}^C$ is a conservative force deriving from a potential exerted on particle i by the j -th particle, $\mathbf{F}_{ij}^D$ is a dissipative force modeling exchange of momentum due to friction, and $\mathbf{F}_{ij}^R$ is a random force inducing the random motion of particles. Again, this is an explicit form of Newton's second law, and the dissipative and random forces bear a resemblance to the selfsame forces employed in the Brownian approach. The physical interpretation, however, is quite different: as the fluid is modeled explicitly, the forces are no longer stand-ins for the effects of thermal fluctuations in a background fluid. Instead, the forces of DPD interaction follow directly from the nature of the dissipative particles as “clumps” of atoms and molecules.

When two particles are moved closer together, more of their components become close enough to interact, and so the particles should increasingly repulse each other. However, since the dissipative particles are mostly empty space, this repulsion should not diverge as they are forced together, since the component molecules could avoid overlap even if their parent dissipative particles occupy the exact same location. This justifies a soft conservative force – pairwise repulsion that does not diverge as the inter-particle separation $\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j \rightarrow 0$.

In addition to center-to-center repulsion due to the degree of overlap, particles should also exhibit

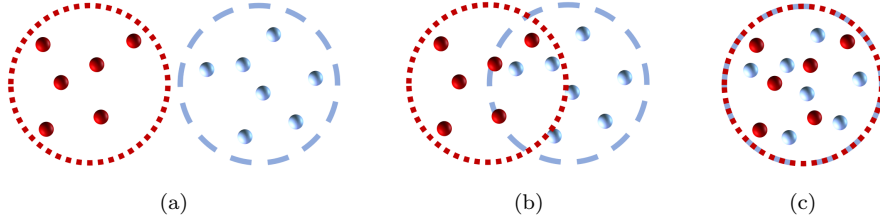


Figure 2.2: Schematic of DPD particles and their degree of interaction. Moving from (a) to (c), the dissipative particles move together until completely overlapping. Despite occupying the same location in space, their component atoms need not overlap, and so the repulsive potential should not diverge.

drag when passing through each other at nonzero velocity. The component atoms and molecules of each particle would be driven together, and the resulting collisions would impart a greater force than if the parent particles were simply held at a fixed distance. This “drag” between particles should thus scale with two different factors: the velocity (affecting the energy imparted when components collide) and the degree of overlap (affecting the number of collisions). This justifies a dissipative force which depends on both the relative velocity $\mathbf{v}_{ij} = \mathbf{v}_i - \mathbf{v}_j$ of particle i with respect to particle j and on the inter-particle separation $\mathbf{r}_{ij}$.

Finally, there should be some correction for the positions and velocities of the components, which are not tracked. Atoms and molecules within the dissipative particles interact in Brownian-like fashion, and so the interaction between particles should have a random component to account for the ensemble of component velocities. This justifies a random force which scales with the temperature T of the system, since higher temperatures increase the magnitude of the Brownian-style interaction between the molecules in each dissipative particle. Note that, like the other two forces, the random force $\mathbf{F}_{ij}^R$ is a force between two particles, and does not affect individual particles in isolation – though the atoms and molecules within a single dissipative particle also interact, any exchange of momentum does not affect the momentum of the dissipative particle which contains them.

Defined in this fashion, the forces are consistent with Galilean invariance, as they depend only on the relative positions and velocities of the particles. On physical grounds, it is also assumed that the

dissipative force $\mathbf{F}_{ij}^D$ depends linearly on the momentum and the random force $\mathbf{F}_{ij}^R$ is independent of the momentum (see, e.g., [31] or [59] for a more detailed discussion of these assumptions). The following functional forms for $\mathbf{F}_{ij}^D$ and $\mathbf{F}_{ij}^R$ satisfy these assumptions and are usually adopted in the context of the DPD methodology:

$$\begin{cases} \mathbf{F}_{ij}^D = -\gamma_{ij} w_D(|\mathbf{r}_{ij}|)(\mathbf{e}_{ij} \cdot \mathbf{v}_{ij})\mathbf{e}_{ij}, \\ \mathbf{F}_{ij}^R = \sigma_{ij} w_R(|\mathbf{r}_{ij}|)\mathbf{e}_{ij}\xi_{ij}, \end{cases} \quad (2.10)$$

where $\mathbf{e}_{ij} = \mathbf{r}_{ij}/|\mathbf{r}_{ij}|$ is the center-to-center unit vector, $w_D(\cdot)$ and $w_R(\cdot)$ are arbitrary weighting functions describing the scaling of each force with the inter-particle distance $|\mathbf{r}_{ij}|$, γ_{ij} and σ_{ij} are adjustable coefficients of interaction, and ξ_{ij} is a Gaussian white-noise term with

$$\mathbb{E}\xi_{ij}(t) = 0 \text{ and } \mathbb{E}(\xi_{ij}(t)\xi_{k\ell}(\tau)) = (\delta_{ik}\delta_{j\ell} + \delta_{i\ell}\delta_{jk})\delta(t - \tau), \quad (2.11)$$

where the δ terms are defined as in Section 2.2. Unlike the Brownian approach, which allows transfer of momentum from modeled particles to the background fluid, the DPD approach should conserve momentum since the entire system is modeled explicitly: in order that this hold, we require that $\xi_{ij}(t) = \xi_{ji}(t)$ for all $t > 0$.

By substituting (2.10) into (2.9) we obtain a system of stochastic differential equations (SDEs) that form the core of the DPD approach:

$$\begin{cases} d\mathbf{r}_i = \mathbf{v}_i dt \\ d\mathbf{v}_i = \frac{1}{m_i} \left(\sum_{j \neq i} \mathbf{F}_{ij}^C(\mathbf{r}_{ij}) - \sum_{j \neq i} \gamma w_D(|\mathbf{r}_{ij}|)(\mathbf{e}_{ij} \cdot \mathbf{v}_{ij})\mathbf{e}_{ij} \right) dt + \frac{1}{m_i} \sum_{j \neq i} \sigma w_R(|\mathbf{r}_{ij}|)\mathbf{e}_{ij} dW_{ij} \end{cases} \quad (2.12)$$

where for each i, j , W_{ij} is a Wiener process and $W_{ij} = W_{ji}$.

To ensure the system is properly thermostatted, it should be checked that the long-time asymptotics converge to the Gibbs canonical ensemble of particles interacting via the conservative force alone at a target temperature T – in other words, the dissipative force (which removes energy from the system) and random force (which can influence the energy in either direction) should be balanced

such that, on average, the behavior is what would be predicted for particles bouncing around at the specified temperature. Español and Warren [31] proved this result for *any* conservative force field $\mathbf{F}_{ij}^C$ given that the following two conditions, which relate the dissipative and random forces (and are thus another example of the fluctuation-dissipation theorem), are satisfied:

$$\begin{cases} w_D(|\mathbf{r}_{ij}|) = w_R^2(|\mathbf{r}_{ij}|) \\ \sigma^2 = 2\gamma k_B T. \end{cases} \quad (2.13)$$

The process of implementing the DPD method, in net, thus consists of numerically approximating the solution of (2.12) while ensuring that the conditions in (2.13) are satisfied. A schematic of a DPD polymer simulation appears in Fig. 2.1(C).

There are a number of free parameters which can be tuned to yield desired physical properties and types of interaction. The conservative force between i and j is most commonly taken to decay linearly from a specified value a_{ij} to zero at a distance of r_{ij}^c , making a_{ij} the conservative coefficient (i.e., coefficient of repulsion) and r_{ij}^c a cutoff radius. The weighting function $w_D(\cdot)$ (and thus $w_R(\cdot)$ via fluctuation-dissipation) is also usually taken to fall to zero at the cutoff radius r_{ij}^c . A final important parameter is the coefficient γ_{ij} of dissipative interaction, which can influence physical properties such as viscosity. Many physical observables have been derived in terms of these free parameters (see, e.g., [98]), and so a fluid with certain desired properties such as compressibility or viscosity can be simulated accurately. The DPD approach thus brings the flexibility of the molecular dynamics approach, which explicitly models all system components, to the longer length and time scales of the Brownian dynamics approach – for systems where nonequilibrium behavior in the fluid is a fundamental part of the system dynamics and when it is computationally feasible, the DPD approach is a natural choice for particle modeling.

With the description of the method established, it remains only to solve the equations (2.12); the following section details the approach used to approximate solutions to the DPD equations of motion in this thesis.

2.4 Numerical Solutions

Though the stochastic equations given by Eq. 2.12 are a complete description of the DPD approach, solving the equations analytically presents a practical issue: as the equations involve the differential of Wiener processes W_{ij} , there will not exist a unique solution, but a probabilistic distribution on the space of solutions. For this reason, analytic literature on DPD and other stochastic particle methods often focuses on the density $f(\mathbf{r}, \mathbf{v}, t)$ in the state space of all positions $\mathbf{r}_i$ and velocities $\mathbf{v}_i$ augmented with the time t , wherein the SDEs of Eq. 2.12 become a partial differential equation

$$\begin{aligned} \frac{\partial f}{\partial t} + \sum_i \frac{\partial}{\partial \mathbf{r}_i} \cdot (\mathbf{v}_i f) + \sum_{i,j} \frac{\partial}{\partial \mathbf{p}_i} \cdot (\mathbf{F}_{ij}^C f) &= \gamma \sum_{i,j} w_D \mathbf{e}_{ij} \frac{\partial}{\partial \mathbf{p}_i} \cdot (\mathbf{e}_{ij} \cdot \mathbf{v}_{ij} f) \\ &+ \gamma k_B T \sum_{i,j} w_D \mathbf{e}_{ij} \cdot \frac{\partial}{\partial \mathbf{p}_i} \cdot \left(\mathbf{e}_{ij} \cdot \left(\frac{\partial f}{\partial \mathbf{p}_i} - \frac{\partial f}{\partial \mathbf{p}_j} \right) \right), \end{aligned} \quad (2.14)$$

where $\mathbf{p}_i = m\mathbf{v}_i$ is the linear momentum of particle i . This PDE is known as the Fokker-Planck equation or Kolmogorov forward equation for the SDEs (2.12) and can be used to derive properties of the system such as the kinetic pressure tensor and stress tensors from conservative and dissipative forces [98]. A solution to Eq. 2.14 would yield a density f corresponding to the position $\mathbf{r}_i$ and velocity $\mathbf{v}_i$ of all particles for each time t .

In practice, the well-defined solution f of the Fokker-Planck interpretation of DPD is often infeasible to calculate, even numerically, given the massive dimensionality of the state space. For this reason, simulation via stochastic particle methods often relies on individual realizations of the system, which are understood to be representative of the distribution f of possible outcomes. Here, we step the realization forward in time with the DPD velocity-Verlet scheme: given positions $\mathbf{r}^n$ and velocities $\mathbf{v}^n$ at step n and a timestep Δt , compute the half-step velocities

$$\mathbf{v}_i^{n+1/2} = \mathbf{v}_i^n + \frac{1}{2m_i} \left(\mathbf{F}_i^C(\mathbf{r}^n) \Delta t + \mathbf{F}_i^D(\mathbf{r}^n, \mathbf{v}^n) \Delta t + \mathbf{F}_i^R(\mathbf{r}^n) \sqrt{\Delta t} \right), \quad (2.15)$$

then calculate the next step as

$$\mathbf{r}_i^{n+1} = \mathbf{r}_i^n + \mathbf{v}_i^{n+1/2} \Delta t, \quad (2.16)$$

$$\mathbf{v}_i^{n+1} = \mathbf{v}_i^{n+1/2} + \frac{1}{2m_i} \left(\mathbf{F}_i^C(\mathbf{r}^{n+1}) \Delta t + \mathbf{F}_i^D(\mathbf{r}^{n+1}, \mathbf{v}^{n+1/2}) \Delta t + \mathbf{F}_i^R(\mathbf{r}^{n+1}) \sqrt{\Delta t} \right). \quad (2.17)$$

Note that the stochastic term scales with $\sqrt{\Delta t}$ due to the standard result that t -increments of the Wiener process are Gaussian with variance t [67]. This scheme is very similar to traditional velocity-Verlet [1], with the exception that the dissipative force term $\mathbf{F}^D$ must be calculated twice per step since it is both position- and velocity-dependent. With the conservative force alone, the algorithm would be order $\mathcal{O}(\Delta t^2)$; because of the stochastic nature of the random terms, the order of the algorithm is less clear, though it has proven practical for a wide range of applications [96].

Given the soft nature of the conservative forces $\mathbf{F}_i^C$, DPD systems have been empirically shown to remain numerically stable up to much larger values of the timestep Δt than are stable for, e.g., molecular dynamics simulations with the Lennard-Jones potential which diverges as particles overlap; combined with the scaled nature of the dissipative beads, the large time steps allow the DPD approach to simulate systems on massively longer timescales than can be achieved with basic molecular dynamics interpretations. In the following chapters, we will rely on this fact heavily in order to use modern parallel computation together with the DPD approach to simulate systems which are not practical to simulate via other means.

Chapter 3

Dissipative Particle Dynamics Models of Biological Soft Matter

Soft matter, i.e., matter which is characterized by large responses to modest stresses, is present in a wide variety of biological systems of interest, especially those at the cellular level. Lipid membranes, proteins and polymers such as hemoglobin and DNA, and gels such as the cross-linked actin networks of eukaryotic cells are important components of the mechanisms underlying a number of prominent life processes. As soft matter is easily deformed and its behavior is often coupled with flows in the surrounding fluid, studying these systems experimentally can be challenging; it is for this reason that the predictions of simplified mathematical models are necessary to understand many of the behaviors of these nanoscopic structures. In this chapter, we introduce a selection of common models for polymers and membranes, then couple the models with the DPD approach of Section 2.3 to simulate several systems of experimental interest. Predictions and analysis of the DPD models will be used to answer a number of questions about mechanics which are impossible to observe directly, providing a detailed look into the behavior of biological soft matter at the nanoscale.

3.1 Polymer Models

From a mathematical perspective, linear polymers are usually characterized as curve segments with a contour length ℓ_c and negligible thickness. Parametrizing the polymer in terms of its arc length s and defining $\mathbf{r}(s)$ as the position vector of a point on the backbone of the polymer, one of the most fundamental quantities which can be defined is the persistence length

$$\ell_p = \frac{|s_1 - s_2|}{\ln \left\langle \frac{\partial \mathbf{r}}{\partial s} \Big|_{s_1} \cdot \frac{\partial \mathbf{r}}{\partial s} \Big|_{s_2} \right\rangle}, \quad (3.1)$$

calculated in terms of the correlation between the unit tangent vectors at two points s_1 and s_2 (here, angle brackets denote the time average of a fluctuating property). Intuitively, the persistence length is a measure of rigidity which corresponds to the distance two points on the polymer must be apart for the orientation of the polymer at those points to be statistically uncorrelated. A second fundamental quantity is the radius of gyration $\langle r_g \rangle$, defined as

$$\langle r_g \rangle^2 = \frac{1}{\ell_c^2} \int_{s_2=0}^{\ell_c} \int_{s_1=0}^{s_2} \left\langle |\mathbf{r}(s_1) - \mathbf{r}(s_2)|^2 \right\rangle ds_1 ds_2, \quad (3.2)$$

which is a statistical measure of the distances between all pairs of points on the polymer backbone. $\langle r_g \rangle$ is often used as a measure of linear size (and $\langle r_g \rangle^3$ as volume) of the polymer in diffusion. The following basic polymer models, adapted from Refs. [51, 66], are characterized in terms of these fundamental quantities.

3.1.1 Wormlike Chain

The wormlike chain model, also referred to as the Kratky-Porod model, considers the polymer to be an idealized flexible beam (Figure 3.1(A)) with flexural rigidity YI in terms of Young's modulus Y and the second moment I of the cross-sectional area. The persistence length ℓ_p and radius of gyration $\langle r_g \rangle$ can be derived as

$$\ell_p = \frac{YI}{k_B T}, \quad (3.3)$$

$$\langle r_g \rangle = \sqrt{\frac{\ell_c YI}{3k_B T}}, \quad (3.4)$$

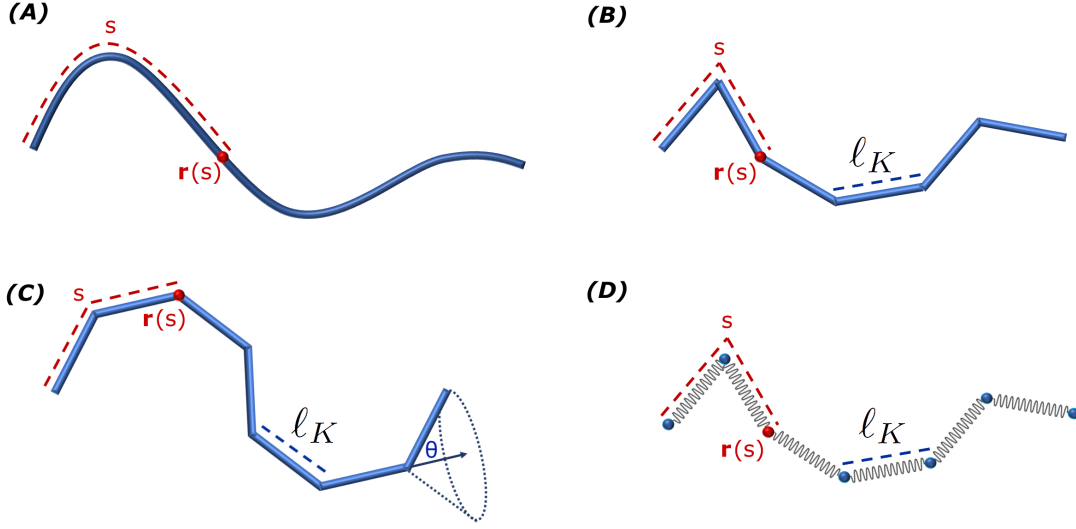


Figure 3.1: Polymer models with example point $\mathbf{r}(s)$ at arc length s . (A) Wormlike chain. (B) Freely jointed chain with Kuhn length ℓ_K . Joints between rods are idealized ball bearings. (C) Freely rotating chain with fixed colatitudinal angle θ . Links are free to rotate only on the given cone, i.e., out of the page. (D) Bead-spring chain. Unlike in (B) and (C), the links do not have fixed length ℓ_K , but are instead Hookean springs with resting length ℓ_K .

in terms of the Boltzmann constant k_B and the beam temperature T , under the assumption that the contour length ℓ_c is much larger than ℓ_p . As the wormlike chain model is based physically on the polymer angle and curvature, it is effective at predicting orientation (via the persistence length) and angles, though as with all simplified mathematical polymer models, the local structure of the polymer on small spatial scales is lost.

3.1.2 Freely Jointed Chain

The freely jointed chain, in contrast to the wormlike chain, does not make the assumption that the curve is smooth and does not describe the orientation or curvature of individual elements of the chain. Instead, the model assumes the polymer to be a series of rigid rods connected by idealized ball joints (Figure 3.1(B)), in the sense that there is no potential associated with the colatitudinal or azimuthal angles between consecutive rods. The length of a single rod is referred to as the Kuhn length ℓ_K ; though ℓ_K is not a physical length in the sense that the actual polymer does not possess

rigid sections, its loose physical interpretation is the length scale at which the orientation of the polymer segments are uncorrelated. It thus plays the role that the persistence length ℓ_p played for the wormlike chain, as ℓ_p itself is undefined for the freely jointed chain (at all points $|s_1 - s_2| > \ell_K$, $\left\langle \frac{\partial \mathbf{r}}{\partial s} \Big|_{s_1} \cdot \frac{\partial \mathbf{r}}{\partial s} \Big|_{s_2} \right\rangle = 0$). In practice, the Kuhn length is set based on experimental observations of the radius of gyration, which (under the assumption $\ell_c \gg \ell_K$) can be derived as

$$\langle r_g \rangle = \sqrt{\ell_c \ell_K / 6}. \quad (3.5)$$

While the freely jointed chain is less physically motivated than the wormlike chain model, it has the advantage of providing explicit expressions for the probability distributions associated with the distance between chain elements using the theory of random walks, making the model well suited for handling the physics of macromolecular confinement.

3.1.3 Freely Rotating Chain

The freely rotating chain is a restriction of the freely jointed chain to the case where only the azimuthal angles between rods are free, while the colatitudinal angles are fixed to a specified value θ (Figure 3.1(C)), a restriction inspired by the structure of alkane polymers [9]. As the fixed colatitudinal angles provide a correlation between rods extending beyond the Kuhn length ℓ_K , the freely rotating chain has a well-defined persistence length, which can be derived in the limit of $|s_1 - s_2| \gg \ell_K$ as

$$\ell_p = \frac{1}{2} \ell_K \frac{1 + \cos \theta}{1 - \cos \theta}. \quad (3.6)$$

Again, the Kuhn length is usually derived from observations of the radius of gyration, which has the form

$$\langle r_g \rangle = \sqrt{\frac{\ell_c \ell_K}{6} \cdot \frac{1 + \cos \theta}{1 - \cos \theta}} \quad (3.7)$$

under the usual assumption that $\ell_c \gg \ell_K$. Though the freely rotating chain is more physically motivated than the freely jointed chain given its correlations between rods at distances larger than ℓ_K , the details of the microscopic model (e.g., the angle θ) often have little effect on the macroscopic

predictions of the model for long polymers. Thus, it is common in the literature to adopt the following simpler model which is mathematically convenient.

3.1.4 Bead-Spring Chain

The bead-spring chain, or Gaussian chain, follows in the footsteps of the freely jointed chain by modeling the polymer as a series of identical segments connected by idealized ball joints. Rather than rigid rods, however, the basic segment of the bead-spring chain is a harmonic spring with spring constant $k = 3k_B T \ell_K^{-2}$ in terms of the Boltzmann constant and the temperature T (Figure 3.1(D)). The model thus suffers from the same drawback as the freely jointed chain in that the persistence length is not well defined, but has the advantage that, due to the choice of k , the distribution of bead-bead distances is Gaussian for all $|s_1 - s_2|$, not just in the limit of spatial scales much longer than ℓ_K . As with the freely jointed chain, the radius of gyration is given by

$$\langle r_g \rangle = \sqrt{\ell_c \ell_K / 6}. \quad (3.8)$$

3.2 DNA in an Array of Entropic Barriers

Due to the nanoscopic size and correspondingly short characteristic time scales of most polymers, studying the mechanics and forces underlying behaviors of interest can be experimentally challenging with traditional imaging devices. It is for this reason that mathematical models for polymers, such as those described in the previous section, and computer simulations incorporating these models, such as the dissipative particle dynamics approach, are essential to the understanding of polymer mechanics. Here, I describe a joint work with experimental collaborators Daniel Kim and Derek Stein of the Brown Department of Physics in which we applied the DPD approach to the entropically constrained motion of DNA in a nanofluidic device. Kim and Stein were specifically interested in “giant acceleration of diffusion” (GAD), which refers to a non-equilibrium phenomenon that Brownian particles exhibit in a tilted periodic potential, like the one shown later in Fig. 3.2(A)

[21, 106]; showing such a phenomenon to exist for DNA in a constrained flow is a curious clash between entropy, an inherently equilibrium phenomenon, and the non-equilibrium nature of a fluid flow. The experimental results described here are the work of Kim, in whose dissertation the physical methodology is described in great detail [64]. The experimental setup and results are reproduced below to provide context for the particle simulations, which allowed for observation and measurement of the polymer mechanics at the microscale. Further details can be found in the supplemental material associated with the joint publication of this work with Kim and Stein in the *Physical Review Letters* [65].

In the context of a tilted Brownian potential, the “acceleration” in giant acceleration of diffusion refers to the Brownian particle’s effective diffusivity as a function of the tilt, which exhibits a peak at some critical value which can exceed the self-diffusivity D_0 by orders of magnitude. Intuitively, a Brownian particle in an untilted periodic potential will largely remain in its initial “potential well,” occasionally crossing over a potential barrier into a neighboring well at a rate governed by the height of the barrier. At the opposite extreme, a Brownian particle in a potential which has been tilted too far will slide over the barriers between wells effortlessly, tumbling down the potential slope with little regard for the comparatively small fluctuations of the potential wells. In both cases, the free diffusion of the particle is comparable to its diffusivity in a uniform potential. There is a regime in between these extremes where the behavior changes significantly: at moderate tilts, the barrier to making a jump to an adjacent well is reduced, but not eliminated, by the tilt, resulting in a behavior reminiscent of a Poisson point process in which the particle makes jumps with exponential-like waiting times determined by the relationship between the barrier height and tilt. These “jumps” have the effect of accentuating the diffusion of the particle, resulting in a peak in measured diffusivity as a function of the forward driving force. This behavior is not limited to the simple case of a Brownian particle in a one-dimensional potential: GAD has been observed in such varied systems as trapped particles circling corrugated optical vortices [71], colloidal spheres moving across an undulating surface tilted in a gravitational field [80], and the rotating F_1 -ATPase protein motor

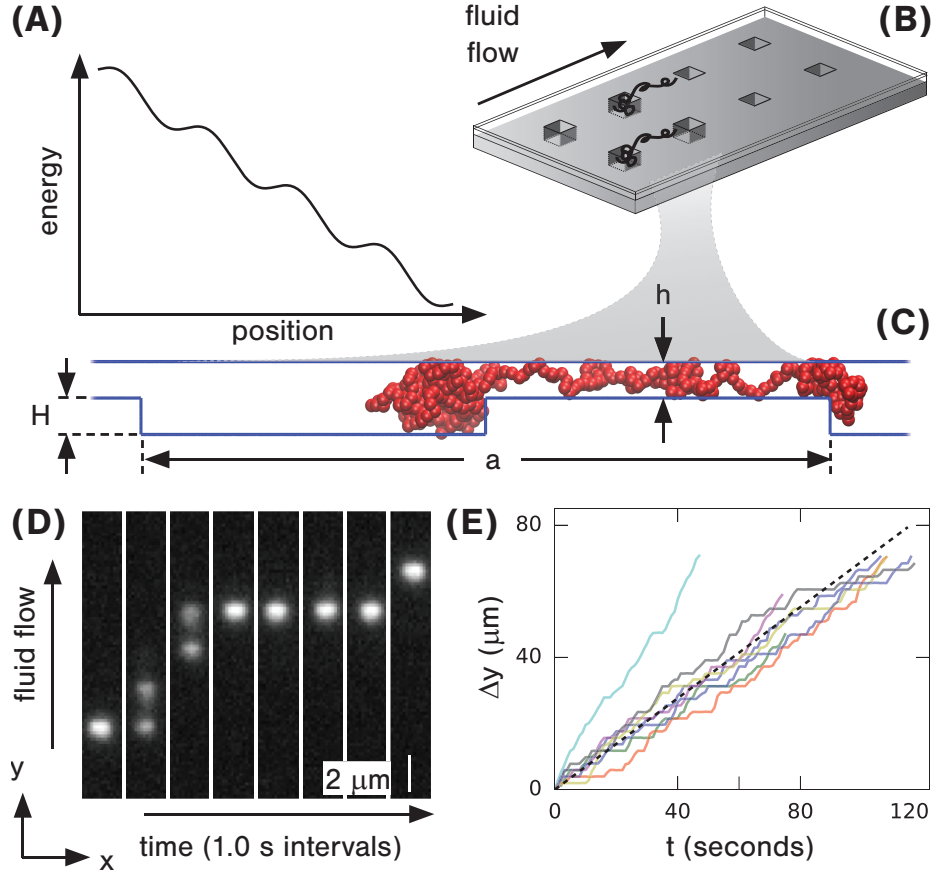


Figure 3.2: (A) Sketch of a tilted periodic energy landscape. (B) Illustration of a nanopit array embedded in a nanoslit, with DNA hopping from pit-to-pit. (C) Side-view of the nanofluidic geometry indicating h , H , and a . A simulated polymer hops between pits under an applied drive of $v = 0.05$. (D) Time-sliced fluorescence images of λ DNA climbing a column of high-barrier pits ($h = 75$ nm, $H = 124$ nm) for $\Delta p = 400$ mbar. (E) DNA trajectories from the same experiment. The dashed line indicates $\Delta y = v_p t$ ($v_p = 0.7 \mu\text{m s}^{-1}$).

[46]. Of particular interest to the experimental setup of Kim and Stein, it is theoretically predicted that Brownian particles conveyed across *entropic* barriers can also exhibit GAD [105, 12, 110], though this had not been shown experimentally. The entropic case is remarkable because entropic barriers are not fixed; they can change or even vanish as the system is driven away from equilibrium. Kim and Stein's device, detailed in the following section, shows GAD to exist in the dynamics of DNA polymers driven across arrays of *entropic* barriers in a nanofluidic structure.

It is already known that a polymer's configurational entropy varies within a nanofluidic device with an inner topography like the one in Fig. 3.2(B, C) [44, 107]. The nanofluidic slit features an

array of depressions called nanopits which allow a compressed polymer to expand; as a result, the polymer experiences entropic barriers at the pit edges, as more configurations are available to a polymer inside the relatively deep pits than inside the shallow slit. The entropic penalty for entering the slit is especially significant when the slit height is much smaller than the polymer’s radius of gyration $\langle r_g \rangle$, which corresponds loosely to the effective size of the polymer in free diffusion. It should be noted that DNA can fully explore the vertical space inside such devices within milliseconds [26], which is much shorter than its typical dwell time in a pit. Previous theoretical studies highlighted the entropic nature of the barriers polymers encounter in similar geometries [95, 93]. Furthermore, in a pressure-driven flow, DNA hops from pit to pit with exponentially distributed dwell times in the pits and a pressure-dependent mobility; this experimentally observed behavior is consistent with thermally activated transport across entropic barriers [24, 136].

Previous studies focused on DNA in nanotopographies have largely focused on the statistics [94, 22, 107, 128, 132, 126, 75, 140] and mobility [44, 45, 24, 140, 23] of the polymer and interpreted the results in terms of the free energy in equilibrium. Here, by contrast, we probe DNA’s diffusivity and its effective energy landscape as it is driven *away* from equilibrium.

3.2.1 Experimental Setup and Results

Kim and Stein created arrays of nanopits within nanofluidic slits on fused silica chips, similar to devices described previously in the literature [24, 114]; the appearance of the device, with a side-view cross section, is shown in Figures 3.2(B,C). The square pits had $1.05 \pm 0.05 \mu\text{m}$ sides and were linearly arrayed with periodicity $a = 2.0 \mu\text{m}$. The depth of a pit relative to the slit controls the height of the entropic barrier; accordingly, “high” barriers were obtained in one device with slit height $h = 75 \text{ nm}$ and pit depth $H = 124 \text{ nm}$, and a second device with dimensions $h = 88 \text{ nm}$ and $H = 79 \text{ nm}$ had “low” barriers. The terms “low” and “high” are relative comparisons; in both devices, the barriers are higher than the thermal energy, $k_B T$, so a pit traps DNA nearly indefinitely without a driving force applied [24]. h and H were measured using a stylus profilometer and an

atomic force microscope, respectively. The slits were 3 mm long and 160 μm wide.

λ DNA molecules (New England Biolabs), whose contour length is $\ell_c = 16.5 \mu\text{m}$ and radius of gyration is $\langle r_g \rangle = 0.73 \mu\text{m}$ [118], were fluorescently stained with YOYO-1 at a 10:1 base-pair-to-dye ratio and suspended in 20 mM Tris-EDTA buffer titrated to $p\text{H} = 8.0$ with HCl, with 3% β mercaptoethanol added. Molecules were driven through the devices by a pressure difference Δp supplied by an air pump. Individual molecules were imaged by epifluorescence microscopy using a 100 $\times$, 1.49 NA oil-immersion objective (Nikon) and an EMCCD camera (Andor iXon). The field of view was $80 \mu\text{m} \times 80 \mu\text{m}$. The excitation light was shuttered with a frequency of 1 Hz and a 50 ms exposure time to minimize photodamage to the DNA. Further details on the experimental methods are described in detail in Ref. [24].

Figure 1(D) shows an example of a DNA molecule moving up a column of pits. At any given time, the bulk of the polymer is contained within one or more nanopits, and so the fluorescence captured by imaging devices is largely constrained to the grid matching the nanopit topography. Of particular note is the fact that the velocity of the DNA is uneven: consecutive time slices can show the same molecule to make several jumps across pits and then remain trapped for equally long in a single pit immediately thereafter. To explicitly quantify this progression, we tracked the DNA center-of-mass position along the flow direction y as a function of time t by performing a weighted spatial averaging of the fluorescence intensity. The integrated fluorescence intensity was also used to identify DNA fragments and concatemers, which were removed from the dataset for consistency. Figure 3.2(E) shows typical trajectories $y(t)$, along with a dashed line loosely corresponding to the mean DNA velocity across the pits, hereafter referred to as v_p .

We tested driving the Poiseuille flow with a pressure difference Δp from 50 to 700 mbar. At each Δp , we recorded 50 to 80 properly sized molecules crossing the pits; more than 2000 total pit-to-pit hops were typically observed in order to obtain good estimates of the mean velocity and the diffusion coefficient. 12 to 20 molecules were also recorded crossing pit-free regions of the slit, where the flow velocity is proportional to Δp . The mean DNA velocity in the slit, hereafter referred to as v_s , was

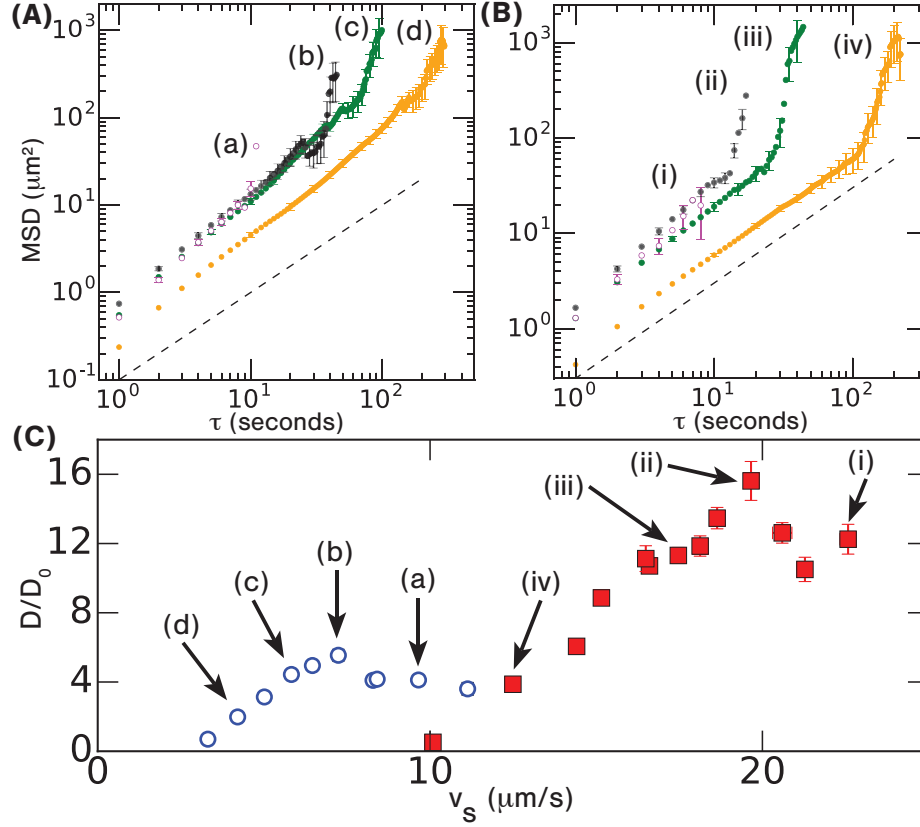


Figure 3.3: Dependence of MSD on τ in (A) the device with low barriers for $\Delta p =$ (a) 250, (b) 140, (c) 90, and (d) 60 mbar; and in (B) the device with high barriers for $\Delta p =$ (i) 750, (ii) 650, (iii) 555, and (iv) 400 mbar. The corresponding D/D_0 and v_s are indicated in (C). The dashed line has a slope of one. (C) Dependence of D/D_0 on v_s for low (blue dots) and high (red squares) barriers with $\tau = 2$ s. Error bars are the standard error from a bootstrap analysis of 200 resamplings of the full dataset.

used to quantify the driving force rather than Δp because we found that the latter could be affected by small changes in the pneumatic system from one experiment to the next.

To quantify the DNA diffusivity, we first evaluated the drift-subtracted mean square displacement (MSD) for each ensemble of trajectories

$$\text{MSD} = \langle (\Delta y(\tau) - v_p \tau)^2 \rangle, \quad (3.9)$$

where $\Delta y(\tau)$ is a molecule's y displacement in the time interval τ , $\langle \dots \rangle$ indicates an ensemble average, and v_p is the mean DNA velocity across the pits. (See the supplementary material of [65], which also references [5, 103], for details and comparisons with alternative methods of analysis.) The effective

diffusivity D is related to MSD by

$$D \equiv \lim_{\tau \rightarrow \infty} \text{MSD}/2\tau. \quad (3.10)$$

Figures 3.3(A, B) show the growth of MSD with τ in devices with “low” and “high” barriers, respectively, for four choices of Δp . On the log-log plot, MSD increased linearly with τ with a slope near one. For each Δp , there was an upper experimental limit to τ , beyond which MSD rose sharply; that limit corresponded to the time it took the fastest molecules to exit the camera’s field of view, which decreased as Δp increased. To compare measurements on an equal basis, we evaluated D using Eq. 3.10 with $\tau = 2$ s. We verified that velocity autocorrelations are negligible for $\tau = 2$ s and that D is relatively insensitive to τ (see supplementary material of [65]). The self-diffusion of DNA in a slit with $\Delta p = 0$ could be observed for long times as the molecules were largely stationary, so we obtained D_0 from the slope of MSD versus τ for the $\Delta p = 0$ case; we found $D_0 = 0.084 \mu\text{m}^2 \text{s}^{-1}$ in the $h = 88$ nm slit and $D_0 = 0.067 \mu\text{m}^2 \text{s}^{-1}$ in the $h = 75$ nm slit.

Figure 3.3(C) shows the dependence of D/D_0 on v_s , which is a proxy for the applied pressure, for both high- and low-barrier devices. In both cases, D/D_0 increased with v_s from zero to a peak, then began to decrease. The peak occurred at $v_s = 7.2 \mu\text{m s}^{-1}$ in the device with low barriers and at $v_s = 19.7 \mu\text{m s}^{-1}$ in the device with high barriers. The height of the D/D_0 peak also depended on the nanotopography; it reached ≈ 5.5 in the device with low barriers and ≈ 15.5 in the one with high barriers.

3.2.2 Giant Acceleration of Diffusion (GAD)

A non-monotonic dependence of D on driving force (i.e., tilt) is a hallmark of *giant acceleration of diffusion* (GAD) [106]. The basic theory of GAD is based on the Langevin equation for an overdamped Brownian particle in a periodic potential, $G(y) = G(y + a)$, with a constant tilt force F :

$$\zeta \frac{dy}{dt} = -\frac{dG(y)}{dy} + F + f(t), \quad (3.11)$$

where ζ is the drag coefficient and $f(t)$ is the usual random force from thermal fluctuations, defined by the moments $\langle f(t) \rangle = 0$ and $\langle f(t)f(t') \rangle = 2\zeta k_B T \delta(t-t')$, where $\delta(t-t')$ is Dirac's delta function centered at t' [147]. Reimann et al. showed that the dynamics of such a system are governed by the form of $G(y)$ near the critical point, which is the dynamical bottleneck [106]. They expanded $G(y)$ about the critical point, chosen to be $y_c = 0$, to obtain the small y behavior

$$G(y) - yF = -\mu y|y|^{q-1} - y\epsilon, \quad (3.12)$$

where $q > 0$, $\mu > 0$ characterizes the gradient of the potential force near y_c , and $\epsilon \equiv F - F_c$.

Equation 3.12 leads to the following expression for D/D_0 [106]

$$\frac{D}{D_0} = \left(\frac{a^q \mu}{k_B T} \right)^{2/q} \frac{\int_{-\infty}^{\infty} dy K^2(y, \gamma) K(-y, \gamma)}{\left[\int_{-\infty}^{\infty} dy K(y, \gamma) \right]^3}, \quad (3.13)$$

where $K(y, \gamma) \equiv \int_0^{\infty} dy' e^{-y|y|^{q-1} + (y-y')|y-y'|^{q-1} - \gamma y'}$ and $\gamma \equiv \epsilon / [\mu^{1/q} (k_B T)^{1-1/q}]$. D is a peaked function of F , with the peak occurring at $F \approx F_c$; this peak corresponds to particles alternating between trapped and freely running states. Namely, when F is very low, particles become trapped between energy barriers and D is suppressed, approaching zero when the barrier height grows large compared with the thermal energy $k_B T$. Increasing F lowers the forward energy barrier and facilitates downhill motion. For relatively low values of the tilt, the particle transitions stochastically between energy minima, but as the tilt increases, the motion evolves into a more deterministic, freely running mode where the particle does not pause at the energy minima of the periodic potential. The critical point in between, i.e., at the critical driving force F_c , is where the barrier vanishes, particles begin to run freely, and D approaches D_0 .

The energy landscape for a DNA molecule in a nanopit array differs fundamentally from the model of Reimann *et al.* because the former is *entropic* and the fluid flow is an out-of-equilibrium, non-conservative driving force. Nonetheless, it is commonly assumed in the literature on polymer physics that the out-of-equilibrium, entropic system can be accurately described by a fixed energy landscape based on the equilibrium situation. The dynamics of DNA in nanopit arrays can be

mapped onto GAD theory by casting the effective free energy landscape for DNA, $\Psi(y)$, in the role of $G(y) - yF$. The entropic barriers give a periodic contribution $S(y)$ to $\Psi(y)$. The viscous driving force is non-conservative, but we can define an effective potential $V(y)$ for it through the work required to translate a molecule adiabatically against the viscous force [24], which is proportional to v_s (our stand-in for the applied pressure). It can be written as the difference of two terms, $V(y) = U(y)v_s - y\zeta^*v_s$, where $U(y)$ is a periodic function related to the periodically varying flow speed inside the nanopit array and ζ^* is an effective drag that quantifies the net force downstream. Combining the contributions, $\Psi(y)$ becomes

$$\Psi(y) = S(y) + U(y)v_s - y\zeta^*v_s. \quad (3.14)$$

For this to be well defined, the lateral relaxation of the polymer must be fast compared with its forward motion across the barriers; this property has been shown to hold in similar nanofluidic channels, where DNA samples the vertical configuration space on the order of milliseconds [26] (in contrast, the molecules in our experiment move between pits on the order of seconds). Equation 3.14 describes the competition between entropy and enthalpy. Inertial effects are absent because the Reynolds number in nanofluidic systems is extremely low. With these definitions, DNA obeys Eq. 3.11 with $G(y) = S(y) + U(y)v_s$ and $F = \zeta^*v_s$. Furthermore, Eq. 3.13 gives the effective diffusivity of DNA with $\mu = -\frac{1}{6}(S'''(0) + U'''(0)v_s)$ and $\epsilon = (\zeta^* - U'(0))(v_s - v_c)$ (primes indicate derivatives; see [65] for the derivation, which includes references [5, 18]).

We fit Eq. 3.13 to D/D_0 using μ and F_c as fitting parameters and assuming $q = 3$. The device with high barriers obtained $\mu = 1.00 k_B T a^{-3}$ and $F_c = 0.18$ pN, while the device with low barriers obtained $\mu = 0.21 k_B T a^{-3}$ and $F_c = 0.08$ pN. Figure 3.4 shows D/D_0 from both devices as a function of the rescaled slit velocity $v_s/v_c - 1$, where v_c is the mean slit speed at peak diffusion. Also shown is Eq. 3.13 with the fit values of μ and F_c , which closely matches the observed data.

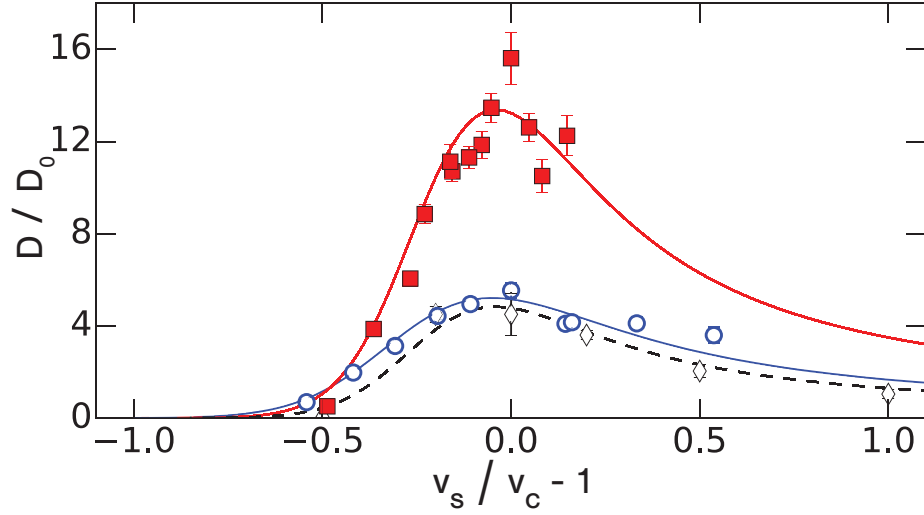


Figure 3.4: Dependence of D/D_0 on $v_s/v_c - 1$ in devices with high (red squares) and low (blue dots) barriers, and for simulation data (diamonds). The lines are fits of Eq. 3.13 to the corresponding data, with fit parameters μ and F_c .

3.2.3 DPD Simulation of the Diffusion Curve

Though experimental results show the existence of giant acceleration of diffusion for λ DNA in the nanofluidic device described in Section 3.2.1, there remain a number of questions which cannot be answered strictly from the observed data. In particular, the extent to which the “force landscape” experienced by the DNA molecules actually maps to the tilted periodic potential and critical tilt of the Brownian particle of Reimann et al. is as of yet unclear beyond heuristic argument, given that the pico-scale forces and potentials in the nanofluidic device cannot be measured with sufficient precision. Changes in the behavior of the DNA molecules below, at, and above the critical applied pressure are also only observable at a 1 Hz sampling rate and at an image resolution which cannot distinguish individual strands of DNA from within the self-intertwined configuration of the polymer diffusing in a nanopit. Here, we use the DPD approach of Section 2.3 in conjunction with the bead-spring polymer model described in Section 3.1 to simulate the nanofluidic system at significantly higher resolution, permitting a closer look at the piconewton forces at play.

For computational efficiency, we chose to model a section of one column of the nanofluidic device comprising three consecutive pits in the slit, partly shown in Fig. 3.2(C). The $80 \times 240 \times 8$

simulation box corresponds to three pits of a device with $1.0\ \mu\text{m}$ wide pits, periodicity $a = 2.0\ \mu\text{m}$, and $h = H = 90\ \text{nm}$ using a scaling factor of 40 LJ units to $1\ \mu\text{m}$ (see [59, 104] for more details on the dimensionless units typically used in DPD simulations, which are scalable to real units by physical arguments), thereby most closely resembling the “low” barrier physical device. The region within the channel was filled with dissipative water particles using the standard parameters established by Groot and Warren [40], while the silica plates were modeled with partial-slip boundary conditions according to the method of Pivkin and Karniadakis, which couples a hard reflecting boundary with a frozen layer of DPD water particles just outside the boundary and has been shown to generate an accurate velocity profile for a pressure-driven flow in a channel [99]. The hard reflecting boundary used for this simulation was imposed via a 12/6 Lennard-Jones potential in the immediate vicinity of the silica plate, given below as a function of distance r from the boundary:

$$E_{\text{LJ}}(r) = 4\epsilon \left[\left(\frac{\sigma}{r} \right)^{12} - \left(\frac{\sigma}{r} \right)^6 \right], r < r_c, \quad (3.15)$$

where $\epsilon = 0.001$ is a scaling factor, $\sigma = 0.3$ is the effective radius of the force, and $r_c = 0.5$ is a cutoff radius. Given the small magnitude of ϵ , the potential is negligible for $r > \sigma$, but quickly blows up to infinity for $r < \sigma$, resulting in a sharp reflective boundary. Boundary conditions in the y direction, i.e., the direction of flow, were periodic to prevent fluid from escaping the system and to allow the DNA molecule to wrap back to the beginning of the channel after traversing the length of the simulation box.

The λ DNA itself was modeled as a bead-spring polymer (as in Section 3.1.4) with 660 beads. Each individual bead was a dissipative particle with conservative repulsion coefficient $a_{ij} = 25.0$ and dissipative coefficient γ_{ij} matching that of the surrounding water; springs between beads had stiffness $k = 500$ and a resting length of 1 unit, making the resting total length of the polymer 660 units, or $16.5\ \mu\text{m}$, matching the contour length ℓ_c of the DNA used in experiment. In order to control the effective size of the simulated polymer, an additional harmonic bending energy was

imposed on colatitudinal angles formed by consecutive bonds:

$$E_{\text{harmonic}}(r) = K (\theta - \theta_0)^2, \quad (3.16)$$

where $\theta_0 = 180^\circ$ is the neutral angle and $K = 0.5$ is the amplitude. The specific value of K was chosen by simulating the polymer in a channel with no pits, time-averaging its radius of gyration after an equilibration period, and matching it to the $\langle r_g \rangle \approx 1.05 \mu\text{m}$ of λ DNA in the channel; an example histogram of the time-dependent simulated radius of gyration can be seen in Figure 3.5.

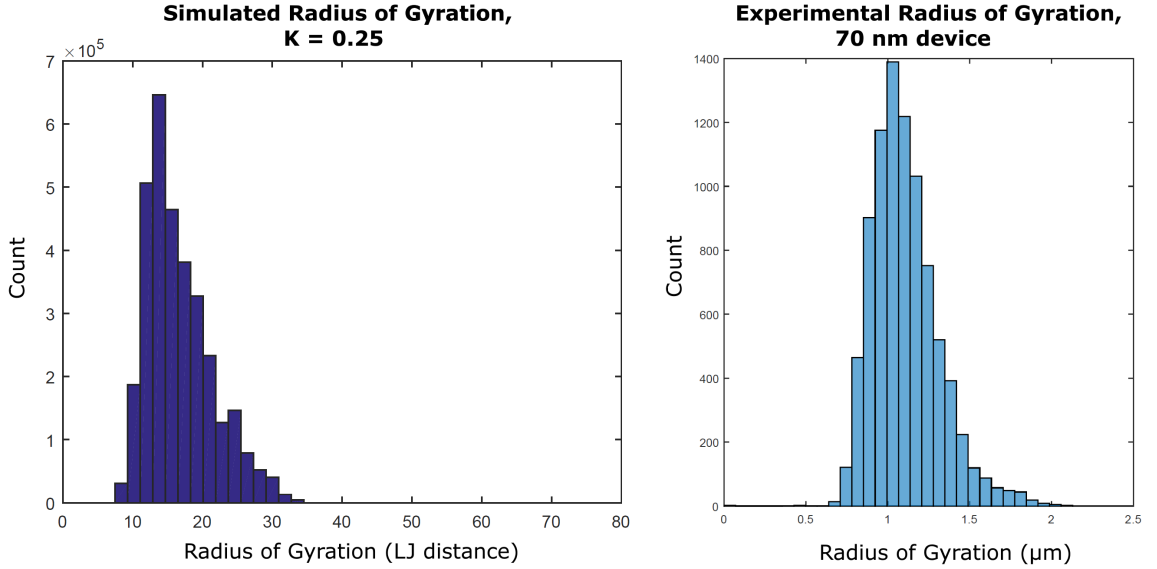


Figure 3.5: Histogram of computed $\langle r_g \rangle$ for 5 million steps of simulation with example bending amplitude $K = 0.25$ in pit-free channel (left) and for Apr. 2, 2015 dataset of experiment with low barriers (right). The mean $\langle r_g \rangle$ from simulation was matched to the low barrier $\langle r_g \rangle$ by adjusting K and using the spatial scaling of 40 LJ units to 1 μm which matched the device geometry.

The simulated DNA thus had two major differences from the standard bead-spring polymer model: first, in addition to consecutive beads interacting via spring potentials, beads at all distances additionally interacted via DPD potentials, providing a notion of volume exclusion lacked by all four basic mathematical models in Section 3.1; and second, the addition of a harmonic bending energy to angles between consecutive bonds ensures nonzero correlation between the orientation of the polymer for all separations $|s_1 - s_2|$, and so the persistence length ℓ_p is well defined. As the vast majority of simulated particles correspond to fluid in the channel, the additional computational

expense of these refinements to the polymer model is negligible.

Finally, a spatially uniform, time-varying force was applied to fluid particles in the upper half of the device (the slit) to create the flow. This force corresponds to the applied pressure Δp , which maintains a constant flow velocity in the channel by counteracting the dissipation of kinetic energy due to friction against the silica plates; for the simulations, a target mean fluid velocity v was specified, with the applied force in each step proportional to the difference between v and the measured mean fluid velocity in the slit. Simulations were evolved in time using the velocity-Verlet scheme described in Section 2.4 with a timestep $\Delta t = 0.001$ for between 50 and 200 million steps.

Simulated data were obtained for a range of flow velocities v from $v = 0.01$ to $v = 0.2$; at the lower end of this range, the polymer remained trapped in the pit for the duration of the simulation, while for high values of v the polymer streamed across pits with little regard for the underlying topography, corroborating the experimental observations and parallel to GAD for the tilted Brownian particle. In addition to the polymer's center-of-mass position, sampled at regular intervals, we also obtained fine-scale data on the linear extent, velocity, and radius of gyration of the DNA, as well as all forces exerted on the DNA by the fluid and walls. The center-of-mass position was used to compute the MSD, determine the diffusion D , and fit the data to Eq. 3.13 as before. The center-of-mass sampling rate, corresponding to τ in the previous analysis, was chosen so that the average number of pit-to-pit hops per τ was comparable to the experimental data. Figure 3.4, which was previously used to display the experimental results for high- and low-barrier devices, also shows D/D_0 and the fit of Eq. 3.13 from the simulations; the data were found to closely match the observed behavior of DNA in the device with low barriers, confirming that the basic physical assumptions of the experiment are sufficient to give rise to GAD.

3.2.4 The DPD Force Landscape

Beyond simply using the DPD model to reaffirm the results of the experiments, we also used measurements from the DPD model to examine the effective force landscape experienced by the DNA

center-of-mass as a function of its position. This implicitly treats the polymer as a single particle located at its center of mass, thereby incorporating the entropic barriers governing its diffusion into a potential term with periodicity matching that of the device. Under this assumption, the force on the DNA can be mapped to a Langevin equation with energy landscape as in Eq. 3.14, where the total force on the simulated polymer $F_{\text{total}}(y, t)$ is given as

$$F_{\text{total}}(y, t) \approx -\zeta \frac{dy}{dt} - \frac{d\Psi(y)}{dy} + f(t) \quad (3.17)$$

in terms of the viscous drag force, the gradient of a periodic energy landscape experienced by the center-of-mass, and the random thermal force. In both the Langevin and DPD settings, the thermal random force scales with a mean-zero Gaussian, and so the term vanishes in an ensemble average:

$$-\frac{d\Psi(y)}{dy} \approx \left\langle \zeta \frac{dy}{dt} + F_{\text{total}}(y, t) \right\rangle_t. \quad (3.18)$$

Namely, the contribution to total force from the periodic energy landscape, which incorporates both entropic effects and the external force due to the fluid, can be approximated as the ensemble average of the recorded total force less the viscous force. As $F_{\text{total}}(y, t)$ and $\frac{dy}{dt}$ are easy to measure in the simulations, it remains only to approximate the viscous coefficient ζ . An additional consequence of treating the polymer as a point mass is that ζ must account for the configurational aspect of the DNA; just as air resistance varies with the cross-section and contours of vehicles, fluid drag for the polymer will vary depending on the DNA's typical extent and orientation. As ζ will only be probed in an ensemble average, we calculate only an average value of ζ over the distribution of configurations experienced by the polymer for long times, though even this value will depend on v (DNA which is trapped in a pit will have a different set of typical configurations than DNA freely streaming through the slit).

We isolated ζ for each choice of flow velocity v by calculating the velocity autocorrelation function $\langle v(t)v(t+\tau) \rangle$, i.e., the average correlation between the polymer's center-of-mass velocity at two times separated by an interval τ . The following result holds for a particle governed by the one-dimensional Langevin equation with periodic forcing term:

Theorem 3.2.1. For $\tau \ll 2\zeta k_B T / |\Psi'|^2$, the velocity autocorrelation function of the one-dimensional Langevin equation in a periodic energy landscape Ψ

$$m \frac{dv}{dt} = -\zeta v + \eta(t) - \frac{d\Psi}{dy} \quad (3.19)$$

is approximated as

$$\langle v(t)v(t+\tau) \rangle \approx \frac{k_B T}{m} e^{-\frac{\zeta}{m}\tau}. \quad (3.20)$$

The proof requires the following lemma:

Lemma 3.2.2. Over timescales $\tau \ll 2\zeta k_B T / |\Psi'|^2$, Eq. 3.19 can be approximated as

$$m \frac{dv}{dt} \approx -\zeta v + \eta(t). \quad (3.21)$$

Proof of Lemma. Writing out the force term in terms of the mass m and acceleration of the particle, Eq. 3.19 becomes

$$m \frac{d^2 y}{dt^2} = -\zeta \frac{dy}{dt} + \eta(t) - \frac{d\Psi}{dy}. \quad (3.22)$$

Define a rescaling $\lambda \bar{t} \equiv t$, where $\bar{t}$ is $\mathcal{O}(1)$, and rewrite Eq. 3.22 in terms of the new variable:

$$\frac{1}{\lambda^2} m \frac{d^2 y}{d\bar{t}^2} = -\frac{1}{\lambda} \zeta \frac{dy}{d\bar{t}} + \eta(\lambda \bar{t}) - \frac{d\Psi}{dy}. \quad (3.23)$$

η also scales with λ : the Langevin random force η has the property $\langle \eta(t)\eta(t') \rangle = 2\zeta k_B T \delta(t-t')$, and the delta function scales as $\lambda \langle \eta(\lambda \bar{t})\eta(\lambda \bar{t}) \rangle = \langle \eta(\bar{t})\eta(\bar{t}) \rangle$. It follows that

$$\frac{1}{\lambda^2} m \frac{d^2 y}{d\bar{t}^2} = -\frac{1}{\lambda} \zeta \frac{dy}{d\bar{t}} + \frac{1}{\sqrt{\lambda}} \eta(\bar{t}) - \frac{d\Psi}{dy}. \quad (3.24)$$

It is clear from Eq. 3.24 that the magnitude of potential force $|d\Psi/dy|$ becomes negligible compared with the magnitude of the stochastic force $\sqrt{2\zeta k_B T / \lambda}$ for sufficiently short times, i.e., when $\lambda \ll 2\zeta k_B T / |\Psi'|^2$, on which timescale the dynamics can be approximated as potential free. $\square$

Proof of Theorem. Using Lemma 3.2.2, we examine the potential-free Langevin equation in terms of the velocity v :

$$m \frac{dv}{dt} \approx -\zeta v + \eta(t). \quad (3.25)$$

Multiplying through by $v(0)$ and taking the ensemble average:

$$m \left\langle \frac{dv(s)}{ds} \right|_t v(0) \rangle \approx -\zeta \langle v(t)v(0) \rangle + v(0) \langle \eta(t) \rangle. \quad (3.26)$$

Since $\langle \eta(t) \rangle = 0$, we have

$$m \left\langle \frac{dv(s)}{ds} \right|_t v(0) \rangle \approx -\zeta \langle v(t)v(0) \rangle. \quad (3.27)$$

Defining a new function $g(s) = \langle v(0)v(s) \rangle$, this equation has the form $g'(t) = -(\zeta/m)g(t)$, which has the well-known solution for $t > 0$

$$g(t) = g(0)e^{-\frac{\zeta}{m}t}. \quad (3.28)$$

It follows that

$$\langle v(0)v(t) \rangle = \langle v(0)^2 \rangle e^{-\frac{\zeta}{m}t}. \quad (3.29)$$

Finally, by the equipartition theorem, we have $\langle \frac{1}{2}mv(0)^2 \rangle = \frac{1}{2}k_B T$, and so

$$\langle v(0)v(t) \rangle = \frac{k_B T}{m} e^{-\frac{\zeta}{m}t}. \quad (3.30)$$

Since ensemble averages are independent of fixed time shifts, it follows that

$$\langle v(t)v(t+\tau) \rangle = \frac{k_B T}{m} e^{-\frac{\zeta}{m}\tau}. \quad (3.31)$$

□

For each simulation, the velocity autocorrelation function was calculated using recorded data of the polymer velocity dy/dt . Figure 3.6 shows the log-scale velocity autocorrelation functions for simulated polymers with fluid velocities $v = 0.04, 0.05$, and 0.06 , all in simulation units, which corresponds to velocities just below, at, and above peak diffusivity, respectively. From Theorem 3.2.1, the short-time autocorrelation should decay with $-\zeta/m$ on a log scale; values of the viscous coefficient ζ can thus be obtained by measuring the slope and multiplying by $-m$.

In Figure 3.6(A), the log autocorrelation functions decrease linearly from $\tau \approx 0.1$ onward, corresponding to the small-time scaling of the log autocorrelation with $-\zeta/m$. In Figure 3.6(B), the

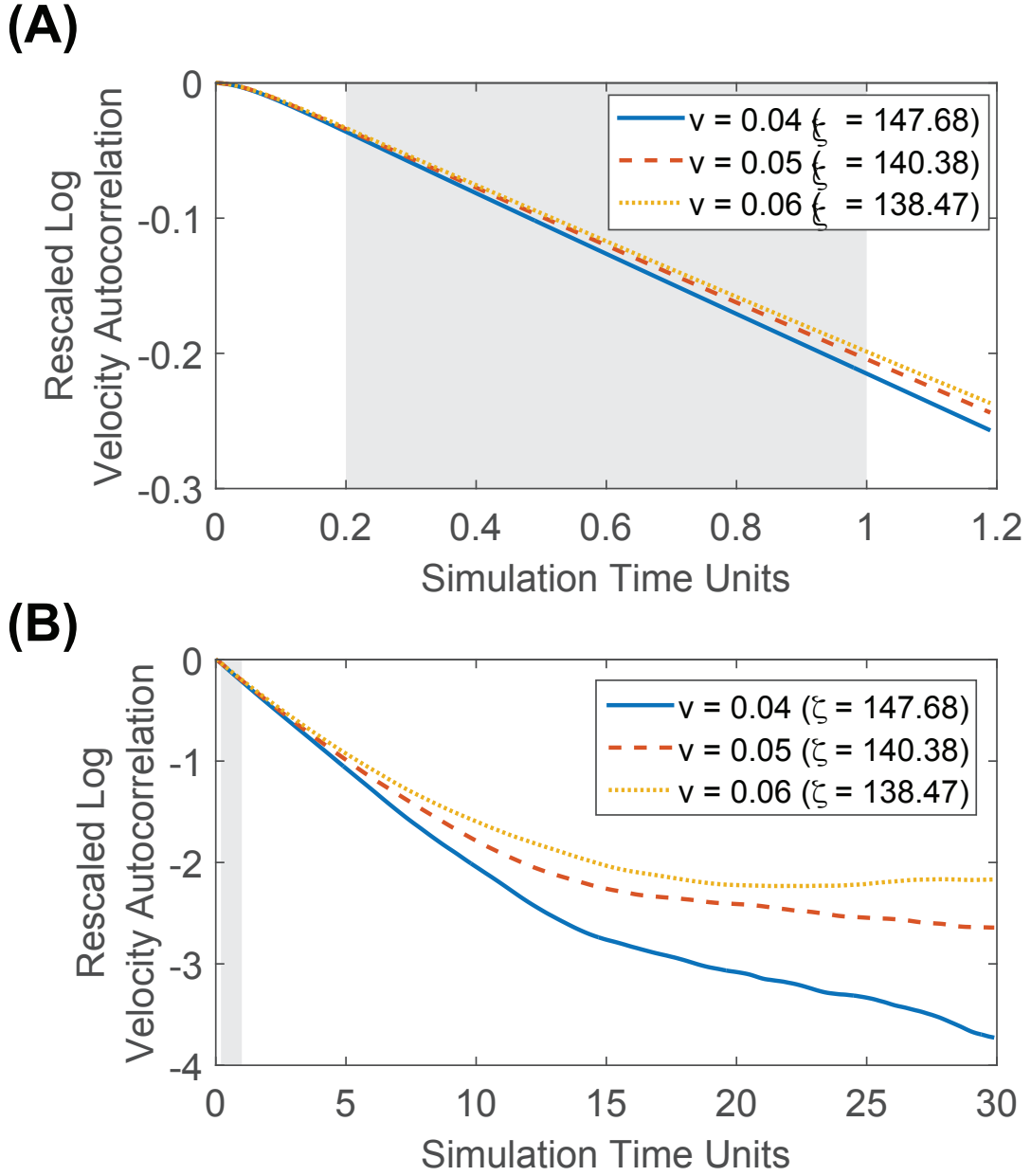


Figure 3.6: Normalized log velocity autocorrelation functions for the simulated polymer with $v = 0.04, 0.05$, and 0.06 plotted on (a) short and (b) long timescales in simulation time units. The viscous drag coefficients were obtained from slopes of log-linear fits of the data in the range $0.2 \leq \tau \leq 1.0$ simulation units, shown shaded in grey.

long-time behavior shows marked departure from linearity as τ reaches around 10 to 15, corresponding to the limit $\tau \approx 2\zeta k_B T / |\Psi'|^2$. We obtained ζ for each v from the slope of the data in the range 0.2 to 1.0 simulation time units, shown in gray. A small but positive lower bound was necessary to avoid the ultra-fine time scale on which neighboring particles interact, while the upper bound was chosen to be an order of magnitude less than $2\zeta k_B T / |\Psi'|^2$. We found $\zeta = 147.7, 140.4$, and 138.5 for $v = 0.04, 0.05$, and 0.06 , respectively. These values, in conjunction with $k_B T = 1$ (LJ units) and $|\Psi'| \approx 5$, yield a limiting time scale of $\tau \approx 12$, in good agreement with the simulation results.

With ζ known, it is possible to calculate the force landscape $-d\Psi(y)/dy$ due to the periodic potential using Eq. 3.18. For each simulation, we binned data (specifically position, velocity, and total force) by center-of-mass y position using adaptive bins which equalized the number of data in each bin, yielding discretized approximations to the ensembles of F_{total} and dy/dt as functions of position y . Using the calculated values of ζ and averaging over values by bin, we generated discretized potential force landscapes according to Eq. 3.18.

Figure 3.7(A) shows a simulated polymer trapped near the critical point and Fig. 3.7(B) plots the potential force landscapes. The landscapes are largely positive, i.e., the gradient of the periodic potential usually results in a forward force (the driving force of the fluid) carrying the particle forward, which is partially balanced by the constant dissipation of kinetic energy due to viscosity and friction against the silica plates. For the subcritical $v = 0.04$, a portion of the potential force landscape is negative, i.e., opposing the direction of motion: this corresponds to the entropic barrier which prevents the polymer from streaming freely. As v increases, the force minimum shifts upward, crossing through zero – in particular, the potential force minimum was closest to zero when the diffusivity reached its peak at $v = 0.05$. For supercritical flow velocities, the entropic barrier is eliminated, and the polymer transitions toward the freely streaming regime wherein its diffusivity asymptotically recedes to D_0 . The minima of all three landscapes occurred around $y \approx 55$, where the DNA molecule would become trapped – pushed up against the forward wall of the pit – before escaping.

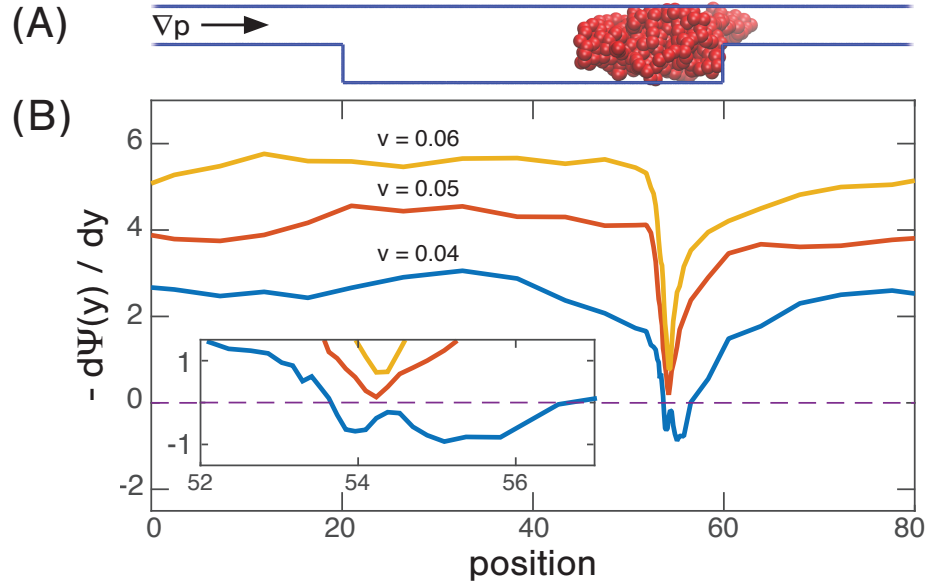


Figure 3.7: (A) Side view of a simulated polymer inside a nanofluidic device, near the critical point, with $v = 0.05$. (B) Effective force landscapes $-d\Psi(y)/dy$ for $v = 0.04, 0.05$, and 0.06 (labeled curves), computed from Eq. 3.18. The position axis is aligned with the nanofluidic geometry in (A). The inset shows an expanded view of the force landscape around the critical point. The dashed line indicates zero force.

The DPD-derived force landscape thus provides a strong argument that the analogy to a Brownian particle in a tilted potential and Langevin approximation are appropriate: as in the case of GAD for the Brownian particle, the observed peak in diffusivity occurs just as the periodic potential reaches the critical tilt where the gradient no longer crosses through zero. The shape of the force landscapes in Fig. 3.7 also change noticeably as v increases, with a sharper barrier developing near the critical point; these effects hint at a changing entropic barrier as the polymer is driven further from equilibrium.

In the theory of Reimann et al., the dynamical bottleneck also governs the drift velocity [106]

$$v_p = \frac{1 - e^{-aF/k_B T}}{\int_{-\infty}^{\infty} dy K(y, \gamma)}. \quad (3.32)$$

As a final test, we compare in Fig. 3.8 the mean drift velocity v_p of the DNA from the experiments and simulations with Eq. 3.32, using the previously fit values of μ and F_c . The data are rescaled by $v_0 = v_s/(1 + H/2h)$, the approximate limiting velocity, and plotted against F/F_c . The results

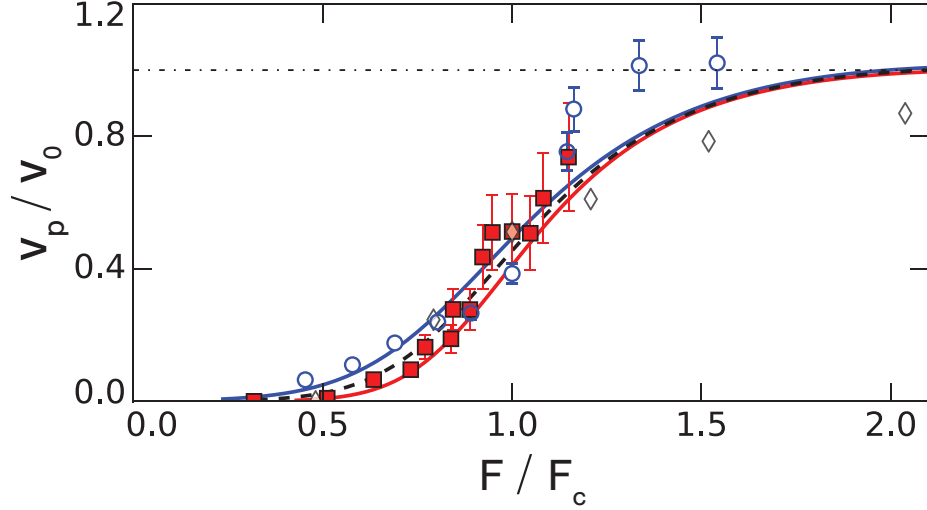


Figure 3.8: Dependence of v_p/v_0 on F/F_c in devices with high (red squares) and low (blue dots) barriers, and simulation data (black diamonds). The corresponding lines plot Eq. 3.32 using the fit values of μ and F_c . The dot-dashed line shows $v_p/v_0 = 1$.

shows good agreement between the model and both experimental and simulation data, especially in the region near the critical force F_c , though we note that Eq. 3.32 seems to become less accurate at high v_s .

3.2.5 Discovery of Design Principles Through DPD

With the agreement of the DPD model to the observed experimental data established, we now turn to a regime which can be probed by simulation alone: a closer look at the small-time mechanics governing the polymer's diffusion. Given the hundreds of thousands of timesteps Δt required for the DNA to transition between nanopits and the real-valued three-dimensional positions, the DPD model provides sufficiently high resolution to observe every minuscule fluctuation contributing to the polymer's macroscale behavior. The DPD model also affords a unique ability to examine how these behaviors change when system parameters, such as the flow velocity, the bending energy of the polymer model, or even the device geometry are changed; in a mathematical model such as a particle simulation, altering system parameters is as simple as changing a value and running the simulation again.

The simulation results presented in this section differ from those used in the previous sections in their boundary conditions; while the previous results employed partial-slip boundary conditions using the method of Pivkin and Karniadakis [99], the simulations here use only the reflecting LJ potential and thus correspond to slip boundary conditions. A consequence of this difference is that the flow velocity around which peak diffusion occurs increases by a factor of four, from $v \approx 0.05$ to $v \approx 0.20$. The slip-boundary simulations exhibit a diffusion peak of similar magnitude and shape to the partial-slip simulations, with the most notable differences in behavior occurring for flows above the critical velocity.

We first define the “occupancy function” $o(t)$ for the polymer as a function of time t :

$$o(t) = \left\lfloor \frac{M(t)}{a} \right\rfloor - \left\lfloor \frac{m(t)}{a} \right\rfloor, \quad (3.33)$$

where $M(t)$ is the y position of the forward-most polymer bead, $m(t)$ is the y position of the rear-most bead, and a is the periodicity as in Fig. 3.2. Defined in this fashion, $o(t) = 0$ when the leading and trailing edge of the DNA are closest to the same pit, $o(t) = 1$ when they are closest to adjacent pits, etc., and so provides a measure of the number of nanopits over which the polymer is extended.

The center-of-mass position $y(t)$, radius of gyration $\langle r_g \rangle$, and occupancy function for a simulation at the critical velocity $v = 0.2$ (i.e., peak diffusion) are illustrated in Fig. 3.9. The DNA jumps from pit to pit at stochastic intervals, alternating between extension during a stochastic jump and compression when becoming trapped in the following pit. Of particular interest is that the geometry allows for “multiple jumps,” where the leading edge of the polymer begins jumping from one pit to the next before the trailing edge has finished its jump from the previous pit (Figs. 3.10 and 3.11). These multiple jumps are a defining feature of peak diffusivity; the simulations of Section 3.2.3 exhibited multiple jumps only when the applied force was near the critical force F_c .

As a point of reference, we also simulated a DNA macromolecule in a channel without nanopits. Without the underlying geometry, there was no trapping effect which led to extension; up to a moving reference frame at the constant mean speed v of the fluid, the behavior of the polymer was

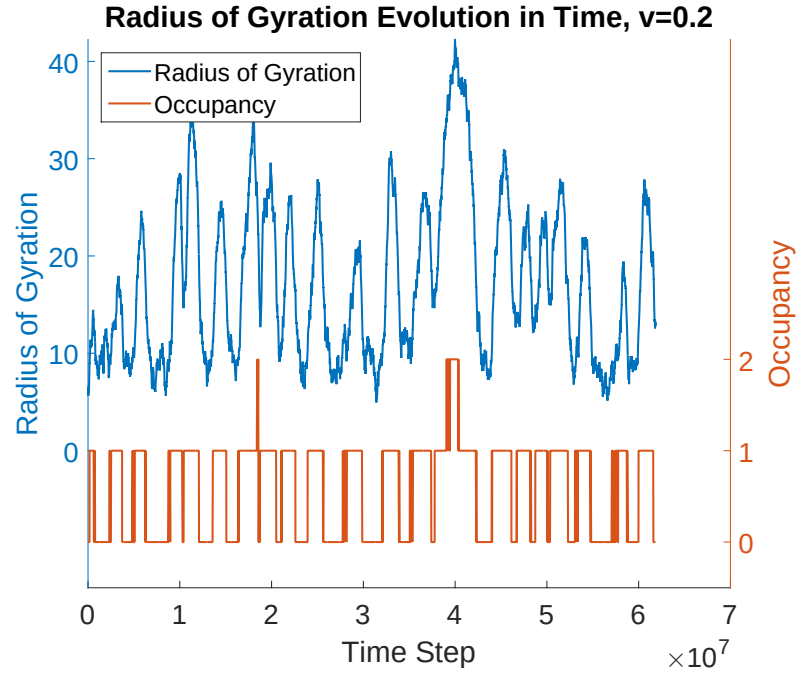


Figure 3.9: Long-time evolution of simulated DNA radius of gyration and occupancy function at the critical velocity $v = 0.2$. The radius of gyration fluctuates as the polymer alternates between extension during jumps and compression during entropic trapping.

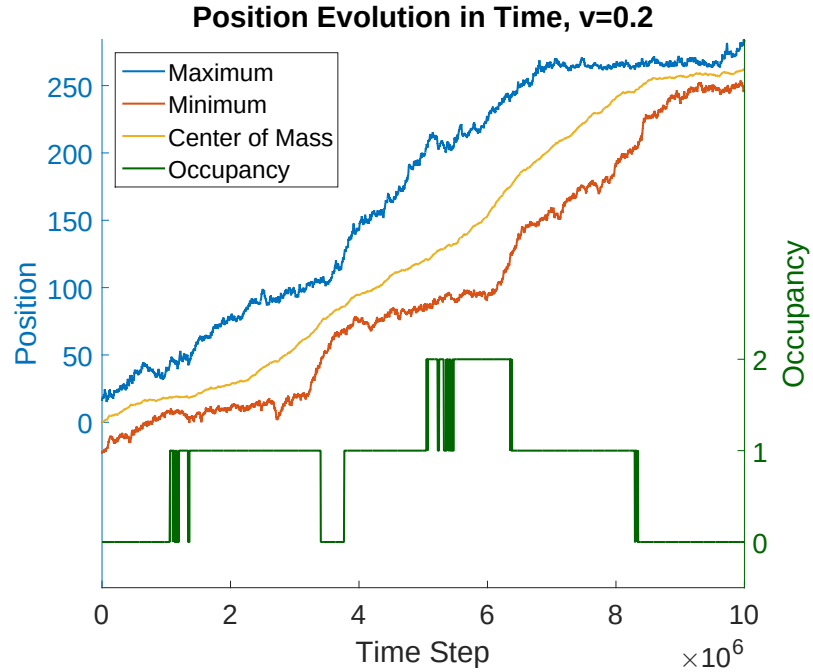


Figure 3.10: Evolution of simulated DNA position and occupancy function in the region surrounding the double jump occurring around time step 40 million in Fig. 3.9.

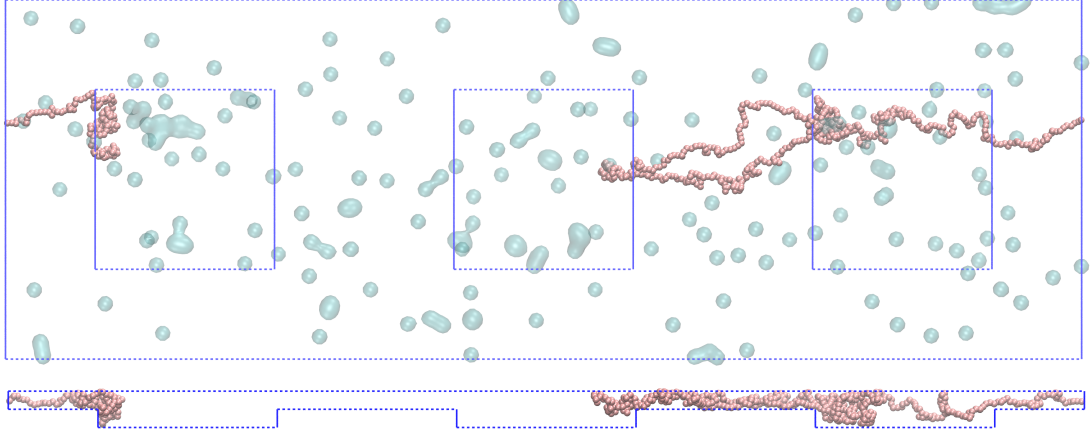


Figure 3.11: Top and side view of a simulated DNA “double jump” at $v = 0.2$ featuring the polymer (red) juxtaposed with a subset of dissipative fluid beads (blue). The DNA at least partially explores the depth of all three pits.

identical to a stationary polymer diffusing in the channel (Fig. 3.12). (Note that the partial-slip boundaries of Section 3.2.3 would distinguish these frames of reference due to friction against the silica plate.) Measurements appear in Figure 3.13, which shows the center of mass to flow linearly forward at the mean flow velocity and the radius of gyration to fluctuate about D_0 in the channel.

A second point of reference used a channel with periodically spaced grooves, i.e., nanopits extended to the periodic boundary in the direction orthogonal to the flow. Friction against water trapped in the grooves results in a “rubber-banding” velocity profile for the fluid, which drives oscillatory behavior in the DNA radius of gyration (Fig. 3.15). The DNA in this flow alternates between periods of extension between grooves, where the flow velocity is higher, and periods of compression within the grooves, where the flow velocity is lower (Fig. 3.14). In both control cases, the device geometry did not exhibit a notable extension of the polymer.

To further explore the effect of the device geometry on the mechanics underlying entropic GAD, we also used the DPD model to alter physical parameters of the system. Increasing the depth of the nanopits should intuitively strengthen the entropic trapping effect; this was observed in experiment with the high-barrier device, which required a significantly higher applied pressure Δp to escape the

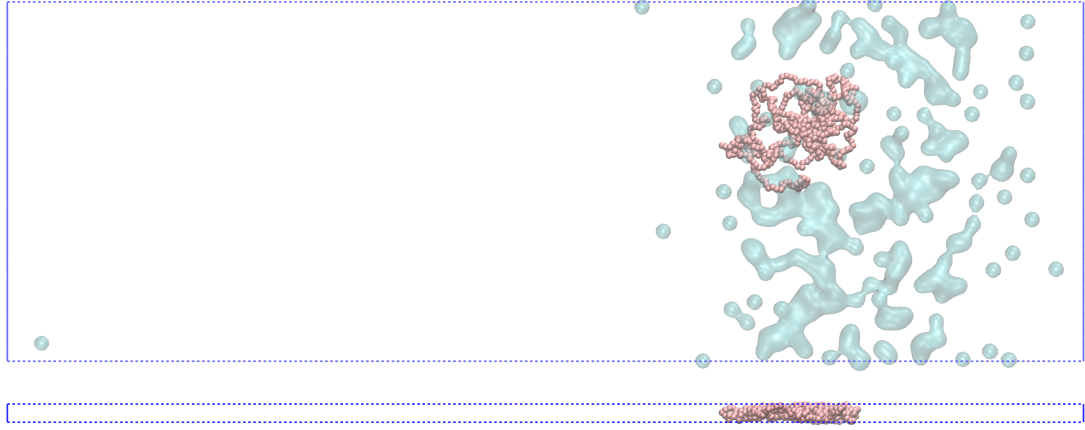


Figure 3.12: Top and side view of simulated DNA in a control channel without nanopits at $v = 0.2$. The DNA moves together with the fluid and diffuses freely.

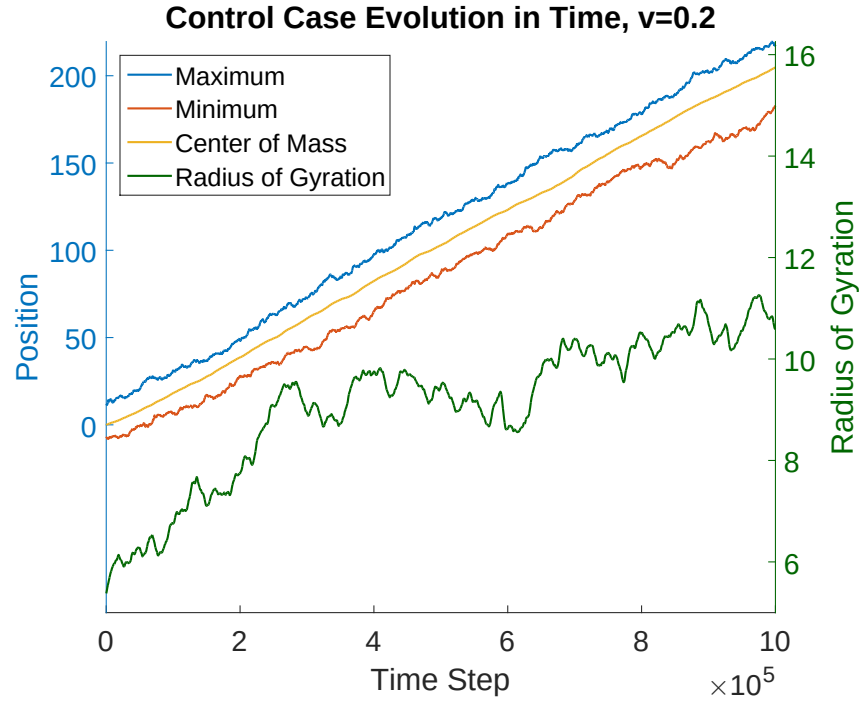


Figure 3.13: Evolution of simulated DNA position and radius of gyration in the control channel without nanopits. After relaxing from its initial configuration, the polymer reaches neutral diffusion in the channel as it is driven forward at the mean flow velocity.

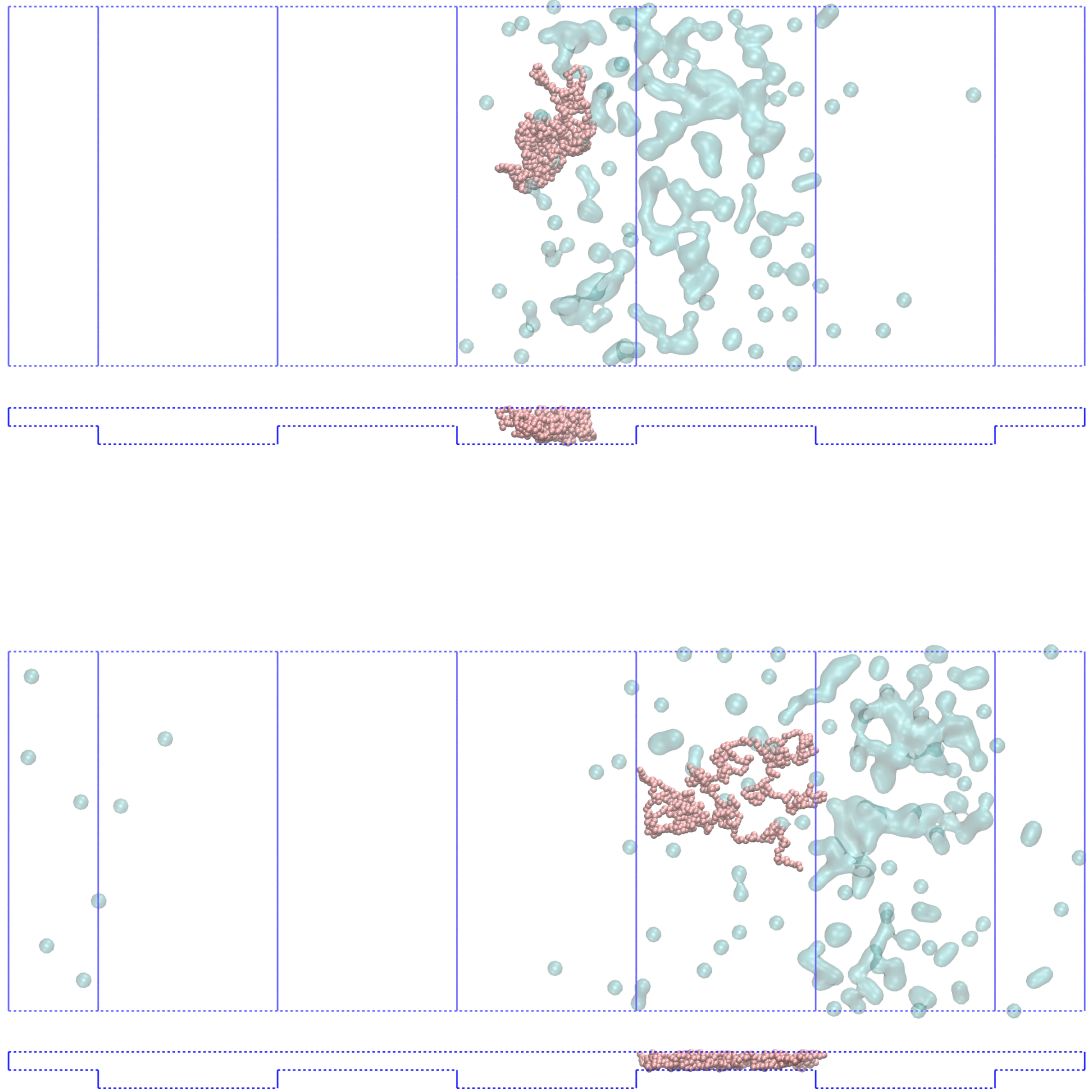


Figure 3.14: Top and side view of simulated DNA in a grooved control channel at $v = 0.2$ in the groove (top) and during a jump (bottom). The “rubber-band” flow profile drives the polymer to oscillate between periods of extension and compression.

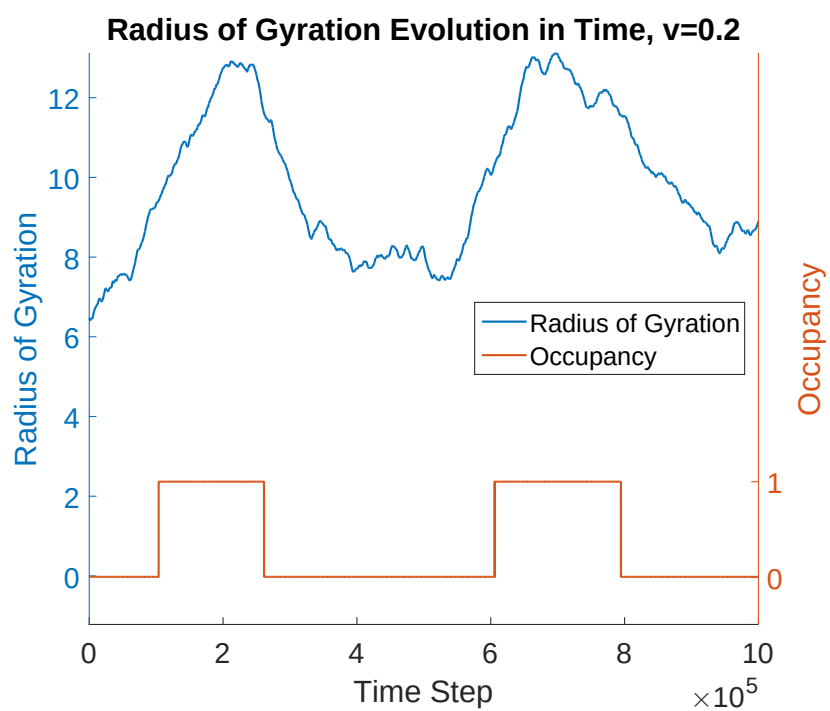


Figure 3.15: Evolution of simulated DNA radius of gyration and occupancy function in the grooved control channel. Peaks in $\langle r_g \rangle$ correspond to jumps between grooves.

nanopits (Fig. 3.3). We consider here a doubling of the nanopit depth, as shown in Fig. 3.16. The former critical flow velocity $v = 0.2$ was sufficient to drive the polymer between pits, but jumps were spaced farther apart and featured visually distinct behavior: the polymer jumping between the deeper pits wraps more tightly around the bridge between nanopits compared to the polymer of Fig. 3.11. Fluid near the bottom of the pits is increasingly isolated from the driving flow in the channel, resulting in slowed progression of the leading edge of the DNA as it enters a new nanopit. These effects are borne in the measurements of Fig. 3.17, which show the polymer to finish its third jump well after 20 million steps, at which time the low-barrier device had seen seven.

A separate simulation explored a scenario with shallow nanopits, here chosen to have half the original depth (Fig. 3.18). The resulting device is the halfway point in the transition from the low-barrier device of Section 3.2 to a nanopit-free control channel; expectedly, the entropic trapping effect is greatly weakened, and the behavior of the polymer most closely resembles the latter case. The long-time evolution of position and occupancy are shown in Fig. 3.19. Over the 20 million step time frame previously used as a point of comparison, DNA in the shallow device completed eight jumps.

The spacing between pits was also found to have a significant effect on the polymer dynamics. A simulation which doubled the separation between nanopits saw increases in the extension of the DNA during jumps, as the leading edge of the polymer continues to be driven by flow in the channel until reaching the following pit, where it moves away from the flow by diffusing into the nanopit (Fig. 3.20). After completion of the jump, the DNA returned to its compact form within the nanopit, at which point the mechanics were no longer affected by the pit separation. The position, radius of gyration, and occupancy function, illustrated in Fig. 3.21, show the mean velocity of the polymer to be significantly higher than in the base device (1100 units traversed in 20 million steps, compared to around 600).

Finally, we returned to the defining parameter of GAD, i.e., the applied pressure Δp , or equivalently, the flow velocity v . The experiments of Section 3.2.1 and simulations of Section 3.2.3 explored

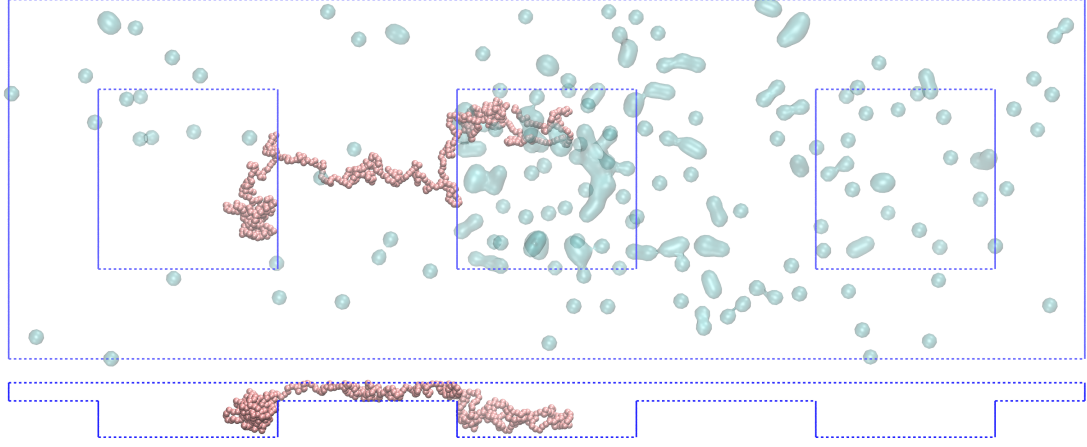


Figure 3.16: Top and side view of simulated DNA in a device with deeper nanopits at $v = 0.2$. The deeper pits provide a stronger entropic trapping effect and reduce the mean forward velocity of the DNA.

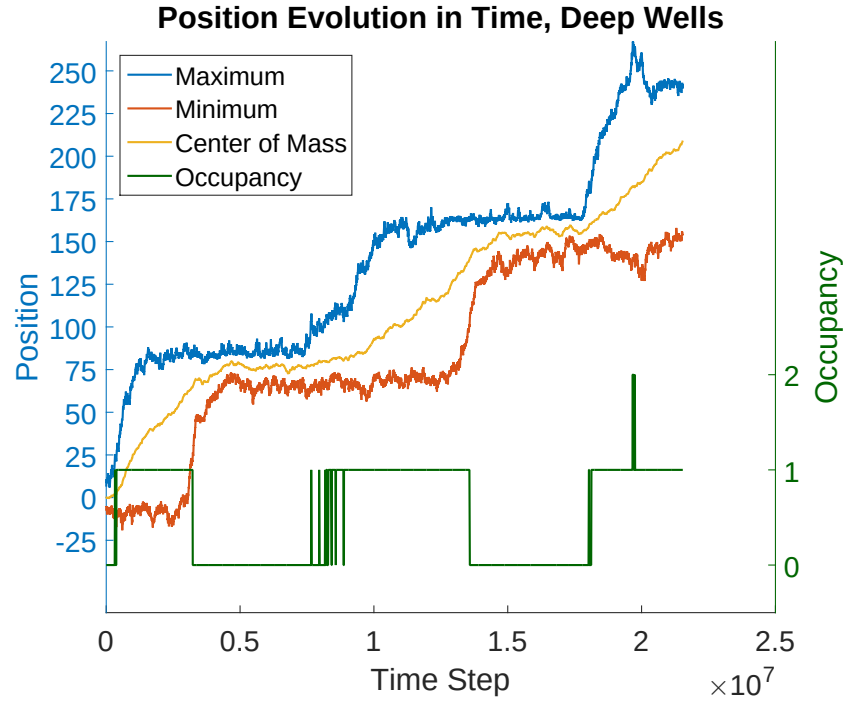


Figure 3.17: Evolution of simulated DNA position and occupancy in a device with deeper nanopits at $v = 0.2$. Single jumps between nanopits occur at stochastic intervals and at a slower rate than the base experiment shown in Fig. 3.11.

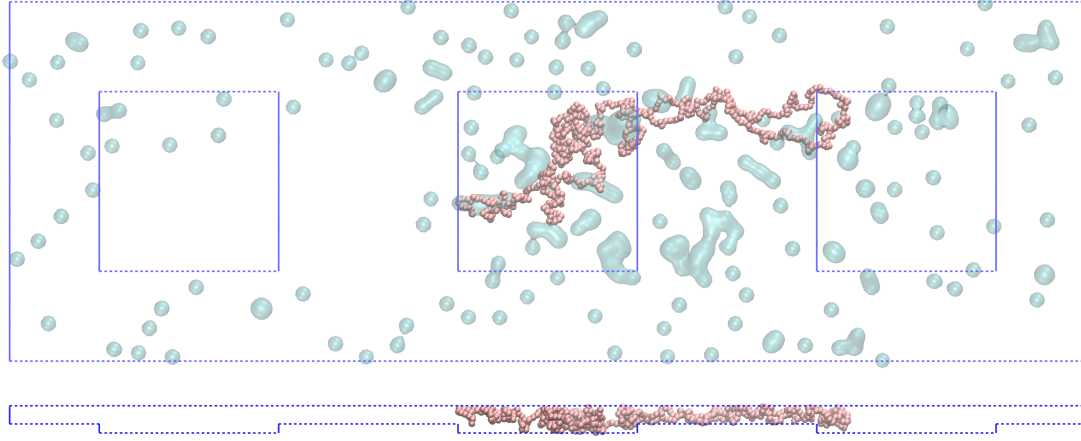


Figure 3.18: Top and side view of simulated DNA in the device with shallow nanopits at $v = 0.2$. The shallow pits weaken the entropic trapping effect and allow the DNA to stream more freely, similarly to observed behavior in the control channel without nanopits.

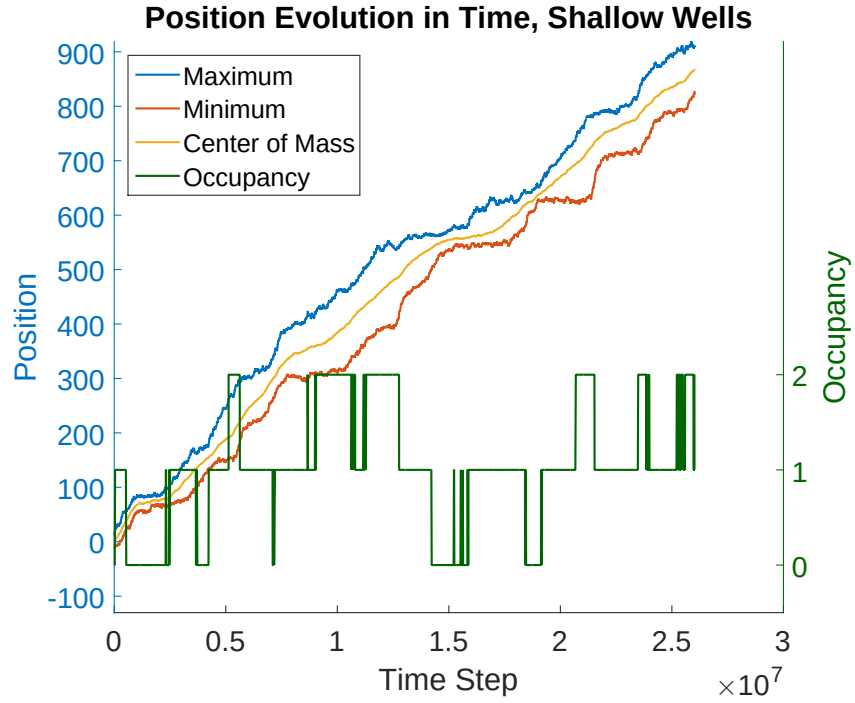


Figure 3.19: Evolution of simulated DNA position and occupancy function in the device with shallow pits at $v = 0.2$. A reduced entropic barrier allows the forward end of the polymer to easily escape nanopits, resulting in a number of extensions across three pits simultaneously.

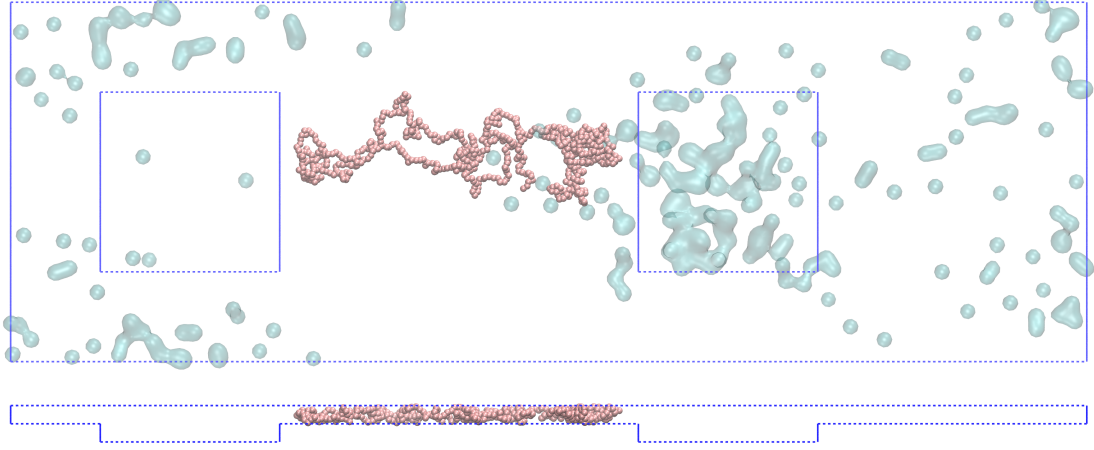


Figure 3.20: Top and side view of simulated DNA in device with doubled pit separation. As the forward end of the DNA must travel farther before entering and exploring the weaker driving flow in the following nanopit, DNA extension during jumps increases significantly.

the effect of v on the diffusion; here, we examine the basic polymer mechanics underlying this effect.

Devices with reduced ($v = 0.1$) and increased ($v = 0.3$) mean flow velocity in the channel appear in Figures 3.22 and 3.24, respectively, while the position and occupancy functions are shown in Figures 3.23 and 3.25. These regimes correspond to $v_s/v_c - 1 = -0.5$ and $v_s/v_c - 1 = 0.5$ in the diffusion peak of Fig. 3.4: namely, $v = 0.1$ is just below the point at which the polymer begins to jump, and $v = 0.3$ is in the freely streaming region where the diffusion D approaches free diffusion in the channel D_0 . The lower velocity flow was unable to overcome the entropic barrier, with the DNA remaining trapped in the pit for the duration of the simulation.

The simulated device with increased flow velocity exhibited the unique property that it resulted in massive extension of the DNA: Figure 3.26 shows the future state of the system in Fig. 3.25. This is a result of the flow profile on the nanopit boundaries, which feature higher-speed flows which catch the forward end of the polymer and accelerate it forward massively, eventually resulting in near-complete extension (the occupancy function, seen in Fig. 3.25, peaks as high as 7). During extension, the DNA did not become trapped nor fully explore the nanopit depth, thereby avoiding the entropic barriers which caught and compressed polymers at lower velocities. It should be noted

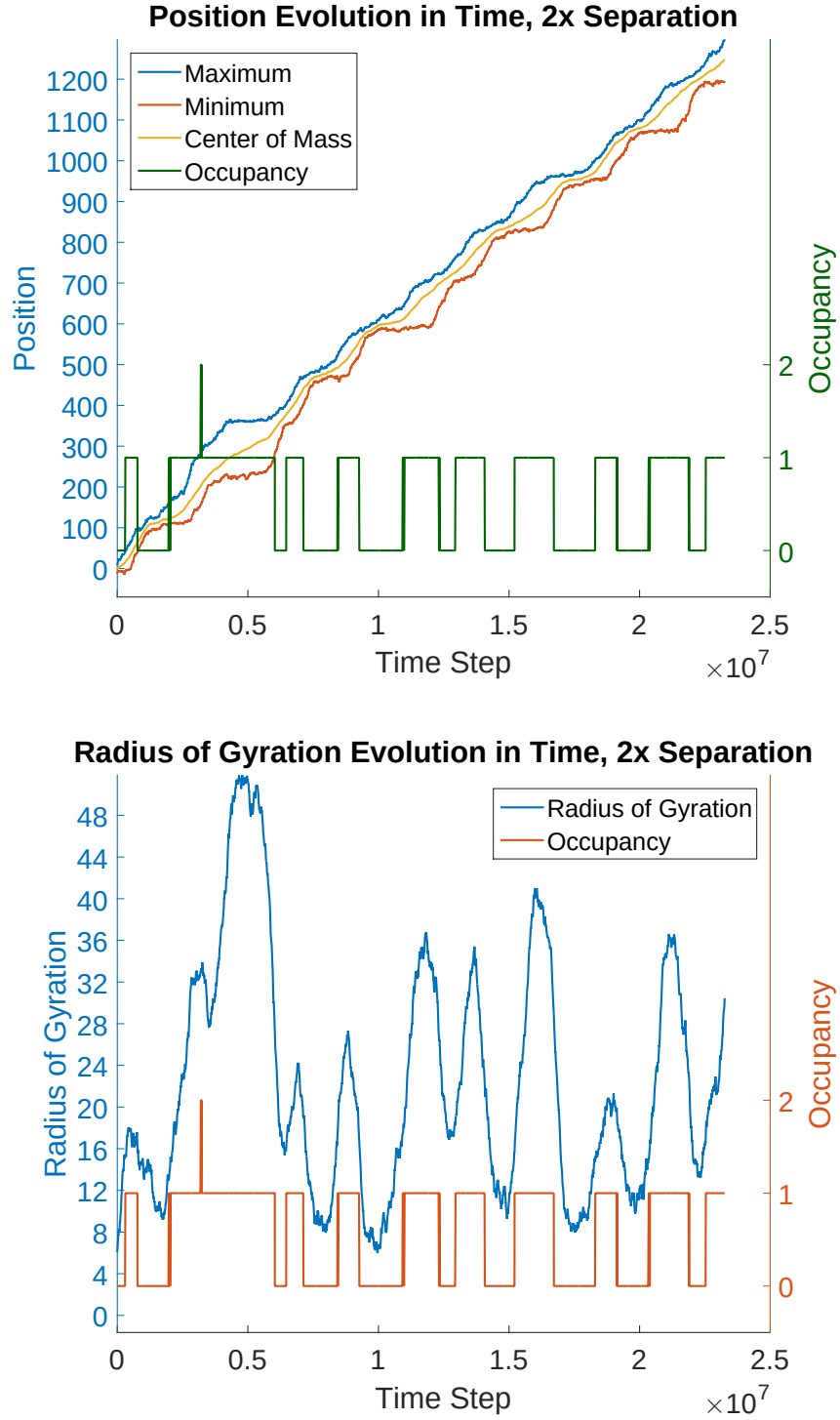


Figure 3.21: Evolution of simulated DNA position and occupancy (top) and radius of gyration (bottom) in device with doubled pit separation. Peaks in $\langle r_g \rangle$ are significantly taller than the base simulation.

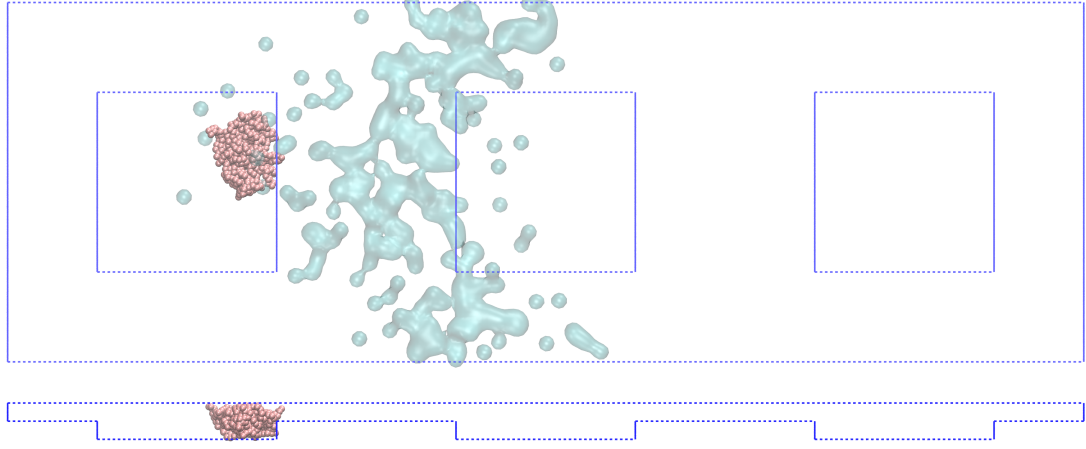


Figure 3.22: Top and side view of simulated DNA in a device with reduced flow velocity $v = 0.1$. At this driving flow rate, the polymer cannot overcome the entropic barrier to forward motion. The illustrated subset of fluid particles (blue), which began in a rough vertical line, show the flow profile wrapping around the nanopits.

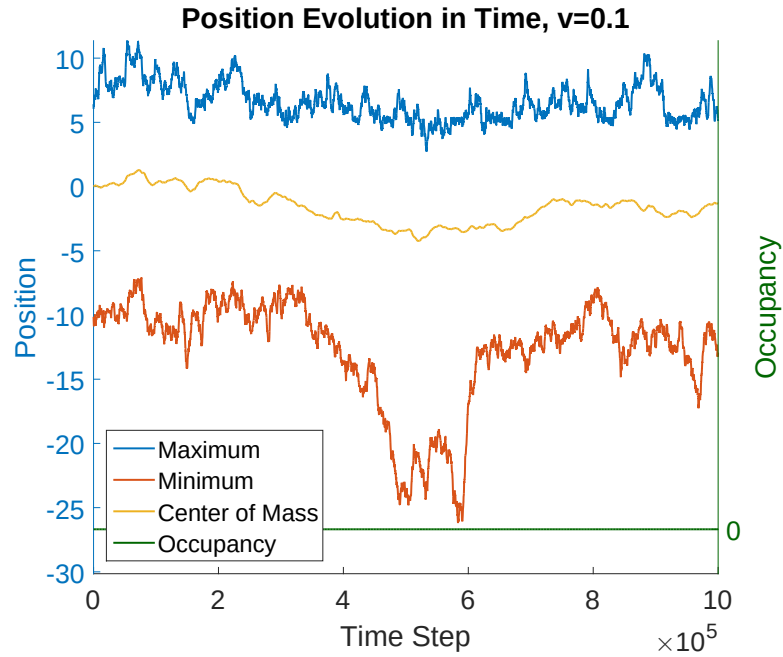


Figure 3.23: Evolution of simulated DNA position and occupation function for a device with reduced flow velocity $v = 0.1$. The rear-most section of the polymer explores backward into the pit, but the forward-most section is unable to overcome the trapping effect.

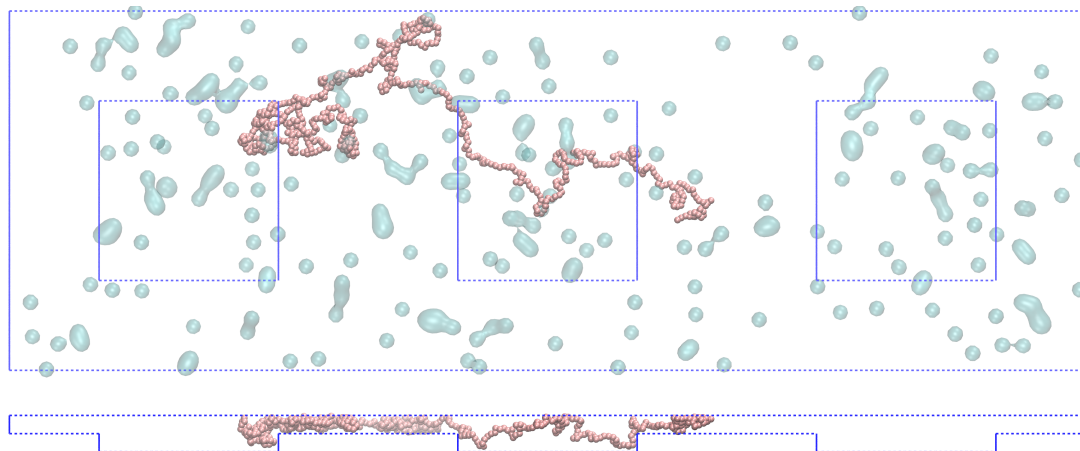


Figure 3.24: Top and side view of simulated DNA in a device with increased flow velocity $v = 0.3$. At this flow rate, the polymer streams nearly freely across pits, with extension resulting from the flow profile of fluid in the device.

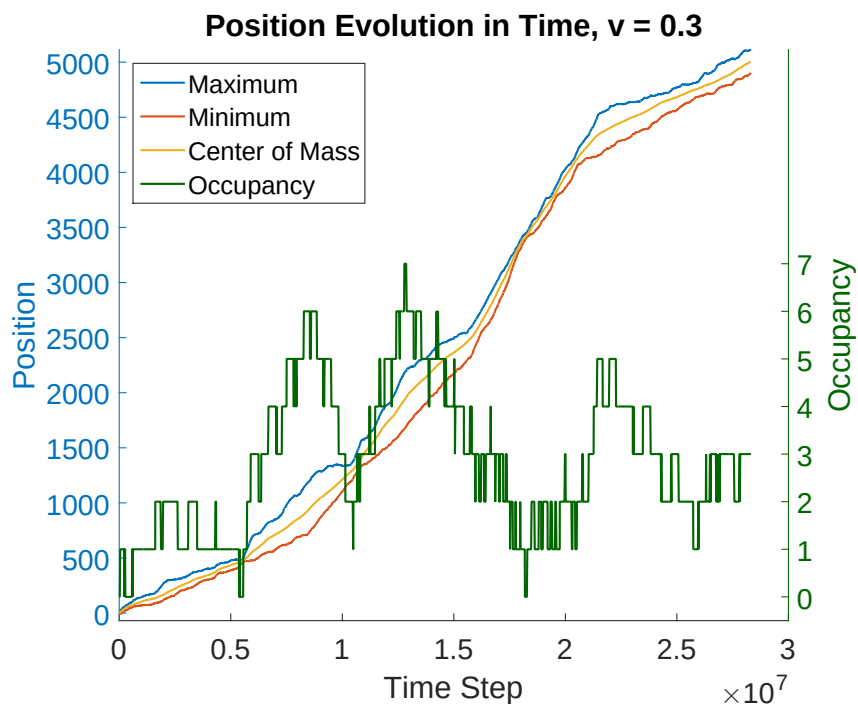


Figure 3.25: Evolution of simulated DNA position and occupation function in a device with increased flow velocity $v = 0.3$. The mean flow rate of the DNA increases sharply as the polymer is pulled into the high-speed flow sections between the nanopit arrays.

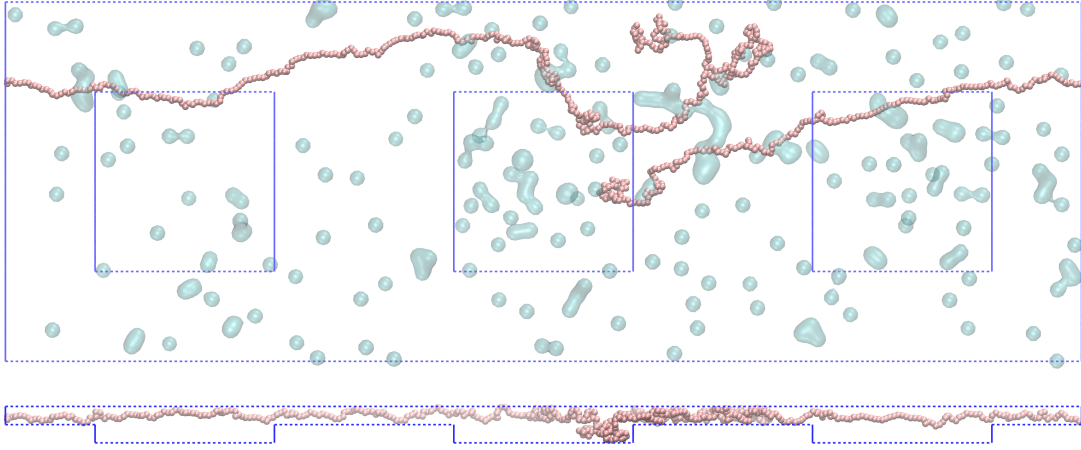


Figure 3.26: Advancing Fig. 3.24 by 800,000 time steps. Accelerated flows on either side of the nanopit array result in massive extension of the DNA.

that this behavior of massive extension is no longer observed with partial-slip boundary conditions, suggesting that a device featuring low-friction plates may aid in the extension of DNA for applications such as sequencing.

In net, the DPD simulations isolated a number of mechanics underlying the observed behaviors in Section 3.2.1: the peak in diffusivity corresponds to the regime in which the polymer can make “multiple jumps,” the flow velocity is a critical factor in driving the forward edge of the polymer from the pit, and the height of the entropic barrier as a function of nanopit depth is determined in part by the ability of the DNA to diffuse into regions of the nanopit isolated from the driving flow in the channel. Given the resolution at which the dynamics can be observed and the ease of making alterations to the device that would require refabricating a physical chip, DPD is thus an essential tool for testing hypotheses and targeting certain behaviors in nanofluidic devices.

3.2.6 Brownian Dynamics and Future Directions

While the accuracy of the DPD simulations in predicting the diffusion curve (Figure 3.3) suggests that the DPD model captures the essential dynamics of the system, the correspondence of the DPD-measured force landscape to the one-dimensional Brownian analog (Figure 3.7) hints that simpler

models may be sufficient for certain applications. In particular, we note that the one-dimensional Langevin equation in the presence of a periodic forcing term (Eq. 3.19) is a slight modification of the Brownian dynamics approach (Eq. 2.8) applied to a single particle corresponding to the polymer's center of mass. Continuing this line of thinking, there exists a hierarchy of possible mathematical models for the nanofluidic device, ordered from most to least complex:

1. DPD approach described in Section 3.2.3, which functions as the ground truth.
2. Brownian approach which models a three-dimensional polymer diffusing in a background flow measured from the DPD approach.
3. Brownian approach which models a three-dimensional polymer diffusing in a simplified background flow, e.g., the target flow velocity v in the upper channel and no flow in the nanopits.
4. Brownian approach which models a one-dimensional polymer diffusing in a tilted periodic potential (i.e., extension of the results of Reimann et al. to a particle chain).
5. Brownian approach which models a single particle, corresponding to the DNA center of mass, diffusing in the measured force landscape of Fig. 3.7.
6. One-dimensional random walk of the polymer ends, related by a nonextensible length constraint and with an asymmetric jump kernel (see, e.g., [87]).

Preliminary results suggest that the first three models produce similar results for the measured diffusion, and so the equilibrium assumption of the Brownian approach (with respect to the fluid flow) appears empirically reasonable. It remains to be seen whether the even simpler models 4–6 are sufficient; depending on which observables they are able to reproduce, it may be possible to make predictions of the polymer evolution with computation time on the order of seconds, or to couple the simpler approximations at longer time scales with higher-accuracy methods for complex behaviors such as multiple jumps over short time scales, similar to multi-scale approaches to numerical PDEs (see, e.g., [50, 28, 27]).

Although the simple one-dimensional models would provide an especially significant reduction in computational complexity, three-dimensional Brownian dynamics in a simple background flow, with runtime on the order of minutes, is already sufficiently cheap to answer a number of more complicated questions about nanofluidic devices. Chapter 4 describes a high-performance parallel implementation of Bayesian uncertainty quantification which could feasibly be applied to these cheaper simulations, providing a rigorous framework for experimental design via a probabilistic solution to the inverse problem. The Brownian approach can also be used to easily test hypotheses about similar devices – for example, is there a force field which can be applied to the DNA via an electric potential which would induce oscillatory behavior? Future investigation may reveal the potential for such questions to yield results worth testing experimentally.

3.3 Membrane Models

While models for biologically relevant polymers (especially DNA, RNA, and a broad range of proteins) are important to the understanding of cellular mechanics, a complete look at cellular biology relies on an understanding of membranes – in particular, the phospholipid bilayer, which is the cell membrane of all living organisms and a number of non-living entities such as viruses. If polymers are mathematical curves, then membranes are mathematical surfaces – lower-dimensional, continuous objects embedded in a higher-dimensional space which separate the space into an interior and exterior. In this section, we briefly describe two approaches for modeling membranes: minimization of a free energy functional and explicit simulation via, e.g., particle methods. Sections 3.3.2 and 3.3.3 describe explicit DPD membrane models in three and two dimensions, respectively, which will be used to explore a number of cell-level phenomena involving membrane deformation. Depending on the behavior of interest, either of these models may be useful: the three-dimensional model is a complex ground-truth, while the two-dimensional model is a cheaper implementation which captures some of the basic mechanics such as curvature. The latter model will prove especially useful in the

high-performance framework of Chapter 5.

3.3.1 Energy Minimization for Surfaces

Surfaces as solutions to energy minimization problems have a long history, tracing back to the work of Lagrange in the 18th century. Lagrange was the first to consider the problem of finding the surface $z = z(x, y)$ of least area stretched across a given closed contour; though he derived an equation for the solution as

$$\frac{d}{dx} \left(\frac{z_x}{\sqrt{1 + z_x^2 + z_y^2}} \right) + \frac{d}{dy} \left(\frac{z_y}{\sqrt{1 + z_x^2 + z_y^2}} \right) = 0 \quad (3.34)$$

via the recently developed Euler-Lagrange formulation, Lagrange was unable to find solutions beyond the case of a planar closed contour, in which case the planar surface is a trivial solution. (A number of solutions would be found over the following centuries in the work of mathematical titans such as Schwarz and Riemann.)

Framing the problem in terms of the free energy $E = \mu A$, where μ is the surface tension and A is the surface area, a minimal surface corresponds to a minimization of the free energy. There are many other possible choices for free energy corresponding to other physical systems, and so the energy-minimization approach can be used to find continuous solutions to more complex membrane models (as opposed to the infinitesimally thin film which is modeled only by surface tension). The chief advantage of this approach is its analytical precision: under the assumption of continuity, the rich fields of calculus of variations and differential geometry can be leveraged to study the membrane shape and its governing shape equation.

The earliest energy functional corresponding to a solid membrane belongs to Poisson, who in 1812 proposed the functional

$$E_{sm} = \frac{k_c}{2} \int (2H)^2 dA, \quad (3.35)$$

corresponding to a closed surface with mean curvature H and bending modulus k_c , written in terms of an integral over the differential area elements dA [101]; the solutions to the Euler-Lagrange

equation corresponding to this energy functional,

$$\nabla^2 H + 2H(H^2 - K) = 0, \quad (3.36)$$

where K is the Gaussian curvature, are called Willmore surfaces [139]. In 1973, Helfrich proposed an energy functional for a vesicle with a lipid bilayer membrane:

$$E_{lb} = pV + \int (f_c + \mu) dA, \quad (3.37)$$

in terms of the osmotic pressure p (the outer pressure less the inner pressure), surface tension μ , enclosed volume V , and a curvature energy per unit area f_c defined as

$$f_c = \frac{k_c}{2} (2H + c_0)^2 + \bar{k}K, \quad (3.38)$$

where k_c and $\bar{k}$ are elastic constants and c_0 is the spontaneous curvature of the membrane surface (a contribution to curvature from the chemical structure of the component lipids) [47]. A first-order variational approach yields the shape equation

$$p - 2\mu H + k_c (2H + c_0) (2H^2 - c_0 H - 2K) + k_c \nabla^2 (2H) = 0 \quad (3.39)$$

for the vesicle corresponding to the free energy E_{lb} [144]. There exist a range of analytical and numerical solutions to this equation in the literature, including spherical vesicles (as observed for some lymphocytes [30]), the $\sqrt{2}$ -torus, and the biconcave disc observed in human red blood cells [25], suggesting that this approach can feasibly solve for macroscale membrane shapes. In cases where the fine-scale structure of the membrane is not relevant, the family of energy-minimization approaches is thus a practical approach for modeling of the cellular membrane.

3.3.2 Three-dimensional DPD Membrane

Energy-minimization approaches to membrane models are inherently continuous: they neglect the fine-scale structure of the membrane, instead focusing on macroscopic observables such as surface tension and curvature. In particular, the Helfrich approach described in Eq. 3.39 makes a number of

assumptions about the membrane structure: most notably, the membrane is assumed homogeneous, with each differential element of the membrane modeled with the same elastic constraints. There are a number of biological mechanisms which might cause these assumptions to fail, especially the presence of proteins embedded in the membrane or an asymmetry in lipid structure between the layers. The continuum approximation is also not appropriate when studying small patches of the membrane, where the lipid structure and thermal fluctuations are significant.

The chief alternative to continuum approaches is particle simulations, which can individually model the many molecules composing a membrane and derive the membrane structure from basic physical interactions, such as repulsion between water molecules and the hydrophobic tails of lipid molecules. While there are a variety of approaches to modeling specific membrane mechanics with molecular dynamics [78, 92, 63, 54, 60], we describe here a DPD formulation of a lipid membrane, which has two main advantages: first, simulations are computationally feasible at much larger spatial scales, and second, the coarse-grained dissipative beads facilitate interactions between structures of vastly different scales, and so water, lipids, and other biological soft matter such as proteins and gels can be modeled jointly.

Defining the DPD membrane model requires specifying the coefficients described in Section 2.3 for the different particle types; because there is a choice of scale for the coarse-graining, DPD coefficients are conventionally specified for the nondimensionalized system. In their foundational paper on DPD, Groot and Warren [40] found that the dimensionless compressibility of water could be matched by using the conservative coefficient $a_{ij} = 25.0$, dissipative coefficient $\gamma_{ij} = 4.5$, cutoff radius $r_c = 1.0$, and numerical density $\rho = 3$ for unit-mass DPD particles (the same values used for the water in Section 3.2.3). To model a lipid bilayer membrane, we additionally introduce particles with modified coefficients to model the lipids, here referred to as “Head” and “Tail” particles. The pairwise a_{ij} and γ_{ij} , chosen to be similar to existing literature on DPD membranes [36, 39, 116] and to reproduce mesoscopic properties such as lateral fluidity and a stable bilayer structure [68], appear in Table 3.1.

Table 3.1: Pairwise Coefficients for 3D DPD Membrane

a_{ij}	Head	Tail	Water
Head	25.0	50.0	35.0
Tail	50.0	15.0	75.0
Water	35.0	75.0	25.0

γ_{ij}	Head	Tail	Water
Head	4.5	9.0	4.5
Tail	9.0	4.5	20.0
Water	4.5	20.0	4.5

Head and tail particles are connected as in Figure 3.27(a) by harmonic bonds with dimensionless energy $E_{ij} = 64(r_{ij} - 0.5)^2$, so that the resting length of a bond is 0.5. Along the tails, three-body potentials $E_{ijk} = 20(1 + \cos \phi_{ijk})$ are introduced to provide stiffness, where ϕ_{ijk} is the angle between bond ij and bond jk . Copies of the lipids are then placed in two layers at the vertices of a square lattice, with the tails facing inward. The outside of the membrane is initialized with water particles placed on a cubic lattice with numerical density $\rho = 3$. Periodic boundary conditions are imposed around the simulation box. To evolve the membrane forward in time, we again use DPD velocity-Verlet (see Section 2.4) with a temperature $T = 1.0$ and timestep $\Delta t = 0.005$.

Lipid Schematic & Example Membrane

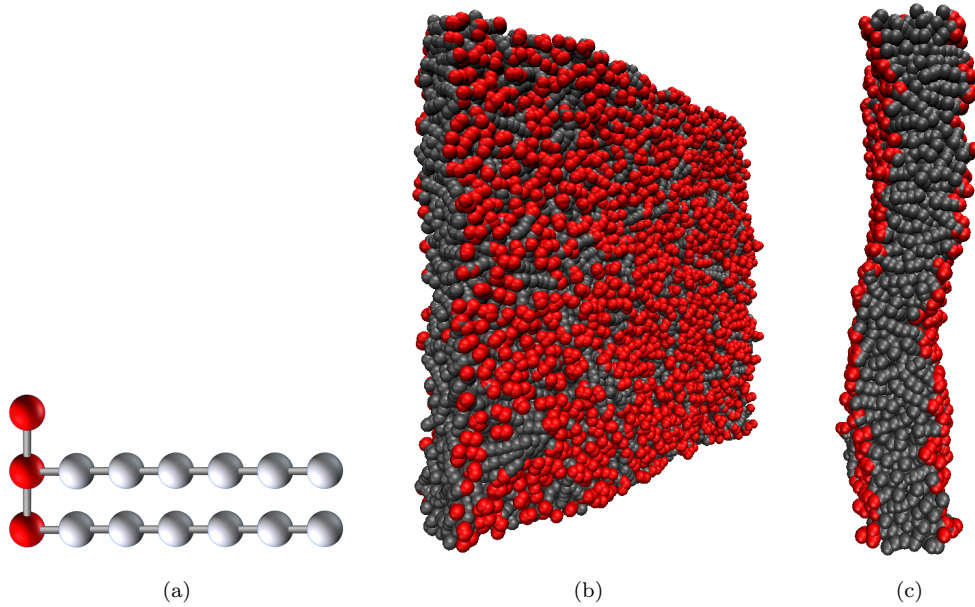


Figure 3.27: (a) Lipid with three heads and two tails comprising six tail particles each. Bonds are shown as thin rods. Three-body potentials are imposed among triplets with consecutive horizontal bonds. (b) 3D view of simulated membrane. Fluid particles (not shown) fill the space on both sides of the membrane. (c) Side view of simulated membrane showing the lipid heads oriented to separate the membrane interior from the surrounding fluid.

An example of the resulting membrane is shown in Fig. 3.27(b,c) after a short equilibration period. As the conservative coefficient a_{ij} between the lipid tails and water is much larger than the corresponding repulsion between lipid heads and water, the configuration with lipid tails turned inward and heads separating the hydrophobic tails from the surrounding fluids is stable. The size of the simulated membrane can be comparatively much larger than lipid membranes modeled with molecular dynamics; the three-dimensional DPD membrane used to study the mechanics of membrane poration in Section 3.4 models $\sim 30,000$ lipids. We will also couple the DPD membrane with the polymer models of Section 3.1 to examine the effect of polymerizing actin filaments driving the membrane forward, a fundamental mechanism for eukaryotic cell motility in organisms such as lamellipodia [102], illustrating the flexibility of the DPD approach in effectively modeling the behavior of soft matter structures at different scales.

3.3.3 Two-dimensional DPD Membrane

For some mechanics of interest, such as the average membrane curvature examined later in Section 5.1, a two-dimensional model may be sufficient to capture the essential behavior and can offer a dramatic reduction in computational expense. We describe here a modification to the three-dimensional DPD model of the previous section to model a cross-sectional membrane slice; the reduction in model complexity makes the two-dimensional model practical for a wider variety of applications, including the long-time simulation of entire vesicles in fluid described in Section 3.6.

As before, the membrane structure is derived wholly from DPD interactions between particles of different types, with the strength of conservative interactions adjusted such that the repulsion between tail particles is less than that between tail and fluid particles. Similarly, hydrophilicity of the lipid heads is achieved by adjusting the repulsion between head and fluid particles to be less than that between head and tail particles. To illustrate the flexibility of the modeling approach, we choose the two-dimensional membrane simulation to use a distinct lipid model with only a single head bead; rather than accounting for the effective size of the lipid head via multiple particles, we

Table 3.2: Pairwise Coefficients for 2D DPD Membrane

a_{ij}	Head	Tail	Water	γ_{ij}	Head	Tail	Water	r_{ij}^c	Head	Tail	Water
Head	15.0	50.0	35.0	Head	4.5	9.0	4.5	Head	2.5	1.75	1.75
Tail	50.0	15.0	75.0	Tail	9.0	4.5	20.0	Tail	1.75	1.0	1.0
Water	35.0	75.0	25.0	Water	4.5	20.0	4.5	Water	1.75	1.0	1.0

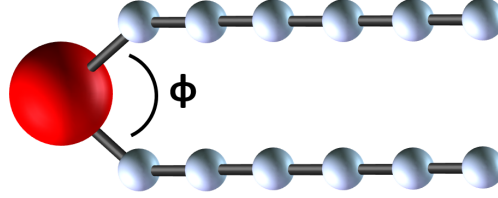


Figure 3.28: Schematic of a lipid with one head particle (red) and two tails comprising six tail particles each (gray). Bonds are shown as thin rods. The angle between tails, labeled ϕ , minimizes energy at a specified neutral angle ϕ_0 .

adjust instead the cutoff radius r_c of the conservative and dissipative forces, thereby scaling the interactions of the particle directly. Particle masses are adjusted accordingly, so that a particle effectively 2.5 times larger in one dimension has a mass $2.5^3 = 15.625$ times greater, with the fluid particles having unit mass. Specific values chosen for the simulation appear in Table 3.2.

Each lipid is composed of a single lipid head connected to two lipid tails of six particles each via harmonic bonds, each of which uses the spring potential $E_{ij} = 64(r_{ij} - 0.5)^2$ so that the neutral length of a bond is 0.5. In the two-dimensional membrane, we use a distinct three-body potential $E_{ijk} = 50(1 - \cos(\phi_{ijk} - \phi_0))$ imposed along consecutive bonds to provide stiffness, where ϕ_{ijk} is again the angle between bonds ij and jk and ϕ_0 is a resting angle; this form of the bending energy facilitates specification of the neutral angle between the membrane tails, i.e., the angle preferentially formed between the line segments connecting the head particle to the first particle in each tail, labeled as ϕ in Figure 3.28. The neutral angle between tail bonds is fixed at 180° .

Finally, lipids are placed in a two-dimensional periodic simulation box in two adjacent rows with head particles toward the outside, as in Figure 3.29(a). The separation between lipids, spacing between tail particles within a lipid, and total length of the membrane patch are free parameters, here chosen to be 0.8, 0.5, and 64, respectively. The remainder of the simulation box is filled with

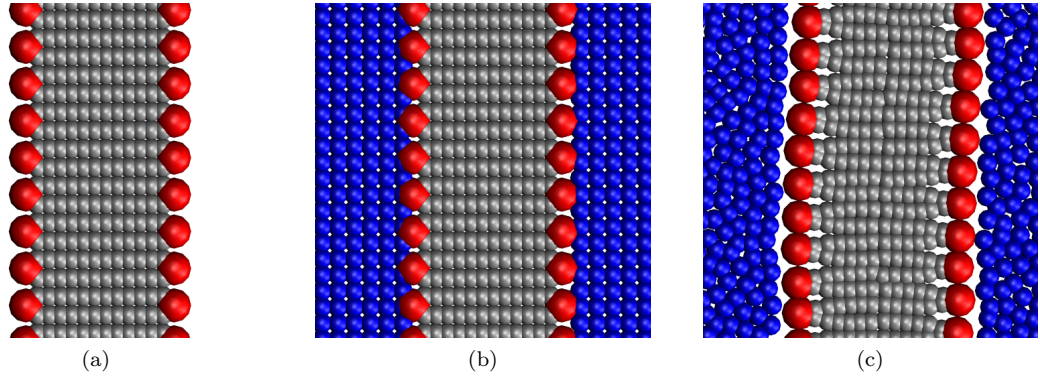


Figure 3.29: (a) Two-dimensional membrane initialization. Head particles (red) line the outside; each has two lipid tails of six tail particles (gray) directed inward. (b) Fluid initialization. Water particles are placed on the vertices of a square lattice outside the membrane. (c) Membrane after 2500 steps.

fluid particles, placed on the vertices of a square lattice with numerical density $\rho = 2.1$. The system is then evolved in time using a velocity-Verlet scheme (see, e.g., [98]) with timestep $\Delta t = 0.03$ and nondimensional temperature $T = 0.1$. The phases of initialization and resulting membrane at a future time $t = 75$ appear in Figure 3.29(c).

The following sections detail a number of applications of the two- and three-dimensional DPD membrane models to physical systems. In Section 3.4, a circular section of the three-dimensional membrane is removed in order to study the mechanics of lipid membrane poration, while Section 3.5 will use stiff bead-spring polymers modeling actin filaments to observe the driving mechanism of polymerization-induced mobility. The two-dimensional model will be used in Section 3.6, which models vesicles in a fluid channel driven by artificial forces, and in Section 5.1, which considers spontaneous curvature of lipid membranes due to difference in lipid structure between layers in the context of a high-performance method for probabilistically solving the inverse problem in the presence of noise. These varied applications highlight the flexibility of the DPD approach in modeling diverse structures and interactions at long time scales.

3.4 Critical Pore Size in Films and Membranes

A recent article by Ilton et al. [53] examined the evolution of pores in a model membrane constructed from polystyrene in a water bath. A tightly focused laser was used to create a temperature gradient in the film, decreasing the local surface tension and driving the formation of a hole whose size was determined by the power and exposure time of the laser. It was found that holes below a critical size r_c sealed, while larger holes began to increase in size, a behavior which had previously been observed in solid-state membranes [123].

The liquid polymer films of [53] are easily observable via traditional microscopy techniques due to their large size (the thinnest homopolymer film studied was ~ 100 nm thick, with most films on the order of $1 \mu m$). In contrast, a typical lipid membrane, as might be found in a biological cell, is much smaller (~ 10 nm thick). Pore formation and evolution in a lipid membrane thus has a number of experimental difficulties owing to the small spatial and temporal scales. In this section, we instead use a simulated lipid membrane to study the dynamics in question. The simulated system is computationally feasible and reproduces many of the results observed in [53].

To explicitly simulated a porated lipid membrane, we employ the three-dimensional DPD membrane model described in Section 3.3.2. A periodic simulation box of $144 \times 144 \times 40$ was filled entirely in the first two dimensions with a DPD bilayer membrane comprising 28,700 lipids with three head particles and two tails of six particles each, so that the side length of the square lattice on which lipids were initialized was approx. 1.202. The number of lipids was chosen experimentally to achieve stability in the membrane. To simulate a pore, all lipids intersecting an orthogonal cylinder of fixed radius were removed, and the cylinder was included in the initialization region for fluid particles. The resulting setup at time zero can be seen in Figure 3.30(b).

To track the size of the pore, the locations of all lipids were recorded every 1000 steps. Images of the system were created by rendering spheres of radius 0.5 at the location of every head and tail particle, then projecting the 3D system into a 2D plane tangent to the membrane surface. The

Porated Bilayer Initialization and Thresholding

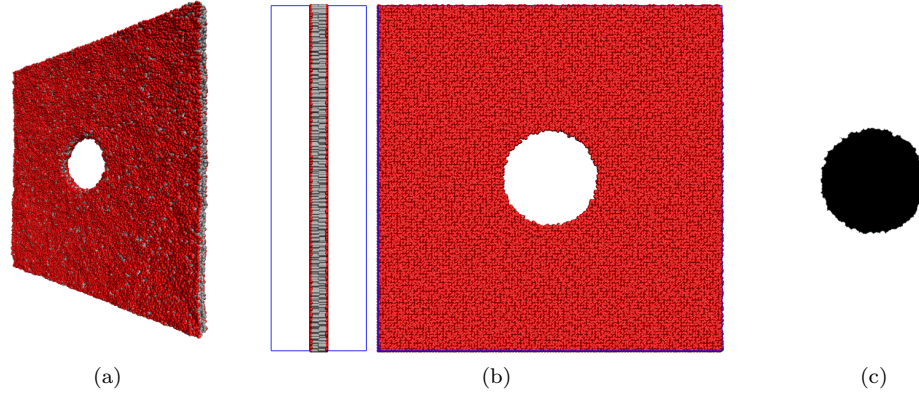


Figure 3.30: (a) 3D view of simulated membrane with pore. Fluid particles (not shown) fill the space on both sides of the membrane and in the pore. (b) Side and front view of initial membrane configuration. Blue lines at boundary show the simulation box. (c) Binary matrix (value 1 shown in black) resulting from smoothing and thresholding the front view in (b).

resulting images were then smoothed and thresholded, resulting in binary matrices M_i with value 1 within the pore and 0 without, as in Figure 3.30(c). Finally, the 2D center of mass position $\bar{\mathbf{x}}_i$ was computed, and the sum $\sum_{\mathbf{x}} M_i(\mathbf{x}) |\mathbf{x} - \bar{\mathbf{x}}_i|$ was calculated. For a circle of radius R , the result should be approximated by the integral

$$\int_0^{2\pi} \int_0^R |r| r dr d\theta = \frac{2\pi}{3} R^3, \quad (3.40)$$

and so the approximate radius of the pore is given by

$$R_i \approx \left(\frac{3}{2\pi} \sum_{\mathbf{x}} M_i(\mathbf{x}) |\mathbf{x} - \bar{\mathbf{x}}_i| \right)^{1/3}. \quad (3.41)$$

To compute the membrane width, the 3D representation was instead projected into a plane bisecting the membrane (i.e., an orthographic side view) as in the side view of Figure 3.30(b). For each row in the projected image, the number of non-background pixels was summed, approximating the width of the membrane at that location; to reduce the effect of the natural thermal fluctuations in the membrane surface, the minimum of all row widths was used as the membrane width for that image. The resulting value was time-averaged over 20 images (one every 1000 steps) to obtain a reference value h for the membrane width.

3.4.1 Existence of a Critical Radius

The time evolution of simulated pores of various initial radii can be seen in Figure 3.31. The behavior of the pore is a function of its radius: sufficiently small pores begin to shrink and eventually seal entirely, while sufficiently large pores begin expanding, eventually severing the membrane. The time-evolving radius of each pore, as measured by Eq. (3.41), is shown in Figure 3.32.

The growth of supercritical pores proceeds in roughly exponential fashion, in agreement with the findings for homopolymer films in [53]. For very large times ($t > 1000$), the pore approaches the order of the simulation box, and so begins interacting with itself across the periodic boundary, resulting in slower growth. The subcritical pore, which closed around $t = 400$, yielded a stable membrane for the remainder of the simulation. The observed critical ratio $1.34 < r_c/h < 1.47$ is larger than the ratio $r_c/h = \pi/4$ derived by Ilton et al. for a homopolymer film., suggesting an additional free energy cost for pore formation due to the lipid structure. Note that we define the ratio in terms of the initial size r_0 , meaning it is possible a pore which evolves to be smaller than the critical radius may still expand as $t \rightarrow \infty$.

A second set of simulations examined the effect of increasing lipid density in the membrane. The number of lipids in the periodic simulation box was increased by 4.4% to 29,970, resulting in a stable membrane with a lipid excess. For short times, a pore opened in the modified membrane begins to close regardless of pore size as the membrane relieves internal pressure. As seen in Figure 3.33, the existence of a critical ratio remains in this scenario, as sufficiently large pores reverse the initial collapse and expand exponentially as in the first simulation. This behavior is governed by equilibrium size, rather than initial size; the pore with $r_0/h = 1.88$, well above the critical ratio, closes regardless in this new scenario despite having expanded in the initial case of Figure 3.32.

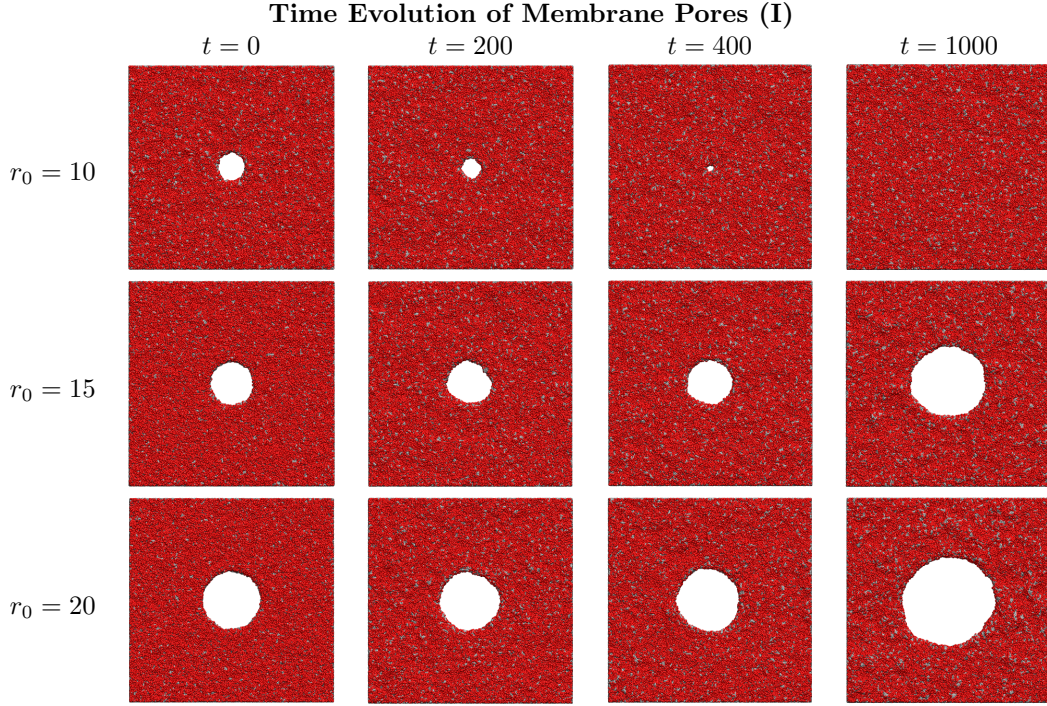


Figure 3.31: Rendered images of pores of three different initial sizes (rows). Four images throughout the simulation are shown, progressing from left to right. Pores shrink or grow exponentially depending on the initial pore radius.

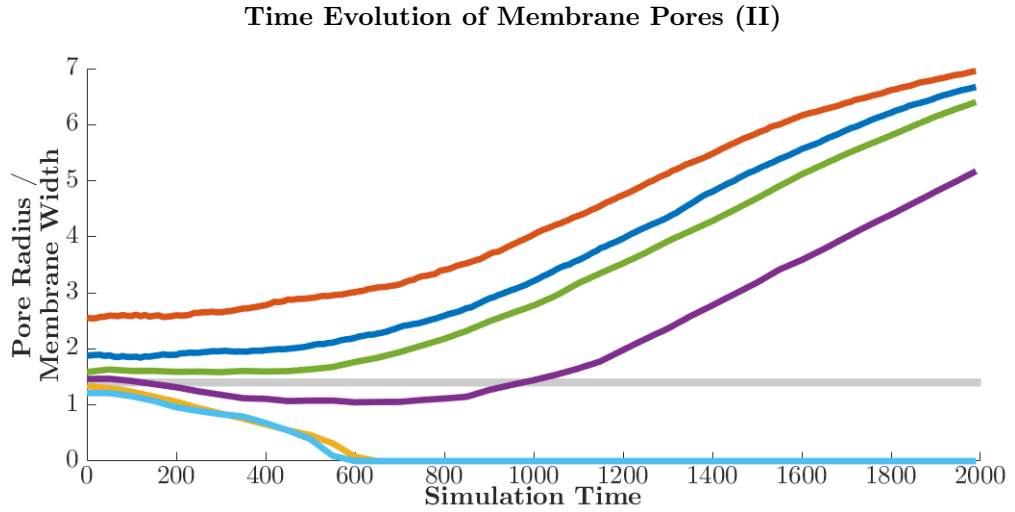


Figure 3.32: Membrane pore size over time as calculated by Eq. (3.41). Size (y axis) is shown in units of the average membrane width, $h = 7.98$. Several initial sizes are shown, ranging from $r_0/h = 1.21$ (bottom) to $r_0/h = 2.56$ (top). The shaded region denotes possible values of the critical ratio r_c/h below which pores seal – $1.34 < r_c/h < 1.47$.

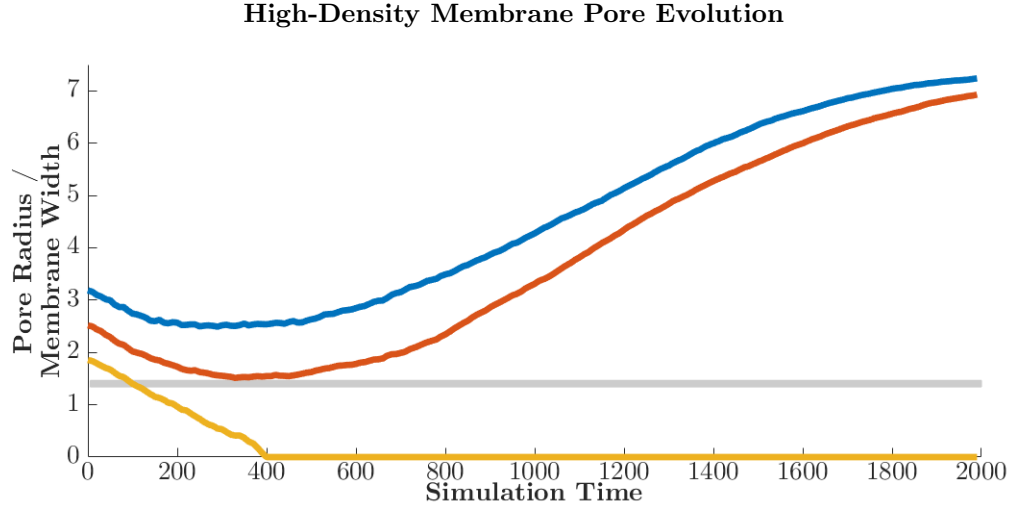


Figure 3.33: Pore size over time via Eq. (3.41) for a membrane with increased lipid density. Three initial sizes are shown: top: $r_0/h = 3.19$, middle: $r_0/h = 2.52$, and bottom: $r_0/h = 1.86$. The shaded region corresponds to the possible range of r_c/h derived in Figure 3.32.

3.4.2 Membranes of Varied Thickness

The curvature of the membrane around the lip of the pore decreases with the membrane width h ; increasing the thickness should decrease the energy cost of pore formation, thereby increasing the critical radius r_c . It was shown in [53] that for a homogeneous film, where the free energy cost of pore formation is derived entirely from edge tension, the scaling should be linear as $r_c = h\pi/4$. To examine this scaling for the simulated lipid membrane, we changed the membrane thickness by altering the number of particles per lipid tail.

Initially, the tail length was increased from 6 to 7 particles. The resulting membrane was found to be stable and at equilibrium with 29,200 lipids in the same simulation box, i.e., a lipid density increase of 1.74%. The resulting membrane had a width of approximately $h = 8.63$ (8.15% thicker than the original membrane).

Results for this set are shown in Figure 3.34. The single additional tail particle induced an upward shift in the critical ratio, to $1.48 < r_c/h < 1.59$, an increase of between 0.68% – 18.66% from the 6 particle case. In addition, there was a notable change in the timescale on which pores evolved away from the critical region. In particular, the pore initialized at $r_0/h = 1.59$ was nearly

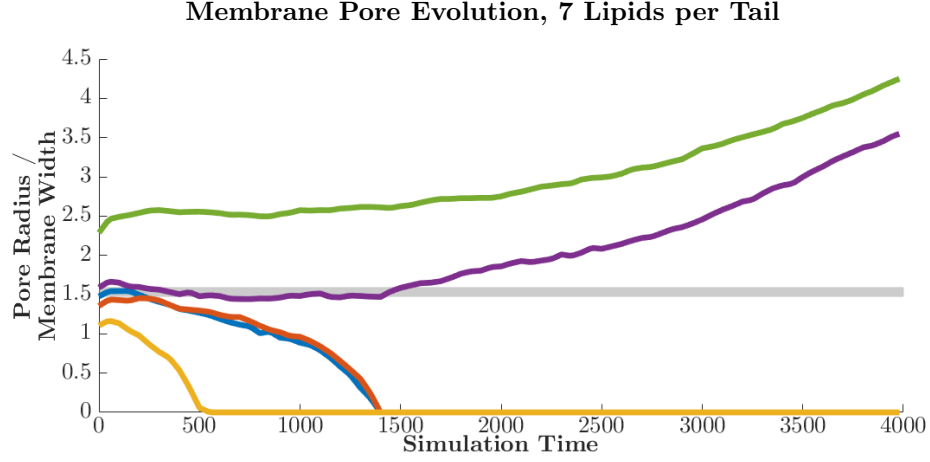


Figure 3.34: Membrane pore size over time from Eq. (3.41). Size (y axis) uses the new average membrane width $h = 8.63$. The shaded region corresponds to the new range for the critical radius, $1.48 < r_c/h < 1.59$.

stable for the entire duration of the previous simulations ($t \approx 2000$) before eventually opening up into supercritical growth. Figure 3.34 also demonstrates the stochastic nature of the dynamics at this spatial scale: in these realizations, the pore with initial size $r_0/h = 1.48$ closed faster than an initially smaller pore with radius $r_0/h = 1.36$.

Next, the tail length was increased further to 12 particles. The resulting time series can be seen in Figure 3.35. For very small pores, the behavior was unchanged, with the hole quickly being sealed. For pores of moderate size, the doubling of lipid tail length afforded significantly increased stability – above some size threshold, all pores in the membrane are stable indefinitely, showing a marked contrast with the behavior of the membranes of Figures 3.32–3.34. The largest simulated pore ($r_0 = 50$) required the simulation box be expanded to $240 \times 240 \times 40$ to avoid interference from periodic boundary effects.

Tail lengths of 8 and 9 lipids were also considered, but were found to produce nearly identical results to the case of 12 lipids per tail. The simulated bilayer membrane thus exhibits a sharp change in stability as the number of lipids per tail increases from 7 to 8.

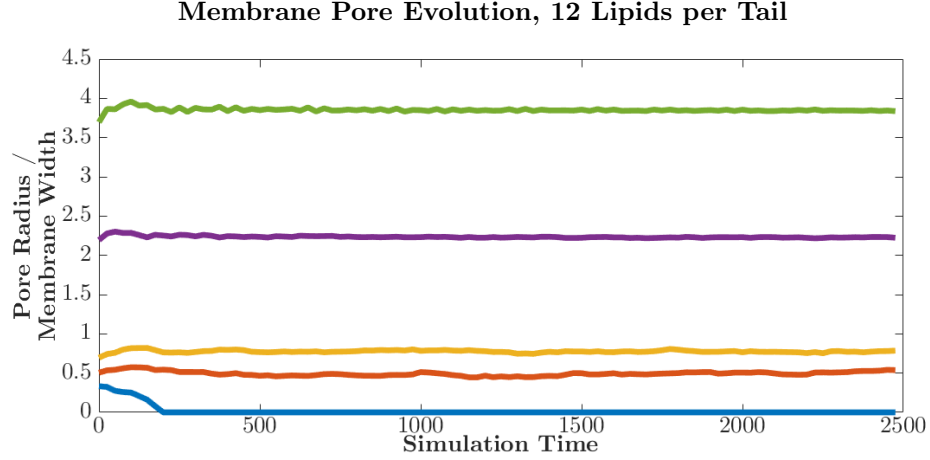


Figure 3.35: Pore size over time via Eq. (3.41) for a membrane with 12 particles per lipid tail. Only the smallest pore simulated was not stable for the duration of the simulation. This membrane had a width of $h = 13.49$, or 69.05% thicker than the membrane of Figures 3.32-3.34.

3.4.3 Pores in Finite Membranes

To further understand this change in stability, we finally considered the case of a finite membrane, i.e., a free-floating square patch of membrane of finite size. To simulate such a membrane, lipids placed on a lattice in the initialization phase were truncated a fixed distance of 5 units away from the edge of the simulation box. The empty space left by truncating the membrane was included in the region of initialization for fluid particles.

All simulated finite membrane pores invariably sealed, regardless of initial radius. The membrane with 6 particles per lipid tail, which formerly exhibited a critical pore radius, transitioned through a metastable torus configuration to a layered cluster (see Figure 3.36). The membrane with 12 particles per lipid tail, whose pores were stable above a small threshold radius, sealed its pore but remained stable in a finite bilayer disc. We hypothesize that the existence of a critical radius of pores in the periodic case corresponds directly to the stability of the finite membrane; a stable finite membrane prevents pores from expanding regardless of their initial size.

In net, many of the experimental findings about the polymer films of [53] were also observed in our dissipative particle dynamics simulation of a bilayer membrane. In particular, the existence

Time Evolution of Perforated Finite Membranes

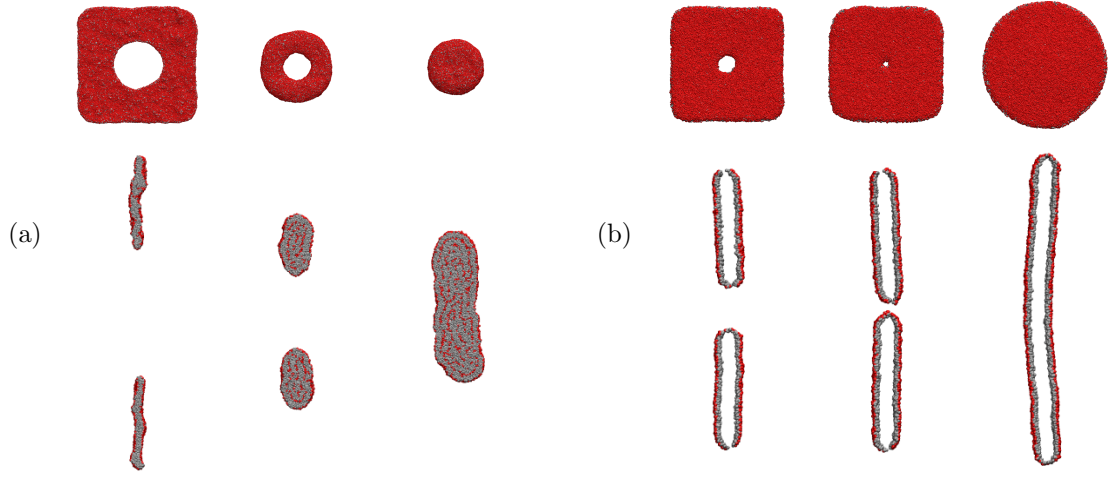


Figure 3.36: Rendered images of simulation results for truncated membranes; time increases left to right. Front and internal side views for the original membrane with 6 particles per lipid tail are shown in (a), while the modified membrane with 12 particles per lipid tail is shown in (b). The original membrane seals its pore but simultaneously transitions into a layered cluster. Conversely, the thicker membrane seals its pore but remains in a stable finite disc indefinitely. The interior of the membrane in (b) comprises only tail particles and is not shown.

of a critical pore radius was observed in the periodic simulations when membrane tails comprised 6–7 monomers. Ilton et al. explain the mechanism for such a phenomenon by writing the energy cost of pore formation $\Delta G(r)$ as a function of the pore radius: $(A - 2\pi r^2)\gamma$, in terms of the surface area A of the pore edge and the per-area surface tension of the film γ . Modeling the pore edge as the inner half-surface of a regular torus with diameter h (the membrane width), the edge surface area A is given by $\pi^2 h r - \pi h^2$. The resulting cost $\Delta G(r)$ is a concave function maximized when $\frac{\partial}{\partial r} \Delta G(r) = (\pi^2 h - 4\pi r)\gamma = 0$, yielding a critical radius $r_c = h\pi/4$ which scales linearly with the thickness h of the film. The cost $\Delta G(r)$ is a barrier for pore formation: once the critical size is reached, further expansion of the pore begins to reduce the free energy.

This expression ignores any energy cost associated with the molecular structure of the membrane; the authors also describe a modified argument for a diblock film, which has been used as a simple model of a lipid bilayer membrane in theoretical work [76]. Due to the additional cost of rearranging molecules on the curved surface around the pore, they predict a critical radius larger than for a

homopolymer film by a factor proportional to the nondimensional curvature L/h , where L is the equilibrium thickness of the lamellar layers. Our simulation results agree in this respect: critical radii of the lipid membranes with 6–7 monomer tails were found to be $1.34h < r_c < 1.47h$ and $1.48h < r_c < 1.59h$, respectively, compared to the homopolymer $r_c = h\pi/4 \approx 0.79h$.

There also exist significant differences between our simulations and the work of Ilton et al., most notably in the case of the thicker membrane. As the diblock copolymer correction term scales with the nondimensional curvature, its influence should decay as the membrane width increases, reaching the same limit of $r_c/h = \pi/4$ as $h \rightarrow \infty$. In contrast, our simulations of a thicker lipid bilayer membrane showed pores above a certain size to be stable indefinitely, i.e., a critical radius above which pores expanded no longer existed. This suggests that the structure of a lipid bilayer membrane is fundamentally different than the structure of a diblock copolymer. Unlike the polystyrene film and PS-*b*-PMMA diblock copolymer of [53], whose critical radius was a continuous function of membrane width, the simulated bilayer membrane exhibits a phase transition from a regime where critical radius relates to thickness (≤ 7 monomer lipid tails) to a regime where arbitrarily large pores are stable (≥ 8 monomer lipid tails).

Our simulations additionally provide insight into the time scale on which growth occurs. Although pores above the critical radius (in simulations where a critical radius existed) were found to increase in size roughly exponentially, the time scale of the exponential growth was markedly different between the membranes with tail lengths of 6 and 7 particles. To compare these timescales, the time series for the radius of the smallest supercritical pore was fit to an exponential $r_0 e^{t/\tau}$ in terms of the characteristic time τ . The thinner membrane (3.4.1) was found to have a characteristic time of $\tau_1 \approx 900$, while the slightly thicker membrane (3.4.2) was found to have a characteristic time of $\tau_2 \approx 2000$. The time scale of pore evolution for lipid bilayer membranes is thus significantly affected by the structure/width, potentially in addition to chemical properties (in this context, the force coefficients for particle interaction, which were not changed between simulations).

Since the DPD method explicitly models the solvent and does not make equilibrium assumptions,

it is also suitable for examining the behavior of lipid membrane pores in the presence of fluid flows or pressure gradients, such as those observed in biological cells during movement. Future work can examine the stability and dynamics of such pores in a variety of contexts of interest in cell biology.

3.5 Polymerization of Actin Filaments

Many of the mechanical interactions of eukaryotic cells are mediated by the cytoskeleton, which is a network of actin filaments, intermediate filaments, and microtubules in the cell cytoplasm. In particular, controlled deformation and remodeling of the cytoskeleton via controlled assembly, cross-linking, and disassembly of the skeleton components are essential mechanisms behind cell motility [52, 102]. In this section, we move toward a DPD model of the actin cytoskeleton driving the membrane forward via polymerization, which exerts and sustains mechanical force due to the high rigidity of actin polymers [90, 145]. As the DPD model explicitly models nonequilibrium flows in the fluid, it may provide insight into the flows which develop as a result of different behaviors in the cytoskeleton, especially the influx at the leading edge of cells which can oppose the cell membrane load and thus increase the rate of polymerization [62, 79].

As a first step, we couple the three-dimensional DPD membrane of Section 3.3.2 with a bead-spring model for actin as in Section 3.1.4. Compared to DNA, actin is quite rigid; the persistence length of an actin filament is about $17.7 \mu m$ [38], over 300 times longer than the corresponding length for double-stranded DNA. To account for this difference, we modify the bead-spring model with a stiff bending energy using the three-body potential $E_{ijk} = 2000(1 - \cos(\phi_{ijk} - \phi_0))$, where ϕ_{ijk} is the angle between consecutive bonds ij and jk and $\phi_0 = 180^\circ$ is a resting angle as in Section 3.3.3. In practice, this potential was not sufficient to achieve sufficient rigidity in the filament, and so another set of three-body potentials $E'_{ijk} = 2000(1 - \cos(\phi'_{ijk} - \phi_0))$ was applied to the angles ϕ'_{ijk} made between beads i, j, k satisfying $i + 5 = j = k - 5$, i.e., every fifth bead. Springs between beads had stiffness $k = 200$ and a resting length of 0.5 units. The pairwise DPD

Flexing Actin Filament Initialization

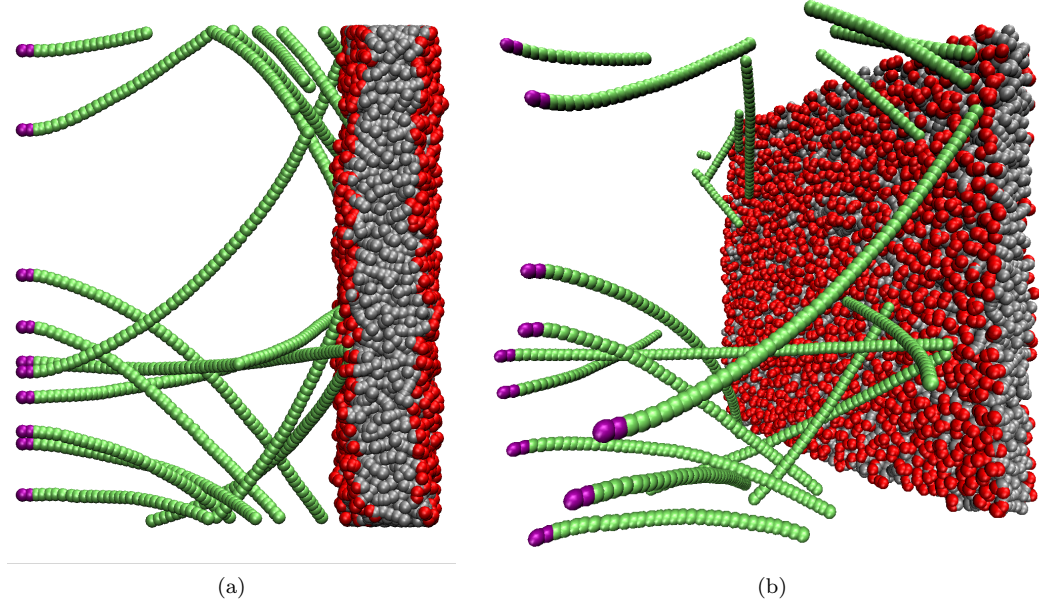


Figure 3.37: (a) Side view of flexed actin filaments (green) and 3D membrane (red lipid heads, silver lipid tails) at initialization. The purple particles on the far left end of each filament are fixed, forcing the filaments to relax into the membrane. (b) 3D view of same system. Fluid particles (not shown) fill the space on both sides of the membrane.

interactions between beads matched the parameters used for the surrounding fluid.

A $48 \times 32 \times 32$ periodic simulation box was filled with a 1480-lipid DPD membrane in the y and z dimensions, with the x position centered in the box at $x = 24$. Ten actin filaments, with potentials described above, were initialized at uniformly random y and z locations in the $x = 0$ plane, then extended outward in the x direction to a length of 70 beads. The filaments were then bent away from the membrane to avoid intersection, as seen in Fig. 3.37; the bent filaments thus have a large amount of potential energy which will be converted to kinetic energy in the membrane as the filaments unfurl, driving the membrane forward. The first two beads in each filament were fixed, ensuring that the actin relaxes in a direction orthogonal to the membrane surface. Empty space surrounding the filaments on either side of the membrane was filled with water beads as in Section 3.3.2, and the velocity-Verlet algorithm was again used to propagate the system forward in time with a step size of $\Delta t = 0.002$.

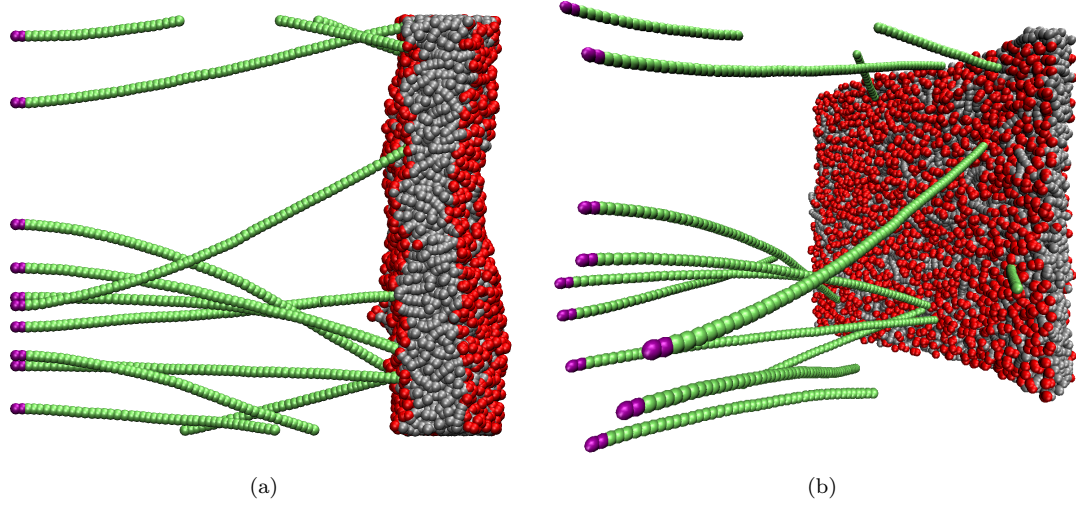
Flexing Actin Filaments and Membrane, $t = 240$ 

Figure 3.38: (a) Side view of flexed actin filaments and 3D membrane at $t = 240$. (b) 3D view of same system. The actin filaments unfurl and exert forward pressure on the membrane, creating a flow of fluid with a feedback loop as the membrane is driven ahead.

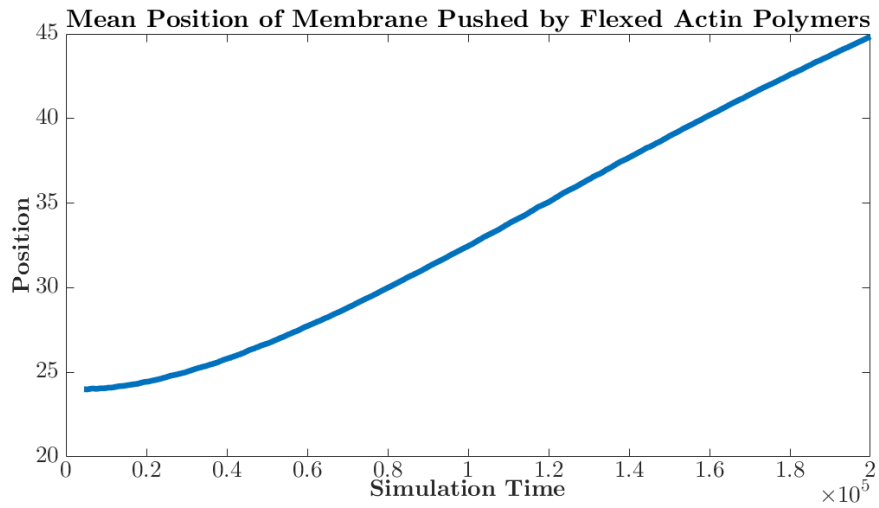


Figure 3.39: Membrane mean x position over time for the simulation shown in Figs. 3.37 and 3.38. After an initial period of acceleration, the forward motion of the membrane achieves a terminal velocity of roughly 12.34 units per 100,000 steps ($v = 0.0617$).

Schematic of DPD Actin Polymerization

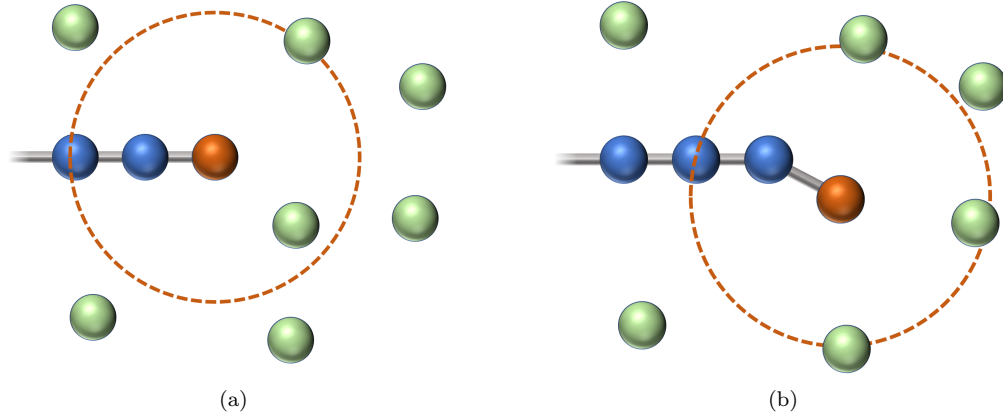


Figure 3.40: (a) Actin filament (blue) with marked terminal end (orange) in the presence of freely diffusing monomers (green). Free monomers within a specified distance of the free end, shown with the dotted circle, result in polymerization with a fixed step-by-step probability. (b) Filament and monomer after polymerization. The newly bonded monomer is the new terminal end, which will then repeat the process by forming new bonds with other free monomers.

Results after 120,000 steps appear in Fig. 3.38. The unfurled actin filaments resulted in net forward motion of the membrane in the x direction, shown in Fig. 3.39. After an initialization period where the filaments accelerate the membrane, the membrane achieves a steady speed of roughly 12.34 units per 100,000 steps (or a unitless velocity $v = 0.0617$), measured via a linear fit to the curve in the range 100000 – 200000 steps. Fluid on the far side of the membrane cycles across the periodic boundary, creating a feedback loop of forward pressure.

While this simple model captures interactions between the rigid filaments and the DPD membrane, actual membrane deformation by the actin cytoskeleton is often a result of polymerization and depolymerization at the actin filament ends facilitated by actin-binding proteins such as gelsolin and cofilin [120]. We thus modify the model to allow for polymerization of the actin polymers; rather than driving the membrane forward with potential energy in the flexed filaments, the membrane will be subject to forward pressure from the polymerizing ends of the filaments.

A schematic of the polymerization approach used is shown in Fig. 3.40. The unbonded end of the filament is marked as a special monomer type which seeks to create additional bonds within some

fixed radius around its position; freely diffusing monomers within this radius have a specified step-by-step probability of forming a bond with the free end of the actin, thereby changing the previous terminal monomer to an internal monomer and the newly bonded monomer to the marked terminal monomer. Here, we choose such bonds to form within a radius 0.8 of the free end (compared to the 0.5 resting length of the harmonic bonds) with probability 1, i.e., any freely diffusing monomer which becomes close enough to the polymerizing end of the filament will immediately join the filament.

A practical consequence of newly formed bonds is the creation of new three-body potentials corresponding to the filament rigidity; as the instantaneous angle formed by the new monomer may be quite sharp, this can result in numerical instabilities due to the large magnitude of the resulting energies. We address this concern with two modifications: first, the long-range three-body potentials E'_{ijk} are removed, with the coefficient of the local three-body potentials E_{ijk} reduced to 100, and second, the bond stiffness k is reduced to 100. This sufficiently reduces the energy of the created bond to be quickly dissipated by frictional forces against the surrounding fluid as it relaxes to the equilibrium position.

A consequence of the changes required for numerical stability is a reduction in the effective stiffness of the model actin filaments. In order to ensure that the polymerizing ends of the filaments push against the membrane, a subset of lipid heads on the near surface of the membrane are tagged as actin attractors (the yellow particles in Fig. 3.41(a)). As lipids diffuse within the membrane surface, these attractors become spread throughout the membrane; attractive potentials between the polymerizing end of the actin and these attractor lipids ensure that the actin polymerizes in a direction orthogonal to the membrane surface. The specific potential used was a V-shaped potential centered at a distance $r_{ij} = 1.5$, which resulted in a fixed force of magnitude $f_v = 100.0$ incentivizing terminal monomers within a distance $r_c = 3$ to remain at the energy-minimizing distance.

Finally, the side of the membrane populated by actin filaments was filled with free monomers at a numerical density $\rho = 0.1$. These monomers diffuse through the water according to the DPD equations of motion until stochastically entering the radius of polymerization, at which point monomers

attach to the free filament ends. The resulting system at initialization can be seen in Figure 3.41, which uses 16 actin filaments of 30 beads with pinned sections spaced evenly at the vertices of a 4×4 grid. A hard reflecting boundary at $x = 0$ applied only to free monomers prevents them wrapping around the periodic simulation box to the opposite side of the membrane; for all other particles, boundaries are periodic as before.

The results of evolving the system forward in time by 120,000 timesteps $\Delta t = 0.001$ are shown in Figure 3.42. As monomers diffuse within the radius of polymerization, they bond to the free end of the simulated filaments, increasing the filament length and exerting forward pressure on the membrane. Fluid on the actin side of the membrane expands into the vacuum as the membrane moves forward, pulling more free monomers to the polymerization interface. The net forward motion of the membrane, seen later in Fig. 3.43(a), follows a similar progression to the flexing filaments, with a period of acceleration followed by a roughly constant terminal velocity; here, the observed long-time velocity of the membrane was roughly 4.42 units per 100,000 steps, corresponding to a unitless velocity of $v = 0.0439$, measured via a linear fit in the range 150000 – 250000 steps.

This model provides a unique opportunity to study the force-velocity relation, i.e., the relationship between the forward velocity of the membrane deformation in a motile cell with the applied force in a direction opposing motion, which is a central concept of cell movement in crowded environments [146]. Though it is clear the relation should be monotonically decreasing (an increased force working against the membrane will result in a lower forward velocity), the specific shape of the force-velocity relation for actin-based membrane protrusions is hotly debated in the literature, with theoretical and experimental work suggesting the relationship may be concave-up or concave-down in different scenarios [88, 111]

The opposing force is easily incorporated as a fixed force applied to all membrane lipids in the $-x$ direction; by increasing the magnitude of this force until it overcomes the forward pressure of polymerization, we can map out the effective force-velocity relation of the polymerizing actin DPD model. The evolution of the mean membrane position for a selection of retarding force magnitudes

Polymerizing Actin Filament Initialization

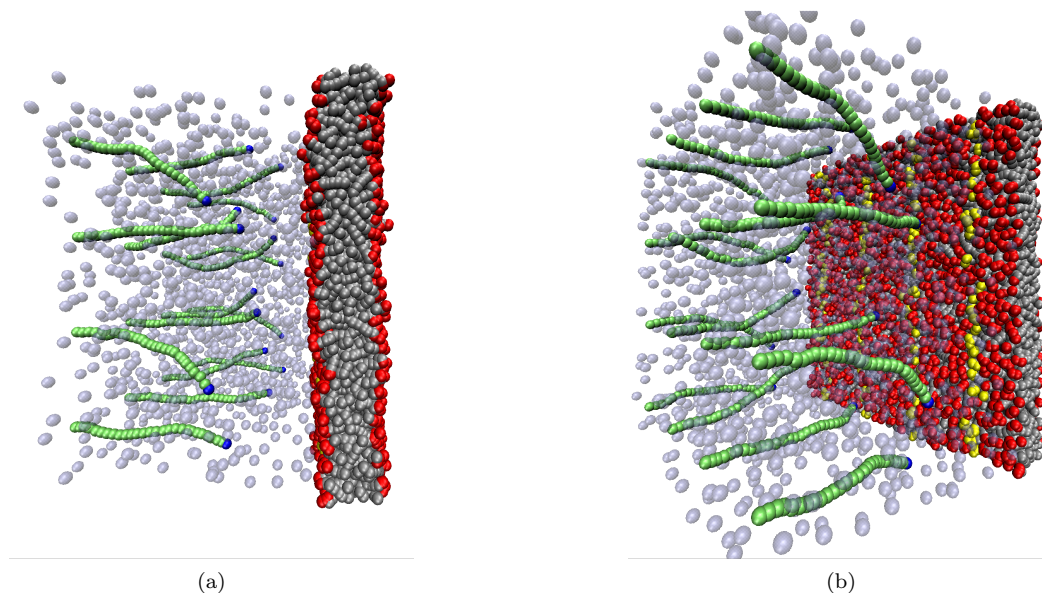


Figure 3.41: (a) Side view of polymerizing actin filaments (green with blue terminal monomers) and 3D membrane (red lipid heads, silver lipid tails) at initialization. Faded particles are freely diffusing monomers. (b) 3D view of same system. Fluid particles not shown.

Polymerizing Actin Filaments and Membrane, $t = 240$

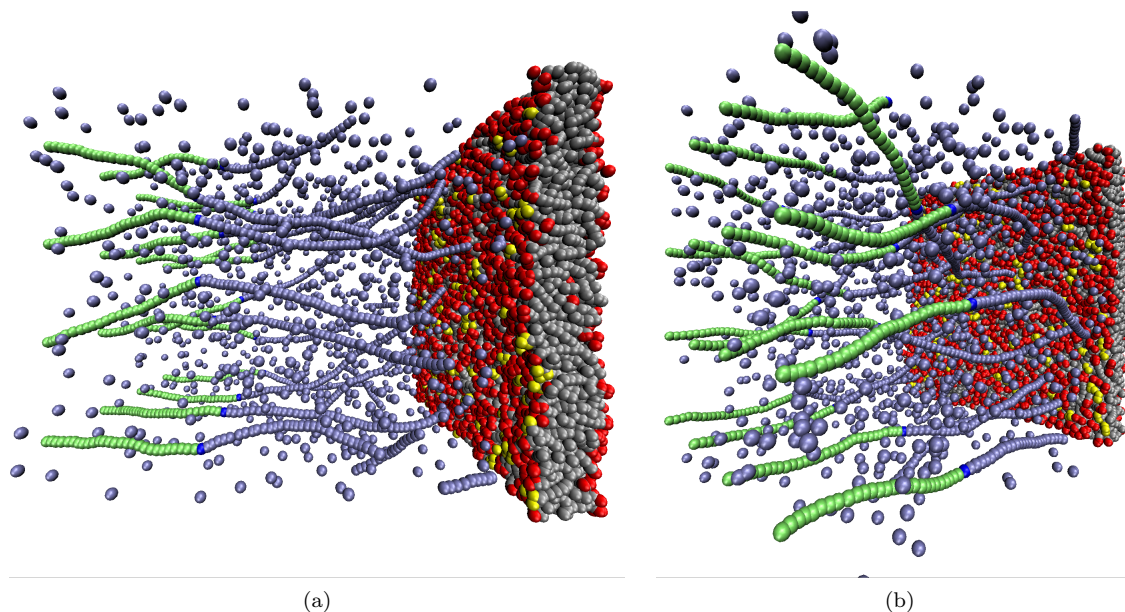
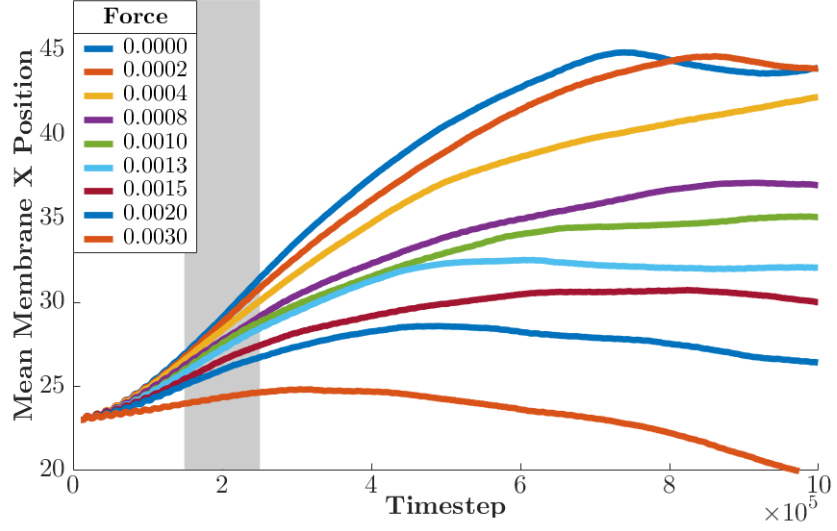
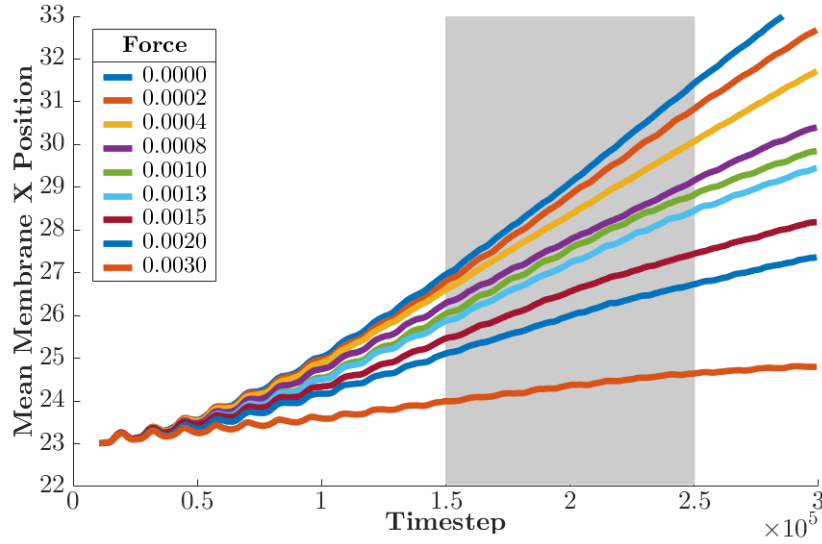


Figure 3.42: (a) Side view of polymerizing actin filaments and 3D membrane at $t = 240$. Diffusing monomers (blue) have been made opaque to show the effects of polymerization. (b) 3D view of same system. The polymerizing filaments exert a forward pressure on the membrane, driving net forward motion which creates a flow drawing additional actin monomers to the polymerization interface.

Evolution of Mean Membrane x Position Due to Polymerizing Actin



(a)



(b)

Figure 3.43: (a) Average membrane x position as a function of timestep for the polymerizing actin model in the presence of a retarding force. Colored curves correspond to force magnitudes specified in the legend applied to each dissipative lipid bead. Shaded region from 150000 – 250000 steps was used to estimate the membrane velocity. (b) Closer view of (a) for small times.

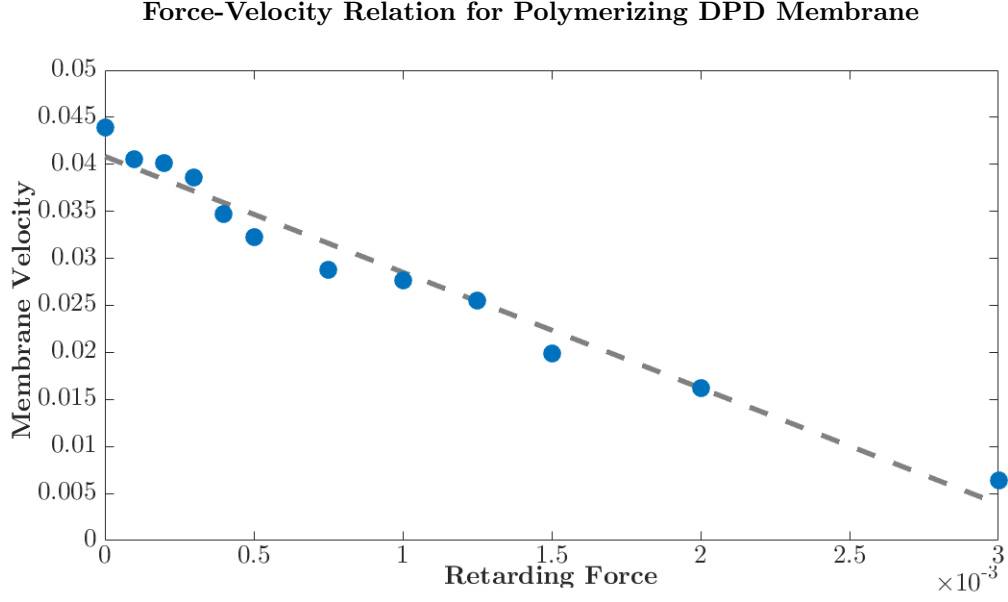


Figure 3.44: Mean membrane velocity v from $t = 150$ to $t = 250$ as a function of the retarding force f . The dotted line shows the least-squares linear fit $v = 0.0408 - 12.315f$.

is shown in Figure 3.43, with the curve corresponding to zero force matching the base model shown in Figures 3.41 and 3.42. The low-amplitude oscillations at small times are a result of the higher particle density on the free-monomer side, which quickly equilibrates as fluid particles wrap across the periodic boundaries. For simulations with small opposing forces, the membrane ultimately strikes the far edge of the simulation box (size 48 in the x dimension), which is a hard reflecting boundary to the membrane (but allows fluid to pass freely). For simulations with large opposing forces, the polymerizing filaments eventually dislodge from the membrane or are bent under the pressure of the applied force, eventually resulting in the membrane moving back in the $-x$ direction.

A linear fit to the position curves in the shaded region yields the force velocity relation shown in Figure 3.44, which includes additional data from intermediate forces. In particular, the data show the force-velocity relation for the DPD polymerizing actin model to be roughly linear. The least-squares linear fit, shown as a dotted line, decays to zero at an applied force $f = 0.0033$.

In contrast to most other particle-based approaches to modeling actin polymerization, which model the membrane as a rigid plate or disc (e.g., [55, 70]), the simulations described here fully

model the membrane lipids, allowing membrane fluctuations to influence the availability of free monomers at the actin filament ends. Future refinements of the model, together with data collected from membranes and model filaments of varied physical properties, may provide additional insight into the true nature of the force-velocity relation governing the forward motion of actin-propelled eukaryotic cells in crowded environments.

3.6 Vesicles Propelled by Actin Networks

One of the major challenges in understanding eukaryotic cell motility is to precisely relate actin polymerization to mechanical force generation. Polymerizing actin networks were first studied in the context of cell motility for bacterial pathogens such as *Listeria*, which were found to use the actin machinery to propel themselves forward [34]. Other objects can also be driven forward by the stresses of a polymerizing actin tail: polystyrene beads and, more recently, synthetic phospholipid vesicles coated with proteins such as ActA which facilitate actin polymerization were also found to observe this behavior [13, 133]. The latter case in particular provides some insight into the underlying mechanical forces, as the lipid vesicles deformed into “tear drop” shapes which were blunted in the direction of motion and stretched toward the actin comet tail, i.e., the network of cross-linked actin polymers created in the wake of forward motion. This shape may suggest a mechanical force profile which effectively squeezes the vesicle forward, with polymerizing actin on the slanted sides and a backward stress at the vesicle rear exerting a net-forward force [134].

While the three-dimensional DPD polymerization model of the previous section is effective at simulating the effect of polymerizing actin filaments on the forward propulsion of a periodic membrane patch, it does not fully model the complicated topography of the actin network at such scales. In this section, we use the two-dimensional DPD membrane of Section 3.3.3 to examine a lipid vesicle subject to the types of position-varying applied forces suggested by experimental observations.

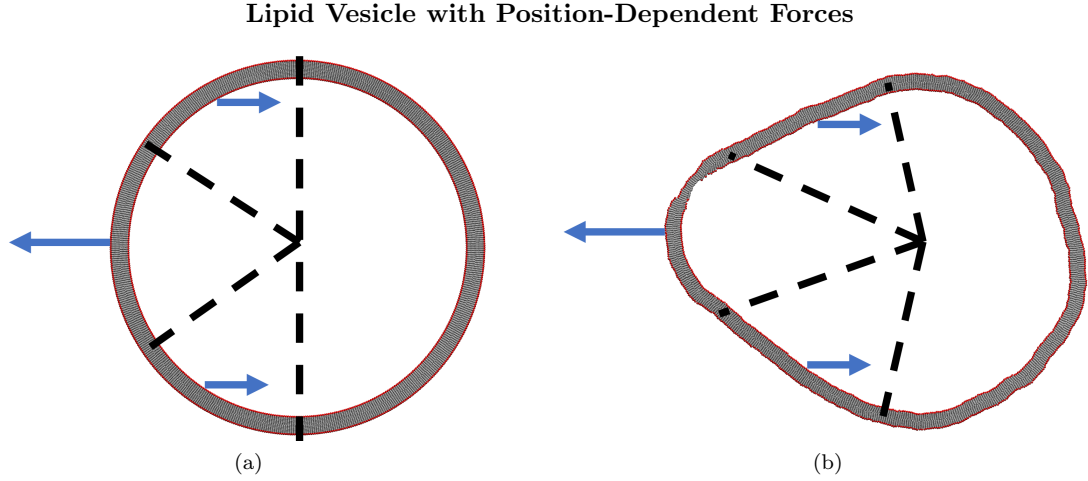


Figure 3.45: (a) 2D lipid vesicle (red lipid heads, silver lipid tails) at circular initialization. Water with numerical density $\rho = 2.1$ inside and outside the vesicle is not shown. The membrane is divided into four regions which experience different applied forces; sections of the top and bottom are driven forward by an applied force $f = 0.15$ in the x -direction, while a section on the left is pulled backward by an applied force $f = -0.25$. As these sections are equal in length, the net force on the vesicle is forward. (b) Lipid vesicle after equilibration under applied forces.

We create the vesicle by using the two-dimensional membrane model with modified lipids possessing one head bead and six tail beads each (i.e., the lipid described in Section 3.3.3 with one tail removed and a head of unit size and mass). The lipids are placed in a 500×200 periodic simulation box with head beads facing toward the fluid interface on both sides of a circular contour with radius 70, shown in Figure 3.45(a). As the contour length of the inner membrane interface is shorter than the corresponding length of the outer interface, the number of lipids in the two layers differs; stability was achieved by targeting a fixed spacing of 0.8 between lipid heads, resulting in a total of 1,049 simulated lipids. The regions inside and outside the membrane are filled with standard DPD water beads at a numerical density $\rho = 2.1$, matching the average inter-particle spacing of the $\rho = 3$ case in three dimensions. The resulting membrane is stable up to small thermal fluctuations when stepped forward via the velocity-Verlet scheme with timestep $\Delta t = 0.03$ at the dimensionless temperature $T = 0.2$.

We divide the vesicle into four sections, as shown in Figure 3.45(a), to apply a position-varying artificial force. The right half of the vesicle is chosen as the side of the vesicle away from the actin

interface, and so is not subject to additional forces. The left half of the vesicle is divided into three equal sections, with the top and bottom sections subject to an applied per-particle force $f = 0.15$ in the positive x direction; these sections correspond to regions of the vesicle surface where actin polymerization exerts a forward pressure on the membrane, as observed for the lipid membrane patch in the previous section. The final membrane section, at the rear of the vesicle, would be directly in the middle of the actin comet tail; while this section of the actin network can also polymerize, we choose to use a retarding per-particle force $f = -0.25$ corresponding to the notion of polymerization as a diffusion-limited reaction. Namely, free monomers coming from outside the network may bond to a filament near the network edge or become otherwise impeded before reaching the rear actin-membrane interface, and so the rate of polymerization may be limited by availability of monomers. In conjunction with the drag force due to pulling the comet tail forward through the fluid medium, this may result in a negative mechanical force locally, though the total force on the vesicle should be nonnegative if its forward motion is unobstructed. The vesicle after equilibration to the applied profile is shown in Figure 3.45(b), which reproduces the tear drop shape.

The forward evolution of the vesicle through the channel is illustrated in Figure 3.46. Due to the net forward force, the vesicle accelerates in the positive x direction, though it maintains a roughly constant shape up to thermal fluctuations. The stability of the tear drop shape under this force profile suggests that the mechanical force profile of a polymerizing actin gel may indeed correspond to the stresses suggested in [134], highlighting the potential of diffusion-limited polymerization in the comet tail center and forward pressures from polymerization at the actin-membrane interface as mechanisms explaining actin-based cell motility.

There are a number of future directions for study of the actin-propelled lipid vesicle. The simulated force model described here is an artificial set of forces corresponding to hypothesized mechanics, chosen to reproduce the vesicle shape observed in experiments; dissipative particle dynamics simulations may be in the unique position of being able to directly model those mechanics at the scale of

Forward Evolution of Lipid Vesicle with Position-Dependent Forces

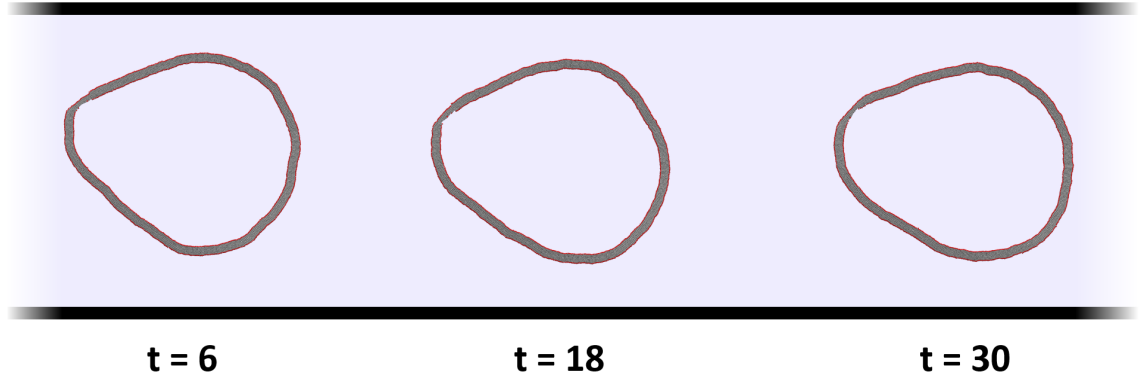


Figure 3.46: Time evolution of the vesicle of Fig. 3.45 rightward in the fluid channel. Water beads (not shown) fill the shaded region of the channel, i.e., both inside and outside the vesicle.

the entire vesicle, which would be the most sophisticated mathematical model of actin-based propulsion in the literature. These mechanisms could be modeled by extending the polymerization model of Section 3.5, both by scaling the size of the membrane patch and by modeling explicit structure in the actin network, such as cross-linkage due to actin-bundling proteins like fimbrin [77]. An explicit DPD model, if feasible, could test the hypothesized mechanism of diffusion-limited polymerization and provide a deeper look into the forces and stresses of polymerization-based propulsion.

Chapter 4

High-Performance Bayesian Uncertainty Quantification

As shown in the varied applications of particle simulations to the nanoscopic systems of Chapter 3, mathematical models are vital to the study of complex materials such as the soft matter in biological cells [72, 89, 122, 130]. From an experimental perspective, the complexity of the systems coupled with the nano- or even pico-scale length and time scales can make direct observation challenging; from an analytical perspective, the ground truth is often unknown, and the sheer number of components may prove attempts at a complete description of the system to be infeasible, even when the individual mechanics involved are well understood. In these scenarios, the comparative simplicity of models affords tractability at the cost of introducing uncertainty.

The most basic question that can be asked of a model, and the sole question which was examined in Chapter 3, is the forward problem: given a particular input, what output will be produced from the system? If the model is deterministic, this question may seem straightforward – the model must simply be applied and the output observed. Yet there are many potential sources of uncertainty: the output may be subject to measurement error, or the model may depend on parameters which

are themselves uncertain (see, e.g., [3, 141]), thereby propagating uncertainty forward. In the case where the model is itself stochastic (e.g., a branching process model of a disease), uncertainty may be a fundamental aspect of the system dynamics [2, 142]. The broad field which describes the mathematical treatment of models in the presence of uncertainty is often referred to as *uncertainty quantification* (UQ) [42, 82, 119].

The mathematical treatment of uncertainty in this context may be trivial – for example, accounting for a measurement error in the output (e.g., a Gaussian noise term) may be as simple as executing the forward problem, then adding a Gaussian random variable. In this case, the forward problem is solved only once, so the resulting distribution is easily sampled from, even when the model itself has considerable complexity.

Conversely, many other problems become significantly more difficult when uncertainty is introduced [142]. The classical inverse problem is one such example: what input (or set of inputs) yields a particular output? Especially if the model is complex, the model itself is unlikely to be invertible; moreover, depending on the form of the uncertainty, the set of inputs which must be considered can be prohibitively large [131]. To illustrate, consider that in the previous example of additive Gaussian noise, which is among the simplest models of measurement uncertainty, the entire input space could feasibly have produced any particular output, so long as the Gaussian noise chanced to be exactly the difference between the true output and the target output. Even if the input space is finite, the number of times the model must be applied makes this “solution” impractical for all but the simplest and most efficient models.

The Bayesian approach to this parameter estimation problem, i.e., Bayesian uncertainty quantification, attempts to leverage tools from Bayesian statistics to make the problem more tractable [6, 141]. As will be explored in the following sections, there are a variety of approaches to sampling from high-dimensional and complex distributions in the Bayesian setting [29]. As the goal of parameter estimation is the probability distribution corresponding to the relative likelihood of particular parameter values generating the observed output, this machinery is readily applied.

In this chapter, we introduce an approach to Bayesian parameter estimation which is practical even for comparatively expensive models, such as those necessary to study in atomistic detail the dynamics of biological polymers, membranes, and gels. Rather than running the forward problem (e.g., a particle simulation of DNA diffusing in a nanofluidic channel) for a prohibitively large set of parameters, the algorithm uses a Markov chain Monte Carlo approach which evaluates the problem only at a discrete set of points. More importantly, the specific algorithm used, known as transitional Markov chain Monte Carlo (TMCMC), is massively parallelizable, and so the high cost of the model can be offset by the ability to execute many copies of the model simultaneously [4, 15, 17, 42].

4.1 Framework

To approach the parameter estimation problem using a Bayesian methodology such as TMCMC, we must first formally introduce the Bayesian formulation of uncertainty quantification [6]. The hallmark of Bayesian statistics is the assumption of prior distributions for unknown parameters – in other words, probabilities are a representation of degrees of belief; any knowledge of the unknowns *a priori*, whether from previous experience or from physical limitations, informs belief in the parameter values even before a statistical model is applied [131].

In the context of parameter estimation, the unknown parameters are the inputs to some mathematical model M (including not only model parameters, but also initial conditions, boundary conditions, etc.), which then produces a set of output quantities governed by the form of the model. A simple bacterial growth model, for example, might be parametrized by the probability p that a particular bacterium divides in a certain time period, but would also take as input the number X_0 of bacteria present at time 0. The Bayesian assumption is that the n inputs, jointly parametrized by $\boldsymbol{\theta} \in \mathbb{R}^n$, are random variables with joint prior density $\pi(\boldsymbol{\theta}|M)$.

Given a particular set of inputs $\boldsymbol{\theta}$, the model M produces a set of m data, referred to jointly as $\mathbf{D} \in \mathbb{R}^m$, according to some likelihood $p(\mathbf{D}|\boldsymbol{\theta}, M)$ – this is the content of the model itself, i.e., the

mapping from inputs to outputs, which is probabilistic owing to uncertainties and stochasticity in the model M . The parameter estimation problem instead targets the posterior $p(\boldsymbol{\theta}|\mathbf{D}, M)$, i.e., the mapping from the outputs back to the inputs, which is also informed by the prior beliefs $\pi(\boldsymbol{\theta}|M)$. In this setting, $\mathbf{D}$ is known and $\boldsymbol{\theta}$ is unknown, and so it should be stressed that $p(\mathbf{D}|\boldsymbol{\theta}, M)$ is purposefully referred to as a likelihood since it is *not* a distribution in $\boldsymbol{\theta}$.

Using Bayes' theorem, the posterior distribution can be computed as

$$p(\boldsymbol{\theta}|\mathbf{D}, M) = \frac{p(\mathbf{D}|\boldsymbol{\theta}, M)\pi(\boldsymbol{\theta}|M)}{\rho(\mathbf{D}|M)}, \quad (4.1)$$

which additionally introduces the evidence $\rho(\mathbf{D}|M)$ of the model class, given by the multi-dimensional integral

$$\rho(\mathbf{D}|M) = \int_{\mathbb{R}^n} p(\mathbf{D}|\boldsymbol{\theta}, M)\pi(\boldsymbol{\theta}|M)d\boldsymbol{\theta}.$$

In the case where the model M is one particular model in a parametrized class of models, the evidence $\rho(\mathbf{D}|M)$ functions as a measure of the degree to which the model matches the observed data, thereby playing a critical role in model selection [7, 143]. For the parameter estimation problem, however, it should be noted that ρ does not depend on $\boldsymbol{\theta}$, and so its role in this chapter is simply to function as a normalization constant for the posterior.

To proceed further, assumptions must be made about the specific form of the likelihood $p(\mathbf{D}|\boldsymbol{\theta}, M)$. Here, we will assume that the measured data $\mathbf{D}$ are generated according to the following model prediction equation:

$$\mathbf{D} = g(\boldsymbol{\theta}|M) + \mathbf{e}, \quad (4.2)$$

where $g(\boldsymbol{\theta}|M)$ is the predicted output for a specific set of parameters $\boldsymbol{\theta}$ and $\mathbf{e}$ is a prediction error due to, e.g., measurement, computational, or modeling errors. Specifically, we will make the simplifying assumption that the error $\mathbf{e}$ is normally distributed with zero mean and covariance matrix $\boldsymbol{\Sigma}$, which may involve additional unknown parameters. Since g is assumed deterministic, it follows that $\mathbf{D}$ is

also normally distributed, and so the likelihood $p(\mathbf{D}|\boldsymbol{\theta}, M)$ of the observed data can be written as

$$p(\mathbf{D}|\boldsymbol{\theta}, M) = \frac{|\boldsymbol{\Sigma}(\boldsymbol{\theta})|^{-1/2}}{(2\pi)^{m/2}} \exp\left[-\frac{1}{2}J(\boldsymbol{\theta}, \mathbf{D}|M)\right], \quad (4.3)$$

where

$$J(\boldsymbol{\theta}, \mathbf{D}|M) = [\mathbf{D} - g(\boldsymbol{\theta}|M)]^T \boldsymbol{\Sigma}^{-1}(\boldsymbol{\theta}) [\mathbf{D} - g(\boldsymbol{\theta}|M)]$$

is the weighted measure of fit between the model predictions and the measured data, $|\cdot|$ denotes determinant, and the parameter set $\boldsymbol{\theta}$ is augmented to include any unknown parameters in the covariance matrix $\boldsymbol{\Sigma}$ [7, 135]. It follows that sets of parameters $\boldsymbol{\theta}$ which reduce $J(\boldsymbol{\theta}, \mathbf{D}|M)$, which is often called the “fitness” of the parameter set $\boldsymbol{\theta}$ to the data $\mathbf{D}$, generally increase the likelihood; thus, the most likely sets of parameters (before consideration of the prior $\pi(\boldsymbol{\theta}|M)$) can be intuitively recognized as those whose output most closely matches the observed data.

While this explicit form of the likelihood can be straightforwardly calculated for any chosen set of parameters $\boldsymbol{\theta}$, each evaluation must reference the model prediction function g , which may be computationally intensive (e.g., a particle simulation). Since the parameters are assumed to be real-valued and the noise $\mathbf{e}$ spans the entire output space, computing the complete posterior $p(\boldsymbol{\theta}|\mathbf{D}, M)$ would then require an infinite number of evaluations of g . By formulating the problem in a Bayesian context, however, we gain access to the Bayesian statistician’s toolkit, which contains a number of methods for approximately sampling such distributions: in particular, the class of Monte Carlo algorithms known as Markov chain Monte Carlo (MCMC).

4.2 Markov Chain Monte Carlo (MCMC)

As we have defined it, the major goal of parameter estimation is the recovery of the posterior distribution $p(\boldsymbol{\theta}|\mathbf{D}, M)$. Via (4.1), there are two distributions which must be known in order to achieve this goal (the evidence ρ does not depend on $\boldsymbol{\theta}$). The first is the prior $\pi(\boldsymbol{\theta}|M)$ associated with the parameters $\boldsymbol{\theta}$ in a given model class – tautologically, this is “easy,” since it is (by definition)

prior knowledge. The second component is more difficult: the likelihood $p(\mathbf{D}|\boldsymbol{\theta}, M)$ must also be known. As has been mentioned, the explicit form of the likelihood can only be poked at by observing specific realizations of the forward problem.

Given infinite time, it would be possible to recover the likelihood by evaluating the model prediction function g for every valid set of parameters $\boldsymbol{\theta}$. To approach the problem in a feasible finite time, we instead turn to a set of stochastic sampling methods which have become omnipresent as computers have become more efficient: Markov chain Monte Carlo, often referred to as MCMC [6, 131]. Such methods rely on constructing Markov chains which have as their invariant distribution a specific distribution of interest (in the context of the parameter estimation problem, the posterior $p(\boldsymbol{\theta}|\mathbf{D}, M)$). There are a number of practical issues which surround the application of MCMC to parameter estimation: to understand the nature of these difficulties, and the ways in which they are addressed in the context of Bayesian UQ, we briefly review a subset of relevant MCMC algorithms.

The most fundamental such algorithm is the Metropolis algorithm [6]. Let $p(\cdot)$ be a target distribution defined on a state space S , and suppose a function $f \propto p$ is known (this is not as absurd an assumption as it might sound at first glance – in the Bayesian framework, if p is a posterior distribution, then a likelihood function can serve the role of f). Begin with any initial guess $X_0 \in S$. The Metropolis algorithm functions by iteratively calculating a new value X_{k+1} given the current value X_k in such a way that, as $k \rightarrow \infty$, the distribution of X_k approaches p . In this way, simply calculating the values X_k up to some large k produces approximate samples from p .

The specific method for updating the guesses X_k is to make a proposal Y , chosen at random according to a symmetric proposal distribution $q(\cdot, X_k)$ (e.g., a multivariate Gaussian centered at X_k). The algorithm then chooses whether to accept the proposal (in which case $X_{k+1} = Y$) or reject it (in which case $X_{k+1} = X_k$). By choosing the acceptance probability to be the ratio $f(Y)/f(X_k)$, the invariant distribution of the chain becomes the target distribution p . Convergence to the stationary distribution can be proven easily if q is chosen carefully – in the case of a Gaussian, its infinite support makes possible the transition to any state x with $p(x) > 0$ (irreducibility), while

the possibility of rejecting a proposal ensures aperiodicity [125].

Even this basic algorithm appears to have ready applications to the parameter estimation problem. Choosing the invariant distribution p to be the posterior $p(\boldsymbol{\theta}|\mathbf{D}, M)$, along with a standard proposal distribution q (a multivariate Gaussian will work for real-valued parameters), the only difficulty seems to be the calculation of some $f \propto p$ – this is the same problem as was encountered initially, where the likelihood function is not known, and it is not computationally feasible to estimate the likelihood over the entire parameter space. The key is that the Metropolis algorithm does not rely on the general form of f , but instead only relies on the calculation of f at various points chosen according to the proposal distribution q . Supposing the algorithm provides a guess Y for the set of parameters, the likelihood $f(Y) = p(\mathbf{D}|Y, M)$ can be estimated as in (4.3), assuming output is generated according to the model prediction equation (4.2).

The main difficulty of this approach has been alluded to already – the convergence to the stationary distribution is asymptotic, and the time to convergence can vary greatly (depending on q , the dimensionality, and the form of the likelihood). Specifically, the algorithm may get “stuck” sampling from a region where the fit appears to be locally optimal for long times, as the probability of transitioning to other regions of high density may be extremely low (this is known as a probabilistic *bottleneck*). This problem can be seen in Figure 4.1, which uses the Metropolis algorithm to approach a simple toy problem in one dimension. The result is that approximate sampling from the posterior $p(\boldsymbol{\theta}|\mathbf{D}, M)$ may require running the chain for longer than is computationally feasible.

Naïvely, one way to address this problem would be to choose an extremely broad proposal distribution q , thereby allowing the chain to jump across bottlenecks in a single step. Unfortunately, this has the second effect of greatly increasing the number of proposals which are rejected. Instead, we borrow ideas from a second MCMC method which is more robust to the problem of probabilistic bottlenecks. Simulated annealing is an approach to nonconvex optimization which introduces a *cooling schedule* to ensure that, in the short term, the chain can transition across regions where the likelihood may be extremely low [43, 48, 109, 125].

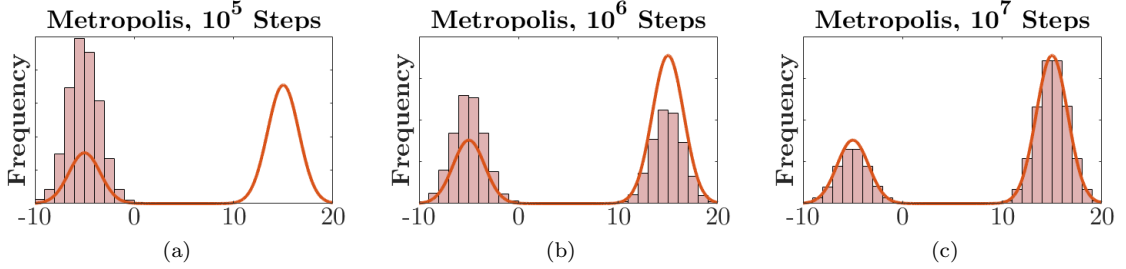


Figure 4.1: The Metropolis algorithm applied to the density $p(x) \propto 0.3e^{-0.2(x+5)^2} + 0.7e^{-0.2(x-15)^2}$ (solid curve). The proposal distribution used is a Gaussian with $\sigma = 4$. Shown are the most recent 10% of samples after the specific number of steps. (a) For short times, the chain gets “stuck” in the smaller peak, unable to cross the bottleneck to explore the full distribution. (b) For intermediate times, although the distribution is explored fully, the chain does not spend the correct proportion of time in each peak. (c) For large times, convergence to the invariant distribution is guaranteed via irreducibility and aperiodicity of the chain.

Simulated annealing bears a strong resemblance to the Metropolis algorithm; it functions by making iterative proposals with a carefully chosen probability of acceptance. Unlike the Metropolis algorithm, the probability depends on the iteration k , such that the probability of transitioning to a region of low probability is much higher when k is small. The acceptance probability is often stated in terms of a temperature $T(k)$ which is monotonically decreasing, owing to its connections to the role of temperature in statistical mechanics (and begetting the term “cooling schedule”). The common explicit acceptance probability $\exp((\log f(Y) - \log f(X_k))/T(k))$ coincides with the Metropolis algorithm in the case that $T(k) = 1$, and so simulated annealing can be thought of as a generalization of Metropolis [43, 109, 131].

Since the transition probability depends on k , the chain $\{X_k\}$ no longer has an invariant distribution. Instead, each step of the chain has its own unique “invariant distribution” p_k . As k increases, $T(k)$ decreases, and if T falls to zero, the weak limit of the p_k should put its entire mass at the global maximum value of f (this is why simulated annealing is effective at optimization). Conversely, for small k , a large temperature greatly increases the probability of accepting a transition to a region where f is much smaller. If the cooling schedule is chosen to scale appropriately, it can thus be very effective at facilitating transitions across bottlenecks like that of Figure 4.1. A simple application of

simulated annealing to the toy sampling problem can be seen in Figure 4.2.

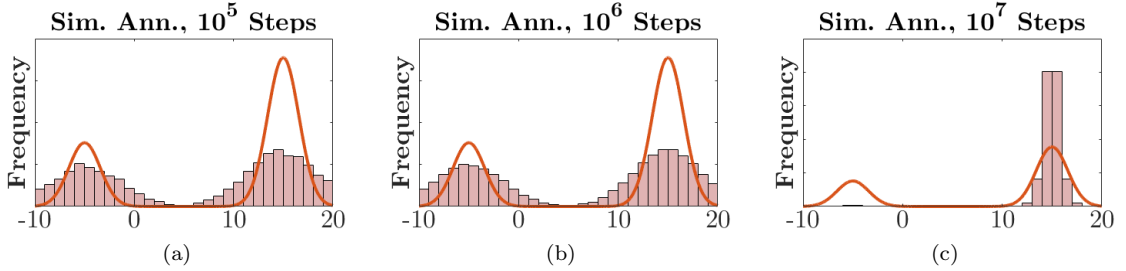


Figure 4.2: Simulated annealing with cooling schedule $T(k) = 5(1 - 10^{-7}k)$ for the toy problem of Figure 4.1. Again, the proposal distribution used is a zero-mean Gaussian with $\sigma = 4$. Shown are the most recent 10% of samples after the specific number of steps. (a, b) Even for short times, the high temperature smooths the target distribution, allowing sampling from both peaks. (c) As the temperature falls to zero, the distribution clusters around the global maximum.

The specific MCMC algorithm which will be applied to the Bayesian parameter estimation problem, known as transitional Markov chain Monte Carlo (TMCMC), makes use of the ideas central to both of these algorithms [42, 141]. Each step of the chain uses the Metropolis algorithm to draw approximate samples from a target intermediate distribution, while the parent chain uses a cooling schedule to *transition* from the prior to the posterior, much as simulated annealing transitions from smoothed to peaked. The specifics of TMCMC, as well as its explicit implementation in the UQ framework, are detailed in Section 4.3.

4.3 Transitional Markov Chain Monte Carlo (TMCMC)

As mentioned in the previous section, MCMC algorithms are often used to generate approximate samples from high-dimensional and complex posterior distributions. While basic algorithms like Metropolis and simulated annealing will function in this regard, more efficient algorithms which improve the convergence rate are more useful in practice, since the convergence times of MCMC algorithms for some distributions (especially those with bottlenecks) can be quite long. Although there do exist more efficient methods with proven convergence results (in particular, delayed rejection adaptive Metropolis (DRAM) [41] and differential evolution random sampling MC (DREAM)

[137] are commonly used), we will use here the TMCMC algorithm, which has the computational advantage of being massively parallelizable due to its use of a large number of independent Markov chains [17, 42, 141].

Like simulated annealing, TMCMC attempts to avoid probabilistic bottlenecks by transitioning to the target distribution from a broad initial distribution (in this case, the prior $\pi(\boldsymbol{\theta}|M)$) that allows the chain to explore regions of low probability. To accomplish this, a series of intermediate probability densities are constructed iteratively:

$$f_j(\boldsymbol{\theta}) \sim [p(\mathbf{D}|\boldsymbol{\theta}, M)]^{q_j} \cdot \pi(\boldsymbol{\theta}|M), \quad j = 0, \dots, \lambda$$

$$0 = q_0 < q_1 < \dots < q_\lambda = 1.$$

Note that f_0 is simply the prior distribution, while f_λ is the posterior distribution (i.e., the product of the prior and likelihood, as in (4.1), up to the normalization constant $\rho(\mathbf{D}|M)$).

The basic idea of TMCMC is as follows: begin with a collection of samples from the prior distribution (which is easy, so long as the prior has a simple form such as a Gaussian or uniform distribution). Then, iteratively re-sample in such a way that the samples at each step j correspond to the intermediate density f_j . At the conclusion of the algorithm, the samples will be asymptotically distributed as f_λ (the posterior), with greater accuracy for small step sizes and large numbers of samples. During each iteration, the re-sampling procedure involves the use of a large number of independent Markov chains which are evolved using the Metropolis algorithm; since the chains are independent, the algorithm is easily parallelizable, and so the potentially expensive cost of evaluating the likelihood $p(\mathbf{D}|\boldsymbol{\theta}, M)$ is offset by the ability to run multiple evaluations concurrently.

The explicit algorithm is summarized below as Algorithm 1:

The algorithm begins by taking N_0 samples $\boldsymbol{\theta}_{0,k}$ (indexed first by generation, then by sample number) from the prior distribution $f_0(\boldsymbol{\theta}) = \pi(\boldsymbol{\theta}|M)$. For each stage j of the algorithm, it uses the current set of samples to evaluate the plausibility weights $w(\boldsymbol{\theta}_{j,k})$ as

$$w(\boldsymbol{\theta}_{j,k}) = \frac{f_{j+1}(\boldsymbol{\theta}_{j,k})}{f_j(\boldsymbol{\theta}_{j,k})} = [p(\mathbf{D}|M, \boldsymbol{\theta}_{j,k})]^{q_{j+1}-q_j}.$$

Algorithm 1 TMCMC

```

1: procedure TMCMC Ref. [42]
2: BEGIN, SET  $j = 0, q_0 = 0$ 
3: Generate  $\{\theta_{0,k}, k = 1, \dots, N_0\}$  from prior  $f_0(\theta) = \pi(\theta|M)$  and compute likelihood  $p(\mathbf{D}|\theta_{0,k}, M)$  for each sample.
4: loop:
5: WHILE  $q_{j+1} \leq 1$  DO:
6:   Analyze samples  $\{\theta_{j,k}, k = 1, \dots, N_j\}$  to determine  $q_{j+1}$ , weights  $\bar{w}(\theta_{j,k})$ , covariance  $\Sigma_j$ , and estimator  $S_j$  of  $\mathbb{E}[w(\theta_{j,k})]$ .
7:   Resample based on samples available in stage  $j$  in order to generate samples for stage  $j+1$  and compute likelihood  $p(\mathbf{D}|\theta_{j+1,k}, M)$  for each.
8:   if  $q_{j+1} > 1$  then
9:     BREAK,
10:  else
11:     $j = j + 1$ 
12:    goto loop.
13:  end
14: END

```

Note that since the plausibility weights are calculated as the likelihood raised to a non-negative power, the observed data $\mathbf{D}$ is more likely to have come from sampled parameters $\theta_{j,k}$ with higher plausibility weights (thus begetting the term). q_{j+1} has not yet been defined; the choice of q_{j+1} is what determines how smoothly the prior distribution transitions to the posterior distribution (with a smoother transition coming at the cost of increased computational requirements). Recent literature suggests that q_{j+1} be chosen to make the covariance of the plausibility weights at stage j smaller than a tolerance covariance, recommending the specific value 1.0 [17, 42].

The algorithm then calculates the average S_j of the plausibility weights, the normalized plausibility weights $\bar{w}(\theta_{j,k})$, and the scaled covariance Σ_j of the $\theta_{j,k}$, which will be used to generate the next generation of samples $\theta_{j+1,k}$:

$$S_j = \frac{1}{N_j} \sum_{k=1}^{N_j} w(\theta_{j,k}),$$

$$\bar{w}(\theta_{j,k}) = w(\theta_{j,k}) / \sum_{k=1}^{N_j} w(\theta_{j,k}) = w(\theta_{j,k}) / (N_j S_j),$$

$$\Sigma_j = \beta^2 \sum_{k=1}^{N_j} \bar{w}(\theta_{j,k}) [\theta_{j,k} - \mu_j][\theta_{j,k} - \mu_j]^T.$$

Here, Σ_j is calculated using the sample mean μ_j of the $\theta_{j,k}$ and a scaling factor β .

As an aside, it is worth noting that the product of the S_j , i.e.,

$$S = \prod_{j=0}^{\lambda-1} S_j,$$

is an asymptotically unbiased estimator of the evidence $\rho(\mathbf{D}|M)$ [17], and so TMCMC is especially efficient for model selection (see Section 4.1).

The algorithm then independently generates N_{j+1} samples $\hat{\boldsymbol{\theta}}_{j+1,k}$ by choosing randomly among the current set of samples $\{\boldsymbol{\theta}_{j,k}\}$ such that $\hat{\boldsymbol{\theta}}_{j+1,\ell} = \boldsymbol{\theta}_{j,k}$ with probability $\bar{w}(\boldsymbol{\theta}_{j,k})$. Since the samples are chosen independently, there is the possibility of selecting the same sample from the current generation multiple times – call $n_{j+1,k}$ the number of times that $\boldsymbol{\theta}_{j,k}$ is chosen (which could range from 0 to N_{j+1}). Each unique sample is used as the starting point of an independent Markov chain whose behavior is governed by the Metropolis algorithm with target distribution f_j and a proposal distribution normally distributed with covariance $\boldsymbol{\Sigma}_j$ and centered at the current value. The scaling parameter β thus plays the role of controlling the rejection rate and jump size; recent literature recommends $\beta = 0.2$ [3, 17].

Finally, the samples $\boldsymbol{\theta}_{j+1,k}$ are drawn from the Markov chains, with each sample being the value obtained by stepping a chain forward once. $n_{j+1,k}$ samples are drawn from the chain which started at $\boldsymbol{\theta}_{j,k}$, yielding N_{j+1} samples total. At this point, the iteration j is complete, and the algorithm either moves forward to generation $j + 1$ or terminates if $q_{j+1} > 1$.

Chapter 5

Uncertainty Quantification for Complex Models

5.1 DPD Lipid Membrane

We demonstrate the effectiveness of the Bayesian approach of Chapter 4 using a two-dimensional particle simulation of a lipid bilayer membrane, which plays a sizeable role in the motility of, e.g., eukaryotic organisms. As in the simulated membranes of Chapter 3, we model both the lipids and fluid directly, and the membrane is held together only by the intermolecular forces between the tails of the lipids and the surrounding water. Two free parameters are selected as input and are passed to the model on initialization, while the simulation output is calculated directly from the particle positions via a time-averaging procedure. The resulting model has considerable complexity and is moderately expensive – by standard Markov chain Monte Carlo approaches, Bayesian uncertainty quantification in such a setting is not feasible.

We apply the parallelized TMCMC approach to the membrane particle simulation in the context of a simple error prediction model which corrupts the output with additive Gaussian noise. To

check that the resulting samples are reasonable, they are compared with a known set of parameters which was used to generate the target output. In all cases tested, the reference parameter values are assigned reasonably high probability under the sampled posterior. Namely, Bayesian parameter estimation is feasible for an example particle simulation with tens of thousands of particles, and so will have applications to a wide range of problems in mathematical and cellular biology, whose models are often too expensive to apply naïve methods.

5.1.1 Membrane Curvature

Bayesian uncertainty quantification is agnostic to its forward problem: it functions exclusively with respect to the parameter space and the problem output. Historically, the high-dimensional optimization literature has largely focused on the particular forward problem of a (usually very complicated) function evaluation via, e.g., convex optimization methods or simulated annealing. In this chapter, we show that the general Bayesian framework (and its ability to discern *distributions*, rather than just critical points) can be applied to a much more complicated system, namely, the output of a comparatively expensive particle simulation. It follows that Bayesian uncertainty quantification has a number of applications in molecular biology, where particle simulations are both feasible and revealing of dynamics which are difficult to study experimentally or with simpler models.

The sample simulation proposed here is the modeling of a finite patch of lipid bilayer membrane composed of lipids with hydrophobic and hydrophilic parts. One of the chief advantages of modeling such a system with dissipative particle dynamics is the difference in spatial scale between solute and solvent: a typical lipid is many times larger than a single water molecule, and so their interaction may be challenging to model by conventional means. In a coarse-grained method such as DPD, the lipid might be modeled as a bead-spring chain, with the beads (i.e., DPD particles) corresponding to different sections of the lipid. Conversely, a DPD particle modeling the solvent, depending on choice of scale, might model several water molecules. In this way, the actual particles which interact via DPD potentials are of similar scale, and their interactions can be easily adjusted via the DPD

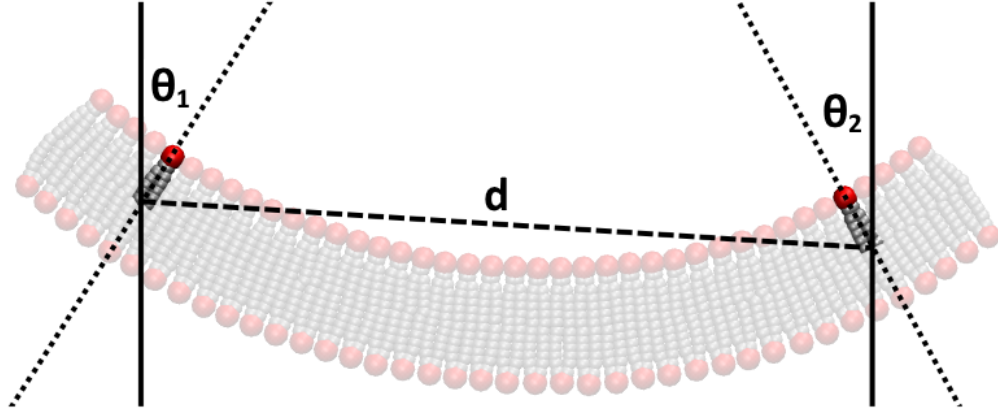


Figure 5.1: Schematic of tracked lipids and method for determination of curvature. Angles are with respect to a fixed reference angle (here shown as vertical lines) and can thus be negative. The derivation for approximate curvature κ relies only on the absolute difference $|\theta_2 - \theta_1|$ and tail-to-tail distance d .

force coefficients, e.g., the conservative coefficient a_{ij} or dissipative coefficient γ_{ij} between particles i and j .

We employ here a finite patch of the two-dimensional membrane described in Section 3.3.3 with 80 lipids in a 96×64 periodic simulation box. To be used as a forward problem where Bayesian uncertainty quantification can be applied, the input and output spaces must be specified; here, we choose the output to be the equilibrium curvature of the membrane, as measured from the particle simulation. Within the upper layer, two lipids, five molecules from the membrane edge on either side, are chosen to be tracked (see Figure 5.1). In each timestep and for each of the two tracked lipids, the position of the head particle and the average position of the furthestmost tail particles are used to compute the angle (relative to vertical; denoted θ_1 and θ_2 for first and second tracked lipid, resp.) and location along the bilayer interface.

The radius of the circle which passes through the given points and subtends the difference in angles $|\theta_2 - \theta_1|$ over that span is given in terms of the distance d between the points as

$$R = \frac{2}{d} \sin \left| \frac{\theta_2 - \theta_1}{2} \right|, \quad (5.1)$$

and the curvature (assuming it is constant along the membrane) is approximated as

$$\kappa \approx \frac{1}{R} = \frac{d}{2 \sin \left| \frac{\theta_2 - \theta_1}{2} \right|}. \quad (5.2)$$

After an equilibration period of 20000 steps, the curvature is measured at each timestep for the following 30000 steps, with the average curvature over this period used as the one-dimensional output of the simulation. A single serial run of the simulation, which tracks ~ 15000 dissipative particles, requires five to ten minutes on an Intel Xeon E5-2670.

There are a number of different biological phenomena which affect the curvature of lipid membranes in eukaryotic phyla; one such mechanism is differences in lipid structure (e.g., tail length or angle) between monolayers [56, 84, 33]. As these structural differences are easily incorporated into the bead-spring model lipids, we choose a set of parameter values describing the lipid shape – specifically, the set of possible neutral angles between lipid tails (ϕ_0 in Fig. 3.28) on the top and bottom layers of the membrane, specified separately – as the parameter space for the forward model. Increasing ϕ_0 in one lipid layer causes the layer to lengthen as the tails are forced further apart, altering the net curvature of the membrane. Since there is no meaningful notion of direction as might be imposed by, e.g., gravity, top and bottom are defined with respect to the y axis at initialization, where the top layer is in the $-y$ direction; the neutral angles on the top and bottom are referred to as ϕ_t and ϕ_b , respectively. Given a target curvature, Bayesian statistics can be used to find the probability distribution in the parameter space which describes the set of parameters which, when accounting for noise in measurement and in the stochasticity of the simulation, could yield the target output.

5.1.2 Parameter Estimation

In Section 4.1, we described a framework for Bayesian uncertainty quantification in the context of the model prediction equation (4.2), reproduced below:

$$\mathbf{D} = g(\boldsymbol{\theta}|M) + \mathbf{e} \quad (5.3)$$

Namely, we assume there exists a function g which, given a particular model M , deterministically maps the parameter space to the output space, and that observable measurement of the output is corrupted by additive Gaussian noise $\mathbf{e}$, which has zero mean and covariance matrix $\boldsymbol{\Sigma}$. In the context of the membrane model, it may be useful intuitively to think of g as the mapping from the parameter space to the mean curvature, while the noise $\mathbf{e}$ corresponds to a hypothetical measurement or modeling error which might have a Gaussian form. We are specifically interested in the subset of Bayesian UQ known as parameter estimation: given a reference curvature κ^* (and assuming, as an example, uniform priors on the lipid tail angles ϕ_t and ϕ_b over some reasonable range), what is the form of the posterior $p(\boldsymbol{\theta}|\kappa^*, M)$?

It is worth noting that we make no assumptions about the covariance $\boldsymbol{\Sigma}$ of the additive noise. Moreover, it is clear that $\boldsymbol{\Sigma}$ should have a significant effect on the posterior distribution. Consider the following example: with a reference curvature $\kappa^* = 0.1$, running the simulation with a particular set of angles $\boldsymbol{\theta}_1$ yielded a mean curvature $\kappa_1 = 0.11$, while a simulation with a second set of angles $\boldsymbol{\theta}_2$ yielded a mean curvature $\kappa_2 = 0.105$. If $|\boldsymbol{\Sigma}|$ were known to be very small, e.g., $|\boldsymbol{\Sigma}| = 10^{-9}$, then the likelihood of the parameters $\boldsymbol{\theta}_2$ should be much greater – being hundreds of standard deviations further from the reference value κ^* is comparatively unlikely. Conversely, if $|\boldsymbol{\Sigma}|$ were known to be very large, e.g., $|\boldsymbol{\Sigma}| = 1$, then the likelihoods of both sets of parameters should be nearly equal – given such a large measurement error in the reference curvature and the proximity of $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$, either is similarly plausible. A natural way of incorporating this effect is to augment the input space with a parametrization of $\boldsymbol{\Sigma}$, as detailed in Section 4.1. In the context of the membrane simulation, the result is a parameter space of dimensionality $n = 3$ which incorporates the two model parameters (ϕ_t and ϕ_b) as well as the standard deviation s of the measurement error $\mathbf{e}$ (since the output – curvature – is one-dimensional, $\boldsymbol{\Sigma}$ is scalar, and so we will hereafter refer only to the estimated standard deviation s and reference standard deviation $\sigma = \sqrt{\boldsymbol{\Sigma}}$ of the noise).

The following sets of results were sampled using a particular high-performance parallel version of TMCMC known as $\Pi 4U$ [42]. In each case, the prior distribution $\pi(\boldsymbol{\theta}|M)$ is taken to be uniform over $[0, 120] \times [0, 120] \times [0, 0.001] \in \mathbb{R}^3$ (the first two parameters, ϕ_t and ϕ_b , are specified in degrees; the third parameter, s , is specified in units of curvature).

Targeting arbitrarily chosen reference curvatures, it would be difficult to evaluate the effectiveness of the algorithm in recovering a meaningful distribution, since we would not *a priori* have any notion of whether the approximate posterior actually generated curvatures close to the specified value. For this reason, we instead use target curvatures derived from arbitrarily chosen angles ϕ_t^* and ϕ_b^* . In order that the generated curvature be consistent with (4.2), a sample from the Gaussian noise term $\mathbf{e}$ (with reference standard deviation σ , here chosen arbitrarily as $\sigma \approx 10^{-5}$) is added to κ^* . In this way, it is possible to check that the recovered approximate posterior contains (and ideally is centered near) the set of values which originally generated the reference curvature κ^* via (4.2). For each of the following sets of results, we specify the reference angles ϕ_t^* and ϕ_b^* which were used to generate the target curvature via one run of the particle simulation. It should be noted that the uniform prior, which assigns equal mass to a wide range of potential parameter values, is fixed before the generation of κ^* – the Bayesian framework does not have any information *a priori* about which parameter values are more or less likely.

5.1.3 Results

The first set of UQ results used reference angles $\phi_t^* = 100$ and $\phi_b^* = 20$, which yielded a curvature of $\kappa^* = 0.03834$ via the model prediction equation with noise level $\sigma \approx 10^{-5}$. Results from the algorithm appear in Figure 5.2 (see caption for a description of the various subfigures). Since curvature does not have a notion of direction, the resulting joint posterior of (ϕ_t, ϕ_b) (center left subfigure) is symmetric about the diagonal. Note that, when one neutral angle is small, changing said angle has little effect on the curvature; the left flank of the upper curve is nearly flat, as is the lower flank of the bottom curve. Conversely, when the magnitudes of both angles are large,

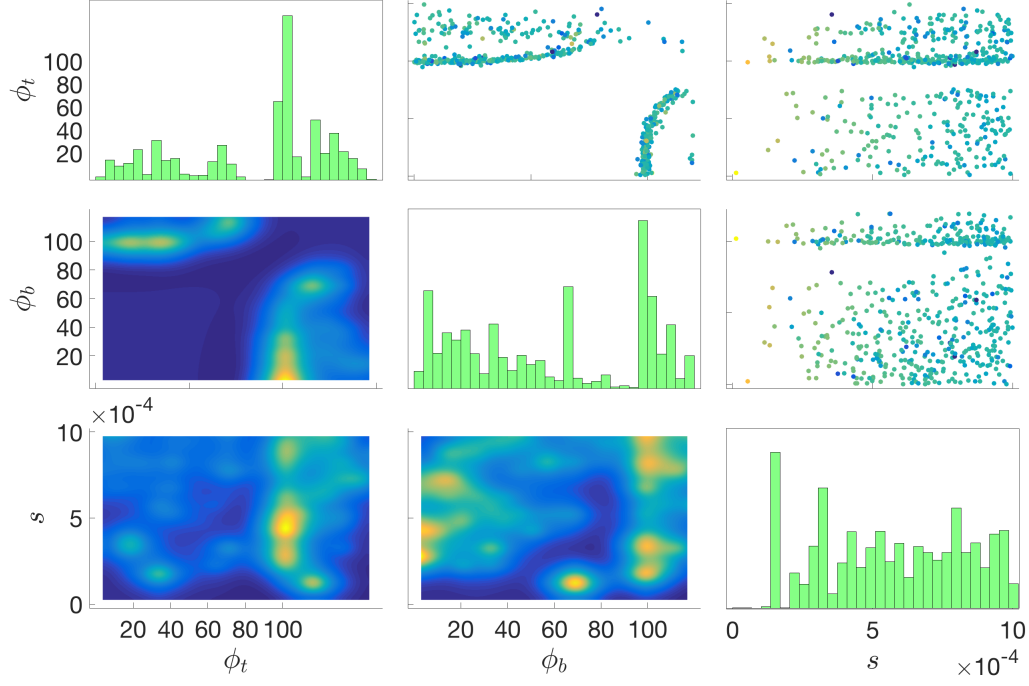


Figure 5.2: 5000 samples from approximate posterior with $\phi_t^* = 100$, $\phi_b^* = 20$ ($\kappa^* = 0.03834$). Histograms on the major diagonal are the empirical marginal distributions of ϕ_t , ϕ_b , s , from left to right. Plots in the upper right are the empirical joint distributions of pairs of parameters; the specific parameters and values appear in the margin. Shaded plots in the lower left are smoothed representations of the joint posterior, color-coded by probability – blue (low) to yellow (high).

the distribution pulls away from the diagonal – once the smaller angle becomes sufficiently similar in magnitude to the larger, the membrane curvature begins to decrease, and so achieving a target positive curvature requires the larger angle to increase further. Since ϕ_b^* and ϕ_t^* are both assigned high probability (up to symmetry), the algorithm was effective at recovering the reference angles.

Unsurprisingly, the parameter s corresponding to the noise term $\mathbf{e}$ was not recovered to great accuracy. The model prediction equation (4.2) makes the assumption of a deterministic prediction function g ; the dissipative particle dynamics simulation, which is inherently stochastic, fails this assumption, although the issue is somewhat circumvented by the time-averaging approach taken to the measurement of the curvature. Conversely, this means that the recovered distribution of s corresponds in some sense to the underlying noise of the DPD approach, which may be of interest in its own right (for a more rigorous treatment of the modeling uncertainty inherent to particle

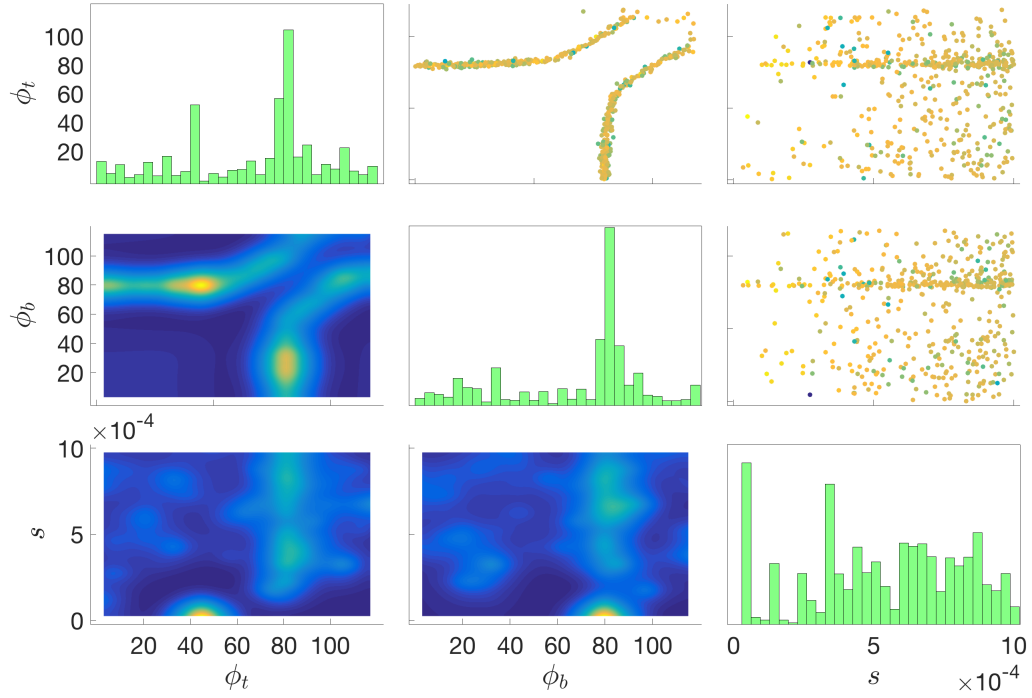


Figure 5.3: 5000 samples from approximate posterior with $\phi_t^* = 80$, $\phi_b^* = 20$ ($\kappa^* = 0.02087$). See Fig. 5.2 for description of subfigures.

simulations, see [3]).

A second set of results, shown in Figure 5.3, used $\phi_t^* = 80$ and $\phi_b^* = 20$, yielding a target curvature of $\kappa^* = 0.02087$. Again, the reference angles were assigned high probability under the posterior, and so the algorithm was effective at parameter recovery. As in the previous set of results, the leftmost region of the joint posterior is close to horizontal, suggesting that small neutral angles are comparatively unimportant in determining the curvature. There are two factors which contribute to this behavior: firstly, the lipid tails do have some repulsion to each other, and so (even in the absence of fluid) a lipid with neutral angle 0 would not actually have overlapping tails at equilibrium. In this way, the energy of extremely small angles may be dominated by the short-scale repulsion between tails. Secondly, the potential imposed is harmonic, and so scales with $(\phi - \phi_0)^2$. As a result, when already close to equilibrium, small perturbations in the lipid angle yield correspondingly small changes in the energy.

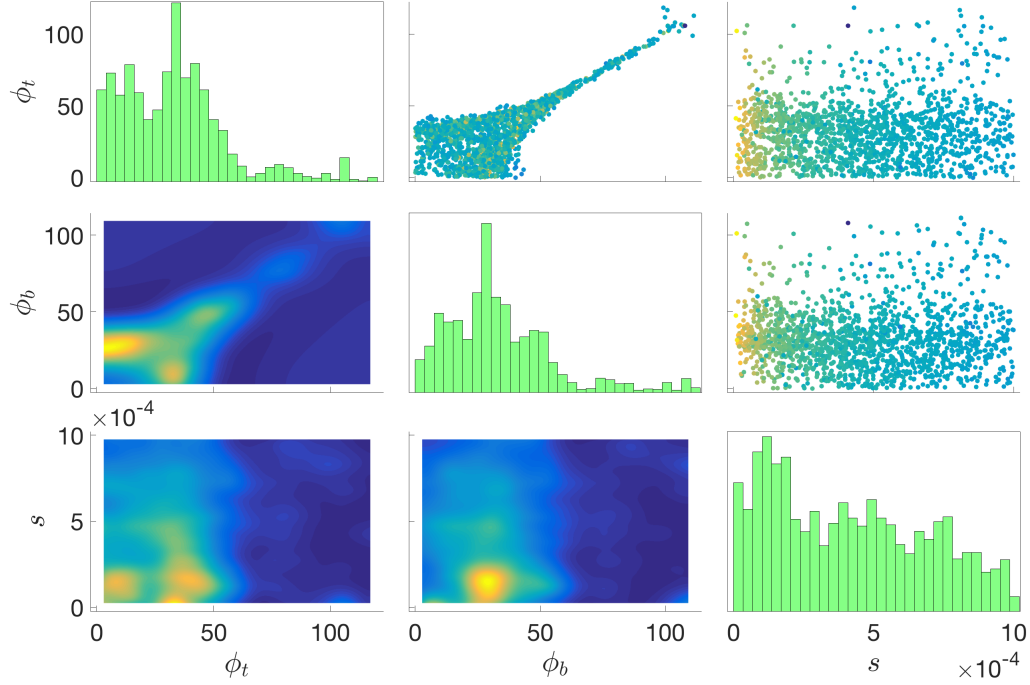


Figure 5.4: 5000 samples from approximate posterior with $\phi_t^* = 20$, $\phi_b^* = 20$ ($\kappa^* = 0.001480$). See Fig. 5.2 for description of subfigures.

The shape of the posterior is markedly different in the case where the reference curvature is very small (i.e., the desired membrane is nearly flat). Figure 5.4 shows the results of the UQ algorithm for identical reference angles $\phi_t^* = \phi_b^* = 20$. (Due to thermal fluctuations, along with the fact that curvature is nonnegative, the time-averaged reference curvature obtained was actually nonzero: $\kappa^* = 0.001480$.) The joint distribution of (ϕ_t, ϕ_b) loosely follows the diagonal, as should be the case by symmetry. Interestingly, the distribution becomes considerably narrower as the neutral angles increase in magnitude. Namely, to achieve a very small reference curvature using large angles, the angles must be extremely close to each other. This is a direct result of the quadratic scaling of the harmonic energy: since the lipids in a flat membrane are restricted to relatively small angles, each lipid angle is far from equilibrium, and so even small perturbations in the neutral angle on one side would significantly upset the balance between the lipid layers.

For small angles, the distribution is more broad; for extremely small values, the fact that the

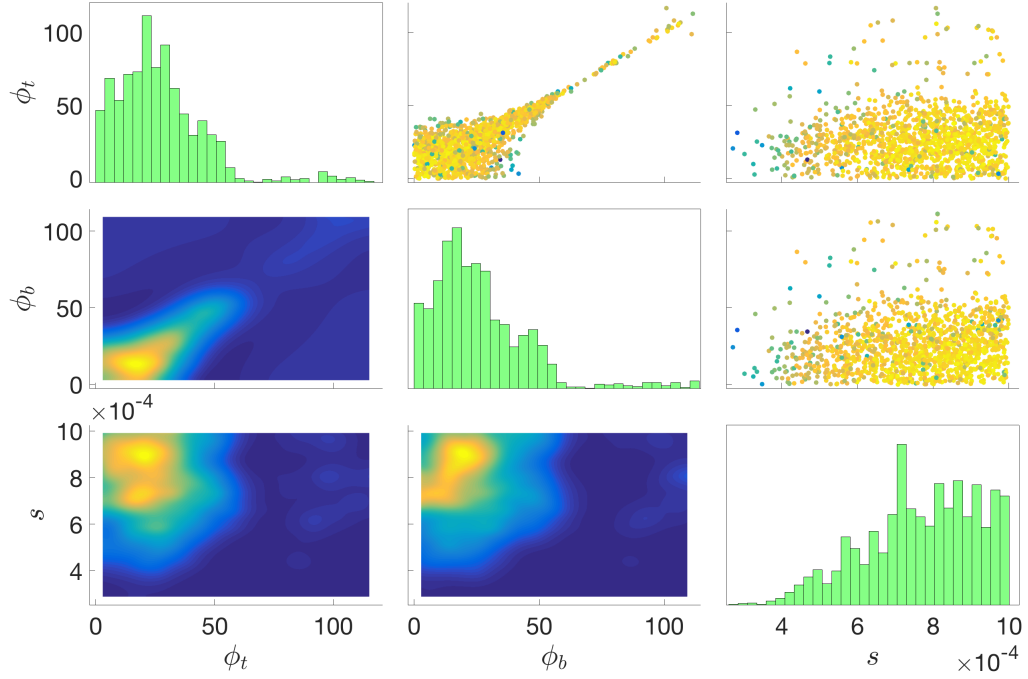


Figure 5.5: 5000 samples from approximate posterior with $\kappa^* = 0$. See Fig. 5.2 for description of subfigures.

reference curvature was nonzero is evident (specifically, the joint distribution of (ϕ_t, ϕ_b) splits away from the diagonal as the angles approach zero). The smaller angles result in lipids being closer to equilibrium, thereby reducing the magnitude of the energy shift when the neutral angle on one side is perturbed. Although it peaks elsewhere, the posterior does put significant mass on the reference angles $\phi_t^* = \phi_b^* = 20$.

As a point of comparison, Figure 5.5 presents the results of a second run directly setting the target curvature κ^* to zero. Unlike Figure 5.4, the joint of (ϕ_t, ϕ_b) here does contain the diagonal (top middle). Because the flat membrane becomes very unstable as the angles are increased on both sides, the posterior appears borderline invisible in that region.

In the three cases presented here, Bayesian UQ via TMCMC was found to be effective at recovering the reference parameters, along with classifying the uncertainty via approximate sampling from the posterior. It is worth restating, in somewhat broader terms, why this result is significant

and how it may be applied to general computational models in cellular biology.

The cell membrane is a central structure in many forms of cell motility, and the simulation method described in Section 3.3.3 has significant complexity and nontrivial dynamics. Nonetheless, it should be thought of as a stand-in for an even more complex problem, whose input and output may be very high-dimensional. For example, a computational model of a prokaryote might consider a broad range of parameters such as the oscillation rate, flexibility, and length of the flagella, the rigidity and size of the cell body, and the temperature and viscosity of the surrounding fluid; a model of a eukaryote might include the density, (de-)polymerization rate, and aspects of the structure of the actin cytoskeleton. Both models may be interested in output such as the speed and trajectory of cell motion, or the robustness of motility to external influences such as fluid flow or obstacles.

Parameter estimation as we have presented it is simply a generalization of the inverse problem: suppose in these examples that we know the comparable output of experiment, or are targeting a specific behavior that is known to have biological relevance. What can be inferred about the input space, i.e., what parameters would generate such behavior? Moreover, what is the degree of uncertainty in these inferences? For extremely complex models as in these examples, many questions in this vein are intractable through naïve methods. The distribution of parameters cannot be sampled exactly, and approximate sampling methods such as Metropolis may require too many evaluations of an already expensive model, or may introduce other problems such as probabilistic bottlenecks.

The most important aspect of the TMCMC approach to Bayesian uncertainty quantification described in Section 4.3 is its parallelizability. Because each generation of the algorithm generates many approximate samples from the target intermediate distribution, the chains can be run simultaneously; with modern parallel architectures growing increasingly more powerful and efficient, this marks an improvement in the computational tractability of uncertainty quantification of many orders of magnitude. As shown here, this allows for practical uncertainty quantification even for complex particle simulations.

5.2 Other Applications

Since the efficient parallel implementation of TMCMC can be applied to an effective black box, the method is accessible to a broad range of existing models in mathematical and cellular biology. The Π 4U approach has been used to perform data-driven inference on the shape of the Lennard-Jones potential for neutral molecule interactions [69] and the pharmacokinetic properties of the cancer drug Aflibercept [32], and it has also been used for applications to other fields such as inferring tension in arch bridge hangers [97]. Of particular relevance to this thesis, many of the systems described in Chapter 3 could be examined under the lens of uncertainty quantification, using the noisy Bayesian inverse setting to simplify experimental design, determine the model parameter distributions matching noisy experimental observations, or perform Bayesian model selection comparing approaches; the high-performance parallel aspect of the framework is vital in extending the Bayesian framework for parameter estimation to the types of expensive computational particle models studied here.

A work in collaboration with Zhizhong Chen, Karen Larson, Panagiotis Hadjidakas, Costas Papadimitriou, Petros Koumoutsakos, and Anastasios Matzavinos studying the application of the high-performance Bayesian uncertainty quantification framework to a network differential-equation model for epidemic spread is currently in submission [16]. Though the model in question is not related to the principal content of this thesis, it nonetheless serves to highlight the flexibility of the Bayesian approach for other types of computationally expensive biological models. We successfully recover synthetic model parameters via limited noisy observations of the population network and use the model selection framework described in Section 4.1 to identify the disease origin; given the increasing availability of such noisy information in the era of “big data,” such a framework has broad practical relevance for the prediction and management of epidemics. Future work in this direction will examine questions about optimal sensor placement, i.e., the information-theoretically defined optimal locations to gather these noisy data, and will consider applications of the approach to real-world epidemic data.

Chapter 6

Conclusions and Directions

Broadly speaking, the main conclusion of this thesis is that dissipative particle dynamics is the natural complement to experimental study of fluidic systems at the mesoscale. Through our varied applications to biological soft matter systems, we showed the ability of the DPD approach to probe the mechanics underlying a wide range of physical phenomena: the explicitly modeled fluid beads can be used to track and observe flows and measure forces between the fluid and other structures, models of polymers and membranes which are easily coupled with the DPD approach can be used to make robust predictions of soft matter evolution over long time scales, and full DPD models of experimental systems can be used to directly observe the physical mechanisms governing the system dynamics. It is important to emphasize that these models need not be *ad hoc*; we have provided several examples in this thesis of tuning DPD parameters to match both quantitative (e.g., a polymer's measured radius of gyration) and qualitative (e.g., the shape of a phospholipid vesicle) experimental observations. DPD thus fills a crucial niche in the mathematical modeling of mesoscale biological structures; after the DPD model is calibrated with observable data, it provides the ability to measure and test physical hypotheses which cannot be studied with experiment alone due to the diminutive time and length scales characteristic of such systems.

After describing a number of applications in this vein, we described in Chapter 4 a Bayesian

uncertainty quantification approach which capitalizes on the increasing amount of parallel computing power available to modern researchers. This high-performance framework allows for parameter estimation and model selection even for expensive computational models such as our DPD models of soft matter systems, thereby broadening the set of questions which can be practically answered by the DPD approach. We described a particular application to lipid membranes which inferred structural properties of the lipids given only a noisy macroscale observable (the curvature): the successful recovery of the underlying parameters highlights the ability of Bayesian UQ to perform data-driven model calibration, automatically tuning free parameters of the DPD system to match experimental observations. The model can also be calibrated based on *targeted* data, i.e., desired properties for a physical system, such as a certain extension of a DNA macromolecule or a certain mean velocity for actin-propelled organisms. In this way, the Bayesian uncertainty quantification framework rigorizes the strength of the DPD modeling approach in experimental design.

As the novel content of this thesis is heavily application focused, we conclude by briefly reviewing below the major results obtained via DPD simulation. In each case, we briefly discuss implications of the results and extensions to related problems, highlighting a few of the more notable future directions for related work.

In Section 3.2, we modeled a nanofluidic device designed by experimental collaborators Daniel Kim and Derek Stein. The device was designed to create energy barriers opposing the forward motion of flow-driven DNA by trapping the polymer in a series of consecutive nanopits etched into the device floor; experimental measurements of the DNA diffusion in this scenario demonstrated giant acceleration of diffusion (GAD), an inherently non-equilibrium dynamical phenomenon, in a system with *entropic* barriers. Using our DPD model and mapping measurements from the model to the one-dimensional Brownian analog, we were able to measure the sub-piconewton forces acting on the molecule and study the entropic barriers defining the system by revealing the shape of the free energy landscape near the critical point. Our measurements revealed that an analogy to Brownian particles in an undulating geometry is well justified; the DPD-measured entropic barrier in the

energy landscape was removed just as the flow velocity was increased past the critical point. We also explored the effect of varying device specifications such as the pit spacing and depth, using the DPD model to directly observe changes in the DNA behavior and diffusion. Future work exploring Brownian-dynamics or even simpler approximations of the system calibrated to measurements from the DPD approach may allow for a further reduction of model complexity, bringing the simulations into the realm of feasibility for the data-driven calibration possible with Bayesian UQ approaches. The DPD model can also be used to study how entropic barriers change with the driving force and the possible emergence of non-ideal behavior, potentially leading to useful new ways of controlling molecules.

Section 3.4 describes a DPD model of a porated lipid membrane, inspired by recent experimental results in the evolution of porated polymer films [53]. The experiments largely used films with thickness on the order of $1\ \mu m$; by contrast, the $\sim 10\ nm$ thick lipid membranes of many biological cells are too thin to pin and measure with the same techniques, and so our DPD model is necessary to study similar mechanics in biologically relevant membranes. The major result of [53] was the identification of a critical radius r_c for the pore size; pores which were initialized at larger sizes grew exponentially, while pores below the critical size were eventually sealed. Though our DPD model showed the same behavior for a membrane comprising lipids with tails of six beads, lengthening the lipid tails resulted in markedly different behavior, with tails of eight beads causing pores to be stable at any radius. This has a number of implications for the translocation of proteins and other molecules across the membrane, some mechanisms of which rely on the controlled creation and deletion of membrane pores [86, 85, 11]; most notably, we demonstrated a strong connection between the pore stability and the lipid membrane thickness, suggesting that pore formation may require distinct mechanisms depending on the lipid structure of the membrane. Future work can investigate the stability of pores in the presence of trans-membrane proteins and other molecules which modify the membrane structure; further advancements in the understanding of pore formation and resealing in these contexts may lead to new medical approaches for controlled modification of

the cell contents.

Sections 3.5 and 3.6 use DPD simulations to examine the interaction between lipid membranes and filamentous actin, which is a major component of the cytoskeleton of eukaryotic cells. Polymerization of the actin filaments applies a forward force on the membrane and is a central concept in the motility of some organisms such as *Listeria* [34]. Using a computational model of rigid actin filaments, we illustrated the ability of actin to push forward a periodic patch of lipid membrane, then modified the actin model to allow for polymerization in the presence of free monomers, showing the modified system to observe the same behavior. Measured data from the DPD simulations of actin-propelled membranes in the presence of a retarding force allowed calculation of a linear force-velocity relation governing the forward motion of the membrane in crowded environments; the literature is divided on the shape of such a relation, finding it to be either concave up or down under different physical or experimental setups [88, 111]. As most other particle-based approaches to actin polymerization model the membrane as a rigid plate or disc, the DPD approach has the unique advantage of explicitly modeling the lipids, allowing membrane fluctuations to influence the availability of free monomers at the actin filament ends; future work should investigate the precise nature of the microflows generated in these circumstances and their influence on monomer availability. Refinements to the model which incorporate more aspects of the actin network structure such as cross-linkage should also be investigated. We also showed the ability of artificial applied forces on a DPD lipid vesicle to reproduce the shape of actin-propelled vesicles in experiment, highlighting some physical mechanisms which may be responsible for the mechanical force profile of polymerizing actin; an explicit DPD model of actin propulsion of a lipid vesicle, if feasible, could test the hypothesized mechanism of diffusion-limited polymerization and provide a deeper look into the forces and stresses of polymerization-based propulsion.

Finally, Section 5.1 discusses parameter estimation for a two-dimensional DPD membrane simulation using the aforementioned high-performance framework for Bayesian uncertainty quantification. As previously stated, the primary implication of this work is that uncertainty quantification and

model selection for DPD models of soft matter are computationally feasible; future work can use this framework to investigate experimental design and data-driven calibration of the many DPD systems studied here, providing a convenient mechanism for the incorporation of experimental data. In particular, this framework may prove useful for inferring the force profile generated by actin networks based on available data collected from lipid vesicles *in vitro* [133]. It may also be of interest to compare the DPD models with comparable models in the literature via Bayesian model selection. Future work involving DPD models of novel systems should continue to leverage the power of this high-performance framework to establish the role of dissipative particle dynamics as an accurate and practical approach for modeling for the large family of mesoscale biofluidic systems.

Funding Acknowledgment

CB gratefully acknowledges the support of NSF CAREER Award 1552903 and NSF/DMS grant DMS-114828 for parts of the research presented in this thesis.

Appendix A

Computational Details

DPD simulations in this work utilized the LAMMPS molecular dynamics package, an open-source C++ framework for particle simulations [100]. Code was run on the computing cluster operated by Brown’s Center for Computation and Visualization, which comprises 450+ high performance nodes, some with extra large memory and some with GPUs for accelerated computing, for a total core count of over 7,000 and peak performance of roughly 125 Teraflops. Most DPD simulations ran on 192 Intel Xeon E5-2670 cores (2.6 GHz) for a time period on the order of 24-48 hours.

The specific TMCMC implementation used in this work was adapted from the open-source framework Π 4U [42], which was modified to perform uncertainty quantification for each of the complex models in Chapter 5. The notable asset of Π 4U is the TORC framework, a software library for programming and running unaltered task-parallel programs on both shared and distributed memory platforms. The parallel framework includes built-in load balancing and fault tolerance.

High-performance uncertainty quantification results from Section 5.1.3 were obtained from parallel runs on 184 Intel Xeon E5-2670 cores (2.6 GHz). TMCMC in Figures 5.2 and 5.3 converged after six generations, with total runtimes of 47.22 and 47.38 hours (34.00 and 34.11 seconds per sample), respectively. TMCMC in Figures 5.4 and 5.5 converged after five generations, with total runtimes of 36.98 and 37.02 hours (26.63 and 26.65 seconds per sample), respectively.

Bibliography

- [1] M.P. Allen and D.J. Tildesley. *Computer Simulation of Liquids*. Oxford Science Publ. Clarendon Press, 1989.
- [2] D.F. Anderson and T.G. Kurtz. *Stochastic Analysis of Biochemical Systems*. Springer, New York, 2015.
- [3] Panagiotis Angelikopoulos, Costas Papadimitriou, and Petros Koumoutsakos. Bayesian uncertainty quantification and propagation in molecular dynamics simulations: A high performance computing framework. *The Journal of Chemical Physics*, 137(14):144103, 2012.
- [4] Panagiotis Angelikopoulos, Costas Papadimitriou, and Petros Koumoutsakos. X-tmcmc: Adaptive kriging for bayesian inverse modeling. *Computer Methods in Applied Mechanics and Engineering*, 289, 02 2015.
- [5] Anthony Balducci, Pan Mao, Jongyoon Han, and Patrick S. Doyle. Double-stranded DNA diffusion in slitlike nanochannels. *Macromolecules*, 39(18):6273–6281, 2006.
- [6] D. Barber. *Bayesian Reasoning and Machine Learning*. Cambridge University Press, 2012.
- [7] James L. Beck and Ka-Veng Yuen. Model selection using response measurements: Bayesian probabilistic approach. *Journal of Engineering Mechanics*, 130(2):192–203, 2004.
- [8] H.C. Berg. *E. coli in Motion*. Biological and Medical Physics, Biomedical Engineering. Springer New York, 2011.

- [9] D.H. Boal. *Mechanics of the Cell*. Mechanics of the Cell. Cambridge University Press, 2012.
- [10] Bernard R. Brooks, Robert E. Bruccoleri, Barry D. Olafson, David J. States, S. Swaminathan, and Martin Karplus. Charmm: A program for macromolecular energy, minimization, and dynamics calculations. *Journal of Computational Chemistry*, 4(2):187–217.
- [11] David Brough, Pablo Pelegrin, and Walter Nickel. An emerging case for membrane pore formation as a common mechanism for the unconventional secretion of fgf2 and il-1. *Journal of Cell Science*, 130:jcs.204206, 09 2017.
- [12] P. S. Burada, G. Schmid, D. Reguera, M. H. Vainstein, J. M. Rubi, and P. Hänggi. Entropic stochastic resonance. *Physical Review Letters*, 101(13):1–4, 2008.
- [13] Lisa A. Cameron, Matthew J. Footer, Alexander van Oudenaarden, and Julie A. Theriot. Motility of acta protein-coated microspheres driven by actin polymerization. *Proceedings of the National Academy of Sciences*, 96(9):4908–4913, 1999.
- [14] Guillaume T. Charras and Ewa K. Paluch. Blebs lead the way: how to migrate without lamellipodia. *Nature Reviews Molecular Cell Biology*, 9:730–736, 2008.
- [15] Z. Chen, K. Larson, C. Bowman, P. Hadjidoukas, C. Papadimitriou, P. Koumoutsakos, and A. Matzavinos. Data-driven prediction and origin identification of epidemics in population networks. *Submitted*, 2018.
- [16] Zhizhong Chen, Karen Larson, Clark Bowman, Panagiotis Hadjidoukas, Costas Papadimitriou, Petros Koumoutsakos, and Anastasios Matzavinos. Data-driven prediction and origin identification of epidemics in population networks. *In submission*, 2018.
- [17] Jianye Ching and Yi-Chu Chen. Transitional markov chain monte carlo method for bayesian model updating, model class selection, and model averaging. *Journal of Engineering Mechanics*, 133(7):816–832, 2007.

- [18] Michel Chipot. *Elliptic Equations: An Introductory Course*. Birkhäuser Basel, 2009.
- [19] W. Coffey, Y.P. Kalmykov, and J.T. Waldron. *The Langevin Equation: With Applications in Physics, Chemistry and Electrical Engineering*. Series in Contemporary Chemical Physics. World Scientific, 1996.
- [20] Wendy D. Cornell, Piotr Cieplak, Christopher I. Bayly, Ian R. Gould, Kenneth M. Merz, David M. Ferguson, David C. Spellmeyer, Thomas Fox, James W. Caldwell, and Peter A. Kollman. A second generation force field for the simulation of proteins, nucleic acids, and organic molecules. *Journal of the American Chemical Society*, 117(19):5179–5197, 1995.
- [21] G Costantini and F Marchesoni. Threshold diffusion in a tilted washboard potential. *Europhysics Letters (EPL)*, 48(5):491–497, 1999.
- [22] Liang Dai, Jeremy J. Jones, R C Van Der Maarel, Patrick S. Doyle, Johan R C Van Der Maarel, Johan R. C. van der Maarel, and Patrick S. Doyle. A systematic study of DNA conformation in slitlike confinement. *Soft Matter*, 8(10):2972, 2012.
- [23] P. G. de Gennes. Flexible Polymers in Nanopores. pages 91–105. Springer Berlin Heidelberg, 1999.
- [24] J T Del Bonis-O'Donnell, W Reisner, and D Stein. Pressure-driven DNA transport across an artificial nanotopography. *New Journal of Physics*, 11(7):075032, jul 2009.
- [25] M Diez-Silva, Ming Dao, J Han, C.T. Lim, and S Suresh. Shape and biomechanics characteristics of human red blood cells in health and disease. *MRS Bulletin*, 35:382–388, 05 2010.
- [26] M Doi and S F Edwards. *The theory of polymer dynamics*. International series of monographs on physics. Clarendon Press, 1986.
- [27] Y. Efendiev and T.Y. Hou. *Multiscale Finite Element Methods: Theory and Applications*. Surveys and Tutorials in the Applied Mathematical Sciences. Springer New York, 2009.

- [28] Yalchin R. Efendiev, Thomas Y. Hou, and Xiao-Hui Wu. Convergence of a nonconforming multiscale finite element method. *SIAM Journal on Numerical Analysis*, 37(3):888–910, 2000.
- [29] B. Efron and T. Hastie. *Computer Age Statistical Inference: Algorithms, Evidence, and Data Science*. Cambridge University Press, 2016.
- [30] S. Eisele, J.M. Lackie, H. Riedwyl, A. Zimmermann, and H.U. Keller. Analysis of lymphocyte shape by visual classification, calculated measures of shape or light scattering. *Journal of Immunological Methods*, 138(1):103 – 109, 1991.
- [31] P Español and P Warren. Statistical Mechanics of Dissipative Particle Dynamics. *EPL (Europhysics Letters)*, 30(4):191, 1995.
- [32] SD Finley, P Angelikopoulos, P Koumoutsakos, and AS Popel. Pharmacokinetics of antivegf agent aflibercept in cancer predicted by datadriven, molecular-detailed model. *CPT: Pharmacometrics & Systems Pharmacology*, 4(11):641–649.
- [33] V.A. Frolov., A.V. Shnyrova, and J. Zimmerberg. Lipid polymorphisms and membrane shape. *CSH Perspect. Biol.*, 3(11):a004747, 2011.
- [34] L G Tilney and D.A. Portnoy. Actin filaments and the growth, movement, and spread of the intracellular bacterial parasite, *listeria monocytogenes*. *The Journal of cell biology*, 109:1597–608, 11 1989.
- [35] G.P. Galdi. *An Introduction to the Mathematical Theory of the Navier-Stokes Equations: Linearized steady problems*. An Introduction to the Mathematical Theory of the Navier-Stokes Equations. Springer-Verlag, 1994.
- [36] Lianghui Gao, Julian Shillcock, and Reinhard Lipowsky. Improved dissipative particle dynamics simulations of lipid bilayers. *The Journal of Chemical Physics*, 126(1):015101, 2007.

- [37] V. Girault and P.A. Raviart. *Finite element methods for Navier-Stokes equations: theory and algorithms*. Springer series in computational mathematics. Springer-Verlag, 1986.
- [38] F Gittes, B Mickey, J Nettleton, and J Howard. Flexural rigidity of microtubules and actin filaments measured from thermal fluctuations in shape. *The Journal of Cell Biology*, 120(4):923–934, 1993.
- [39] Andrea Grafmüller, Julian Shillcock, and Reinhard Lipowsky. The fusion of membranes and vesicles: Pathway and energy barriers from dissipative particle dynamics. *Biophysical Journal*, 96(7):2658–2675, 2008.
- [40] R D Groot and P B Warren. Dissipative particle dynamics: Bridging the gap between atomistic and mesoscopic simulations. *J. Chem. Phys.*, 107(11):4423–4435, 1997.
- [41] H. Haario, M. Laine, A. Mira, and E. Saksman. DRAM: Efficient adaptive MCMC. *Stat. Comput.*, 16:339–354, 2006.
- [42] P.E. Hadjidoukas, P. Angelikopoulos, C. Papadimitriou, and P. Koumoutsakos. 4u: A high performance computing framework for bayesian uncertainty quantification of complex models. *Journal of Computational Physics*, 284:1 – 21, 2015.
- [43] Bruce Hajek. Cooling schedules for optimal annealing. *Mathematics of Operations Research*, 13(2):311–329, 1988.
- [44] J Han and H G Craighead. Separation of long DNA molecules in a microfabricated entropic trap array. *Science (New York, N.Y.)*, 288(5468):1026–1029, 2000.
- [45] J. Han, S. Turner, and H. Craighead. Erratum: Entropic Trapping and Escape of Long DNA Molecules at Submicron Size Constriction [Phys. Rev. Lett. 83, 1688 (1999)]. *Physical Review Letters*, 86(7):1394–1394, 2001.

- [46] Ryunosuke Hayashi, Kazuo Sasaki, Shuichi Nakamura, Seishi Kudo, Yuichi Inoue, Hiroyuki Noji, and Kumiko Hayashi. Giant Acceleration of Diffusion Observed in a Single-Molecule Experiment on F1-ATPase. *Physical Review Letters*, 114(24):248101, 2015.
- [47] W Helfrich. Elastic properties of lipid bilayers: Theory and possible experiments. *Z. Naturforsch. C*, 28, 12 1973.
- [48] R. Holley and D. Stroock. Simulated annealing via sobolev inequalities. *Comm. Math. Phys.*, 115(4):553–569, 1988.
- [49] P. J. Hoogerbrugge and J. M. V. A. Koelman. Simulating microscopic hydrodynamic phenomena with dissipative particle dynamics. *EPL (Europhysics Letters)*, 19(3):155, 1992.
- [50] Thomas Y. Hou and Xiao-Hui Wu. A multiscale finite element method for elliptic problems in composite materials and porous media. *Journal of Computational Physics*, 134(1):169 – 189, 1997.
- [51] J. Howard. *Mechanics of Motor Proteins and the Cytoskeleton*. Sinauer Associates, Publishers, 2001.
- [52] Jifeng Hu, Anastasios Matzavinos, and Hans G. Othmer. A theoretical approach to actin filament dynamics. *Journal of Statistical Physics*, 128(1):111–138, Jul 2007.
- [53] Mark Ilton, Christian DiMaria, and Kari Dalnoki-Veress. Direct measurement of the critical pore size in a model membrane. *Phys. Rev. Lett.*, 117:257801, Dec 2016.
- [54] Wonpil Im and Charles L. Brooks. Interfacial folding and membrane insertion of designed peptides studied by molecular dynamics simulations. *Proceedings of the National Academy of Sciences*, 102(19):6771–6776, 2005.

- [55] Yasuhiro Inoue, Takeji Deji, and Taiji Adachi. Brownian dynamics simulation study on force-velocity relation in actin-based membrane protrusion. *Computational Particle Mechanics*, 2, 12 2015.
- [56] Iris K. Jarsch, Frederic Daste, and Jennifer L. Gallop. Membrane curvature in cell biology: An integration of molecular mechanisms. *The Journal of Cell Biology*, 214(4):375–387, 2016.
- [57] William L. Jorgensen, David S. Maxwell, and Julian Tirado-Rives. Development and testing of the opls all-atom force field on conformational energetics and properties of organic liquids. *Journal of the American Chemical Society*, 118(45):11225–11236, 1996.
- [58] G. Karniadakis, A. Beskok, and N. Aluru. *Microflows and Nanoflows: Fundamentals and Simulation*. Springer, 2005.
- [59] George Em Karniadakis, Ali Beskok, and Narayan Aluru. *Microflows and nanoflows: fundamentals and simulation*, volume 29. Springer Science & Business Media, 2006.
- [60] Shuhei Kawamoto, Michael L. Klein, and Wataru Shinoda. Coarse-grained molecular dynamics study of membrane fusion: Curvature effects on free energy barriers along the stalk mechanism. *The Journal of Chemical Physics*, 143(24):243112, 2015.
- [61] E E Keaveny, I V Pivkin, M Maxey, and G Em Karniadakis. A comparative study between dissipative particle dynamics and molecular dynamics for simple- and complex-geometry flows. *J Chem Phys*, 123(10):104107, 2005.
- [62] K. Keren, P.T. Yam, A. Kinkhabwala, A. Mogilner, and J.A. Theriot. Intracellular fluid flow in rapidly moving cells. *Nat Cell Biol.*, 11(10):1219–1224, 2009.
- [63] Fatemeh Khalili-Araghi, Brigitte Ziervogel, James C. Gumbart, and Benoît Roux. Molecular dynamics simulations of membrane proteins under asymmetric ionic concentrations. *The Journal of General Physiology*, 142(4):465–475, 2013.

- [64] Daniel Kim. *Applications of Entropic DNA Trapping to Studies of Giant Acceleration of Diffusion and a Novel Restriction Mapping Method*. PhD thesis, Brown University, 2017.
- [65] Daniel Kim, Clark Bowman, Jackson T. Del Bonis-O'Donnell, Anastasios Matzavinos, and Derek Stein. Giant acceleration of dna diffusion in an array of entropic barriers. *Phys. Rev. Lett.*, 118:048002, Jan 2017.
- [66] B.J. Kirby. *Micro- and Nanoscale Fluid Mechanics: Transport in Microfluidic Devices*. Cambridge University Press, 2010.
- [67] P.E. Kloeden and E. Platen. *Numerical Solution of Stochastic Differential Equations*. Stochastic Modelling and Applied Probability. Springer Berlin Heidelberg, 2011.
- [68] Marieke Kranenburg, Jean-Pierre Nicolas, and Berend Smit. Comparison of mesoscopic phospholipid-water models. *Phys. Chem. Chem. Phys.*, 6:4142–4151, 2004.
- [69] Lina Kulakova, Georgios Arampatzis, Panagiotis Angelikopoulos, Panagiotis Hadjidoukas, Costas Papadimitriou, and Petros Koumoutsakos. Data driven inference for the repulsive exponent of the lennard-jones potential in molecular dynamics simulations. *Scientific Reports*, 7(16576), 2017.
- [70] Kun-Chun Lee and Andrea J. Liu. Force-velocity relation for actin-polymerization-driven motility from brownian dynamics simulations. *Biophysical Journal*, 97(5):1295–1304, 2009.
- [71] Sang Hyuk Lee and David G. Grier. Giant colloidal diffusivity on corrugated optical vortices. *Physical Review Letters*, 96(19):190601, 2006.
- [72] Huan Lei and George E. Karniadakis. Probing vasoocclusion phenomena in sickle cell anemia via mesoscopic simulations. *Proceedings of the National Academy of Sciences*, 110(28):11326–11330, 2013.

- [73] B. Leimkuhler and C. Matthews. *Molecular Dynamics: With Deterministic and Stochastic Numerical Methods*. Interdisciplinary Applied Mathematics. Springer International Publishing, 2015.
- [74] Benedict Leimkuhler and Xiaocheng Shang. On the numerical treatment of dissipative particle dynamics and related systems. *Journal of Computational Physics*, 280:72 – 95, 2015.
- [75] Stephen L Levy and Harold G Craighead. DNA manipulation, sorting, and mapping in nanofluidic systems. *Chemical Society reviews*, 39(3):1133–52, mar 2010.
- [76] Jianfeng Li, Kyle A. Pastor, An-Chang Shi, Friederike Schmid, and Jiajia Zhou. Elastic properties and line tension of self-assembled bilayer membranes. *Phys. Rev. E*, 88:012718, Jul 2013.
- [77] Oliver Lieleg, Mireille M. A. E. Claessens, and Andreas R. Bausch. Structure and dynamics of cross-linked actin networks. *Soft Matter*, 6:218–225, 2010.
- [78] Erik Lindahl and Mark SP Sansom. Membrane proteins: molecular dynamics simulations. *Current Opinion in Structural Biology*, 18(4):425 – 431, 2008. Membranes / Engineering and design.
- [79] Vesa Loitto, Thommie Karlsson, and Karl-Eric Magnusson. Water flux in cell motility: Expanding the mechanisms of membrane protrusion. *Cell motility and the cytoskeleton*, 66:237–47, 05 2009.
- [80] Xiao Guang Ma, Pik Yin Lai, Bruce J. Ackerson, and Penger Tong. Colloidal dynamics over a tilted periodic potential: Nonequilibrium steady-state distributions. *Physical Review E*, 91(4):042306, 2015.
- [81] Y. Maday and A. T. Patera. Spectral element methods for the incompressible Navier-Stokes equations. pages 71–143, 1989.

- [82] O.L. Maitre and O.M. Knio. *Spectral Methods for Uncertainty Quantification: With Applications to Computational Fluid Dynamics*. Scientific Computation. Springer Netherlands, 2010.
- [83] A man Zhang, Peng nan Sun, Fu ren Ming, and A. Colagrossi. Smoothed particle hydrodynamics and its applications in fluid-structure interactions. *Journal of Hydrodynamics, Ser. B*, 29(2):187 – 216, 2017.
- [84] Hirotami Matsuo, Julien Chevallier, Nathalie Mayran, Isabelle Le Blanc, Charles Ferguson, Julien Fauré, Nathalie Sartori Blanc, Stefan Matile, Jacques Dubochet, Rémy Sadoul, Robert G. Parton, Francis Vilbois, and Jean Gruenberg. Role of lbpa and alix in multivesicular liposome formation and endosome organization. *Science*, 303(5657):531–534, 2004.
- [85] K. Matsuzaki, S. Yoneyama, and K. Miyajima. Pore formation and translocation of melittin. *Biophysical Journal*, 73(2):831 – 838, 1997.
- [86] Katsumi Matsuzaki, Osamu Murase, Nobutaka Fujii, and Koichiro Miyajima. Translocation of a channel-forming antimicrobial peptide, magainin 2, across lipid bilayers by forming a pore. *Biochemistry*, 34(19):6521–6526, 1995. PMID: 7538786.
- [87] Anastasios Matzavinos, Alexander Roitershtein, and Youngsoo Seol. Random walks in a sparse random environment. *Electron. J. Probab.*, 21:20 pp., 2016.
- [88] James L. McGrath, Narat J. Eungdamrong, Charles I. Fisher, Fay Peng, Lakshminarayanan Mahadevan, Timothy J. Mitchison, and Scot C. Kuo. The force-velocity relationship for the actin-based motility of *listeria monocytogenes*. *Current Biology*, 13(4):329 – 332, 2003.
- [89] Florian Milde, Gerardo Tauriello, Hannah Haberkern, and Petros Koumoutsakos. Sem++: A particle model of cellular growth, signaling and migration. *Computational Particle Mechanics*, 1(2):211–227, Jun 2014.
- [90] Alex Mogilner. On the edge: modeling protrusion. *Current Opinion in Cell Biology*, 18(1):32 – 39, 2006. Cell structure and dynamics.

- [91] J.J. Monaghan. Smoothed particle hydrodynamics and its diverse applications. *Annual Review of Fluid Mechanics*, 44(1):323–346, 2012.
- [92] Takaharu Mori, Naoyuki Miyashita, Wonpil Im, Michael Feig, and Yuji Sugita. Molecular dynamics simulations of biological membranes and membrane proteins using enhanced conformational sampling algorithms. *Biochimica et Biophysica Acta (BBA) - Biomembranes*, 1858(7, Part B):1635 – 1651, 2016. New approaches for bridging computation and experiment on membrane proteins.
- [93] Grant I. Nixon and Gary W. Slater. Entropic trapping and electrophoretic drift of a polyelectrolyte down a channel with a periodically oscillating width. *Phys. Rev. E*, 53:4969–4980, May 1996.
- [94] Theo Odijk et al. Scaling theory of DNA confined in nanochannels and nanoslits. *Physical Review Letters*, 77(2):394–401, mar 2008.
- [95] Santtu T. T. Ollila, Colin Denniston, Mikko Karttunen, and Tapio Ala-Nissila. Biopolymer Filtration in Corrugated Nanochannels. *Physical Review Letters*, 112(11):118301, mar 2014.
- [96] H.C. Öttinger. *Stochastic Processes in Polymeric Fluids: Tools and Examples for Developing Simulation Algorithms*. Stochastic Processes in Polymeric Fluids: Tools and Examples for Developing Simulation Algorithms. Springer, 1996.
- [97] Costas Papadimitriou, Konstantina Giakoumi, Costas Argyris, Leonidas Spyrou, and Panagiotis Panetsos. Bayesian estimation of tension in bridge hangers using modal frequency measurements. *Structural Monitoring and Maintenance, An International Journal*, 3:349–375, 10 2016.
- [98] N. Phan-Thien. *Understanding Viscoelasticity: An Introduction to Rheology*. Graduate Texts in Physics. Springer Berlin Heidelberg, 2012.

- [99] Igor V Pivkin and George Em Karniadakis. A new method to impose no-slip boundary conditions in dissipative particle dynamics. *Journal of Computational Physics*, 207(1):114–128, 2005.
- [100] Steve Plimpton. Fast parallel algorithms for short-range molecular dynamics. *J. Comput. Phys.*, 117(1):1–19, March 1995.
- [101] S.D. Poisson. Trait de mecanique. *Bachelier, Paris*, 1833.
- [102] Thomas D Pollard and Gary G Borisy. Cellular motility driven by assembly and disassembly of actin filaments. *Cell*, 112(4):453 – 465, 2003.
- [103] Hong Qian, Michael P Sheetz, and Elliot L Elson. Single particle tracking. Analysis of diffusion and flow in two-dimensional systems. *Biophysical journal*, 60(4):910–21, 1991.
- [104] D.C. Rapaport. *The Art of Molecular Dynamics Simulation*. Cambridge University Press, 2004.
- [105] D. Reguera, G. Schmid, P. S. Burada, J. M. Rubí, P. Reimann, and P. Hänggi. Entropic transport: Kinetics, scaling, and control mechanisms. *Physical Review Letters*, 96(13):130603, 2006.
- [106] P. Reimann, C. Van den Broeck, H. Linke, P. Hänggi, J. M Rubi, and a. Pérez-Madrid. Giant Acceleration of Free Diffusion by Use of Tilted Periodic Potentials. *Physical Review Letters*, 87(1):010602, jun 2001.
- [107] Walter Reisner, Niels B Larsen, Henrik Flyvbjerg, Jonas O Tegenfeldt, and Anders Kristensen. Directed self-organization of single DNA molecules in a nanoslit via embedded nanopit arrays. *Proceedings of the National Academy of Sciences of the United States of America*, 106(1):79–84, jan 2009.

- [108] Alan S Verkman. Solute and macromolecule diffusion in cellular aqueous compartments. *Trends in biochemical sciences*, 27:27–33, 02 2002.
- [109] P. Salamon, P. Sibani, and R. Frost. *Facts, Conjectures, and Improvements for Simulated Annealing*. Monographs on Mathematical Modeling and Computation. Society for Industrial and Applied Mathematics, 2002.
- [110] J. M. Sancho and a. M. Lacasta. The rich phenomenology of Brownian particles in nonlinear potential landscapes. *European Physical Journal: Special Topics*, 187(1):49–62, 2010.
- [111] Christian H. Schreiber, Murray Stewart, and Thomas Duke. Simulation of cell motility that reproduces the force–velocity relationship. *Proceedings of the National Academy of Sciences*, 107(20):9141–9146, 2010.
- [112] Armin Seyfried, Bernhard Steffen, and Thomas Lippert. Basics of modelling the pedestrian flow. *Physica A: Statistical Mechanics and its Applications*, 368(1):232 – 238, 2006.
- [113] T. Shardlow and Y. Yan. Geometric ergodicity for dissipative particle dynamics. *Stoch. Dyn.*, 6(1):123–154, 2006.
- [114] Elijah Shelton, Zhijun Jiang, Shutong Wang, and Derek Stein. Controlling the conformations and transport of DNA by free energy landscaping. *Applied Physics Letters*, 99(26):263112, 2011.
- [115] C. Shen. *Rarefied Gas Dynamics: Fundamentals, Simulations and Micro Flows*. Heat and Mass Transfer. Springer Berlin Heidelberg, 2006.
- [116] Julian C. Shillcock and Reinhard Lipowsky. Equilibrium structure and lateral stress distribution of amphiphilic bilayers from dissipative particle dynamics simulations. *The Journal of Chemical Physics*, 117(10):5048–5061, 2002.

- [117] Teruo Shimmen and Etsuo Yokota. Cytoplasmic streaming in plants. *Current opinion in cell biology*, 16:68–72, 03 2004.
- [118] DE Smith, TT Perkins, and S Chu. Dynamical scaling of DNA diffusion coefficients. *Macromolecules*, pages 1372–1373, 1996.
- [119] R.C. Smith. *Uncertainty Quantification: Theory, Implementation, and Applications*. Computational Science and Engineering. SIAM, 2013.
- [120] Frederick S. Southwick. Gelsolin and adf/cofilin enhance the actin dynamics of motile cells. *Proceedings of the National Academy of Sciences*, 97(13):6936–6938, 2000.
- [121] Volker Springel. Smoothed particle hydrodynamics in astrophysics. *Annual Review of Astronomy and Astrophysics*, 48(1):391–430, 2010.
- [122] Magdalena A. Stolarska, Yangjin Kim, and Hans G. Othmer. Multi-scale models of cell and tissue dynamics. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 367(1902):3525–3553, 2009.
- [123] A.J. Storm, J.H. Chen, X Ling, Henny Zandbergen, and C Dekker. Fabrication of solid-state nanopores with single-nanometre precision. *Nature materials*, 2:537–40, 09 2003.
- [124] J.C. Strikwerda and Wisconsin Univ-Madison Mathematics Research Center. *Finite Difference Methods for the Stokes and Navier-Stokes Equations*. Mathematics Research Center technical summary report. Defense Technical Information Center, 1982.
- [125] D.W. Stroock. *An Introduction to Markov Processes*. Graduate Texts in Mathematics. Springer Berlin Heidelberg, 2004.
- [126] Elizabeth a. Strychalski, Stephen L. Levy, and Harold G. Craighead. Diffusion of DNA in Nanoslits. *Macromolecules*, 41(20):7716–7721, oct 2008.

- [127] L. Taddei, A. Awoukeng Goumtcha, and S. Roth. Smoothed particle hydrodynamics formulation for penetrating impacts on ballistic gelatine. *Mechanics Research Communications*, 70:94 – 101, 2015.
- [128] Jing Tang, Stephen L. Levy, Daniel W. Trahan, Jeremy J. Jones, Harold G. Craighead, and Patrick S. Doyle. Revisiting the Conformation and Dynamics of DNA in Slitlike Confinement. *Macromolecules*, 43(17):7368–7377, sep 2010.
- [129] Yu-Hang Tang and George Em Karniadakis. Accelerating dissipative particle dynamics simulations on gpus: Algorithms, numerics and applications. *Computer Physics Communications*, 185(11):2809 – 2822, 2014.
- [130] Yu-Hang Tang, Lu Lu, He Li, Constantinos Evangelinos, Leopold Grinberg, Vipin Sachdeva, and George Em Karniadakis. Openrbc: A fast simulator of red blood cells at protein resolution. *Biophysical Journal*, 112(10):2030 – 2037, 2017.
- [131] A. Tarantola. *Inverse Problem Theory and Methods for Model Parameter Estimation*. Other titles in applied mathematics. Society for Industrial and Applied Mathematics, 2005.
- [132] Jonas O Tegenfeldt, Christelle Prinz, Han Cao, Steven Chou, Walter W Reisner, Robert Riehn, Yan Mei Wang, Edward C Cox, James C Sturm, Pascal Silberzan, and Robert H Austin. From the Cover: The dynamics of genomic-length DNA molecules in 100-nm channels. *Proceedings of the National Academy of Sciences of the United States of America*, 101(30):10979–10983, 2004.
- [133] Arpita Upadhyaya, Jeffrey R. Chabot, Albina Andreeva, Azadeh Samadani, and Alexander van Oudenaarden. Probing polymerization forces by using actin-propelled lipid vesicles. *Proceedings of the National Academy of Sciences*, 100(8):4521–4526, 2003.
- [134] Arpita Upadhyaya and Alexander van Oudenaarden. Biomimetic systems for studying actin-based motility. *Current Biology*, 13(18):R734 – R744, 2003.

- [135] M. W. Vanik, J. L. Beck, and S. K. Au. Bayesian probabilistic approach to structural health monitoring. *Journal of Engineering Mechanics*, 126(7):738–745, 2000.
- [136] Christian L Vestergaard, Morten Bo Mikkelsen, Walter Reisner, Anders Kristensen, and Henrik Flyvbjerg. Transition state theory demonstrated at the micron scale with out-of-equilibrium transport in a confined environment. *Nature Communications*, 7:10227, 2016.
- [137] A. Vrugt, C. J. F. ter Braak, C. G. H. Diks, B. A. Robinson, J. M. Hyman, and D. Higdon. Accelerating markov chain monte carlo simulation by differential evolution with self-adaptive randomized subspace sampling. *International Journal of Nonlinear Sciences and Numerical Simulation*, 10:273, 2011.
- [138] Scott J. Weiner, Peter A. Kollman, David A. Case, U. Chandra Singh, Caterina Ghio, Guliano Alagona, Salvatore Profeta, and Paul Weiner. A new force field for molecular mechanical simulation of nucleic acids and proteins. *Journal of the American Chemical Society*, 106(3):765–784, 1984.
- [139] T.J. Willmore. *Total curvature in Riemannian geometry*. Mathematics and Its Applications Series. Ellis Horwood, 1982.
- [140] Lingling Wu and Stephen Levy. Fluctuations of DNA mobility in nanofluidic entropic traps. *Biomicrofluidics*, 8(4):044103, jul 2014.
- [141] S. Wu, P. Angelikopoulos, C. Papadimitriou, R. Moser, and P. Koumoutsakos. A hierarchical Bayesian framework for force field selection in molecular dynamics simulations. *Philosophical Transactions of the Royal Society of London Series A*, 374:20150032, February 2016.
- [142] D. Xiu. *Numerical Methods for Stochastic Computations: A Spectral Method Approach*. Princeton University Press, 2010.
- [143] K.V. Yuen. *Bayesian Methods for Structural Dynamics and Civil Engineering*. Wiley, 2010.

- [144] Ou-Yang Zhong-can and Wolfgang Helfrich. Bending energy of vesicle membranes: General expressions for the first, second, and third variation of the shape energy and applications to spheres and cylinders. *Phys. Rev. A*, 39:5280–5288, May 1989.
- [145] Jie Zhu and Alex Mogilner. Mesoscopic model of actin-based propulsion. *PLOS Computational Biology*, 8(11):1–12, 11 2012.
- [146] Juliane Zimmermann, Claudia Brunner, Mihaela Enculescu, Michael Goegler, Allen Ehrlicher, Josef Ks, and Martin Falcke. Actin filament elasticity and retrograde flow shape the force-velocity relation of motile cells. *Biophysical Journal*, 102(2):287 – 295, 2012.
- [147] R Zwanzig. *Nonequilibrium Statistical Mechanics*. Oxford University Press, USA, 2001.