NUMERICAL STUDIES ON COLLECTIVE MOTION AND POLYMER STATISTICS

WANG NAN

NATIONAL UNIVERSITY OF SINGAPORE

2014

NUMERICAL STUDIES ON COLLECTIVE MOTION AND POLYMER STATISTICS

WANG NAN

(B.Sc.(Hons), National University of Singapore)

A THESIS SUBMITTED

FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

DEPARTMENT OF MATHEMATICS

NATIONAL UNIVERSITY OF SINGAPORE

2014

DECLARATION

I hereby declare that this thesis is my original work and it has been written by me in its entirety.
I have duly acknowledged all the sources of information which have been used in the thesis.

This thesis has also not been submitted for any degree in any university previously.

王楠

Wang Nan
20 Aug 2014

Acknowledgements

I would like to express my deepest appreciation to professor Bao Weizhu for his inspiration, guidance and encouragement. He also gave me the chance and lead me to the world of computational mathematics. Without his help, this thesis and many others would not be possible.

I would like to thank Prof Pierre Degond and Prof Wang Zhisong for their support and guidance during the collaboration. They provide constructive suggestions along my research.

I would like to thank my research fellows Ngoc and Hou Ruizheng for their discussion and inspirations.

Special thanks are given to Zhao Xiaofei for helping me check the thesis.

I would also like to thank friends Yuan Zihong, Huang Mengmin, Jia Xiaowei, Wang Yan and many others for their friendship and support.

Finally I would like to thank my parents and my wife for their understanding, patience and love during the past several years.

Contents

Acknowledgements **i**

Summary **v**

List of Figures **vii**

1 Introduction **1**

1.1 Problems 1

1.1.1 Collective motion 1

1.1.2 Polymer statistics 4

1.2 Scope and outline of the thesis 6

2 Models and Methods for Collective Motion of Particles **8**

2.1 Existing models 8

2.2 Modified Vicsek model 11

2.2.1 Microscopic model and the mean field limit 12

2.2.2 Scaling 15

2.2.3 Hydrodynamic limit 18

2.3 Particle model generated from Navier-Stokes system 31

2.3.1 Macroscopic model 31

2.3.2 Scaling 33
2.3.3 Particle method for fluid and microscopic model 34
2.4 Numerical methods 43
2.4.1 GPU parallelization 43
2.4.2 Numerical methods for microscopic modified Vicsek model . . 44
2.4.3 Numerical methods for the macroscopic model 51
2.4.4 Numerical methods for particle model from the Navier-Stokes system 57
2.5 Numerical results 60
2.5.1 Microscopice modified Vicsek model 60
2.5.2 Comparison between the microscopic and macroscopic model . 61
2.5.3 Particle model from the Navier-Stokes system 66

3 Models and Methods for Collective Motion of Polymers 68
3.1 Existing models 68
3.2 Self-propelling polymer model 70
3.2.1 Numerical methods 71
3.2.2 Numerical results 74
3.3 Polymer fluid interaction 74
3.3.1 Numerical methods 78
3.3.2 Numerical results 80

4 Single Polymer Statistics 82
4.1 Existing models 84
4.2 Worm Like Chain model 88
4.3 Methods for the WLC end to end distribution 90
4.3.1 1D case 90
4.3.2 3D case 98
4.4 Numerical results and applications 103
4.4.1 Single-polymer ‘flyfishing’ by a local alignment at one end . . 103

4.4.2 Single-polymer power stroke and intra-chain force transmission 106
4.4.3 Site-selective dissociation by intra-chain force 108
4.4.4 New force-extension formula 109

5 Conclusion and Future Perspectives 118

Bibliography 121

Summary

Mathematical models have been applied to various biological problems for a long history, including the studies of populations, DNA sequences, pattern formations and protein structures. This thesis aims to study two areas in computational biology, namely the collective motion and polymer statistics.

The first part of the thesis focuses on collective motion. Collective motion, or flocking behaviour studies the common coordinated behaviour which is observed in many scenarios. For example, animal society like schools of fish, herds of sheep, swarm of locusts, and even a collection of micro-organisms like bacteria or sperms perform collective motion. While individual may only react to their neighbours, the overall structure obtained can be complex. It is therefore interesting to find suitable particle interaction rules. Models have been proposed in both microscopic and macroscopic levels.

In this thesis, we begin with a review of microscopic models for particles. Different interaction rules and models have been proposed to match different senarios with different complexity. We try to understand the link between micro models and macro models. Two approches are used. The first one is a bottom-up approach. We focus on the Vicsek model with repulsion. Starting for a mean-field description, we build a fluid limit or continuum limit to the system. The result is a set of non-conservative hydrodynamic equations. Numerical schemes are proposed and the results for microscopic and macroscopic models are compared to validate the derivation. The second one is a top-down approach. Starting from the fluid model, we review particle methods for fluid simulation. We try to discretize the active fluid model to particle level and yield an interaction rule knowing the global structure. Furthermore, the simulation for a large system is achieved with the help of GPU acceleration.

For micro-organisms living in fluid, volume exclusion effect and hydrodynamic forces are important for the collective motion pattern formation. We simulate a large system of rigid self-propelling rods. Extensive numerical simulations are performed in rectangular, circular or annulus domains with different boundary conditions, leading to different patterns. We then review some methods to simulate particles in viscous fluids and try to understand the flow field generated by micro swimmers.

The second part of the thesis deals with polymer statistics. Polymers are chains made up with repeating units. For example, polymers include DNA, collagens, actin filaments, microtubules and motor proteins such as kinesin. We will review some models used in polymer theory. The most popular model for semi-flexible polymer is the worm like chain model. Despite its simplicity, it can model soft chains as well as rigid rods. An understanding of the statistics of the worm like chain is the basics for applications. After reviewing the existing polymer models, we use a path integral approach to map the problem to a quantum rotor on a unit sphere to get the 3d end to end distribution of the worm like chain. With the distrubution at hand, we can get the force extension relationship and free energies of the chains with different conformations.

List of Figures

1.1 A gallery of images related to collective motion 2

2.1 Three zone model 11

2.2 Technical Circle 46

2.3 Illustration of Verlet table algorithm 48

2.4 Illustration of cell linked list algorithm 49

2.5 Simulation for 2×10^4 particles at time $T = 0, 0.5, 1.0, 3.0$ respectively. 61

2.6 Simulation for 2×10^4 particles at time $T = 1$ with $\mu = 0, 10, 100, 300$ respectively. The initial orientation is randomized and the same initial data is used for all the tests. 62

2.7 Relative error between the macroscopic and the microscopic model for density (left) and θ (right) as a function of the number of averages for different values of ϵ. The error decreases with both decreasing ϵ and increasing number of averages, showing that the microscopic model provides a valid approximation of the individual based model for ρ and θ. 64

2.8 Solution of the Riemann problem along the x-axis for the macroscopic model (blue line) and for the microscopic model with $\epsilon = 0.05$ (red line) at $T = 1$. 65

2.9 Density ρ for the Green Taylor Problem at $T = 0.6$. Left:microscopic model. Right:macroscopic model 65

2.10 Mean orientation Ω for the Green Taylor Problem at $T = 0.6$. Left:microscopic model. Right:macroscopic model 66

2.11 Particle simulations in 2D domains. Top left: rectangle domain $L_x = L_y = 1$ with periodic boundary condition. Top right: rectangle domain $L_x = L_y = 1$ with periodic boundary condition in x direction and Neumann boundary condition in y direction. Bottom left: circle domain, radius=1, Neumann boundary condition. Bottom right: annulus domain, outer radius=1, inner radius=0.4, Neumann boundary condition. 67

3.1 Rigid rod simulations. Top left: 2D rectangle domain $L_x = L_y = 1$ with periodic boundary condition. Top right: 2D annulus domain, outer radius=1, inner radius=0.3, repulsive boundary condition. Bottom figures show the corresponding 3D domain with $L_z = 0.3$. 75

3.2 Rigid rod simulations. Left: n=10, 600 rods, highly local alignment observed for long rods. Right: n=4, 500 rods, alignment normal to the boundary observed near the boundary. 75

3.3 The mesh and streamlines for a ellipsoid particle. 81

3.4 The interaction for 2 particles. They approach each other and align. . 81

4.1 The probability distribution integrated over a plane $P(z)$. Left column: both ends are free. Middle column: the initial end is fixed along the z axis. Right column: both ends are fixed along the z axis. 111

4.2 The probability distribution $P(x, y, z)$ taken at different y planes for different values of β, fixing $l_c = 1$ 112

4.3 The probability distribution $Q(z)$, energy $F(z)$ and force $f(z)$ for the case $l_p = 4$nm and $l_c = 10$nm, with the starting direction being free or make an angle θ with the z axis. 113

4.4 The critical z where $Q(z)$ reaches its maximum and the full width half maximum for different θ where the starting vector makes an angle θ with the z-axis. Here $l_c = 10$nm and $l_p = 4$nm. 114

4.5 The critical z where $Q(z)$ reaches its maximum and the full width half maximum for different l_p 114

4.6 The probability distribution Q(z) (blue curve) together with the Gauss approximation for different l_p 114

4.7 The energy difference ΔF for a chain with one end fixed along the z axis. In the left figure, for each value of z, $\Delta F \to \Delta F^*$ as $l_p \to 0$. The dependence of ΔF^* on z is plotted in the right figure(blue curve). The probability $Q(z)$ can be well described by the Gaussian function. Taking $l_p \to 0$, we can compute $\Delta F_{Gauss}(z) = 0.30478z$(green curve). The energy difference agrees near $z = 0$nm. 115

4.8 The force at $z = -7$nm and $z = 7$nm, for different θ 115

4.9 The comparison between the new fit formula and the force $f(z)$ obtained by numerical methods for different β. 116

4.10 The probability distribution, energy, force at position $z = 5$nm and $z = 8$nm, for different l_p 117

Chapter 1

Introduction

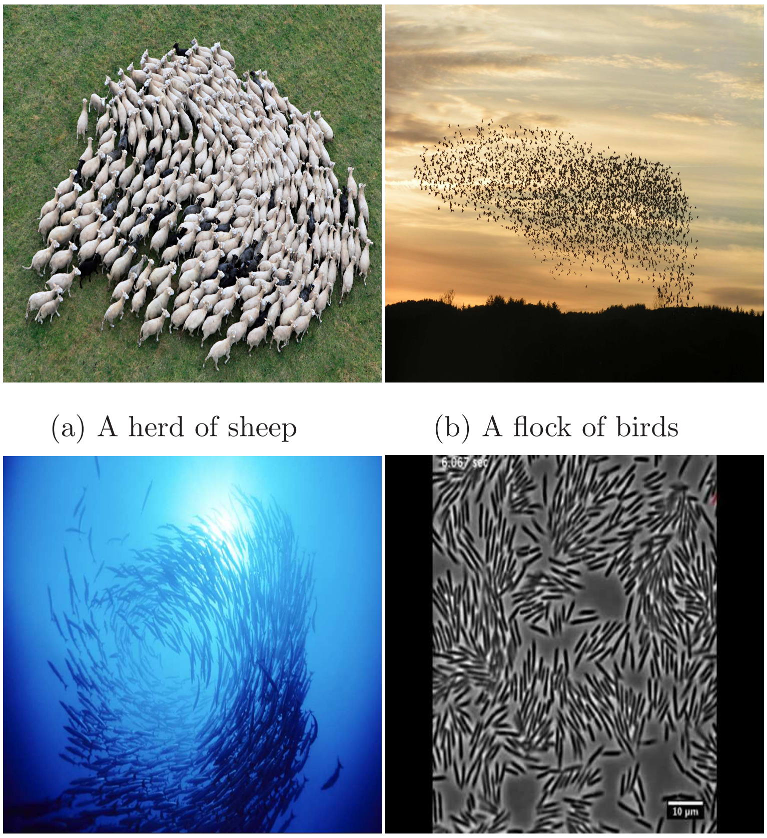
1.1 Problems

1.1.1 Collective motion

Collective motion is a very common behaviour in nature. Many biological systems with different scales show the coherent motion of large number of individuals. For example,a school of fish [21], a flock of birds [82]. The scale can be as large as kilometers (herds of beasts) and can be as small as micrometers (bacteria) [112].

In particular, we are interested in the collective motion of spermatozoa. Under a phase contrast microscope, for a drop of undiluted semen sample, we can observe that millions of spermatozoa move together forming whirlpools and circular waves. The collective phenomenon is termed massal motility. It is believed that the massal motility is the only parameter which has a good correlation with male fertility, but not the individual motility of a single spermatozoa. An understanding of the collective motion can help us predict the semen fertility.

The study of collective behaviour has a long history and different approaches has been taken. Different parameters describing the system can be extracted, including density, polarity, packing fraction and so on. Experiments are carried out to identify the collective motions. These include non-living systems(for example, shaken



(a) A herd of sheep (b) A flock of birds

(c) A school of fish (d) A colony of bacteria

Figure 1.1: A gallery of images related to collective motion

metallic rods, simple robots, etc) [110], macromolecules, bacteria colonies, cells [30], insects, fish, birds, mammals and even human. The data collection techniques are not discussed in details here.

In the mathematical point of view, different models are proposed, that range from single particles to those with complex geometries, and from microscopic level to macroscopic level. The most simple form is the microscopic description of particles, which are termed self-propelled particles (SPP). The individual is described by its position, velocity and orientation, and the main feature of collective motion is that the individual behaviour is dominated by the influence of others. Exact rules for the motion of each individual are defined, possibly with a noise term, which would then yield a stochastic process. These rules are termed "collision rules", which describe how individual would react to their neighbours. Based on the numerous observations for different systems, the following hypotheses can be made about collective motion: the tendency to adopt the motion of the neighbours is the main reason for collective motion, and there is a possible universal class of patterns since similar observations can appear in very different origins.

Based on the observations, different collision rules are proposed. The first widely-known flocking simulation was published by Reynolds in 1987, who just want to visualize birds like flying objects, or "boids". Collision rules include alignment and avoidance. After that, a statistical physics type of approach to collective motion, or the Vicsek model was introduced in 1995 by Vicsek. A random perturbation of the direction is included in their system. A lot of research has been done by varying some parameters or initial settings. The simulations exhibit a rich variety of collective behaviours.

More sophisticated models consider individual agents not only as particles, but with specific shape and volume. The shape selected depends on the objects to be modelled. To study the collective motion of micro-swimmers, since many bacteria and sperms have elongated body, usually we consider them as polymers and use hard rod or linked beads to model them. The most simple model is the active self-propelled particles in the paper of Baskaran and Marchetti. The particles are basically asymmetric rigid dumbbells, where two spheres are connected with a infinitely rigid rod. Peruani simulated self propelled hard rods with a purely physical mechanism. It is shown that the steric interaction due to volume exclusion leads to the formation of moving clusters. Experiments with granular particles, i.e. artificial self-propelled rods, have been performed to confirm that such a physical mechanism is indeed enough to produce a variety of collective motion pattern.

Furthermore, for micro-organisms swimming in a fluid medium, the hydrodynamics effect is often enough to generate a collective motion pattern, e.g., clusters, vortices. Many works have been done to study the fluid mechanics. One of the first general method for computing the hydrodynamic interactions among an suspension of particles was presented by Brady in 1988. Their method is actually the basis of the Stokesian dynamics, a technique to study the velocity field for a particle system in Stoke flow. Another approach, the so called slender-body theory was used by Saintillan and Shelley to study the dynamics of self-propelled slender rods. Several other different factors can be included in the bacteria model, including chemotaxis, polarity and adhesion. It is still an open problem to get a realistic simple model to capture the behaviour of micro-swimmers.

1.1.2 Polymer statistics

A polymer is a chain made of many repeating units, they are created by polymerization of many small molecules, which is termed monomers. Examples of polymers include the synthetic plastics such as polystyrene, as well as natural biopolymers such as DNA and proteins. Due to the free rotations between the single bonds in a polymer molecule, a single polymer molecule can have an enormous number of different configurations, which are referred as polymer configurations. The difficulty of a complete description of a single polymer configuration arises from the huge number of degree of freedoms, and we can see that the description of a single polymer molecule is already a many body problem. However in real life experiments, we cannot identify and measure a single polymer in molecular level, and we can only consider some average over many different configurations. Thus we will study polymer properties by means of statistic mechanics.

The study of polymer solution science started with the celebrated book by Flory in 1953, where the concept of the excluded-volume effect was accepted. After that, the study focused on flexible polymers within the Flory framework, which consists the concept of Gaussian chain. If the chain length is decreased, then the stiffness of the polymer becomes an important factor even for ordinary flexible chain polymers as well as stiff or semi-flexible macromolecules such as DNA and the α-helical polypeptides. In atomic level, the stiffness result in the hindrances to internal rotations within the chain and other structural constraints. However, the details are usually unnecessary to consider and cannot be treated easily. Therefore continuous models were proposed. The first of them is the worm like chain model proposed by Kratky and Porod in 1949 [73]. Other modifications such as the helical worm like chain were also developed later. Although the worm like chain model cannot mimic exactly the dimensional behaviour of a real chain, it offers a rather good approximation for polymers with a wide range of stiffness, and it acts as a fundamental tool to predict the physical behaviour of many biopolymers. Examples include polymer liquid crystals [125], polyelectrolytes [50], protein networks and DNA molecules.

One of the key descriptor of the worm like chain statistics is the end to end distribution function. For the Gaussian chain, the end to end distribution is well known. However, it is not a trivial task to find the end to end distribution for a worm like chain. Several studies have addressed the problem in different limits. In the flexible limit, the chain tends to a Gaussian chain, and the behaviour of finite rigidity captures the first moments [36]. In the rigid rod limits, the chain statistics are obtained using a path integral formulation with a fixed end orientation [129]. Another approach for the rigid rod limit is evaluating the partition function by summing over fluctuations about a nearly straight chain.

Beside the studies for the chain statistics, the exact end to end distribution is only obtained in the last decade. Several different works tried different approaches. One approach is a numerical study by direct diagolization of the truncated scattering matrix [108]. A result for the end to end distribution in three dimensions in Fourier-Laplace space (Fourier-transformed end position and Laplace transformed chain length) was obtained using algebraic techniques [117]. Another approach used diagrammatic methods and got the chain statistics in Fourier-Laplace space in the form of infinite continued fractions [113]. For the latter method, the statistical behaviour in real space still requires an inversion from Fourier-Laplace space.

1.2 Scope and outline of the thesis

As shown in the previous sections, a large number of models have been proposed to study the collective motion of animal societies. However, a model which can successfully describe the behaviour of spermatozoon is still missing. Most existing microscopic models will produce aggregation of particles and the formation of colonies, while a suspension of sperms is a rather homogeneous solution. In chapter 2, we try to understand the relationship between the individual motion and global fluid like motion they can produce. Two approaches are used. The first is a bottom-up approach. We will start with the Vicsek model and add repulsion between individuals. Its hydrodynamic limit is derived to understand the global structure of the motion. The second approach make use of the particle methods for fluids. Knowing the suspension behaves as an self-propelling active fluid, we can get a microscopic model out of it. Different particle methods will be discussed. Numerical simulations will be carried out to justify the derivation.

In chapter 3, we will move to more sophisticated models for a better description of the collective motion. By making use of GPU acceleration, we can model the sperms as self propelling rods and take account of the volume exclusion effect. We will study how the shape of the rods and different boundary conditions can affect the motion of the system. For more precise descriptions, we will also include the hydrodynamic forces and try to understand the fluid particle interactions.

Chapter 4 then studies a single polymer statistics. We will review some famous existing polymer models and then focus on the worm like chain model. Numerical methods are proposed to get the 3d end to end distributions with different end conformations. The results suggest the possibility of a surprising accurate flyfish-like control in which tilting one end of a semiflexible polymer enables positioning of the other diffusing end to a remote location within an error of 1nm. With the exact statistics at hand, we can easily get the free energy and force-extension relationships. A new force-extension formula that is valid for polymer with different rigidity is obtained. The formula provides a convenient tool to estimate direction and magnitude of intra-chain force, which are critical in site selective dissociation in nanomotors.

Finally the main results will be summarized in chapter 5. We will also address some interesting topics for future works.

Chapter 2

Models and Methods for Collective Motion of Particles

In this chapter, we study the collective motion of particles where each individual is represented by a point and the shape of the object is not considered. We will firstly review the existing models in the literature. Then we will focus on selected models and study them on both microscopic level and macroscopic level. Then accurate and efficient numerical methods are proposed for the models for comparison.

2.1 Existing models

By our knowledge, Vicsek is the first in understanding collective motion [122]. He showed order-disorder phase transitions can be achieved using a simple model, which is the discrete Vicsek model. The discrete Vicsek model considers N point particles $x_i^n \in \mathbb{R}^3$ at discrete times t_n. Each particle has orientation ω_i^n which belongs to the unit sphere $\mathbb{S}^2 = \{\omega, |\omega|^2 = 1\}$ and velocity $v_i^n \in \mathbb{R}^3$. The system is updated in the following manner.

$$v_i^{n+1} = v_0\bar{\omega}_i^n + \text{perturbation}, \tag{2.1.1}$$

$$x_i^{n+1} = x_i^n + v_i^{n+1}. \tag{2.1.2}$$

Here $\bar{\omega}_i^n$ represents an average orientation of the neighbour particles near particle i, and v_0 is the self-propulsion speed. Such particles are termed "self propulsion particles"(SPP). Note that the interaction is short ranged. It is interesting that in the limit v_0 goes to 0, the model becomes analogous to a classical Ising model in the study of ferromagnet. One key finding of this model is that, as the number of particles increase, there is a sharp phase transition from a disordered state to an aligned state [63], and this observation is confirmed using locusts [17]. Even without an common quadratic Lyapunov function, the system is proved to be stable [68]. Although the Vicsek model is not a very realistic model, it can be easily modified and is applied in many areas due to its simplicity, examples include school of fish [49], robotic swarms [7] and even human trails [57]. Variations for the Vicsek model include changing symmetry, adding local cohesion and considering fluid [23].

Another model which received great attention is proposed by Cucker and Smale [32]. Instead of interaction with the neighbours, each individual adjusts its velocity as a weighted average of the whole population as follow:

$$\frac{v_i^{n+1} - v_i^n}{\delta t} = \frac{\lambda}{N} \sum_{i=1}^{N} a_{ij}(v_j^n - v_i^n).$$

Here a_{ij} indicate the weight of how particle j will affect particle i. In [32], it is defined as

$$a_{ij} = \frac{1}{(1 + ||x_i - x_j||^2)^\beta}.$$

for some $\beta > 0$.

Denote

$$\Gamma(x) = \frac{1}{2} \sum_{i \neq j} ||x_i - x_j||^2 \text{ and } \Lambda(v) = \frac{1}{2} \sum_{i \neq j} ||v_i - v_j||^2.$$

The main result in [32] is that when $\beta < \frac{1}{2}$, the flock will converge to a constant velocity unconditionally, where the initial configuration is not important. However when $\beta \geq \frac{1}{2}$, the initial velocity and position have to satisfy certain compatible conditions for collective behaviour. Another simple proof based on the explicit construction of a Lyapunov functional can be found in [55]. A remarkable application is that a spacecraft control law is designed based on the Cucker Smale model [99]. Similar as the Vicsek model, the Cucker Smale model can be easily modified. For example, [111] studies the emergent behavior under hierarchical leadership. [34] further considered some random interactions.

Both Vicsek model and Cucker Smale model are "Individual Based Model(IBM)" where the individual behaviour is studied based on the interaction with other individuals. Despite their success in modelling various fields, it becomes computational expensive and time consuming without parallel computing. Many works has been done to understand the models in macroscopic level. The statistic version of the models are also call kinetic models. Kinetic model of the Vicsek model can be found in [15, 16] while kinetic model of the Cucker Smale model can be found in [20]. It is interesting to see that the kinetic Cucker Smale model actually converge to the kinetic Vicsek model [18]. With an assumption of weak anisotropy of the velocity distribution function, a hydrodynamic model of the Vicsek model is derived [10, 11]. [103] offered another try to get a fluid model directly from the Vicsek model. The first work which derives a fluid limit via the mean-field kinetic version can be found in [41]. They make use of the concept called the generalized collision invariant(GCI). Further works can be found in [38–40].

To build more realistic models, many models proposed in the biological literature focus on three interaction rules between individuals. Basically they are repulsion, alignment and attraction. When animals get too close, they will avoid each other. When they are at intermediate distances they will try to align with each other. Lastly when they will attract those who are very far away. These models are known as three zone models [5, 29, 67]. Different elements can be added into the three-zone models depending on the species of the animals. For example, the position of the eyes of a bird is much different from those of a locust. And cone of visibility can be taken into account for the interaction. Closed-neighbor interaction, noise and other factors can also be considered. Other works include simulations considering the speed change when joining or leaving a group [54], avoiding predators [121] or

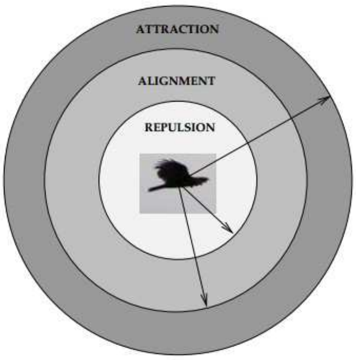


Figure 2.1: Three zone model

clustering when a predator approaches [124]. With different parameters, different zoology patterns are classified rather than a simple flocking behaviour, including rotating single and double mills, rings and clumps [33].

2.2 Modified Vicsek model

To understand the behaviour of a collection of sperm cells. The most simple model considered would be the Vicsek model. In [41], the hydrodynamic limit for the Vicsek model is proposed. We will name this system as the Self-Organized Hydrodynamic (SOH) model. To be precise, SOH model reads:

$$\partial_t \rho + c_1 v_0 \nabla_x \cdot (\rho\Omega) = 0, \tag{2.2.1}$$

$$\rho\partial_t \Omega + c_2 v_0 \rho(\Omega \cdot \nabla_x)\Omega + v_0 d P_{\Omega^\perp} \nabla_x \rho = \gamma P_{\Omega^\perp} \Delta_x(\rho\Omega), \tag{2.2.2}$$

$$|\Omega| = 1. \tag{2.2.3}$$

Note that (2.2.1) describes the conservation of mass, (2.2.2) controls the orientation which is the mean of neighbouring particles, and finally, (2.2.3) is a constraint term. If (2.2.3) is satisfied initially, it will be automatically satisfied for all times due to the projection operator $P_{\Omega^\perp}$. Note that this system is similar as the Navier-Stokes system in the sense that both contains a non-linear hyperbolic part followed by a diffusion term. The difference is that the momentum is not conserved in the

SOH model. Another important difference is that the convection velocities for the density and the orientation, v_0c_1 and v_0c_2 are different while they are the same for the Navier-Stokes system. As a consequence, the propagation of sound waves is anisotropic for this type of fluids [120].

Although the Vicsek model results in a fluid type system, it does not take consideration of the volume exclusion effect and there could be a formation of very high particle concentration. For a suspension of sperm cells, each is an elongated body and the repulsion between each individual is important resulting in a rather homogeneous suspension. Therefore a more reasonable model would be adding repulsion to the Vicsek model.

2.2.1 Microscopic model and the mean field limit

In the microscopic level, the model is stated as below.

$$\frac{dx_i}{dt} = v_i, \tag{2.2.4}$$

$$\frac{dv_i}{dt} = \mu_1(v_0\omega_i - v_i) + \frac{\mu_2}{N}\sum_j F_{ij}, \tag{2.2.5}$$

$$d\omega_i = P_{\omega_i^\perp}(\nu\bar{\omega}_i dt + \sqrt{2D}dB_t + \alpha v_i dt). \tag{2.2.6}$$

Here (2.2.4) simply describes the spatial motion of particle i with velocity v_i. (2.2.5) describes the force. The first term is a self propulsion force where each particle tends to move towards its orientation ω_i with a fixed speed v_0, and the second term is a repulsive force where F_{ij} represents the pairwise interaction between particles i and j. It can be written as $F_{ij} = -\nabla_x\phi(x_i - x_j)$. The support of the smooth potential ϕ is a ball with radius r. μ is the mobility coefficient and N is the total number of particles. Finally (2.2.6) describes the time evolution of the orientation. Here $P_{\omega_i^\perp} = Id - \omega_i \otimes \omega_i$ is the projection operator onto the orthogonal plane of ω_i. Here Id is the identity matrix and $\otimes$ represents the tensor product of two vectors. This projection operator ensures that ω_i is always a unit vector.

The mean orientation $\bar{\omega}_i$ is the mean orientation around particle i and is defined as

$$\bar{\omega}_i = \frac{J_i}{|J_i|}, \quad J_i = \sum_j K(|x_i - x_j|)\omega_j. \tag{2.2.7}$$

It is constructed in a way such that it is the normalization of the "total orientation" J_i, which sums up all the orientation vectors ω_j within the observation kernel K, which is the indicator function of a ball centred at origin and with radius R.

The second term describes a white noise with intensity $\sqrt{2D}$. Note that the noise here is against collective motion thus competes with the first term. In 2-dimensional space, the orientations belong to the unit circle $\mathbb{S}^1$ and one can write $\omega = e^{i\theta}$. In the original version, a uniform noise in a small interval of angles $[-a, a]$ can be added to θ. In [122], it is shown that there exists a threshold value a^*. For $a < a^*$, a coherent dynamic structure is achieved while for $a > a^*$, the system becomes disordered at all times.

The last term models the relaxation of the particle orientation towards the direction of the particle velocity v_i with rate α.

For a suspension of micro-swimmers in a fluid with very small Reynolds number, we will consider the velocity in the overdamped regime. In (2.2.5), we consider the case when μ_1 goes to infinity and divide both sides by μ_1, and let $\mu = \frac{\mu_2}{\mu_1}$. The system can be rewritten as

$$\frac{dx_i}{dt} = v_i, \tag{2.2.8}$$

$$v_i = v_0\omega_i + \frac{\mu}{N}\sum_j F_{ij}, \tag{2.2.9}$$

$$d\omega_i = P_{\omega_i^\perp}(\nu\bar{\omega}_i dt + \sqrt{2D}dB_t + \alpha v_i dt). \tag{2.2.10}$$

We now consider the mean-field kinetic equation which describes the time evolution of the particle system in the large N limit. The unknown is a distribution function which depends on the position $x \in \mathbb{R}^n$, orientation $\omega \in \mathbb{S}^{n-1}$, as well as time t. Note that in the overdamped regime, the velocity can be readily computed once the orientation is determined. We consider the case without the Gaussian noise first.

Introduce the so-called empirical distribution $f^N(x,\omega,t)$ defined by:

$$f^N(x,\omega,t) = \frac{1}{N}\sum_i^N \delta(x - x_i(t))\delta(\omega,\omega_i(t)). \tag{2.2.11}$$

Note that $\delta(\omega,\omega_i(t)) \neq \delta(\omega - \omega_i(t))$ because the sphere $\mathbb{S}^2$ is not left invariant by subtraction. We can show that

$$\partial_t f + \nabla_x \cdot (v_f f) + \nu\nabla_\omega \cdot (P_{\omega^\perp}\bar{\omega}_f f) + \alpha\nabla_\omega \cdot (P_{\omega^\perp} v_f f) = 0. \tag{2.2.12}$$

Proof. Define $\langle\psi\rangle_f(t)$ to be $\langle f,\psi\rangle$ where $\psi(x,\omega)$ is a smooth test function. Therefore $\langle\psi\rangle_f(t) = \frac{1}{N}\psi(x_i(t),\omega_i(t))$,

$$\begin{aligned}\frac{d}{dt}\langle\psi\rangle_f &= \langle\frac{\partial f}{\partial t},\psi\rangle \\ &= \frac{1}{N}\sum_{i=1}^N(\nabla_x\psi\frac{dx_i}{dt} + \nabla_\omega\psi\frac{d\omega_i}{dt}) \\ &= \frac{1}{N}\sum_{i=1}^N(\nabla_x\psi\cdot v_i + \nabla_\omega\psi\cdot P_{\omega_i^\perp}(\nu\bar{\omega}_i + \alpha v_i)) \\ &= \langle f, v_f\cdot\nabla_x\psi + P_{\omega^\perp}(\nu\bar{\omega}_f + \alpha v)\cdot\nabla_\omega\psi\rangle \\ &= -\langle\nabla_x\cdot(v_f f) + \nabla_\omega\cdot(P_{\omega^\perp}(\nu\bar{\omega}_f f + \alpha v_f f)),\psi\rangle.\end{aligned} \tag{2.2.13}$$

□

Together with the noise, we will admit that the evolution of f will be governed by the following system:

$$\partial_t f + \nabla_x \cdot (v_f f) + \nu\nabla_\omega \cdot (P_{\omega^\perp}\bar{\omega}_f f) + \alpha\nabla_\omega \cdot (P_{\omega^\perp} v_f f) - D\Delta_\omega f = 0, \tag{2.2.14}$$

$$v_f(x,\omega,t) = v_0\omega - \mu\int_{\mathbb{S}^{n-1}\times\mathbb{R}^n}\nabla_x\phi(x-y)f dy d\omega, \tag{2.2.15}$$

$$\bar{\omega}_f = \frac{J_f(x,t)}{|J_f(x,t)|}, \tag{2.2.16}$$

$$J_f(x,t) = \int_{\mathbb{S}^{n-1}\times\mathbb{R}^n} K(|x-y|)\omega f dy d\omega. \tag{2.2.17}$$

Here Δ_ω denotes the Laplace-Belltrami operator on the sphere:

$$\Delta_\omega f = \frac{1}{\sin\theta}\partial_\theta(\sin\theta\partial_\phi f) + \frac{1}{\sin^2\theta}\partial_{\phi\phi}f.$$

Equation (2.2.14) is a Fokker-Planck type equation. With the noise added, the rigorous convergence of the particle system to the above Fokker-Planck equation is still an open problem. We recall that, the derivation of the kinetic equation for the Vicsek model without repulsion has been done in [41] in a slightly modified context.

This system is the kinetic counterpart to the system (2.2.8) to (2.2.10). The second term in (2.2.14) describes particle transport with velocity v_f. The third term describes the alignment interaction. The fourth term describes the relaxation force towards the velocity. And finally the last term describes the diffusion due to the Brownian noise in the orientation space.

2.2.2 Scaling

In order to highlight the role of each terms, we firstly write the system in dimensionless form. Introduce the dimensionless variables:

$$\tilde{f} = \frac{f}{f_0}, \quad \tilde{x} = \frac{x}{x_0}, \quad \tilde{t} = \frac{t}{t_0}, \quad \tilde{\phi} = \frac{\phi}{\phi_0}.$$

Choose

$$x_0 = v_0 t_0, \quad f_0 = \frac{1}{x_0^n}, \quad \phi_0 = \frac{v_0 x_0}{\mu},$$

together with the dimensionless parameters

$$\tilde{\nu} = t_0 \nu \quad \tilde{\alpha} = x_0 \alpha, \quad \tilde{D} = t_0 D.$$

Then in the new set of variables $(\tilde{x}, \tilde{t})$, dropping all the tildes for simplicity, (2.2.15) becomes

$$v_f(x, \omega, t) = \omega - \int_{\mathbb{S}^{n-1} \times \mathbb{R}^n} \nabla_x \phi(x-y) f dy d\omega.$$

while $f, \bar{\omega}_f, J_f$ are still given by (2.2.14,2.2.16,2.2.17). Note that the radius of the interacting kernels ϕ and K is also scaled by x_0.

We now define the regime we are interested in. We assume that the ranges R and r of the interaction kernels K and ϕ are both small but with R much larger than r. Also we assume that the diffusion coefficient D and the relaxation rate to the mean orientation ν are large and of the same orders of magnitude, while the relaxation to the velocity α stays of order 1.

More specifically, we assume the existence of a small parameter ϵ and assume the following scaling:

$$D = \frac{1}{\epsilon}\hat{d}, \quad \nu = \frac{1}{\epsilon}, \quad K(|x-y|) = \hat{K}(|\frac{x-y}{\sqrt{\epsilon}}|), \quad \phi(x-y) = \frac{1}{\epsilon^n}\hat{\phi}(\frac{x-y}{\epsilon}).$$

We can see that the D and ν are of order $\frac{1}{\epsilon}$ while the kernels K and ϕ have radius with order $\sqrt{\epsilon}$ and ϵ respectively. Again dropping all the hats for simplicity, in this new set of variables, the system for the distribution function f^ϵ (The superscript ϵ shows that f is dependent on ϵ) can be written as:

$$\epsilon(\partial_t f^\epsilon + \nabla_x \cdot (v^\epsilon f^\epsilon) + \nabla_\omega \cdot (\alpha P_{\omega^\perp} v^\epsilon f^\epsilon)) = -\nabla_\omega \cdot (P_{\omega^\perp}\bar{\omega}^\epsilon f^\epsilon) + d\Delta_\omega f^\epsilon, \tag{2.2.18}$$

$$v^\epsilon = \omega - \nabla_x \int_{\mathbb{S}^{n-1}\times\mathbb{R}^n} \frac{1}{\epsilon^n}\phi(\frac{x-y}{\epsilon}) f^\epsilon dy d\omega, \tag{2.2.19}$$

$$\bar{\omega}^\epsilon = \frac{J^\epsilon(x,t)}{|J^\epsilon(x,t)|}, \tag{2.2.20}$$

$$J^\epsilon = \int_{\mathbb{S}^{n-1}\times\mathbb{R}^n} K(|\frac{x-y}{\sqrt{\epsilon}}|)\omega f^\epsilon dy d\omega. \tag{2.2.21}$$

Now making use of Taylor expansion, we can get rid of the kernels ϕ and K in our system, and expand v^ϵ, J^ϵ in orders of ϵ.

We firstly rewrite the velocity v, using $y = x + \epsilon z$,

$$\begin{aligned} v^\epsilon(x,\omega,t) &= \omega - \nabla_x \int_{\mathbb{S}^{n-1}\times\mathbb{R}^n} \frac{1}{\epsilon^n}\phi(\frac{x-y}{\epsilon}) f^\epsilon(y,\omega,t) dy d\omega \\ &= \omega - \nabla_x \int_{\mathbb{S}^{n-1}\times\mathbb{R}^n} \phi(z)[f^\epsilon(x,\omega,t) + \epsilon z \cdot \nabla_x f^\epsilon(x,\omega,t) + O(\epsilon^2)] dz d\omega \\ &= \omega - \int_{\mathbb{R}^n} \phi(z) dz \cdot \nabla_x \int_{\mathbb{S}^{n-1}} f^\epsilon(x,\omega,t) d\omega + O(\epsilon^2) \\ &= \omega - \Phi \cdot \nabla_x \rho^\epsilon(x,t) + O(\epsilon^2). \end{aligned} \tag{2.2.22}$$

with $\Phi = \int_{\mathbb{R}^n} \phi(z) dz, \quad \rho^\epsilon(x,t) = \int_{\mathbb{S}^{n-1}} f^\epsilon(x,\omega,t) d\omega.$

Similarly, using $y = x + \sqrt{\epsilon}z$, we see that

$$\begin{aligned} J^\epsilon &= \int_{\mathbb{S}^{n-1}\times\mathbb{R}^n} K(\frac{|x-y|}{\sqrt{\epsilon}})\omega f^\epsilon(y,\omega,t)dyd\omega \\ &= \int_{\mathbb{S}^{n-1}\times\mathbb{R}^n} K(|z|)\omega f^\epsilon(x+\sqrt{\epsilon}z,\omega,t)\sqrt{\epsilon}^n dzd\omega \\ &= \sqrt{\epsilon}^n \int_{\mathbb{S}^{n-1}\times\mathbb{R}^n} K(|z|)\omega[f^\epsilon(x,\omega,t) + \sqrt{\epsilon}z\cdot\nabla_x f^\epsilon(x,\omega,t) \\ &\quad + \frac{\epsilon}{2}z^T H_f^\epsilon(x,\omega,t)z + O(\epsilon^2)]dzd\omega \\ &= \sqrt{\epsilon}^n(A+\epsilon B + O(\epsilon^2)). \end{aligned} \tag{2.2.23}$$

Here

$$A = \int_{\mathbb{R}^n} K(|z|)dz \int_{\mathbb{S}^{n-1}} \omega f^\epsilon(x,\omega,t)d\omega = k_0 j(x,t), \tag{2.2.24}$$

$$k_0 = \int_{\mathbb{R}^n} K(|z|)dz, \quad j(x,t) = \int_{\mathbb{S}^{n-1}} \omega f^\epsilon(x,\omega,t)d\omega. \tag{2.2.25}$$

$$\begin{aligned} B &= \frac{1}{2}\int_{\mathbb{S}^{n-1}\times\mathbb{R}^n} K(|z|)\omega z^T H_f^\epsilon(x,\omega,t)zdzd\omega \\ &= k_1 \int_{\mathbb{S}^{n-1}} \omega\Delta f^\epsilon(x,\omega,t)d\omega. \end{aligned} \tag{2.2.26}$$

$$k_1 = \frac{1}{2n}\int_{\mathbb{R}^n} K(|z|)|z|^2 dz. \tag{2.2.27}$$

Finally using the identity

$$\frac{A+\epsilon B}{|A+\epsilon B|} = \Omega + \epsilon P_{\Omega^\perp}\frac{B}{|A|} + O(\epsilon^2), \quad \Omega = \frac{A}{|A|}. \tag{2.2.28}$$

We can get

$$\bar{\omega}^\epsilon = \frac{j(x,t)}{|j(x,t)|} + \epsilon P_{\Omega^\perp}\frac{k_1\int\omega\Delta f^\epsilon d\omega}{|k_0 j(x,t)|}. \tag{2.2.29}$$

We collect all the results and define the local density ρ, local current density j and local average orientation Ω, neglecting the superscript ϵ:

$$\rho(x,t) = \int_{\mathbb{S}^{n-1}} f(x,\omega,t)d\omega, \tag{2.2.30}$$

$$j(x,t) = \int_{\mathbb{S}^{n-1}} f(x,\omega,t)\omega d\omega, \tag{2.2.31}$$

$$\Omega(x,t) = \frac{j(x,t)}{|j(x,t)|}. \tag{2.2.32}$$

Together with the following terms

$$G(x,t) = P_{\Omega^\perp}\frac{k_1\Delta j(x,t)}{|k_0 j(x,t)|}, \tag{2.2.33}$$

$$Q(f) = -\nabla_\omega \cdot (P_{\omega^\perp}\Omega f) + d\Delta_\omega f^\epsilon. \tag{2.2.34}$$

Here $Q(f)$ is called the collision operator. Dropping all the $O(\epsilon^2)$ terms, the system can be written as below:

$$\epsilon(\partial_t f + \nabla_x \cdot (vf) + \nabla_\omega \cdot (\alpha P_{\omega^\perp} vf) + \nabla_\omega \cdot (P_{\omega^\perp} G(x,t) f)) = Q(f), \tag{2.2.35}$$

$$v = \omega - \Phi \cdot \nabla_x \rho(x,t). \tag{2.2.36}$$

2.2.3 Hydrodynamic limit

The purpose of this section is to consider the hydrodynamic limit for the system (2.2.35,2.2.36) as ϵ goes to 0, where the local density, the local current and local average orientation are defined by (2.2.30) to (2.2.32). Firstly we introduce the von Mises-Fisher (VMF) probability distribution $M_\Omega(\omega)$ of $\omega \in \mathbb{S}^{n-1}$ where Ω is a parameter.

$$M_\Omega(\omega) = C\exp(\frac{\omega \cdot \Omega}{d}). \tag{2.2.37}$$

This is a distribution where the peak is centered at Ω. While d measures the intensity of noise, we can see that if d goes to 0, M_Ω converge to δ_Ω which means a perfect alignment, and if d goes to infinity, M_Ω converges to a uniform distribution representing a totally disordered state. Here C is a normalizing constant to ensure that $\int M_\Omega(\omega)d\omega = 1$, clearly,

$$C = \frac{1}{\int \exp(\frac{\omega\cdot\Omega}{d})d\omega}. \tag{2.2.38}$$

An important parameter would be the flux of the VMF distribution, which is $\int M_\Omega(\omega)\omega d\omega$. By simple symmetry properties, we can see that the flux is aligned with Ω, or

$$\int M_\Omega \omega d\omega = c_1\Omega, \quad c_1(d) = \int M_\Omega(\omega)(\omega \cdot \Omega)d\omega. \tag{2.2.39}$$

Theorem 2.2.1. *Let f^ϵ be the solution of (2.2.73) and assume that*

$$f^\epsilon \to f \text{ as } \epsilon \to 0. \tag{2.2.40}$$

Then there exists $\rho(x,t)$ and $\Omega(x,t)$ such that

$$f(x,\omega,t) = \rho(x,t)M_\Omega(\omega). \tag{2.2.41}$$

Furthermore, the functions $\rho(x,t)$ and $\Omega(x,t)$ satisfy the following equations:

$$\partial_t\rho + \nabla_x \cdot (\rho U) = 0, \tag{2.2.42}$$

$$\rho(\partial_t\Omega + (V \cdot \nabla_x)\Omega) + P_{\Omega^\perp}\nabla_x(p(\rho)) = \gamma P_{\Omega^\perp}\Delta(\rho\Omega). \tag{2.2.43}$$

With

$$U = C_1\Omega - \Phi\nabla_x\rho, \tag{2.2.44}$$

$$V = C_2\Omega - \Phi\nabla_x\rho, \tag{2.2.45}$$

$$p(\rho) = d\rho + \alpha\Phi((n-1)d + C_2)\frac{\rho^2}{2}, \tag{2.2.46}$$

$$\gamma = \frac{k_1}{k_0}((n-1)d + C_2). \tag{2.2.47}$$

The coefficient $C_1(d)$ and $C_2(d)$ are defined in the proof.

To prove the theorem, we will take three steps. The first step is to determine the equilibrium state. The second step is to compute the generalized collision invariant. The final step is to get the hydrodynamic limit.

step1: determination of the equilibrium state:

The aim is to cancel the right hand side of (2.2.35) and thus the equilibrium state is the elements in the null space of Q. Note that Q is an operator on f defined in (2.2.34). To do this, we will study the property of Q first.

With M_Ω defined, we have the following lemma:

Lemma 2.2.1. *The operator Q can be written as*

$$Q(f) = d\nabla_\omega \cdot [M_\Omega\nabla_\omega(\frac{f}{M_\Omega})]. \tag{2.2.48}$$

and we have

$$H(f) = \int Q(f)\frac{f}{M_\Omega}d\omega = -d\int M_\Omega|\nabla_\omega(\frac{f}{M_\Omega})|^2 d\omega \leq 0. \tag{2.2.49}$$

Proof. In 3D, we choose a reference frame $\Omega = e_3$, thus

$$\omega = \begin{pmatrix} \sin\theta\cos\phi \\ \sin\theta\sin\phi \\ \cos\theta \end{pmatrix}, \quad e_\theta = \begin{pmatrix} \cos\theta\cos\phi \\ \cos\theta\sin\phi \\ -\sin\theta \end{pmatrix}, \quad e_\phi = \begin{pmatrix} -\sin\phi \\ \cos\phi \\ 0 \end{pmatrix}, \tag{2.2.50}$$

and we have the following identities for scalar f and vector $\vec{A} = A_\theta e_\theta + A_\phi e_\phi$:

$$\nabla_\omega f = f_\theta e_\theta + \frac{1}{\sin\theta}f_\phi e_\phi, \quad \nabla_\omega \cdot \vec{A} = \frac{1}{\sin\theta}\partial_\theta(A_\theta\sin\theta) + \frac{1}{\sin\theta}\partial_\phi A_\phi. \tag{2.2.51}$$

Therefore $M_\Omega(\omega) = Cexp(\frac{\cos\theta}{d})$ and

$$\begin{aligned} \nabla_\omega(lnM_\Omega) &= \nabla_\omega[lnCexp(\frac{\cos\theta}{d})] \\ &= \frac{1}{d}\nabla_\omega(\cos\theta) \\ &= -\frac{1}{d}\sin\theta e_\theta \\ &= \frac{1}{d}P_{\omega^\perp}\Omega, \end{aligned} \tag{2.2.52}$$

and we deduce that

$$\begin{aligned} d\nabla_\omega \cdot [M_\Omega\nabla_\omega(\frac{f}{M_\Omega})] &= d\nabla_\omega \cdot [\nabla_\omega f - f\nabla_\omega(lnM_\Omega)] \\ &= d\Delta_\omega f - d\nabla_\omega \cdot (f \cdot \frac{1}{d}P_{\omega^\perp}\Omega) \\ &= Q(f). \end{aligned} \tag{2.2.53}$$

The case in 2D is similar. And the second part of the lemma follows directly from the first part as well as Stoke's theorem. □

Now we can define the set of equilibria of Q to be

$$\mathcal{E} = \{f \in H^1(\mathbb{S}^{n-1}) | f \geq 0 \quad \text{and} \quad Q(f) = 0\}. \tag{2.2.54}$$

With the previous lemma, we can see that if $Q(f) = 0$, then $H(f) = 0$. Note that $H(f)$ is the integral of a non-negative quantity and the only possibility is that

$\nabla_\omega(\frac{f}{M_\Omega})$ is identically 0, which means that $f = \rho M_\Omega$ for some conveniently constant ρ.

Finally, we can state the lemma:

Lemma 2.2.2. *The set of equilibria of Q such that $Q(f) = 0$ forms a three-dimensional manifold $\mathcal{E}$ and is given by*

$$\mathcal{E} = \{\rho M_\Omega(\omega) | \rho \in \mathbb{R}+, \Omega \in \mathbb{S}^2\}. \tag{2.2.55}$$

Using the fact that $\int M_\Omega d\omega$ is normalized, we see that ρ is the total local density. Also Ω is the direction of the flux of $\rho M_\Omega(\omega)$.

$$\rho = \int \rho M_\Omega(\omega) d\omega, \tag{2.2.56}$$

$$\Omega = \frac{j[\rho M_\Omega]}{|j[\rho M_\Omega)]}, \quad j(\rho M_\Omega) = \int \rho M_\Omega(\omega)\omega d\omega. \tag{2.2.57}$$

The lemma can be interpreted as follows: For the equilibrium solution of f, the dependence on position x and orientation ω can be decoupled. The distribution f is the product of the local density ρ and the Von-Mises Fisher distribution in the orientation space.

Step2: Generalized Collision Invariants (GCI)

We will begin this section by defining the term Collision Invariant(CI).

Definition 2.2.3. *A collosion invariant (CI) is a function $\psi(\omega)$ such that for all the function f with sufficient regularities. We have*

$$\int Q(f)\psi d\omega = 0. \tag{2.2.58}$$

With (2.2.48), this means that

$$\int \nabla_\omega \cdot [M_\Omega \nabla_\omega (\frac{f}{M_\Omega})]\psi d\omega = 0, \tag{2.2.59}$$

so we have $\forall f$,

$$\begin{aligned}\int \nabla_\omega \cdot [M_\Omega \nabla_\omega (\frac{f}{M_\Omega})]\psi d\omega &= -\int M_\Omega \nabla_\omega (\frac{f}{M_\Omega})\nabla_\omega \psi \\ &= \int \frac{f}{M_\Omega}\nabla_\omega \cdot (M_\Omega \nabla_\omega \psi)d\omega \\ &= 0.\end{aligned} \tag{2.2.60}$$

We are looking for function ψ such that (2.2.60) holds. Clearly the set of constants are collision invariant. Physically, this corresponds to the conservation of mass during particle interactions. Note that for our system, the momentum is not conserved, and we cannot hope for more physical conservations. The set of constants is not large enough for us to derive the evolution of ρ and Ω. To overcome this problem, we introduce collision invariant in a weaker sense, which is the so called generalized collision invariant. The concept is firstly introduce in [41]

Now fix a particular Ω and $\Omega[f]$ be the average local orientation for a given distribution f. The generalized collision invariant(GCI) is defined to be function ψ such that

$$\int \frac{f}{M_\Omega}\nabla_\omega \cdot (M_\Omega \nabla_\omega \psi)d\omega = 0, \quad \forall f \text{ such that } \Omega[f] = \Omega. \tag{2.2.61}$$

Remember that Ω is the normalization of $j[f]$, therefore saying $\Omega[f] = \Omega$ is the same as saying that $j[f]$ is aligned with Ω. Or we can stated that

$$0 = \Omega \times j[f] = \int f(\Omega \times \omega)d\omega = \int \frac{f}{M_\Omega}(\sin\theta M_\Omega)d\omega. \tag{2.2.62}$$

Viewing this as a linear constraint and introducing the Lagrange multiplier β of this constraint. Let β be a vector which is perpendicular to Ω, the problem of finding the generalized collision invariant can be restated as below: Given $\Omega \in \mathbb{S}^2$, find all ψ such that there exist $\beta \perp \Omega$, and

$$\int \frac{f}{M_\Omega}\{\nabla_\omega \cdot (M_\Omega \nabla_\omega \psi) - \beta \cdot (\Omega \times \omega)M_\Omega\}d\omega = 0. \tag{2.2.63}$$

Now (2.2.63) holds for all f without any constraint, and therefore we can get the following equation for ψ:

$$\nabla_\omega \cdot (M_\Omega \nabla_\omega \psi) = \beta \cdot (\Omega \times \omega)M_\Omega. \tag{2.2.64}$$

Now we again use the reference frame where $\Omega = e_3$

$$\omega = \begin{pmatrix} \sin\theta\cos\phi \\ \sin\theta\sin\phi \\ \cos\theta \end{pmatrix}, \quad \beta = \begin{pmatrix} \beta_1 \\ \beta_2 \\ 0 \end{pmatrix}, \quad \Omega \times \omega = \begin{pmatrix} -\sin\phi\sin\theta \\ \cos\phi\sin\theta \\ 0 \end{pmatrix}. \tag{2.2.65}$$

The problem of finding ψ is obviously linear and we can see that ψ_1 and ψ_2 can be solved separately, with the right hand side of (2.2.64) being $-\sin\phi\sin\theta M_\Omega$ and $\cos\phi\sin\theta M_\Omega$ respectively.

With the right hand side fixed, we firstly prove the uniqueness of the generalized collision invariant. And we have the following lemma:

Lemma 2.2.4. *Let $\chi \in L^2(\mathbb{S}^2)$ such that $\int \chi d\omega = 0$, then*

$$\nabla_\omega \cdot (M_\Omega \nabla_\omega \psi) = \chi, \tag{2.2.66}$$

has a unique weak solution in the quotient of the space $H^1(\mathbb{S}^2)$ by the space spanned by the constant functions, endowed with the quotient norm.

The proof of the lemma is based on applying the Lax-Milgram theorem to the variational form of (2.2.66) which is

$$\int M_\Omega \nabla_\omega \psi \cdot \nabla_\omega \phi d\omega = \int \chi \phi d\omega, \tag{2.2.67}$$

for a test function ϕ.

Therefore for $\chi = -\sin\phi\sin\theta M_\Omega$ and $\chi = \cos\phi\sin\theta M_\Omega$, we are able to solve for ψ_1 and ψ_2 uniquely up to factor of a constant. If we impose the condition $\int \psi_k d\omega = 0$ for $k = 1, 2$, we can single out the unique solutions.

If we can find ψ_1 and ψ_2, together with the set of constants, we can state that the set of collision invariant forms a three dimensional space which is spanned by $\{1, \psi_1, \psi_2\}$.

Actually we can find the form of ψ_1 and ψ_2 explicitly. $\psi_1 = -g(\cos\theta)\sin\phi$ and $\psi_2 = g(\cos\theta)\cos\phi$ where the dependence on θ and ϕ is separated. Making use of the differential operators in spherical coordinate, we see that the function $g(\mu)$ satisfies the differential equation below:

$$-(1-\mu^2)\partial_\mu[(1-\mu^2)g_\mu M_\Omega] + gM_\Omega = -(1-\mu^2)^{\frac{3}{2}}M_\Omega. \tag{2.2.68}$$

Since the operator on the left hand side degenerates at the boundary $\mu = \pm 1$, there is no need to specify a boundary condition for the differential equation.

Again using the Lax-Milgram theorem, to prove that problem (2.2.68) has a unique solution in the space V defined by

$$V = \{g|(1-\mu^2)^{-1/2}g \in L^2(1,1), (1-\mu^2)^{1/2}\partial_\mu g \in L^2(1,1)\}. \tag{2.2.69}$$

Furthermore, the Maximum Principle shows that g is non-positive.

For the convenience of the next step, we introduce $h(\mu) = (1-\mu^2)^{-1/2}$. In terms of θ, it is $h(\cos\theta) = g(\cos\theta)/\sin(\theta)$. Using the spherical reference frame, we can compute $(\Omega \times \omega)h(\Omega\cdot\omega) = \psi_1 e_1 + \psi_2 e_2$.

Here we call $\vec{\psi}$ the vector form of the generalized collision invariant associated with orientation Ω.

The above computation of the collision invariant assumes the dimension $n = 3$. For $2D$ case, the idea is similar. Using a reference frame $\Omega = (1,0)$ and $\omega = (\cos\theta, \sin\theta)$, from the relation (2.2.64), now we get only one generalized collision invariant ψ with the corresponding $\chi = \sin\theta M_\Omega$, i.e.

$$\nabla_\omega \cdot (M_\Omega \nabla_\omega \psi) = \sin\theta M_\Omega. \tag{2.2.70}$$

This ψ can be explicitly written as $\psi = g(\cos\theta)$ and $g(\mu)$ satisfies

$$\partial_\mu((1-\mu^2)^{1/2}g_\mu M_\Omega) = M_\Omega. \tag{2.2.71}$$

In this case, we can introduce $h(\cos\theta) = g(\cos\theta)/\sin(\theta)$ and reach our conclusion $(\Omega\times\omega)h(\Omega\cdot\omega) = \psi$.

In summary, we found the generalized collision invariant in the form of

$\psi =$ constants or $\psi = (\Omega\times\omega)h(\Omega\cdot\omega)$ which guarantees that

$$\int Q(f)\psi d\omega = 0, \quad \forall f \text{ such that } \Omega[f] = \Omega. \tag{2.2.72}$$

making use of this, we can move to the next step of getting the hydrodynamic limit.

step3: Hydrodynamic limit

Finally we go back to our system (2.2.73,2.2.74),

$$\epsilon(\partial_t f + \nabla_x \cdot (vf) + \nabla_\omega \cdot (\alpha P_{\omega^\perp} vf) + \nabla_\omega \cdot (P_{\omega^\perp} G(x,t) f)) = Q(f), \tag{2.2.73}$$

$$v = \omega - \Phi \cdot \nabla_x \rho(x,t), \tag{2.2.74}$$

$$G = P_{\Omega^\perp} \frac{k_1 \Delta_x j(x,t)}{k_0 j(x,t)}. \tag{2.2.75}$$

and we consider the limit when ϵ goes to 0. Clearly this implies that $Q(f)$ equals to 0. Thanks to our lemma (2.2.1), this implies that $f(x,\omega,t)$ has the form of $\rho(x,t)M_\Omega(\omega)$. What remains is to find the evolution equation for ρ as well as ω. Note that now the operator Q acts on ω alone, which does not specify the dependence of f on (x,t), consequently, ρ and Ω are functions of x,t.

To find the dependence, we can make use of the generalized collision invariant and integrate (2.2.73) together with ψ. This helps us to cancel the right hand side by the definition of the generalized collision invariant. What we can get is, for a collision invariant ψ

$$\int [\partial_t f + \nabla_x \cdot (vf) + \nabla_\omega \cdot (\alpha P_{\omega^\perp} vf) + \nabla_\omega \cdot (P_{\omega^\perp} G(x,t) f)] \psi d\omega = 0. \tag{2.2.76}$$

For our numerical comparisons, we will deal with the 2D case. Since the collision invariants is spanned by $\{1, g(\cos\theta)\}$, we firstly consider the constant collision invariant. By integrating with 1 with respect to ω, the right hand side is 0 while the third and fourth terms in the left hand side also becomes 0 since they are the divergence form. We get

$$\int (\partial_t f + \nabla_x \cdot (vf)) d\omega = 0. \tag{2.2.77}$$

In another word, with (2.2.55) this is

$$\partial_t \rho + \nabla_x \cdot (v\rho M_\Omega d\omega) = 0, \tag{2.2.78}$$

$$v = \omega - \Phi \nabla_x \rho. \tag{2.2.79}$$

substituting the expression of v, we get the first equation governing the evolution of ρ, which is

$$\partial_t\rho + c_1\nabla_x \cdot (\rho\Omega) - \Phi\Delta(\frac{\rho^2}{2}) = 0, \tag{2.2.80}$$

where c_1 is defined in (2.2.117). Now we consider collision invariants other than the constant. Note that $\psi = (\Omega \times \omega)h(\Omega \cdot \omega)$ thus (2.2.76) can be written as

$$\Omega \times \int [\partial_t f + \nabla_x \cdot (vf) + \nabla_\omega \cdot (\alpha P_{\omega^\perp} vf) + \nabla_\omega \cdot (P_{\omega^\perp} G(x,t)f)]\omega h d\omega = 0. \tag{2.2.81}$$

We can write it in the form $\Omega \times X$=0, with

$$\begin{aligned} X &:= \int [\partial_t f + \nabla_x \cdot (vf) + \nabla_\omega \cdot (\alpha P_{\omega^\perp} vf) + \nabla_\omega \cdot (P_{\omega^\perp} G(x,t)f)]\omega h d\omega \\ &= X_1 + X_2 + X_3 + X_4. \end{aligned} \tag{2.2.82}$$

consisting of four terms. This means that $P_{\Omega^\perp}X = 0$ and we compute each term $P_{\Omega^\perp}X_k(k = 1,2,3,4)$ successively and let the sum be 0. Then we can get the evolution equation we want.

Elementary differential geometry gives the derivative of M_Ω with respect to Ω acting on a tangent vector $d\Omega$ as follows:

$$\frac{\partial M}{\partial \Omega}(d\Omega) = \frac{1}{d}(\omega \cdot d\Omega)M_\Omega. \tag{2.2.83}$$

And we can deduce the following identities:

$$\partial_t(\rho M_\Omega) = M_\Omega(\partial_t\rho + \frac{1}{d}\rho(\omega \cdot \partial_t\Omega)), \tag{2.2.84}$$

$$(\omega \cdot \nabla_x)(\rho M_\Omega) = M_\Omega((\omega \cdot \nabla_x)\rho + \frac{1}{d}\rho\omega \cdot ((\omega \cdot \nabla_x)\Omega)). \tag{2.2.85}$$

In the following computations, we will use a moving reference polar frame $\Omega = (1,0)$ and $\omega = (\cos\theta, \sin\theta)$ in 2D case. The computation in 3D is similar with a spherical reference frame e_3. An easy computation shows that

$$\int \omega \otimes \omega d\omega = \int_0^{2\pi} \cos^2\theta d\theta P_\Omega + \int_0^{2\pi} \sin^2\theta d\theta P_{\Omega^\perp}. \tag{2.2.86}$$

Firstly we consider X_1,

$$\begin{aligned} X_1 &= \int \partial_t(\rho M_\Omega)\omega h d\omega \\ &= \int \rho_t M_\Omega \omega h d\omega + \int \frac{1}{d}\rho M_\Omega (\omega \cdot \partial_t \Omega)\omega h d\omega \\ &= X_{11} + X_{12}. \end{aligned} \tag{2.2.87}$$

Here due to symmetry X_{11} is along Ω, clearly $P_{\Omega^\perp} X_{11} = 0$, and

$$X_{12} = \frac{\rho}{d}\int (\omega \otimes \omega)\partial_t \Omega M_\Omega h d\omega. \tag{2.2.88}$$

Therefore

$$P_{\Omega^\perp} X_1 = P_{\Omega^\perp} X_{12} = \frac{\rho}{d}\int\limits_0^{2\pi} \sin^2\theta h M_\Omega d\theta \partial_t \Omega = c\rho \partial_t \Omega, \tag{2.2.89}$$

where

$$c = \frac{1}{d}\int\limits_0^{2\pi} \sin^2\theta h M_\Omega d\theta. \tag{2.2.90}$$

Next we consider X_2

$$\begin{aligned} X_2 &= \int \nabla_x \cdot (v\rho M_\Omega)\omega h d\omega \\ &= \int \nabla_x \cdot (\omega \rho M_\Omega)\omega h d\omega - \int \nabla_x \cdot (\Phi \nabla_x \rho \cdot \rho M_\Omega)\omega h d\omega \\ &= X_{21} + X_{22} + X_{23} + X_{24}. \end{aligned} \tag{2.2.91}$$

with

$$X_{21} := \int (\omega \cdot \nabla_x \rho) M_\Omega \omega h d\omega, \tag{2.2.92}$$

$$X_{22} := \int \frac{\rho}{d}\omega \cdot (\omega \cdot \nabla_x)\Omega M_\Omega \omega h d\omega, \tag{2.2.93}$$

$$X_{23} := -\Phi \int \Delta(\frac{\rho^2}{2}) M_\Omega \omega h d\omega, \tag{2.2.94}$$

$$X_{24} := -\Phi \int \frac{1}{d} M_\Omega \omega \cdot (\nabla_x(\frac{\rho^2}{2} \cdot \nabla_x))\Omega \omega h d\omega. \tag{2.2.95}$$

We can compute

$$\begin{aligned} P_{\Omega^\perp} X_{21} &= \int ((\omega \otimes \omega)\nabla_x \rho) h M_\Omega d\omega \\ &= \int_0^{2\pi} \sin^2\theta h M_\Omega d\theta P_{\Omega^\perp}(\nabla_x \rho), \end{aligned} \tag{2.2.96}$$

$$P_{\Omega^\perp} X_{22} = \frac{\rho}{d}\int_0^{2\pi} \sin^2\theta \cos\theta h M_\Omega d\theta (\Omega \cdot \nabla_x)\Omega, \tag{2.2.97}$$

$$P_{\Omega^\perp} X_{23} = 0, \tag{2.2.98}$$

$$P_{\Omega^\perp} X_{24} = -\frac{\Phi}{d}\int_0^{2\pi} \sin^2\theta h M_\Omega d\theta (\nabla_x(\frac{\rho^2}{2}) \cdot \nabla_x)\Omega. \tag{2.2.99}$$

Combining the above four terms, we get

$$P_{\Omega^\perp} X_2 = c_3 P_{\Omega^\perp}(\nabla_x \rho) + c_4 \rho (\Omega \cdot \nabla_x)\Omega + c_5 (\nabla_x(\frac{\rho^2}{2}) \cdot \nabla_x)\Omega, \tag{2.2.100}$$

with

$$c_3 = \int_0^{2\pi} \sin^2\theta h M_\Omega d\theta, \tag{2.2.101}$$

$$c_4 = \frac{1}{d}\int_0^{2\pi} \sin^2\theta \cos\theta h M_\Omega d\theta, \tag{2.2.102}$$

$$c_5 = -\frac{\Phi}{d}\int_0^{2\pi} \sin^2\theta h M_\Omega d\theta. \tag{2.2.103}$$

The next term is X_3 which is

$$\begin{aligned} X_3 &= \int \nabla_\omega \cdot (\alpha P_{\omega^\perp} v \rho M_\Omega) \omega h d\omega \\ &= \int \nabla_\omega \cdot (\alpha P_{\omega^\perp}(\omega - \Phi \nabla_x \rho)\rho M_\Omega)\omega d\omega \\ &= -\alpha\Phi \int -(\omega \otimes \omega)(\nabla_x(\frac{\rho^2}{2})) M_\Omega h d\omega \\ &= -\alpha\Phi \int \nabla_\omega \cdot (P_{\omega^\perp}(\nabla_x(\frac{\rho^2}{2})) M_\Omega)\omega h d\omega - \alpha\Phi \int (\omega \otimes \vec{b})(\nabla_x(\frac{\rho^2}{2}))\frac{1}{d} M_\Omega h d\omega \\ &= X_{31} + X_{32}. \end{aligned}$$

$$\tag{2.2.104}$$

Here $\vec{b} = (\sin^2\theta, -\sin\theta\cos\theta)$. We find that

$$P_{\Omega^\perp}X_{31} = \alpha\Phi\int_0^{2\pi}\sin^2\theta M_\Omega h d\theta P_{\Omega^\perp}(\nabla_x(\frac{\rho^2}{2})), \tag{2.2.105}$$

$$P_{\Omega^\perp}X_{32} = \frac{\alpha\Phi}{d}\int_0^{2\pi}\sin^2\theta\cos\theta M_\Omega h d\theta P_{\Omega^\perp}(\nabla_x(\frac{\rho^2}{2})). \tag{2.2.106}$$

Combining these two terms, we have

$$P_{\Omega^\perp}X_3 = c_6 P_{\Omega^\perp}(\nabla_x\frac{\rho^2}{2}), \tag{2.2.107}$$

with

$$c_6 = \alpha\Phi\int_0^{2\pi}(\sin^2\theta + \frac{1}{d}\sin^2\theta\cos\theta)hM_\Omega d\theta. \tag{2.2.108}$$

Finally the last term is X_4 and

$$\begin{aligned} X_4 &= \int \nabla_\omega\cdot(P_{\omega^\perp}G\rho M_\Omega)\omega h d\omega \\ &= \int -(\omega\otimes\omega)G\rho M_\Omega h d\omega + \int(\omega\otimes\vec{b})GM_\Omega\frac{1}{d}h d\omega \\ &= X_{41} + X_{42}. \end{aligned} \tag{2.2.109}$$

We find that

$$P_{\Omega^\perp}X_{41} = -\int_0^{2\pi}\sin^2\theta M_\Omega h P_{\Omega^\perp}(\rho G), \tag{2.2.110}$$

$$P_{\Omega^\perp}X_{42} = -\int_0^{2\pi}\sin^2\theta\cos\theta M_\Omega h(\frac{1}{d})P_{\Omega^\perp}(\rho G). \tag{2.2.111}$$

We can further compute

$$P_{\Omega^\perp}X_4 = c_7 P_{\Omega^\perp}\Delta(\rho\Omega). \tag{2.2.112}$$

where

$$c_7 = -\frac{k_1}{k_0}\int_0^{2\pi}(\sin^2\theta + \sin^2\theta\cos\theta(\frac{1}{d})hM_\Omega)d\theta. \tag{2.2.113}$$

We can do the same computation in 3D and define

$$\langle g \rangle_{M_\Omega} := \int_{\mathbb{S}^{n-1}} M_\Omega(\omega) g d\omega. \tag{2.2.114}$$

where n is dimension.

Collecting all the terms we have, we get the complete system of equations

$$\partial_t \rho + c_1 \nabla_x \cdot (\rho\Omega) + c_2 \Delta(\frac{\rho^2}{2}) = 0, \tag{2.2.115}$$

$$\begin{aligned} &c\rho\partial_t\Omega + c_3 P_{\Omega^\perp}(\nabla_x \rho) + c_4 \rho(\Omega \cdot \nabla_x)\Omega + c_5(\nabla_x(\frac{\rho^2}{2}) \cdot \nabla_x)\Omega \\ &+ c_6 P_{\Omega^\perp}(\nabla_x(\frac{\rho^2}{2})) + c_7 P_{\Omega^\perp}(\Delta(\rho\Omega)) = 0. \end{aligned} \tag{2.2.116}$$

With the coefficients

$$c_1 = \int_{\mathbb{S}^{n-1}} M_\Omega(\omega \cdot \Omega) d\omega, \tag{2.2.117}$$

$$c_2 = -\Phi, \tag{2.2.118}$$

$$c = \frac{1}{d(n-1)} \langle \sin^2 \theta h \rangle_{M_\Omega}, \tag{2.2.119}$$

$$c_3 = \frac{1}{n-1} \langle \sin^2 \theta h \rangle_{M_\Omega}, \tag{2.2.120}$$

$$c_4 = \frac{1}{d(n-1)} \langle \sin^2 \theta \cos\theta h \rangle_{M_\Omega}, \tag{2.2.121}$$

$$c_5 = -\frac{\Phi}{d(n-1)} \langle \sin^2 \theta h \rangle_{M_\Omega}, \tag{2.2.122}$$

$$c_6 = \alpha\Phi(\langle \sin^2 \theta h \rangle_{M_\Omega} + \frac{1}{d(n-1)} \langle \sin^2 \theta \cos\theta h \rangle_{M_\Omega}), \tag{2.2.123}$$

$$c_7 = -\frac{k_1}{k_0}(\langle \sin^2 \theta h \rangle_{M_\Omega} + \frac{1}{d(n-1)} \langle \sin^2 \theta \cos\theta h \rangle_{M_\Omega}). \tag{2.2.124}$$

This can be rewritten in the form

$$\partial_t \rho + \nabla_x \cdot (\rho U) = 0, \tag{2.2.125}$$

$$\rho(\partial_t \Omega + (V \cdot \nabla_x)\Omega) + P_{\Omega^\perp} \nabla_x (p(\rho)) = \gamma P_{\Omega^\perp} \Delta(\rho\Omega). \tag{2.2.126}$$

With

$$U = C_1\Omega - \Phi\nabla_x\rho, \tag{2.2.127}$$

$$V = C_2\Omega - \Phi\nabla_x\rho, \tag{2.2.128}$$

$$p(\rho) = \frac{c_3}{c}\rho + \frac{c_6}{c}\frac{\rho^2}{2} = d\rho + \alpha\Phi((n-1)d + C_2)\frac{\rho^2}{2}, \tag{2.2.129}$$

$$\gamma = -\frac{c_7}{c} = \frac{k_1}{k_0}((n-1)d + C_2). \tag{2.2.130}$$

where $C_1 = c_1$ and $C_2 = \frac{c_4}{c}$.

This ends our proof of the Theorem 2.2.1.

2.3 Particle model generated from Navier-Stokes system

2.3.1 Macroscopic model

In the previous section we added repulsion to the Vicsek model and successfully derived a fluid system. The momentum is not conserved since the alignment is achieved by measuring the average of the neighbours. In contrast to the Navier-Stokes equation, the lack of conservation of momentum is a consequence of the absence of Galilean invariance.

To find better individual based models, we consider the global fluid structure and try to derive a microscopic model from it. There has been a history of numerical studies on fluid where the fluid is replaced by a set of particles. Mathematically, these particles are just interpolation points where the properties of the fluid can be calculated. Physically, the particles can be treated as material particles just like any particle system. Clearly this is an analogue of our microscopic model. We will start with a macroscopic model and try to apply particle discretization on it.

The collective motion can be understood by constructing the Navier-Stokes equation with a proper self propelling force. The Vicsek family models display true-long-range order even in two dimensions. How this happens was shown by Toner and

Tu [119], [120], using continuum field-theoretic approach.

For an incompressible assumption, the flow field $v(x,t)$ is divergence free, i.e. $\nabla \cdot v = 0$, it is postulated that the dynamics of v is governed by a generalized d-dimensional Navier-Stokes equation of the form

$$(\partial_t + v \cdot \nabla)v = -\nabla p - (\alpha + \beta |v|^2)v + \nabla \cdot E. \tag{2.3.1}$$

The second term in (2.3.1) is a typical local driving terms found in Toner-Tu model. For stability reasons, it is required that $\beta > 0$ and for $\alpha > 0$, the fluid will reach an equilibrium state with $v = 0$, for $\alpha < 0$, the fluid will reach a polar ordering state with characteristic speed $v_0 = \sqrt{|\alpha|/\beta}$.

And the strain tensor E should have the following form:

$$E_{ij} = \Gamma_0(\partial_i v_j + \partial_j v_i) - \Gamma_2 \Delta(\partial_i v_j + \partial_j v_i) + S q_{ij}, \tag{2.3.2}$$

with $q_{ij} = v_i v_j - \frac{\delta_{ij}}{d}|v|^2$.

Note that (2.3.2) reduces to the usual rate of strain tensor if $S = 0 = \Gamma_2$, the additional S term in (2.3.2) is from the theories of active nematics. With the strain tensor E and defining $\lambda_0 = 1 - S$, $\lambda_1 = S/d$, (2.3.1) becomes

$$(\partial_t + \lambda_0 v \cdot \nabla)v = -\nabla p + \lambda_1 \nabla v^2 - (\alpha + \beta |v|^2)v + \Gamma_0 \Delta v - \Gamma_2 \Delta^2 v. \tag{2.3.3}$$

In this model, the two Γ terms describe the high order derivatives in the Swift-Hohenberg theory [115] and they cause the formation of quasi chaotic pattern. Since our focus is the collective motion of the micro-organism where they perform motion in a regular pattern within a confined domain, we will drop the two terms for simplicity.

Note also it is impossible to get an individual based model which satisfies the divergence free condition. We will model our suspension as a weakly compressible fluid, and the modified model reads

$$\rho(\frac{\partial v}{\partial t} + v \cdot \nabla v) = -\nabla p + \rho \nu \nabla^2 v + \rho F, \tag{2.3.4}$$

with

$$p = p_0 \rho, \tag{2.3.5}$$

$$F = \lambda(v_\infty^2 - |v|^2)v, \tag{2.3.6}$$

$$v_\infty = max\{v_0(1 - \frac{\rho_c + \rho_0}{\rho + \rho_0}), 0\}. \tag{2.3.7}$$

Here ρ is introduced as the density, p is then pressure which depends on ρ, and F is the modified self propelling force.

Instead of having a characteristic velocity $\sqrt{|\alpha|/\beta}$, it is found experimentally that the characteristic velocity v_∞ is dependent on ρ. It is interesting that no net collective motion can be observed when ρ is sufficiently small, or $\rho < \rho_c$, where ρ_c is the critical density. Other than this, it is found that v_∞ is an increasing function of ρ. We assume that as $\rho \to \infty$, $v_\infty \to v_0$. Thus we get the expression in (2.3.7).

2.3.2 Scaling

We will firstly introduce the following dimensionless variables:

$$\tilde{x} = \frac{x}{x_s}, \quad \tilde{v} = \frac{v}{v_s}, \quad \tilde{v_0} = \frac{v_0}{v_s}, \quad \tilde{t} = \frac{t}{x_s/v_s},$$

$$\tilde{\rho} = \frac{\rho}{\rho_s}, \quad \tilde{\rho_c} = \frac{\rho_c}{\rho_s}, \quad \tilde{\rho_0} = \frac{\rho_0}{\rho_s},$$

$$\tilde{p} = \frac{p}{v_s^2 \rho_s}, \quad \tilde{p_0} = \frac{p_0}{v_s^2}, \quad \tilde{\nu} = \frac{\nu}{x_s v_s}, \quad \tilde{\lambda} = x_s v_s \lambda, \quad \tilde{F} = \frac{x_s F}{v_s^2}.$$

Here $x_s = 1100\mu m$ is the typical size of an experimental domain. $v_s = 237\mu m/s$ is the typical velocity of an undiluted suspension. $\rho_s = 2.96 \times 10^{-3}/\mu m^3$ is the typical particle density. With this new set of variables, we can rewrite our system as

$$\frac{Dv}{Dt} = -\frac{\nabla p}{\rho} + \nu \nabla^2 v + F. \tag{2.3.8}$$

With $\frac{Dv}{Dt} = \frac{\partial v}{\partial t} + v \cdot \nabla v$ being the material derivatives. We will use this Lagrangian descriptions of continuum matter to get our individual based model.

2.3.3 Particle method for fluid and microscopic model

Fluid mechanism has two philosophies of fluid flow, namely Lagrangian and Eulerian. In the Eulerian concept, the observer stays at a fixed position and all the flow quantities are functions of time and space. The Eulerian methods have been well studied for more than fifty years and are widely applied in many flow simulation areas. Standard methods include the finite volume method (FVM), the finite difference method (FDM) and the finite element method (FEM). In the Lagrangian description, the fluid can be viewed as individual particles which can move freely inside the domain, and they carry all the fluid's information. There is a fundamental similarity between the Lagrangian methods with the molecular dynamics [64] which helps us understand the physical process in the atomic level. Note that the individual based model also has a Lagrangian description and the Lagrangian methods for fluid give us exactly an individual model which will produce the global structure that we want. Existing particle methods for fluids include the vortex method [27], the finite point or finite pointset method (FPM) [118], the radial basis function method(RBFM) [70, 71], the finite volume particle method (FVPM) [95] and smoothed particle hydrodynamics(SPH). The advantage of Lanrangian methods is that they dont suffer from mesh distortion problems and can deal with complex domain. Also the mass and other quantities that carried by each particle are automatically conserved.

We will borrow the idea from the SPH method, which was invented to solve astrophysical problem in 1977 by Gingold [51] and Lucy [81]. The formulation of SPH is usually divided into two key steps. The first step is the integral representation or kernel representation which approximates the field variables. The second step is particle approximation which sums up the values for all the nearest neighbours.

Now we show the basic idea of SPH. For a variable A, in SPH formulation, it can be written as a convolution product of the variable A with the Dirac δ function.

$$A(r) = \int A(r')\delta(r - r')dr'. \tag{2.3.9}$$

Now if we replace the delta function $\delta(r-r')$ by some smoothing functions, then we can get the integral representation of a function A:

$$A(r) \approx \int A(r')W(r-r',h)dr' = \langle A(r) \rangle. \tag{2.3.10}$$

Here the kernel approximation operator is marked by the angle braket <>. The kernel function should be chosen with the following three properties: Firstly, W should have the normalization condition

$$\int W(r-r')dr' = 1. \tag{2.3.11}$$

Secondly, it needs to satisfy the Dirac function property when the smoothing length h goes to 0, that is

$$\lim_{h \to 0} W(r-r',h) = \delta(r-r'). \tag{2.3.12}$$

The last condition is the compact condition which requires that

$$W(r-r') = 0 \text{ if } |r-r'| > h. \tag{2.3.13}$$

This means that the kernel has a compact support with radius h such that the integration over the whole domain reduces to the domain of the kernel. In this way, the individual model we obtained will have only local interaction.

Furthermore, we would like to make W an even function, i.e. $W(r-r') = W(r'-r)$. This requirement that W is an even function ensures an h^2 order accuracy for the kernel approximation. The error of the integral representation can be roughly estimated using the Taylor expansion

$$\begin{aligned} < A(r) > &= \int [A(r) + A'(r)(r-r') + O(r'-r)^2]W(r-r',h)dr' \\ &= A(r) \int W(r-r',h)dr' + A'(r) \int (r'-r)W(r-r',h)dr' + O(h^2). \end{aligned} \tag{2.3.14}$$

The second term is an integral of an odd function which vanishes and we have the estimate.

With this integral representation, the gradient and divergence operator can be written in a straightforward manner, i.e

$$\langle \nabla A(r) \rangle = \int A(r') \nabla W(r - r') dr', \tag{2.3.15}$$

$$\langle \nabla \cdot A(r) \rangle = \int A(r') \cdot \nabla W(r - r') dr'. \tag{2.3.16}$$

Once we have the kernel representation, the second key step is the particle approximation. In this step, the continuous integral is discretized to be the summation of the particle quantities within the domain. The infinitesimal volume dr' can be changed to a finite volume ΔV_i which is related to the mass of a particle i by the following relationship:

$$m_i = \Delta V_i \rho_i, \tag{2.3.17}$$

where ρ_i is the local density near particle j.

Therefore we can get

$$\begin{aligned} A(r) &= \int A(r') W(r - r', h) dr' \\ &= \sum_j A(r_j) W(r - r_j, h) \Delta V_j \\ &= \sum_j A(r_j) W(r - r_j, h) \frac{1}{\rho_j} (\rho_j \Delta V_j) \\ &= \sum_j \frac{m_j}{\rho_j} A(x_j) W(r - r_j, h). \end{aligned} \tag{2.3.18}$$

This gives us the particle summation formula for a quantity A at any position r. In particular, the local density ρ can be obtained by

$$\rho_i = \rho(r_i) = \sum_j m_j W(r_i - r_j, h). \tag{2.3.19}$$

Also the derivatives are given in the form

$$\nabla A(r) = \sum_j m_j \frac{A_j}{\rho_j} \nabla W(r - r_j, h), \tag{2.3.20}$$

$$\Delta A(r) = \sum_j m_j \frac{A_j}{\rho_j} \Delta W(r - r_j, h). \tag{2.3.21}$$

There are some problems about this direct discretization for the derivative, in the sense that these equations do not satisfy certain physical principals such as symmetry of force and conservation of momentum. Therefore they are not suitable for our macroscopic model (2.3.8), so we discuss some modifications.

One of the modifications for the gradient operator can be found in [89]. Consider a function f, we know $\nabla(\rho f) = \rho\nabla f + f\nabla\rho$, which can be rewritten as $\rho\nabla f = \nabla(\rho f) - f\nabla\rho$. Applying the SPH discretization, we can have the following expression:

$$\rho_i\nabla f_i = \sum_i m_i(f_j - f_i)\nabla W(r_i - r_j, h). \tag{2.3.22}$$

To deal with the $-\frac{\nabla p}{\rho}$ term in (2.3.8), keep in mind that this represents the repulsive force in the microscopic model. Therefore, we seek for a symmetric form of interaction. The most simple repulsive force takes the form

$$\nabla f = \sum_j m_j \frac{f_i + f_j}{2\rho_j}\nabla W(r_i - r_j, h). \tag{2.3.23}$$

This is a direct average for the force interacting between particles i and j. For a better construction, we will make use of the identity

$$\nabla(\frac{f}{\rho}) = \frac{\rho\nabla f - f\nabla\rho}{\rho^2}, \tag{2.3.24}$$

which can be written as

$$\frac{\nabla f}{\rho} = \nabla(\frac{f}{\rho}) + \frac{f\nabla\rho}{\rho^2}. \tag{2.3.25}$$

Let f be our pressure p, and apply the SPH discretization. We can obtain our formula for the repulsive force in the microscopic level, which is

$$(-\frac{\nabla p}{\rho})_i = -\sum_j m_j(\frac{p_i}{\rho_i^2} + \frac{p_j}{\rho_j^2})\nabla W(r_i - r_j, h). \tag{2.3.26}$$

We can see that in this way, particle i and particle j interaction with each other with a repulsive force which follows Newton's second law.

Next we shall deal with the viscosity term $\nabla^2 v$. Viscosity is a phenomenon from friction, and would reduce the velocity difference and kinetic energy among the particles in fluid. This is the term which would give us the alignment interaction. While the standard discretization (2.3.21) does not capture this feature, we can consider the following modifications: One possible discretization considers the following identity:

$$\nabla^2(\rho f) = f(\nabla^2 \rho) + \rho(\nabla^2 f) + 2\nabla f \nabla \rho. \tag{2.3.27}$$

i.e.

$$\rho(\nabla^2 f) = \nabla^2(\rho f) - f(\nabla^2 \rho) - 2\nabla f \cdot \nabla \rho. \tag{2.3.28}$$

An direct discretization for the above term leads to

$$\nabla^2 f_i = \frac{1}{\rho_i}\sum_j m_j(f_j - f_i)(\nabla^2 W(r_i - r_j, h) - \frac{2}{\rho_i}\nabla W(r_i - r_j, h)\cdot\nabla\rho_i). \tag{2.3.29}$$

Although this is a difference form, it is not suitable for an individual model due to the complex terms at the right hand side.

Artificial viscosity has also been introduced in literature. By simple argument about its form and relationship to the gas viscosity, in [74], a typical form of artificial viscosity is given the form

$$\Pi_{ij} = \begin{cases} \frac{-\alpha\bar{c_{ij}}\mu_{ij}+\beta\mu_{ij}^2}{\bar{\rho_{ij}}}, \text{ if } (v_i - v_j)\cdot(r_i - r_j) \le 0, \\ 0, \text{ otherwise.} \end{cases} \tag{2.3.30}$$

where

$$\mu_{ij} = \frac{h(v_i - v_j)\cdot(r_i - r_j)}{|r_i - r_j|^2 + \epsilon h^2} \tag{2.3.31}$$

is an estimate for the velocity divergence. The notation $\bar{c_{ij}} = \frac{1}{2}(c_i + c_j)$ is used for scalars. This viscosity works well except that this give rise to large entropy when in a pure shear flow. A modification is the Balsara viscosity which reads

$$\mu_{ij} = \frac{h(v_i - v_j)\cdot(r_i - r_j)}{|r_i - r_j|^2 + \epsilon h^2}\frac{f_i + f_j}{2}, \tag{2.3.32}$$

where

$$f_i = \frac{|\nabla \cdot v|_i}{|\nabla \cdot v|_i + |\nabla \times v|_i + \epsilon c_i / h}. \tag{2.3.33}$$

And finally, for our purpose, there is a natural way to symmetrize the viscosity term by writing

$$\nabla^2 v_i = \nabla^2 v_i - v_i \nabla^2 1, \tag{2.3.34}$$

where the second term is 0 and can be added for free. In this way, we apply the SPH descritization and we can obtain

$$\nabla^2 v_i = \sum_j m_j \frac{v_j - v_i}{\rho_j} \nabla^2 W(r_i - r_j, h). \tag{2.3.35}$$

We can see that this is in the form $a_{ij}(v_j - v_i)$ where a_{ij} is a constant depending on the density of particles i and j as well as the distance between them. This is very similar to the Cucker Smale model where the alignment is achieved by applying a force in the form of the velocity difference.

Next we shall discuss about the choice of the kernel W. Other than the three key properties, we also enforce the non-negativity and decay condition. We require $W(r - r') > 0$ for any r' within the support domain. Although it is not mathematically essential for convergence, it is important to ensure a physically meaningful representation of a quantity. The smoothing function value should also be monotonically decreasing as the distance away from the particle increase, indicating that a nearer neighbour should have a larger influence on a particle. Any function having all the above properties can be employed as a smoothing kernel function.

In Lucy's original paper [81], a bell-shaped function is used.

$$W(r - r', h) = W(R, h) = \alpha_d \begin{cases} (1 + 3R)(1 - R)^3 \text{ if } R \leq 1 \\ 0 \text{ if } R > 1 \end{cases} ., \tag{2.3.36}$$

where α is a normalizing constant, which is $5/4h, 5/\pi h^2$ and $105/16\pi h^3$ in one, two and three dimensions respectively. Here $R = \frac{|r-r'|}{h}$.

In Gingold and Monaghan's original paper [51], Gaussian kernel is used.

$$W(R,h) = \alpha_d e^{-R^2}. \tag{2.3.37}$$

With $\alpha_d = 1/\pi^{1/2}h, 1/\pi h^2, 1/\pi^{3/2}h^3$ respectively. Although it is not really compact, piratically it approaches zero numerically very fast. It is also very stable and accurate for disordered particles.

The most commonly used kernel function is the cubic B-spline function, used in [88],

$$W(R,h) = \alpha_d \begin{cases} \frac{2}{3} - R^2 + \frac{1}{2}R^3, \text{ if } 0 \leq R < 1, \\ \frac{1}{6}(2-R)^3, \text{ if } 1 \leq R < 2, \\ 0, \text{ if } R \geq 2. \end{cases} \tag{2.3.38}$$

Other choices of the kernels are available for simulating different problems. For our individual based model, the kernels are designed in the form

$$W(R,h) = \frac{1}{Ch^d} \begin{cases} f(R), \text{ if } 0 \leq R < 1, \\ 0, \text{ if } R \geq 1. \end{cases} \tag{2.3.39}$$

Direct computation gives

$$\nabla W(r-r',h) = \frac{r-r'}{Ch^{d+2}} \begin{cases} R^{-1}f'(R), \text{ if } 0 \leq R < 1, \\ 0, \text{ if } R \geq 1. \end{cases} \tag{2.3.40}$$

and

$$\nabla^2 W(R,h) = \frac{1}{Ch^{d+2}} \begin{cases} f''(R) + (d-1)R^{-1}f'(R), \text{ if } 0 \leq R < 1, \\ 0, \text{ if } R \geq 1. \end{cases} \tag{2.3.41}$$

Here the normalization constant C can be computed as $C = \int\limits_0^1 2\pi R f(R) dR$ in 2D and $C = \int_0^1 4\pi R^2 f(R) dR$ in 3D.

For the computation of density, any smooth kernels would suffice. We adapt the choice in [93], and choose a degree 6 polynomial $f_{\text{poly6}}(R) = (1-R^2)^3$. We can get $C = \pi/4$ and $64\pi/315$ respectively in 2D and 3D.

For the computation of pressure, however, the usual kernels would not work. The reason is that the repulsive force vanishes as particles get very close to each other, since the gradient of the kernel approaches zero at the center. Therefore, a reasonable choice of the kernel would have a spiky shape which has a large gradient near zero and the gradient should vanish for $R = 1$. This would describe the correct repulsive interaction among individuals. [43] describes a kernel to fit our purpose. We will choose $f_{\text{spicky}}(R) = (1 - R)^3$ for the computation of the repulsive force, which yields $C = \pi/10$ in 2D and $C = \pi/15$ in 3D.

Finally, for the viscosity term, remembering this is the alignment term which is similar as the Cucker-Smale model except for a density term. We would require the weight function $\nabla^2 W$ to be positive everywhere for an alignment effect. Also $\nabla^2 W$ should be decreasing as R increase representing a greater interaction as particles becomes nearer. Therefore, a simple choice of $\nabla^2 W$ is $\frac{1}{Ch^{d+2}}(1 - R)$ for $0 \leq R < 1$. In this way, we can find that in $2D$,

$$f_{\text{vis}}(R) = -\frac{R^3}{9} + \frac{R^2}{4} - \frac{lnR}{6} - \frac{5}{36}, \tag{2.3.42}$$

with $C = \pi/40$. And in $3D$, $f(R)$ has a slightly different format with

$$f_{\text{vis}}(R) = -\frac{R^3}{12} + \frac{R^2}{6} + \frac{1}{12R} - \frac{1}{6} \tag{2.3.43}$$

Collecting the results, and considering dimension $d = 2$, we can write out the results explicitly as follows, considering only $0 \leq R < 1$ for simplicity:

$$W_{\text{poly6}}(r, h) = \frac{4}{\pi h^8}(h^2 - |r|^2)^3, \tag{2.3.44}$$

$$\nabla W_{\text{poly6}}(r, h) = -\frac{24r}{\pi h^8}(h^2 - |r|^2)^2, \tag{2.3.45}$$

$$W_{\text{spiky}}(r, h) = \frac{10}{\pi h^8}(h - |r|)^3, \tag{2.3.46}$$

$$\nabla W_{\text{spiky}}(r, h) = -\frac{30}{\pi h^4}\frac{(1 - R)^2}{R}r, \tag{2.3.47}$$

$$W_{\text{vis}}(r, h) = \frac{40}{\pi h^2}(-\frac{R^3}{9} + \frac{R^2}{4} - \frac{lnR}{6} - \frac{5}{36}), \tag{2.3.48}$$

$$\nabla^2 W_{\text{vis}}(r, h) = \frac{40}{\pi h^4}(1 - R). \tag{2.3.49}$$

Put everything into our macroscopic model (2.3.8), and assuming a constant mass for all the particles m, we derive our individual based model as:

$$\frac{dx_i}{dt} = v_i, \tag{2.3.50}$$

$$\frac{dv_i}{dt} = \sum_j F_{rij} + \sum_j F_{aij} + F_{p_i}, \tag{2.3.51}$$

$$F_{rij} = \frac{30m}{\pi h^4}(\frac{p_i}{\rho_i^2} + \frac{p_j}{\rho_j^2})\frac{(1-R)^2}{R}(x_i - x_j), \tag{2.3.52}$$

$$F_{aij} = \frac{40m\nu}{\pi h^2}\frac{v_j - v_i}{\rho_j}, \tag{2.3.53}$$

$$F_{p_i} = \lambda(v_\infty^2 - |v|^2)v. \tag{2.3.54}$$

Together with

$$\rho_i = \sum_j \frac{4m}{\pi h^2}(1-R^2)^3, \quad p_i = p_0\rho_i, \quad v_\infty = max\{v_0(1 - \frac{\rho_c + \rho_0}{\rho + \rho_0}), 0\}. \tag{2.3.55}$$

This is an individual based model. Each particle will conduct self propulsion due to the force F_p. The limiting velocity is chosen such that below a critical density ρ_c, the velocity is 0. If the dnesity is above ρ_c, the limiting velocity is an increasing function of ρ which approaches v_0 as ρ goes to infinity. The particles also interact with their neighbours. The term F_r is a force responsible for the repulsion among individuals, which is analogue to the pressure in the fluid. F_a is an alignment force, where particles try to adapt their velocity with their neighbours, and this give rise to the viscosity in the fluid. Due to its derivation, we can predict the global behaviour which would follow the Navier-Stokes equation. In similar manners, we can derive other microscopic models as long as the we have a global description. The difference between this model with classical individual based model is that a local density is introduced. Although its highly unlikely each particle can compute its local density in reality, it helps to produce a more homogeneous global structure.

2.4 Numerical methods

2.4.1 GPU parallelization

Similarly as the Molecular Dynamics (MD), the individual based models simulate a large number of particles at atomic level. The computational time is too long to bear if each individual is computed sequentially. In recent years, the computation speed of graphics processing unit(GPU) has increased rapidly. The modern GPU is not only a graphic engine but also a highly parallel programmable processor [128].The use of GPU in non-graphic applications is termed general-purpose computing on the GPU (GPGPU). Compared to modern CPUs which usually contain two or four cores, modern graphic card can contain as many as 480 cores, and each works in parallel. This parallelism enables us to calculate large data sets in real time. In the computation of this thesis, a NVIDIA GeForce gtx 560 ti is used, with 384 cores.

The traditional GPGPU development is based on graphics function library, examples include OpenGL and Direct 3D, thus only professional users who are familiar with graphics API can access the power of the GPU computing. The only way to make use of the GPUs abilities was to carefully cast the algorithm and data structures to be represented as individual pixels being written to an image via fragment shaders, which makes programming very cumbersome. Furthermore, each thread of execution can only write a single output value to a single memory location.

To meet the demand of GPGPU for common users, CUDA (Compute Unified Device Architecture) technology is developed. Users no longer need to map programs into graphic API and use shader languages. CUDA brings C-like development environment and use a C-compiler so that users who are not familar with the graphics knowledge can do GPU computing [77]. Now CUDA can run on all current NVIDIA GPUs, including the HPC-oriented Tesla product line. Such nature of these GPUs makes them a compelling platform for accelerating high-performance computing.

CUDA has been applied in various research fields, examples include image processing [134], finite difference [84] and molecular dynamics [4].

Now we will discuss about the structure of a GPU. A GPU is a set of multiprocessors, and each has its own stream processors and shared memories. The stream processors are capable of executing integer and single precision floating point calculation. Additional cores are also available for double-precision. In CUDA, parallel kernel programs are performed on parallel threads. All the threads will execute the same sequential program, with different input data. Each thread has its own variables (registers) and processor state.

Threads are then grouped into thread blocks. Note that threads within the same block can communicate. There are two mechanisms by which they can communicate. First is a barrier mechanism which is thread synchronization. Once it is called, any thread in the block would be delayed until all other threads in the block reach the synchronization point. Second, each block has a shared memory that exists on the multiprocessor. Any thread in the block can access this shared memory area without communication with the global memory. Note that all the blocks and thread have their unique ID number.

2.4.2 Numerical methods for microscopic modified Vicsek model

In this part, we describe the numerical schemes used to solve the individual based model(2.2.4,2.2.5,2.2.6) with CUDA.

$$\frac{dx_i}{dt} = v_i, \tag{2.4.1}$$

$$\frac{dv_i}{dt} = \mu_1(v_0\omega_i - v_i) + \frac{\mu_2}{N}\sum_j F_{ij}, \tag{2.4.2}$$

$$d\omega_i = P_{\omega_i^\perp}(\nu\bar{\omega}_i dt + \sqrt{2D}dB_t + \alpha v_i dt). \tag{2.4.3}$$

$$\tag{2.4.4}$$

For each particle i, the position x_i^0, orientation ω_i^0 are initialized in CPU. And the initial velocity is given as $v_i^0 = v_0\omega_i$. The grid cells for neighbour searching are also initialized and stored in CPU. The data stored in CPU memory is then copied to GPU memory. And the positions are allocated in an OpenGL vertex array object (VBO) so that direct rendering of the particles are available. Whenever we need to retrieve data, we can copy the data back into CPU memory. Numerically each block handles one cell and each tread handles one particle. The interaction with a particle is computed from all the particles in its own cell and the neighbouring cells

Integration

The domain can be a rectangle, a circle or an annulus in 2D or 3D.

The particles $i = 1..N$ are initialized with their positions, velocities and orientations stored in arrays in the CPU memory.

To update the orientation, we firstly consider the case without the random noise. Note that $P_{\omega_i^\perp}(w_i) = 0$ that (2.2.6) can be written as

$$d\omega_i = P_{\omega_i^\perp}(\nu dt(\bar{\omega}_i + \frac{\alpha}{\nu}v_i - \omega_i)), \tag{2.4.5}$$

where the last term is added for free. This can be discretized as

$$\omega_i^{n+1} - \omega_i^n = (Id - \omega_i^{n+\frac{1}{2}} \otimes \omega_i^{n+\frac{1}{2}})(\nu dt(\bar{\omega}_i{}^n + \frac{\alpha}{\nu}v_i^n - \omega_i^n)). \tag{2.4.6}$$

Here

$$\omega_i^{n+\frac{1}{2}} = \frac{\omega_i^n + \omega_i^{n+1}}{|\omega_i^n + \omega_i^{n+1}|}. \tag{2.4.7}$$

Without knowing ω^{n+1}, this still can be done using geometry. See the figure below. We write

$$\omega_i^{n+1} - \omega_i^n = P_{\Gamma^\perp} a_i^n. \tag{2.4.8}$$

Here Γ is the plane orthogonal to the $\omega_i^{n+\frac{1}{2}}$ Let point B be the center of the circle with diameter joined by the two points ω_i^n and $\omega_i^n + a_i^n$, the intersection points between this circle and the unit circle will give us ω_i^n and ω_i^{n+1} respectively.

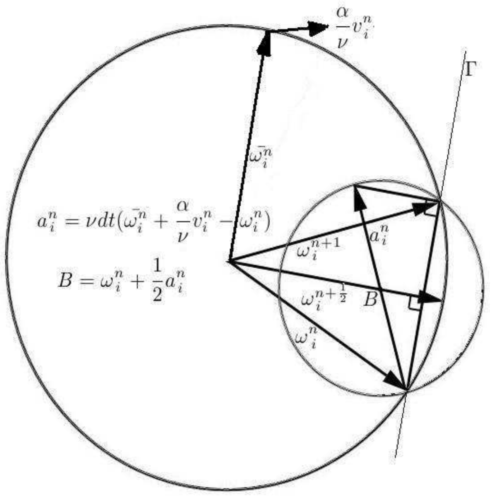

$\frac{\alpha}{\nu}v_i^n$
Γ
$\bar{\omega}_i^n$
$a_i^n = \nu dt(\bar{\omega}_i^n + \frac{\alpha}{\nu}v_i^n - \omega_i^n)$
$B = \omega_i^n + \frac{1}{2}a_i^n$
ω_i^{n+1}
a_i^n
$\omega_i^{n+\frac{1}{2}}$
B
ω_i^n

Figure 2.2: Technical Circle

In 2D, denoting the angle of the vector ω_i^n by θ_i^n, this geometry implies that

$$\theta_i^{n+1} = \theta_i^n + 2(\widehat{\omega_i^n, B}). \tag{2.4.9}$$

To take account of the noise, we simply add a random variable

$$\theta_i^{n+1} = \theta_i^n + 2(\widehat{\omega_i^n, B}) + \sqrt{2Ddt}\epsilon, \tag{2.4.10}$$

where ϵ is drawn from a unit normal distribution. Finally ω_i^{n+1} is defined by the angle θ_i^{n+1}.

In 3D, similarly, we can compute the center of the technical sphere B in a similar manner. Writing $B = (x_B, y_B, z_B)$. we determine the standard Euler angles.

$$\phi_B = \arctan(\frac{y_B}{x_B}), \quad \theta_B = \arccos(z_B). \tag{2.4.11}$$

Denoting the angle of the vector ω_i^n by θ_i^n, ϕ_i^n, we get

$$\theta_i^{n+1} = 2\theta_B - \theta_i, \quad \phi_i^{n+1} = 2\phi_B - \phi_i. \tag{2.4.12}$$

To add a white noise, we firstly apply a rotation operation R which will rotate θ_i^{n+1} to the z axis. In general, to apply a rotation about $u = (u_x, u_y, u_z)$ by an angle of

θ, the rotation matrix takes the following form:

$$R = \begin{pmatrix} \cos\theta + u_x^2(1-\cos\theta) & u_x u_y(1-\cos\theta) - u_z\sin\theta & u_x u_z(1-\cos\theta) + u_y\sin\theta \\ u_y u_x(1-\cos\theta) + u_z\sin\theta & \cos\theta + u_y^2(1-\cos\theta) & u_y u_z(1-\cos\theta) - u_x\sin\theta \\ u_z u_x(1-\cos\theta) - u_y\sin\theta & u_z u_y(1-\cos\theta) + u_x\sin\theta & \cos\theta + u_z^2(1-\cos\theta) \end{pmatrix}. \tag{2.4.13}$$

Applying a rotation $-\phi_i^n$ around the z-axis followed by a rotation θ_i^n around the y-axis yields the desired rotation matrix:

$$R = \begin{pmatrix} \cos\theta\cos\phi & \cos\theta\sin\phi & -\sin\theta \\ -\sin\phi & \cos\phi & 0 \\ \sin\theta\cos\phi & \sin\theta\sin\phi & \cos\theta \end{pmatrix}. \tag{2.4.14}$$

The noise can be added on ϕ in the same pattern as in 2D where as a random angle for θ is drawn from a uniform distribution in $[0, 2\pi]$. And finally ω_i^{n+1} is obtained by an inverse rotation.

The force $\sum_j F_{ij}^n$ can be computed exactly by summing up all the interaction forces near particle i, the computation of local interactions requires spatial subdivision which would be discussed later.

Once we have ω_i^{n+1} and $\sum_j F_{ij}^n$, the velocity and the position can be updated using Euler method:

$$\frac{v_i^{n+1} - v_i^n}{dt} = \mu_1(v_0\omega_i^{n+1} - v_i^n) + \frac{\mu_2}{N}\sum_j F_{ij}^n, \tag{2.4.15}$$

$$\frac{x_i^{n+1} - x_i^n}{dt} = v_i^{n+1}. \tag{2.4.16}$$

Neighbour Search Algorithm

To compute the interaction, a particle would interact with all the particles within the radius. Checking the distance with all the other particles requires $O(N^2)$ operations. To avoid the unnecessary distance computations, we combine the advantage of Verlet table [114] and cell linked list [132] to accelerate the neighbour list construction.

The idea of Verlet table algorithm is that, for each particle in the system, a table is constructed and maintained which stores all its neighbours. In this way, the total CPU time is scaled from $O(N^2)$ to $O(N \cdot N_{\text{neighbours}})$ where $N_{\text{neighbour}}$ is the average number of neighbours in the Verlet table and is independent of N. The table is initialized and then updated periodically for every few time steps.

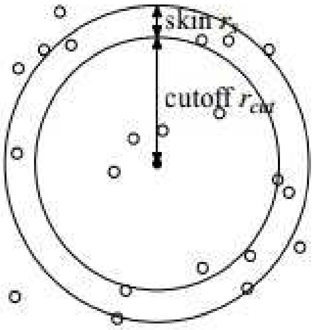


Figure 2.3: Illustration of Verlet table algorithm

While the physical cutoff interaction radius is denoted by r_{cut}, the neighbours considered in the Verlet Table has a larger radius where a skin layer r_s is considered. For each particle i, particle j is considered as a neighbour in the Verlet table if $r_{ij} < r_{\text{cut}} + r_s$. The skin r_s is chosen such that $r_s > v_{\text{max}} dT$. Here v_{max} is maximum velocity of a particle and dT is the time interval to update the Verlet table. In this way, a particle cannot penetrate the skin layer during one update interval. Verlet table is efficient when number of particles is small and the motion is slow. Else dT needs to be reduced or r_s needs to be increased, both results dramatic increase in computation time. Another drawback is the construction of the Verlet neighbour table still requires a computation of $O(N^2)$.

For a large particle system, for a true $O(N)$ scheme, we use the cell linked list algorithm. The domain is partitioned into $N_x \times N_y \times N_z$ cells which are cubes with edge length $r = r_{\text{cut}} + r_s$. Note that we used uniform grid in this case. In this way, each individual will only interact with others in neighbouring cells.

Each cell has a unique index. Firstly, the grid position of each particles is

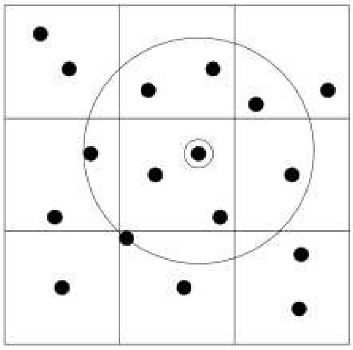


Figure 2.4: Illustration of cell linked list algorithm

computed. In a 3D domain with a corner at the origin, the grid position $\{p_1, p_2, p_3\}$, is defined as:$\{\lfloor \frac{x_1}{r} \rfloor, \lfloor \frac{x_2}{r} \rfloor, \lfloor \frac{x_3}{r} \rfloor\}$ and the cell index corresponding to this particle is $p_z N_x n_y + p_y N_x + p_x$. In this way two arrays are created in global memory, one is the particle index and the other is the corresponding cell index for each particle.

To get a fast access to the neighbours of individual, we can sort the particle index using the cell index as key. Since the cell indexes are integers, a quick radix sort is applied, which sorts data by grouping keys by the individual digits which share the same significant position and value. The processing of the keys begins at the least significant digit (i.e., the rightmost digit), and proceeds to the most significant digit (i.e., the leftmost digit), therefore it is also named least significant digit (LSD) radix sort. Note that an LSD radix sort operates in $O(nk)$ time, where n is the number of keys, and k is the average key length.

In this way, we can record the starting particle index and the ending particle index in each cell. When looking for the neighbours, we will only need to find the neighbouring cell index and consider the particles inside.

Boundary Conditions

We firstly consider the domains in 2D. Four types of domains are considered: rectangle domain with periodic boundary conditions in both directions, or periodic boundary conditions in x direction and repulsive boundary conditions in y direction, circle domain and annulus domain. For a rectangle domain $[0, L_x] \times [0, L_y]$ with periodic boundary condition, we use the grid method and divide the domain into $N_x \times N_y$ cells. A layer of ghost cells with ghost particles are created around the domain. For example, for a cell labelled by $(k, 1)$ which is at the bottom of the domain, we create a corresponding ghost cell labelled by $(k, N_y + 1)$ which is above the domain. For each particle i in cell $k, 1$, with position $X_i = (X_i(1), X_i(2))$, we shall create a corresponding particle in cell k, N_y+1, with position $X_i(1), X_i(2)+L_y$. The velocity and orientation of the corresponding particle remains the same. In the same manner, we create ghost cells for all the cells near the boundary. Finally, the interactions are computed with all the real particles together with the ghost particles.

For a rectangle domain $[0, L_x] \times [0, L_y]$ with periodic boundary conditions in x direction and repulsive boundary in the y direction, ghost cells are created for cells labelled by $1, k$ and N_x, k in the same way as the above case. We consider a particle near the bottom boundary, with position $X_i = (X_i(1), X_i(2))$. If $X_i(2) < R_b$, the threshold interaction radius near the boundary, we apply an upward force $(0, F_b(R))$, where $R = X_i(2)$ is the distance between the particle and the boundary and $F_b(R) = -2\frac{F_1}{R_b}(R) + 2F_1$ for $R < R_b$ and F_1 being a constant, which has the same form as the interaction force among particles. Particles near the top boundary is treated with the same manner.

For a circle domain centred at the origin with radius L, when a particle is near the boundary at X_i, a repulsive force is applied in the direction $-X_i$, which is normal to the wall, with magnitude $F_b(R)$ where $R = L - |X_i|$ is the distance between the particle and the wall. The annulus domain is treated in the same way except the repulsive force is along the direction X_i for particles near the inner boundary.

For 3D domains considered, repulsive forces are added along the z direction which is normal to the horizontal wall. If a particle is near a corner in the 3D domain, the horizontal repulsive force and the vertical repulsive force are both considered and the resultant force is the sum of them. The boundary force together with the particle particle interaction completely determines the motion in the microscopic level.

2.4.3 Numerical methods for the macroscopic model

Relaxation Model

In this section we develop numerical approximation for the system (2.2.42-2.2.47). The difficulty of solving the system is that the system is not conservative, due to the geometry constraint $|\Omega| = 1$. Weak solutions of non-conservative systems are not unique because jump relations across discontinuities are not uniquely defined. To develop the numerical scheme, we will firstly see that the system can be seen as the relaxation limit of a hyperbolic system. More precisely, we introduce the following relaxation model, in dimension $n = 2$:

$$\partial_t \rho^\eta + \nabla_x \cdot (\rho^\eta U^\eta) = 0, \tag{2.4.17}$$

$$\partial_t(\rho^\eta \Omega^\eta) + \nabla_x \cdot (\rho^\eta V^\eta \otimes \Omega^\eta) + \nabla_x p(\rho^\eta) - \gamma \Delta(\rho^\eta \Omega^\eta) = \frac{\rho^\eta}{\eta}(1 - |\Omega^\eta|^2)\Omega^\eta, \tag{2.4.18}$$

where

$$U^\eta = C_1 \Omega^\eta - \Phi \nabla_x \rho^\eta, \tag{2.4.19}$$

$$V^\eta = C_2 \Omega^\eta - \Phi \nabla_x \rho^\eta, \tag{2.4.20}$$

$$p(\rho^\eta) = d\rho^\eta + \alpha \Phi (d + C_2) \frac{(\rho^\eta)^2}{2}, \tag{2.4.21}$$

$$\gamma = \frac{k_1}{k_0}(d + C_2). \tag{2.4.22}$$

We can prove that as η goes to zero, the relaxation model will converge to our macroscopic model (2.2.42,2.2.43). To see this we will start with our system (2.2.115,2.2.116). Here the second equation is divided by c and c_4 to c_7 are the corresponding new coefficients, and the corresponding relaxation model reads:

$$\partial_t \rho^\eta + c_1 \nabla_x \cdot (\rho^\eta \Omega^\eta) + c_2 \Delta(\frac{\rho^{\eta 2}}{2}) = 0, \tag{2.4.23}$$

$$\begin{aligned}&\partial_t(\rho^\eta\Omega^\eta)+c_3\nabla\rho^\eta+c_4\nabla\cdot(\rho^\eta\Omega^\eta\otimes\Omega^\eta)+c_5\nabla\cdot(\nabla(\frac{\rho^{\eta 2}}{2}\otimes\Omega^\eta))\\&+c_6\nabla(\frac{\rho^{\eta 2}}{2})+c_7\Delta(\rho^\eta\Omega^\eta)=\frac{\rho^\eta}{\eta}(1-|\Omega^\eta|^2)\Omega^\eta.\end{aligned}\tag{2.4.24}$$

To proof the convergence of the above relaxation model, we define

$$R^\eta=\rho^\eta(1-|\Omega^\eta|^2)\Omega^\eta.\tag{2.4.25}$$

By assumption, the left-hand side of equation (2.4.24) is bounded independently of η, therefore multiplying (2.4.24) by η and taking he limit $\eta\to 0$ yields $R^\eta\to 0$. This implies that $|\Omega^\eta|^2=1$ (except where $\rho^\eta\Omega^\eta=0$ which one assumes to be a negligible set).

By definition of R^η, we know that R^η is aligned with Ω^η, we have $R^\eta\times\Omega^\eta=0$, thus (2.4.24) leads to:

$$\begin{aligned}&(\partial_t(\rho^\eta\Omega^\eta)+c_3\nabla\rho^\eta+c_4\nabla\cdot(\rho^\eta\Omega^\eta\otimes\Omega^\eta)+c_5\nabla\cdot(\nabla(\frac{\rho^{\eta 2}}{2}\otimes\Omega^\eta))\\&+c_6\nabla(\frac{\rho^{\eta 2}}{2})+c_7\Delta(\rho^\eta\Omega^\eta))\times\Omega^\eta=0.\end{aligned}\tag{2.4.26}$$

Basically this implies that the projection of the sum of all the terms onto Ω^η yields 0. In another word, assuming ρ^η and Ω^η will converge to ρ^0 and Ω^0 respectively, as $\eta\to 0$

$$\begin{aligned}&(Id-\Omega^0\otimes\Omega^0)(\partial_t(\rho^0\Omega^0)+c_3\nabla\rho^0+c_4\nabla\cdot(\rho^0\Omega^0\otimes\Omega^0)\\&+c_5\nabla\cdot(\nabla(\frac{\rho^{02}}{2}\otimes\Omega^0))+c_6\nabla(\frac{\rho^{02}}{2})+c_7\Delta(\rho^0\Omega^0))=0.\end{aligned}\tag{2.4.27}$$

We can compute the projection of each term and we find that

$$(Id-\Omega^0\otimes\Omega^0)(\rho^0\Omega^0)=\rho^0\partial_t\Omega^0,\tag{2.4.28}$$

$$(Id-\Omega^0\otimes\Omega^0)(\nabla\cdot(\rho^0\Omega^0\otimes\Omega^0))=\rho^0(\Omega^0\cdot\nabla)\Omega^0,\tag{2.4.29}$$

$$(Id-\Omega^0\otimes\Omega^0)(\nabla\cdot(\nabla(\frac{\rho^{02}}{2}\otimes\Omega^0)))=(\nabla(\frac{\rho^{02}}{2})\cdot\nabla)\Omega^0.\tag{2.4.30}$$

Inserting the above expressions into (2.4.27) will yield (2.4.24). And this system can be rewritten as (2.4.18)

Time splitting scheme

With this relaxation model, we get rid of the constraint $|\Omega| = 1$ and develop a time splitting scheme. The system (2.4.31,2.4.18) can be split into two parts. The conservative part reads

$$\partial_t \rho^\eta + \nabla_x \cdot (\rho^\eta U^\eta) = 0, \tag{2.4.31}$$

$$\partial_t(\rho^\eta \Omega^\eta) + \nabla_x \cdot (\rho^\eta V^\eta \otimes \Omega^\eta) + \nabla_x p(\rho^\eta) - \gamma \Delta(\rho^\eta \Omega^\eta) = 0. \tag{2.4.32}$$

And the non-conservative part or relaxation part reads

$$\partial_t \rho^\eta = 0, \tag{2.4.33}$$

$$\partial_t(\rho^\eta \Omega^\eta) = \frac{\rho^\eta}{\eta}(1 - |\Omega^\eta|^2)\Omega^\eta. \tag{2.4.34}$$

Note that (2.4.34) can be reduced to

$$\partial_t \Omega = \frac{1}{\eta}(1 - |\Omega|^2)\Omega. \tag{2.4.35}$$

In terms of Ω^2, it becomes

$$\frac{1}{2}\partial_t |\Omega|^2 = \frac{1}{\eta}(1 - |\Omega|^2)|\Omega|^2. \tag{2.4.36}$$

(2.4.36) is an ODE for $|\Omega^2|$ and can be solved explicitly: $|\Omega|^2 = (1 + C_0 e^{-2/\eta t})^{-1}$ with $C_0 = (\frac{1}{|\Omega_0|^2} - 1)$. Numerically, taking the limit $\eta \to 0$, the relaxation part (2.4.34) yields a mere normalization:

$$\Omega^{n+1} = \frac{\Omega^n}{|\Omega^n|}. \tag{2.4.37}$$

To solve the conservative part, we will drop all the η for simplicity and will let $\Omega = (\Omega_1, \Omega_2), U = (U_1, U_2), V = (V_1, V_2)$, the conservative part becomes

$$\rho_t + (\rho U_1)_x + (\rho U_2)_y = 0, \tag{2.4.38}$$

$$\partial_t(\rho\Omega_1) + (\rho V_1 \Omega_1)_x + (\rho V_2 \Omega_1)_y + p_x - \gamma((\rho\Omega_1)_{xx} + (\rho\Omega_1)_{yy}) = 0, \tag{2.4.39}$$

$$\partial_t(\rho\Omega_2) + (\rho V_1 \Omega_2)_x + (\rho V_2 \Omega_2)_y + p_y - \gamma((\rho\Omega_2)_{xx} + (\rho\Omega_2)_{yy}) = 0. \tag{2.4.40}$$

We let $u = \rho\Omega_1$ and $v = \rho\Omega_2$, and use the coefficients in (2.2.115,2.2.116), the system can be written in the following form:

$$Q_t + (F(Q, Q_x))_x + (G(Q, Q_y))_y = 0. \tag{2.4.41}$$

Here $Q = (\rho, u, v)$

$$F(Q, Q_x) = \begin{pmatrix} c_1 u + c_2 \rho\rho_x \\ c_3\rho + c_4\frac{u^2}{\rho} + c_5\rho_x u + c_6\frac{\rho^2}{2} + c_7 u_x \\ c_4\frac{uv}{\rho} + c_5\rho_x v + c_7 v_x \end{pmatrix}, \tag{2.4.42}$$

and

$$G(Q, Q_y) = \begin{pmatrix} c_1 v + c_2 \rho\rho_y \\ c_4\frac{uv}{\rho} + c_5\rho_y u + c_7 u_y \\ c_3\rho + c_4\frac{v^2}{\rho} + c_5\rho_y v + c_6\frac{\rho^2}{2} + c_7 v_y \end{pmatrix}. \tag{2.4.43}$$

We will consider the following numerical scheme: Let the domain be a 2D rectangle $[(0, L_x) \times (0, L_y)]$, with $\Delta x = \frac{L_x}{n_x}$ and $\Delta y = \frac{L_y}{n_y}$, the quantity we are interested is $Q^n_{i,j}$, which is the approximation of Q at time $t^n = n\Delta t$ and position $x_i = i\Delta x, y_j = j\Delta y$, with $i = 0, 1 ... n_x + 1$ and $j = 0, 1 ... n_y + 1$.

The local Lax-Friedrich scheme can be written as

$$\frac{Q^{n+1}_{i,j} - Q^n_{i,j}}{\Delta t} + \frac{F^n_{i+\frac{1}{2},j} - F^n_{i-\frac{1}{2},j}}{\Delta x} + \frac{G^n_{i,j+\frac{1}{2}} - G^n_{i,j-\frac{1}{2}}}{\Delta y} = 0. \tag{2.4.44}$$

Here the flux F is given as

$$F^n_{i+\frac{1}{2},j} = \frac{F(Q^n_{i,j}, {Q_x}^n_{i,j}) + F(Q^n_{i+1,j}, {Q_x}^n_{i+1,j})}{2} - \frac{1}{2}|A(Q^n_{i+\frac{1}{2},j}, {Q_x}^n_{i+\frac{1}{2},j})|(Q_{i+1,j} - Q_{i,j}). \tag{2.4.45}$$

With

$${Q_x}^n_{i,j} = \frac{Q^n_{i+1,j} - Q^n_{i,j}}{\Delta x}, \quad Q^n_{i+\frac{1}{2},j} = \frac{Q^n_{i,j} + Q^n_{i+1,j}}{2}, \quad {Q_x}^n_{i+\frac{1}{2},j} = \frac{{Q_x}^n_{i,j} + {Q_x}^n_{i+1,j}}{2}.$$

(2.4.46)

In (2.4.45), $A(Q, Q_x)$ is the Jacobian matrix and can be write out explicitly as

$$\begin{aligned} A(Q, Q_x) &= \frac{\partial F(Q, Q_x)}{\partial Q} \\ &= \begin{pmatrix} c_2\rho_x & c_1 & 0 \\ c_3 - \frac{c_4 u^2}{\rho^2} + c_6\rho & \frac{2c_4 u}{\rho} + c_5\rho_x & 0 \\ -\frac{c_4 uv}{\rho^2} & \frac{c_4 v}{\rho} & \frac{c_4 u}{\rho} + c_5\rho_x \end{pmatrix} = \begin{pmatrix} a & b & 0 \\ c & d & 0 \\ e & f & g \end{pmatrix}. \end{aligned} \tag{2.4.47}$$

If the matrix can be diagonalized as $A = P\Lambda P^{-1}$, then $|A|$ is defined as $P|\Lambda|P^{-1}$ where $|\Lambda|$ is a diagonal matrix with diagonal entries the absolute values of the eigenvalues of A. In our case the eigenvalues λ can be calculated explicitly easily as

$$\lambda_1 = g, \quad \lambda_{2,3} = \frac{1}{2}(a + d \pm \sqrt{(a+d)^2 - 4(ad - bc)}). \tag{2.4.48}$$

We will use a scheme which is a simple approximation of A, which does not require the information of the spectral decomposition of A. To be precise, we will use a polynomial of a matrices $P_m^{i+\frac{1}{2}}$ with degree m to approximate $|A(Q_{i+\frac{1}{2},j}, Q_{x i+\frac{1}{2},j})|$.

The idea of the polynomial scheme aims to find $p_m(x)$ such that $|\Lambda|$ can be approximated by $p_m(\Lambda)$. Due to the diagonalization, we have

$$|A| = P|\Lambda|P^{-1} \approx P p_m(\Lambda) P^{-1} = p_m(A). \tag{2.4.49}$$

To build a polynomial approximation of A, we define $a_+ = \max\{\lambda\}$, the largest eigenvalue. $a_- = \min\{\lambda\}$, the smallest eigenvalue and $a_{\max} = a_+$ if $|a_+| > |a_-|$, $a_{\max} = a_-$ otherwise.

For $m = 0$, the polynomial p_0 is just a constant where $p(x) = |a_{\max}|$. Then $|A|$ is approximated by $|a_{\max}|Id$ where Id is the identity matrix.

For $m = 1$, the polynomial p_1 is a linear polynomial satisfies $p(a_-) = |a_-|$ and $p(a_+) = |a_+|$.

Finally, in our simulation, we will use a degree 2 polynomial scheme. In this case, the polynomial p_2 is a degree 2 polynomial which satisfies $p(a_-) = |a_-|$, $p(a_+) = |a_+|$

as well as $p'(a_{\max}) = sign(a_{\max})$. Numerically, we write $p_2(A) = \alpha A^2 + \beta A + \gamma$ where α, β, γ can be obtained by solving the following linear system

$$\begin{pmatrix} |a_+| \\ |a_-| \\ sign(a_{\max}) \end{pmatrix} = \begin{pmatrix} a_+^2 & a_+ & 1 \\ a_-^2 & a_- & 1 \\ 2a_{\max} & 1 & 0 \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix}. \tag{2.4.50}$$

Similarly, the flux G is given as

$$G_{i,j+\frac{1}{2}} = \frac{G(Q_{i,j}, Q_{y_{i,j}}) + G(Q_{i,j+1}, Q_{y_{i,j+1}})}{2} - \frac{1}{2}|B(Q_{i,j+\frac{1}{2}}, Q_{y_{i,j+\frac{1}{2}}})|(Q_{i,j+1} - Q_{i,j}). \tag{2.4.51}$$

with

$$Q_{y}{}_{i,j}^{n} = \frac{Q_{i,j+1}^n - Q_{i,j}^n}{\Delta y}, \quad Q_{i,j+\frac{1}{2}}^n = \frac{Q_{i,j}^n + Q_{i,j+1}^n}{2}, \quad Q_{y}{}_{i,j+\frac{1}{2}}^{n} = \frac{Q_{y}{}_{i,j}^{n} + Q_{y}{}_{i,j+1}^{n}}{2}. \tag{2.4.52}$$

And the Jacobian matrix B has the form

$$\begin{aligned} B(Q, Q_y) &= \frac{\partial G(Q, Q_y)}{\partial Q} \\ &= \begin{pmatrix} c_2\rho_y & 0 & c_1 \\ -\frac{c_4 uv}{\rho^2} & \frac{c_4 v}{\rho} + c_5\rho_y & \frac{c_4 u}{\rho} \\ c_3 - \frac{c_4 v^2}{\rho^2} + c_6\rho & 0 & \frac{2c_4 v}{\rho} + c_5\rho_y \end{pmatrix} = \begin{pmatrix} a & 0 & b \\ e & g & f \\ c & 0 & d \end{pmatrix}. \end{aligned} \tag{2.4.53}$$

So the eigenvalues of B have the same expressions as those of A.

In summary, we will initialize the density ρ and orientation θ in the 2D domain described above. Thus $\Omega_1 = \cos\theta$ and $\Omega_2 = \sin\theta$ and we can compute $Q_{i,j}^0$ for $i = 0, 1..n_x + 1$, $j = 0, 1..n_y + 1$.

Given the information on timestep n, the scheme (2.4.44) is firstly applied for $i = 1, 2..n_x$ and $j = 1, 2..n_y$, i.e. the inner grid points in the domain.

For a periodic boundary condition, where $Q(0, y) = Q(L_x, y)$ and $Q(x, 0) = Q(x, L_y)$, We will update the boundary as $Q_{0,j}^{n+1} = Q_{n_x,j}^{n+1}$, $Q_{n_x+1,j}^{n+1} = Q_{1,j}^{n+1}$, $Q_{i,0}^{n+1} =$

Q^{n+1}_{i,n_y}, $Q^{n+1}_{i,n_y+1} = Q^{n+1}_{i,1}$. The derivatives $Q_{x}{}^{n}_{0,j}, Q_{x}{}^{n}_{n_x+1,j}$ and $Q_{y}{}^{n+1}_{i,0}$, $Q_{y}{}^{n+1}_{i,n_y+1}$ can be obtained in the same manner.

For a Neumann homogeneous boundary condition, where $Q_x(0,y) = Q_x(L_x,y) = 0$ and $Q_y(x,0) = Q_y(x,L_y) = 0$, we will update the boundary as $Q^{n+1}_{0,j} = Q^{n+1}_{1,j}$, $Q^{n+1}_{n_x+1,j} = Q^{n+1}_{n_x,j}$, $Q^{n+1}_{i,0} = Q^{n+1}_{i,1}$, $Q^{n+1}_{i,n_y+1} = Q^{n+1}_{i,n_y}$. And the derivatives at the outer grid points are given as $Q_{x}{}^{n}_{0,j} = Q_{x}{}^{n}_{n_x+1,j} = 0$ and $Q_{y}{}^{n+1}_{i,0} = Q_{y}{}^{n+1}_{i,n_y+1} = 0$.

After computing Q^{n+1}, we can normalize Ω and proceed for the next time step.

Finally we will briefly discuss about the determination of the coefficients c_1 to c_7 as in (2.2.117) to (2.2.124). While Φ, k_1 and k_0 can be computed by direct integration, the rest of the terms involve the computations of h, which is defined as $g(\cos(\theta))/\sin(\theta)$. Note that in 2D, the generalized collision invariant $\psi(\theta) = g(\cos(\theta))$ satisfies (2.2.70), and this can be written as

$$\partial_\theta(e^{\frac{\cos\theta}{d}}\partial_\theta\psi) = \sin\theta e^{\frac{\cos\theta}{d}}. \tag{2.4.54}$$

This is an ODE which can be solved analytically by direct integration with respect to θ and together with the boundary condition $\psi(0) = \psi(2\pi)$. We can get

$$\psi = -d\theta + \frac{2\pi d\int\limits_0^\theta e^{-\frac{\cos x}{d}}dx}{\int\limits_0^{2\pi} e^{-\frac{\cos x}{d}}dx}. \tag{2.4.55}$$

This expression can be evaluated using trapezoidal rule numerically.

2.4.4 Numerical methods for particle model from the Navier-Stokes system

In this section we discuss the numerical methods by the particle model discretized using the SPH approach. The model is described by (2.3.51) to (2.3.55). Since it is already a full description of the microscopic model. Therefore the particle initialization and GPU parallelization is the same as the modified Vicsek model.

Note that we do not need the orientation in this case and the speed of each particle is adjusted by the self propulsion force. In addition, the “mass” and density need to be initialized. The “mass” of each particle is given by $\frac{1}{N} \times \rho_{\text{average}} \times |\Omega|$, where Ω is the volume of the domain.

Integration

Note that the force is decomposed into interaction force and self-propulsion force, where the sole purpose of the self-propulsion force is to adjust the speed of individual particles. A time splitting scheme is then applied for the integration of the system.

For each time step, the equations are split into two parts. The first part is

$$\frac{dv}{dt} = \sum_j F_{rij} + \sum_j F_{aij}. \tag{2.4.56}$$

The second part is

$$\frac{dv}{dt} = F_{p_i}. \tag{2.4.57}$$

In each time step, we solve the first part. In order to compute F_r, we need to compute ρ first. This is based on (2.3.55). We collect the contribution from all the neighbouring particles and the neighbour search algorithm is the same as the modified Vicsek model. After we compute all the individual density, we can get the interaction force and update

$$v_i^{n+\frac{1}{2}} = v_i^n + dt(\sum_j F_{rij} + \sum_j F_{aij}). \tag{2.4.58}$$

The second part is an ODE which can be solved exactly, thus we use the analytical solution of the ODE to update the velocity.

if $\rho < \rho_c$, ODE is $|v|' = -\lambda |v|^3$, we get $|v^{n+1}| = \dfrac{1}{\sqrt{2\lambda dt + k}}$, $k = \dfrac{1}{|v^{n+\frac{1}{2}}|^2}$.

Similarly,

if $\rho > \rho_c$, ODE is $|v|' = \lambda(c - |v|^2)|v|$, where $c = v_0^2(1 - \dfrac{\rho_c + \rho_0}{\rho + \rho_0})^2$.

If $c > |v^{n+\frac{1}{2}}|^2$ we get $|v^{n+1}| = \dfrac{\sqrt{c}e^{c(\lambda dt + k)}}{\sqrt{1 + e^{2c(\lambda dt + k)}}}$, $k = \dfrac{1}{c}\ln(\dfrac{|v^{n+\frac{1}{2}}|}{\sqrt{c - |v^{n+\frac{1}{2}}|^2}})$.

If $c < |v^{n+\frac{1}{2}}|^2$ we get $|v^{n+1}| = \dfrac{\sqrt{c}e^{c(\lambda dt + k)}}{\sqrt{e^{2c(\lambda dt + k)} - 1}}$, $k = \dfrac{1}{c}\ln(\dfrac{|v^{n+\frac{1}{2}}|}{\sqrt{|v^{n+\frac{1}{2}}|^2 - c}})$.

Boundary Condition

For a periodic boundary condition, the treatment is exactly the same as the modified Vicsek model. However, when a wall is present, the boundary condition maybe complicated. Since the development of SPH, the treatment for boundary conditions include the boundary force method [35], the dummy particle method [76] and the mirror particle method [90]. Using the boundary force method, it is almost impossible to treat some standard boundary conditions such as homogeneous Neumann boundary condition.

In the dummy particle method, several extra layers of particles are placed inside the wall. One layer of particles is placed on the boundary with equal distances between neighbours. More than one layer of boundary particles are required for the correction of density summation near the boundary.

The density of the dummy particles needs to be chosen carefully to match the density of the fluids. And for a curved domain or near a corner, the placement of the dummy particles needs to be designed. Dummy particles are easy to implement in the sense that they are set up at the beginning and do not move for a fixed domain.

To implement a Neumann boundary condition, slip or non-slip, we will use the mirror particle method. The boundary condition is $\frac{\partial v}{\partial \vec{n}} = 0$, where $\vec{n}$ is the outward normal of the domain. At each time step, mirror particles are generated as the reflection of the real fluid particles. Note that mirror particles are only required for a particle which is near the wall, with a distance to wall smaller than R, where R is the interaction radius.

We consider 2D domains for simplicity. For a rectangle domain, if a fluid particle is near a corner, an extra mirror particle is created as a point reflection to the corner. This is to prevent the density deficiency.

For a circle domain centered at the origin with radius L, consider a particle i near the boundary, a set of the mirror particles are generated for each fluid particle in the neighbourhood of particle i. This set of mirror particles only interact with particle i but not the others. Suppose the position of the ith particle is $X_i = (\cos\theta, \sin\theta)$, for each particle j near particle i at position X_j, the position of the mirror particle is given as

$$\tilde{X}_j = R(\theta)(2(L,0) - R(-\theta)X_j). \tag{2.4.59}$$

Here $R(\theta)$ is the rotation matrix for an anticlockwise rotation θ.

The velocity of the mirror particles require special treatment based on the boundary condition. Considering a mirror particle of the fluid particle j with velocity v_j, we firstly decompose v_j into the tangential and normal components. Here $v_j^{\text{normal}} = (v_j \cdot \vec{n})\vec{n}$ and $v_j^{\text{tangential}} = v_j - v_j^{\text{normal}}$. For a Neumann boundary condition, the mirror particle is given the velocity $\tilde{v}_j = v_j^{\text{tangential}} - v_j^{\text{normal}}$. If a non-slip boundary condition is used instead which require $v = 0$ at the boundary, the mirror particle is given the velocity $\tilde{v}_j = -v_j^{\text{tangential}} - v_j^{\text{normal}}$ instead. The negative sign for the tangential force makes sure the velocity along the tangential direction is 0 once the alignment force is applied.

2.5 Numerical results

The goal of this section is to provide the numerical results which would validate the numerical schemes described in the previous section.

2.5.1 Microscopice modified Vicsek model

We firstly present the particle simulations for the Microscopic Modified Vicsek Model. 2×10^4 particles are simulated in a 2D box domain with periodic boundary condition in both directions. The parameters used are:

$$v_0 = 1, \quad \mu = 50, \quad \alpha = 1, \quad D = 0.1, \quad L_x = 1, \quad L_y = 1. \tag{2.5.1}$$

We can see the formation of clusters after some time (Figure 2.5). To compare with the Vicsek model and study the effect of the repulsive force, we present the result for different μ at time $T = 1.0$, other parameters are fixed as the previous test (Figure 2.6).

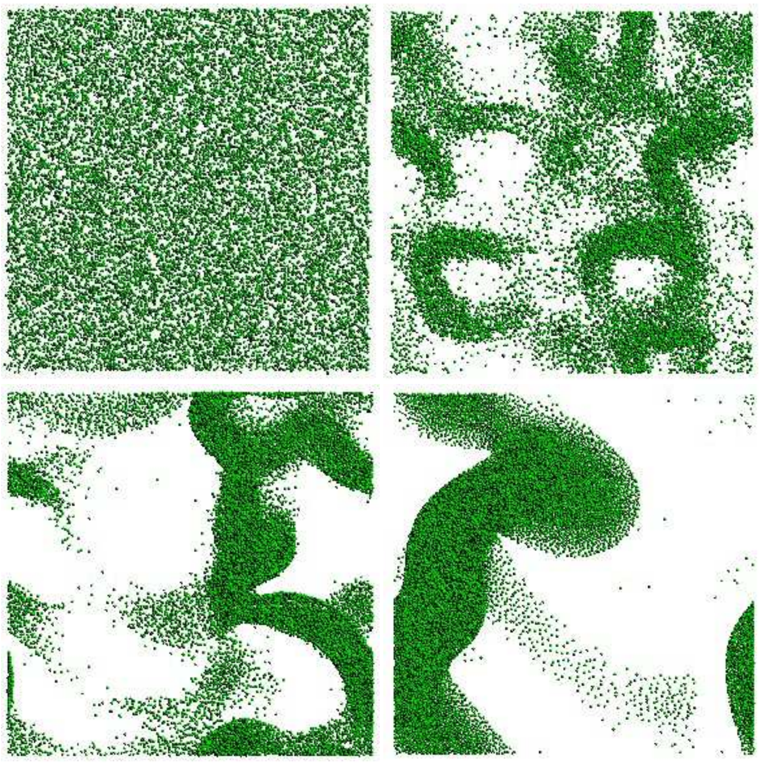


Figure 2.5: Simulation for 2×10^4 particles at time $T = 0, 0.5, 1.0, 3.0$ respectively.

2.5.2 Comparison between the microscopic and macroscopic model

In the microscopic Vicsek model with repulsion, we choose the potential kernel as

$$\phi(x) = \begin{cases} \frac{|x|^2}{r} - 2|x| + r, \text{ if } |x| \leq r, \\ 0, \text{ if } |x| > r. \end{cases} \tag{2.5.2}$$

With $r = R = 1$, which are the interaction radius for repulsion and alignment respectively before scaling, we can compute in dimension two:

$$\Phi = \int \phi dx = \frac{\pi}{6}, \quad k_1 = \frac{1}{4}\int K(|z|)|z|^2 dz = \frac{\pi}{8}, \quad \text{and } k_0 = \int K(|z|)dz = \pi. \tag{2.5.3}$$

The other parameters, which are fixed for all simulations if not differently stated

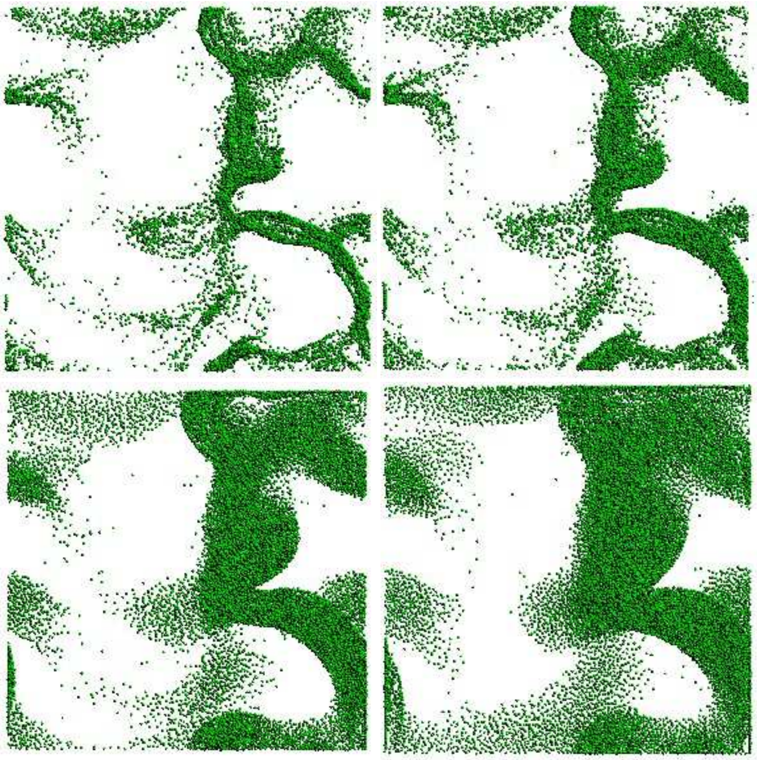


Figure 2.6: Simulation for 2×10^4 particles at time $T = 1$ with $\mu = 0, 10, 100, 300$ respectively. The initial orientation is randomized and the same initial data is used for all the tests.

are:

$$v_0 = 1, \quad \mu = \frac{1}{2}, \quad \alpha = 1, \quad D = 0.1, \quad L_x = 10, \quad L_y = 10. \tag{2.5.4}$$

After numerical computation of the coefficients, we have that in (2.4.23), (2.4.24):

$$c_1 = 0.9486, \quad c_2 = -0.5236, \quad c_3 = 0.1, \quad c_4 = 0.8486, \tag{2.5.5}$$

$$c_5 = -0.5236, \quad c_6 = 0.4967, \quad c_7 = -0.1186. \tag{2.5.6}$$

Note that the microscopic model is the continuum limit in the sense that $\epsilon \to 0$.

The scaled individual based model reads as

$$\frac{dX_i}{dt} = v_i, \quad v_i = \omega_i - \nabla_x \Phi(X_i(t), t), \tag{2.5.7}$$

$$d\omega_i = P_{\omega_i^\perp}(\frac{1}{\epsilon}\bar{\omega}(X_i(t), t)dt + \alpha v_i dt + \sqrt{2D}\epsilon dB_t^i), \tag{2.5.8}$$

$$\Phi(x, t) = \frac{1}{\epsilon^2 N}\sum_{j=1}^{N} \nabla\phi(\frac{|x - x_j|}{\epsilon}), \tag{2.5.9}$$

$$\bar{\omega}(x, t) = \frac{J(x, t)}{|J(x, t)|}, J(x, t) = \frac{1}{N}\sum_{j=1}^{N} K(\frac{|x - X_j|}{\sqrt{\epsilon}})\omega_j. \tag{2.5.10}$$

Only 2D models are compared. In the microscopic model, once the position of the particles are determined, we can compute the density by distributing the weight of a particle onto the neighbouring four grid points. Consider a particle within a 2D grid box. The horizontal line and the vertical line divide the box into four rectangles, and the weight contributed to each grid point is proportional to the area of the rectangle which is furthest away from the grid point. After normalization, we can get the density profile of a particle system.

Similarly for a given density, to initialize the particles, we allocate $N \times \rho$ particles in the region with density ρ randomly. The positions are drawn with a uniform distribution.

Furthermore, an average is taken over several simulation to reduce the statistical error since white noise is implemented in the microscopic model. $N = 10^5$ particles are used in each simulation. The interaction radius in the microscopic model are fixed to $r = 0.0625$ for the repulsive range and $R = 0.25$ for the alignment interaction range.

Two problems are considered. We firstly consider the Riemann problem. The initial data is

$$(\rho_l, \theta_l) = (0.0067, 0.7), \quad (\rho_r, \theta_r) = (0.0133, 2.3). \tag{2.5.11}$$

Periodic boundary condition is used both in the direction of x and y. The time step is $\Delta t = 0.01$ and the mesh size is $\Delta x = \Delta y = 0.25$. The final time is chosen that $T = 1$. We firstly show the L^1 norm of the relative error in Figure 2.7. As shown in the figure, as we increase the number of realizations in the microscopic model, or decrease ϵ, the error decreases and the microscopic model provides a good approximation for the individual based model.

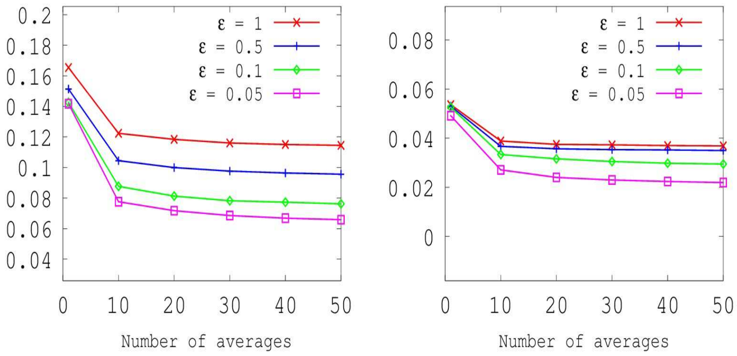


Figure 2.7: Relative error between the macroscopic and the microscopic model for density (left) and θ (right) as a function of the number of averages for different values of ϵ. The error decreases with both decreasing ϵ and increasing number of averages, showing that the microscopic model provides a valid approximation of the individual based model for ρ and θ.

Note that in the Riemann problem, the result is independent of y. Therefore in Figure 2.8 we show the density ρ and the flux direction θ for the same Riemann problem along the x-axis for $\epsilon = 0.05$ at time $T = 1s$. Again we can see that these two models produce close results. The difference maybe due to the different schemes used for different models.

Another problem considered for the comparison is the Taylor-Green vortex problem. This initial condition is chosen trying to mimic the behaviour of the sperm suspension. The initial data is given as $\rho_0 = 0.01$ being a constant density, and

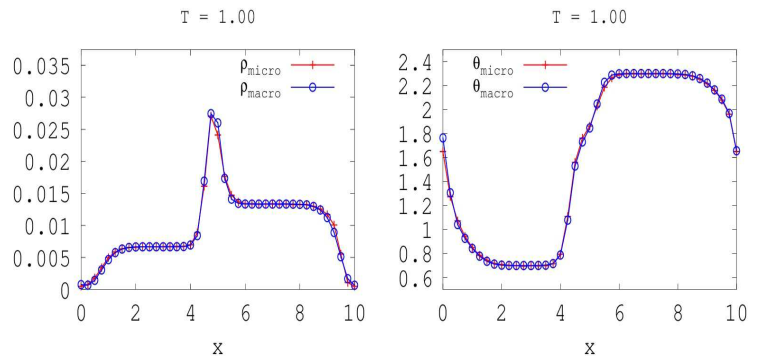


Figure 2.8: Solution of the Riemann problem along the x-axis for the macroscopic model (blue line) and for the microscopic model with $\epsilon = 0.05$ (red line) at $T = 1$.

$\Omega_0(x,y) = \frac{\tilde{\Omega}_0(x,y)}{|\tilde{\Omega}_0(x,y)|}$ is a normalization of the Green vortex $\tilde{\Omega} = (\tilde{\Omega}_1, \tilde{\Omega}_2)$ given as

$$\tilde{\Omega}_1(x,y) = \frac{1}{3}\sin(\frac{\pi}{5}x)\cos(\frac{\pi}{5}y) + \frac{1}{3}\sin(\frac{3\pi}{10}x)\cos(\frac{3\pi}{10}y) + \frac{1}{3}\sin(\frac{\pi}{2}x)\cos(\frac{\pi}{2}y), \tag{2.5.12}$$

$$\tilde{\Omega}_2(x,y) = -\frac{1}{3}\cos(\frac{\pi}{5}x)\sin(\frac{\pi}{5}y) - \frac{1}{3}\cos(\frac{3\pi}{10}x)\sin(\frac{3\pi}{10}y) - \frac{1}{3}\cos(\frac{\pi}{2}x)\sin(\frac{\pi}{2}y). \tag{2.5.13}$$

The parameters are chosen as $\epsilon = 0.05, r = 0.04, R = 0.3$, finial time $T = 0.6$ and number of particles $N = 10^5$. Again a periodic boundary condition is used. The following figures show the density comparison together with the orientation.

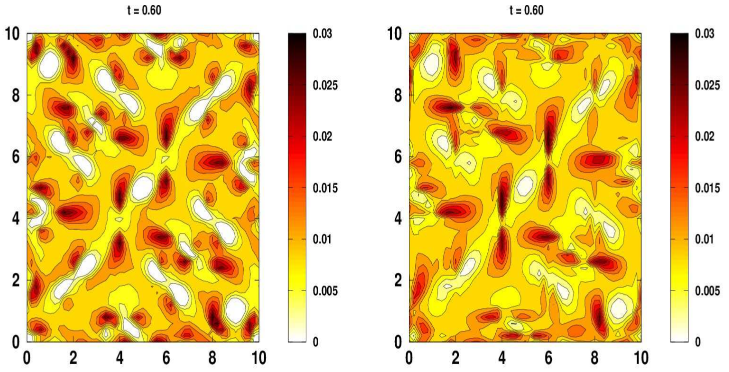


Figure 2.9: Density ρ for the Green Taylor Problem at $T = 0.6$. Left:microscopic model. Right:macroscopic model

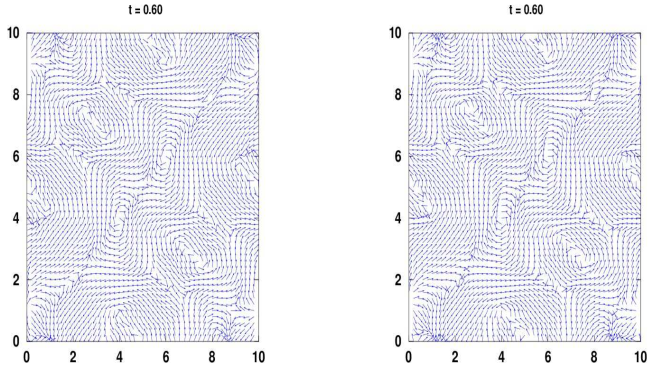


Figure 2.10: Mean orientation Ω for the Green Taylor Problem at $T = 0.6$. Left:microscopic model. Right:macroscopic model

2.5.3 Particle model from the Navier-Stokes system

Finally we present the particle simulation with the interaction rules derived from the Navier-Stokes equations. We could expect a more homogeneous structure due the derivation of the model. We firstly show the 2D particles with 2×10^4 particles in different domains. (figure 2.11)

The parameter used are:

$$dt = 0.001, v_0 = 5, p_0 = 5, \nu = 0.1 \text{ and } \lambda = 2000. \tag{2.5.14}$$

Note that the large λ results in a constant speed while the stability is ensured in our spilitting scheme. Note that in a domain with a periodic boundary condition or in the annulus domain, we can observe collective motion towards a certain direction, whereas in a confined domain (the circle case), the particles form a homogeneous structure.

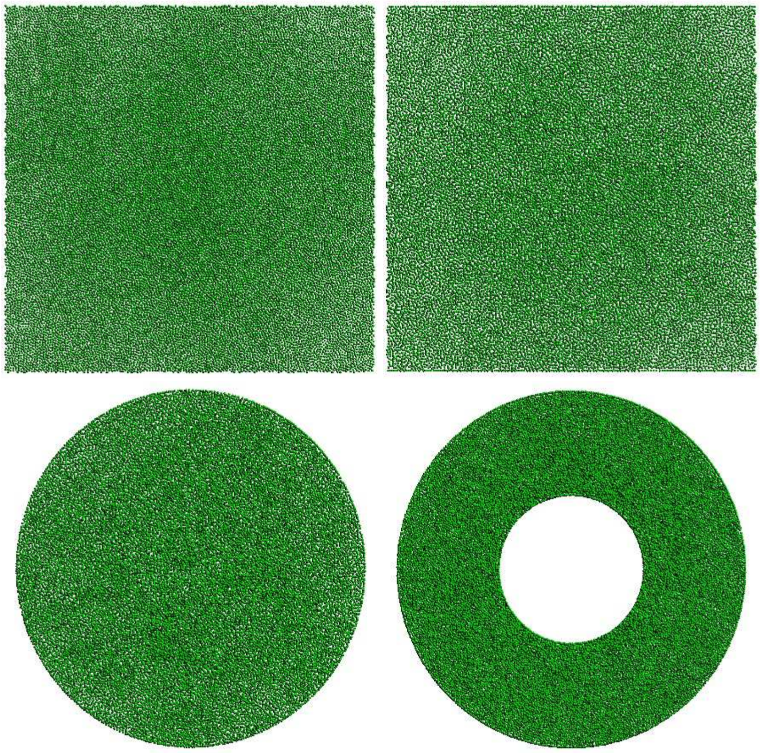


Figure 2.11: Particle simulations in 2D domains. Top left: rectangle domain $L_x = L_y = 1$ with periodic boundary condition. Top right: rectangle domain $L_x = L_y = 1$ with periodic boundary condition in x direction and Neumann boundary condition in y direction. Bottom left: circle domain, radius=1, Neumann boundary condition. Bottom right: annulus domain, outer radius=1, inner radius=0.4, Neumann boundary condition.

Chapter 3

Models and Methods for Collective Motion of Polymers

3.1 Existing models

The above particle models add specific interaction rules for individuals. However, in reality, there is no way for animals or micro-organisms to make accurate measurement for an alignment. Understanding the mechanisms such that cellular organization can be achieved still remains as an open question. It is found that chemoattractant can guides cell motion and leads to complex patterns in some systems [13]. For example E.coli as well as amoebae like D.discoideum can move due to the chemical sigmals produced by other cells [12]. [45] showed that in some bacteria, there is not evidence of the chemoattractant mechanism, as the physical contact between bacteria or sperms play a role for the alignment of such suspensions. As the volume exclusion effect becomes significant, we should no longer model the individual polymer by points without a volume. [44] has shown that the shape of micro-swimmers is essential for individual swimming of certain cell types, and it remains unclear how the shape will affect the collective behaviour.

These interactions result in a large varieties of pattern. The easiest observable patterns in a bacteria suspension are large scale vortices and jets with size and speeds much more than those swimming individuals. Other observations include the formation of spatial inhomogeneities, locally correlated motions.

Different models are available with different shapes and large scale simulations have been carried out to model the dynamics of the particles directly. A “minimal” swimmer model can be found in [60]. Each polymer is represented as a rigid dumbbell. The dumbbell exert equal and opposite force on the fluid, creating a far field dipole induced by self-propulsion. In [56], the dumbbell model is generalized to an ellipsoid model. A similar description of self-propulsion was used in recent numerical simulations in two dimensions by Decoene.

Another different class of models consider the propulsion not as the result of flagellum on the fluid but rather as a prescribed surface slip velocity. This is so called the “squirmer” model, which was initially developed by Lighthill [78] and Blake. The classical squirmer model prescribes the velocity on a sphere for motion. The squirmer model is then refined by Pedley and coworkers, who generalize the squirmer model to anisotropic shapes, for example, spheroids, and also with more complex boundary conditions, where the tangential stress is prescribed instead of slip.

Such particle-based models are invaluable since they result in qualitative agreement with experiments and can be analysed statistically.

In order to understand the hydrodynamic interactions, computational fluid dynamics are involved. For a viscose fluid with low Reynold number, Stokesian dynamics can be applied. The key feature is that it can handle a large number of particles and produce fairly accurate result. The particle motion is described by the coupled N body Langevin equation in the following form:

$$m\frac{dU}{dt} = F^H + F^P + F^B. \tag{3.1.1}$$

U is the translational/rotational vector with dimension $6N$, the three terms in the right hand side represent hydrodynamic forces, non hydrodynamic forces and noise. The relation between the forces and the motion is assemble in the grand mobility matrix resulting in a large linear system.

To study fluid particle interaction, we can also adopt the boundary integral method. This method allows the computation of flows associated with complex geometries. It discretize the integral form of the original Navier-Stokes equation, and it reduces the computation from a 3D domain to a 2D domain on a surface.

3.2 Self-propelling polymer model

In this section we will consider each individual polymer as rigid linked spheres with radius R moving in a 3D domain. Steric rod rod interactions are described by a segment model, where each rod is partitioned into n equidistant segments, or spheric beads. All the segments of the rods interact each other.

Each individual rod α has the mass of nm with m being the mass of individual beads. The center of mass has position x^α and the velocity of the center of mass is v^α. The orientation of the rod is an unit vector u^α.

Consider a pair of rods α and β. Their pairwise potential is given by

$$\Phi^{\alpha\beta} = \Phi_0 \sum_{i=1}^{n} \sum_{j=1}^{n} \Phi(r_{ij}^{\alpha\beta}) \tag{3.2.1}$$

Here $\Phi(r)$ is a repulsive potential same as the modified Vicsek model, Φ_0 is a constant and $r_{ij}^{\alpha\beta}$ is the distance between the ith segment of rod α, and the jth segment of rod β, given as

$$r_{ij}^{\alpha\beta} = |\Delta r^{\alpha\beta} + (l_i u^\alpha - l_j u^\beta)|, \tag{3.2.2}$$

where $\Delta r^{\alpha\beta}$ is the distance between the center of mass of the two rods, and $l_i = (-(n-1) + 2(i-1))R$.

Each individual rod also experience a self propelling force f which would relax its current velocity towards its current orientation and a fixed speed v_0

$$f^\alpha = \mu(v_0 u^\alpha - v^\alpha) \tag{3.2.3}$$

This gives us a complete description of a self propelling rod.

3.2.1 Numerical methods

The aim of this section is to provide the detailed method of simulating a large system of self propelling rigid rods. For each rod α, we drop the superscript α for simplicity, the simulation basics at each time step involves the position of center of mass $x(t)$, the rotation matrix $R(t)$, the momentum $P(t)$ as well as the angular momentum $L(t)$.

Here $x(t)$ and $R(t)$ describes the transform from the translation and rotation from the body space to the world space. $x(t)$ is the center of mass and $R(t)$ is the rotation matrix. In the body space, we assume the center of mass is positioned at the origin and the rod is aligned with the x axis, with the orientation of the rod point towards the positive x axis. For any bead i at x_{i0} in the body space, the corresponding position in the world space is $x_i(t) = R(t)x_{i0} + x(t)$. In this way, we can see that the orientation $u(t)$ can be written as $R(t)(1,0,0)^T$ which is the first column of $R(t)$. Finally the position of each bead is given as $x_i(t) = x(t) + l_i u$.

The translating is easy to deal with. The governing equation for $x(t)$ is $\frac{dx(t)}{dt} = v(t)$. The velocity $v(t)$ is related to the linear momentum $P(t) = Mv(t)$, and the governing equation for $P(t)$ is $\frac{dP(t)}{dt} = F(t)$. Here $F(t)$ is the total force acting on the rod, which include the interaction forces summed over all the beads together with the self propelling force. Note that the only algorithm that determines the motion is the steric interaction.

Although there is no interaction among individual beads within the same rod, we still can take the interaction into consideration, since the equal and opposite forces can cancel each other. Therefore, for the computation of the total force on each bead F_i, we can consider the individual beads as a particle system and compute the all interaction forces. GPU parallelization can be applied, where each tread store the data for individual beads, instead of individual rods. Finally for each rod, we have $F = \sum F_i + f$, where f is the self propelling force.

The rotation needs to be handled carefully. To find the time derivative of the rotation matrix $R(t)$, we consider a vector $r(t)$ in world space. We have the following indentity: $\frac{dr(t)}{dt} = \omega(t) \times r(t)$, where $\omega(t)$ is the angular velocity, a vector which the rigid body rotates about. Therefore by writing the rotation matrix as $R = (R_1, R_2, R_3)$, we have

$$\frac{dR(t)}{dt} = (\omega(t) \times R_1, \omega(t) \times R_2, \omega(t) \times R_3) = \omega(t)^* R(t), \tag{3.2.4}$$

with a^* for a vector a is defined by the matrix:

$$a^* = \begin{pmatrix} 0 & -a_z & a_y \\ a_z & 0 & -a_x \\ -a_y & a_x & 0 \end{pmatrix}. \tag{3.2.5}$$

To compute $\omega(t)$, we consider the angular momentum. We have the total angular momentum $L(t) = I(t)\omega(t)$, with $I(t)$ being a 3×3 matrix called the inertia tensor. This tensor is a 3D assemble of the scalar moment of inertia in the case where the axis of rotation is fixed. $I(t)$ describes how the mass in a body is distributed relative to the center of mass. By computing $I(t)$ and $L(t)$, we can get $\omega(t) = I(t)^{-1}L(t)$.

Analogue to the relation $\frac{dP(t)}{dt} = F(t)$, the relation between $L(t)$ and the total torque is $\frac{dL(t)}{dt} = \tau(t)$. The total external torque is $\tau(t) = \sum \tau_i(t) = \sum (x_i(t) - x(t)) \times F_i(t)$. Again the virtual force added among particles within the same rod does not affect $\tau(t)$.

Finally the inertia tensor has the following form, denoting $r_i(t) = x_i(t) - x(t)$:

$$I(t) = \sum_i \begin{pmatrix} m_i(r_{iy}^2 + r_{iz}^2) & -m_i r_{ix} r_{iy} & -m_i r_{ix} r_{iz} \\ -m_i r_{iy} r_{ix} & m_i(r_{ix}^2 + r_{iz}^2) & -m_i r_{iy} r_{iz} \\ -m_i r_{iz} r_{ix} & -m_i r_{iz} r_{iy} & m_i(r_{ix}^2 + r_{iy}^2) \end{pmatrix}. \tag{3.2.6}$$

In actual implementation, this can be written as

$$I(t) = R(t)I(0)R(t)^T, \quad I(0) = \sum_i m_i((x_{i0}^T x_{i0})Id - x_{i0}x_{i0}^T). \tag{3.2.7}$$

As a result, we have

$$I(t)^{-1} = R(t)I(0)^{-1}R(t). \tag{3.2.8}$$

Note that $I(0)^{-1}$ is independent of time and can be used through out the simulation.

We can put everything in a state vector $X(t)$ and summarize our result.

$$\text{Let } X(t) = \begin{pmatrix} x(t) \\ R(t) \\ P(t) \\ L(t) \end{pmatrix}, \quad \text{then } X'(t) = \begin{pmatrix} v(t) \\ \omega(t)^* R(t) \\ F(t) \\ \tau(t) \end{pmatrix}. \tag{3.2.9}$$

An Euler scheme is used for time integration: $\frac{X^{n+1}-X^n}{dt} = X'^n$ For each time step, we firstly compute all the force on individual beads F_i^n. Only for F_i^n, the computations for individual beads are done in parallel. For the rest, only the computations for individual rods can be done in parallel.

With F_i^n, we compute the total force F^n and torque τ^n. Together with

$$v^n = \frac{P^n}{M}, \quad I^n = R^n I(0) R^{nT}, \quad \omega^n = (I^n)^{-1} L^n, \tag{3.2.10}$$

we can get X'^n and proceed with the simulation.

To reduce the numerical error accumulated in the computation of $R(t)$, a unit quaternion $q = [s, v] = [s, v_x, v_y, v_z]$ representing a rotation of s radians about an axis v can be used to store the rotation. We have

$$q'(t) = \frac{\omega(t)}{q(t)}, \tag{3.2.11}$$

which is analogue to $R'(t) = \omega^* R(t)$

Now define the quaternion multiplication by:

$$[s_1, v_1][s_2, v_2] = [s_1 s_2 - v_1 \cdot v_2, s_1 v_2 + s_2 v_1 + v_1 \times v_2]. \tag{3.2.12}$$

If q_1, q_2 are rotations, $q_2 q_1$ indicates a composite rotation q_1 followed by q_2. Therefore we have $q^{n+1} = dq^n \times q^n$.

From (3.2.11), following by a normalization of q, we get the expression for dq which is

$$dq = [\cos(\frac{\theta}{2}), a \sin(\frac{\theta}{2})]. \tag{3.2.13}$$

Here $a = \frac{\omega}{|\omega|}$, $\theta = |\omega dt|$, and finally, for the computation of ω, we still need R^n in terms of $q^n = [s^n, v^n]$. Dropping the superscript n, we have

$$R = \begin{pmatrix} 1 - 2v_y^2 - 2v_z^2 & 2v_x v_y - 2sv_z & 2v_x v_z + 2sv_y \\ 2v_x v_y + 2sv_z & 1 - 2v_x^2 - 2v_z^2 & 2v_y v_z - 2sv_x \\ 2v_x v_z - 2sv_y & 2v_y v_z + 2sv_x & 1 - 2v_x^2 - 2v_y^2 \end{pmatrix}. \tag{3.2.14}$$

This completes our time integration. Once we get the position and orientation of a rod, we can compute the position of all the beads on it. Note that the boundary conditions and grid generation are exactly the same as the modified Vicsek model.

3.2.2 Numerical results

In this section we show the simulation result of the self propelling rods. By changing the number of rods, the shape, the interaction force, as well as the shape of the domain, a large number of patterns can be observed. A dense suspension is crucial for the collective motion in most cases, and a laning phase can be observed for rigid rods with large aspect ratios. We also observe cluster formation near the boundary of circular domains.

Figure 3.1 shows the result of 2000 rods with $n = 3$ representing 6000 particles in different domains. Particle interaction radius is chosen as 0.05, the potential amplitude $\Phi_0 = 100$ and speed $v_0 = 0.5$.

Figure 3.2 show some selected scenarios varying the aspect ratio.

3.3 Polymer fluid interaction

In the previous sections we treat the suspension of micro swimmers as an active rigid polymers where the swimmer fluid interactions are ignored. However, in recent experiments carried out to understand the large scale structure that is obtained by individual swimming mechanism [26, 75], it is found that the hydrodynamics play an essential role and should not be easily ignored also. Complex flow structure such

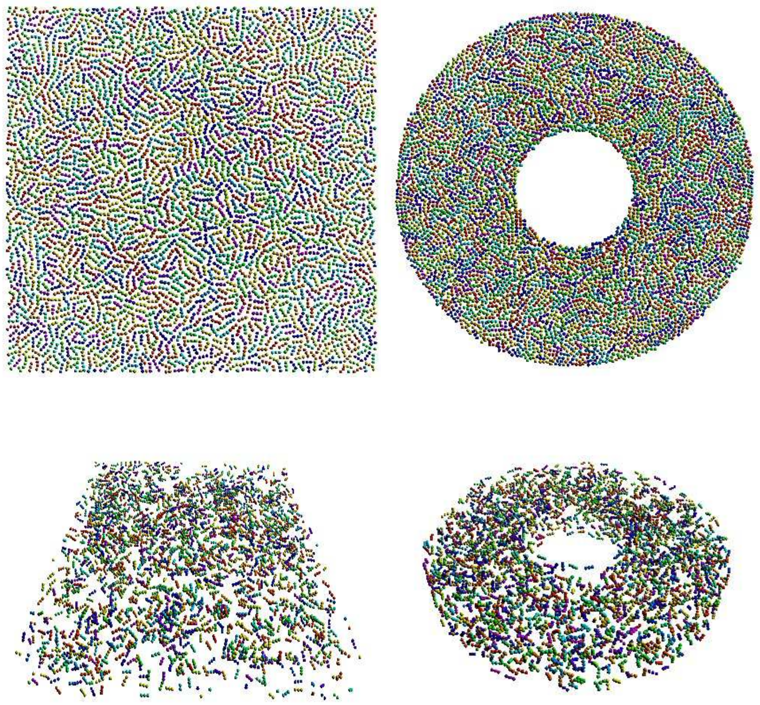


Figure 3.1: Rigid rod simulations. Top left: 2D rectangle domain $L_x = L_y = 1$ with periodic boundary condition. Top right: 2D annulus domain, outer radius=1, inner radius=0.3, repulsive boundary condition. Bottom figures show the corresponding 3D domain with $L_z = 0.3$.

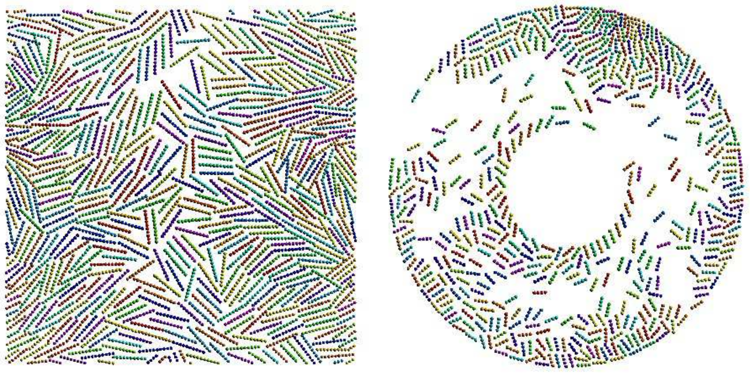


Figure 3.2: Rigid rod simulations. Left: n=10, 600 rods, highly local alignment observed for long rods. Right: n=4, 500 rods, alignment normal to the boundary observed near the boundary.

as large-scale vortices and jet can be induced. It is also found the speed of the flow can be much faster than each swimming individuals [83].

Therefore in this section we will consider rigid particles embedded in a Newtonian fluid using a finite element approach. For a direct simulation of particles immersed in fluid. One series of work begins with [65], followed by [66]. Their approach replies on a moving mesh which follows the fluid domain. Another method for fluid simulation extend the fluid domain to a fictitious domain, thus have the name of fictitious domain method or domain embedding method. These methods use a Cartesian mesh which would cover the whole domain (both fluids and particles). Rigid motion can be enforced using a Lagrange Multiplier, for example, in [97], the strain tensor is constraint to be zero. Rigid motion can also be enforced by penalty methods, where a penalty term for the strain tensor is applied.

The domain considered is a 2D domain $\Omega \in \mathbb{R}^2$, and we assume that it should be bounded, connected and regular. Polymers are modelled as rigid particles immersed in the fluid. Self propulsion is modelled as propelling force along their axis of motion, and volume exclusion is considered by repulsive force. The polymers are denoted by P_i, $i = 1..N$ and the whole rigid domain is $P = \cup_i P_i$, The domain $\Omega \backslash \bar{P}$ is the fluid domain with Newtonian fluid. Whereas in [79], the particles are rigid spheres, we will model swimming polymers as rigid ellipses.

To model the force, we will investigate the mechanism by which the polymer may propagate at low Reynolds numbers. In many cases, such as sperms or Escherichia coli, the thrust comes from the rear part of the body. For example, E coli swims using five to six flagella that are rotated by molecular motors embedded in the cell membrane. This kind of swimmer are named “pushers”. In contrast, another type of swimmer are “pullers”, example include algal species C. reinhardtii. Pullers perform a breaststroke-like motion to “pull” liquid from the front and force it to the sides of the micro-organism. In short, both types of micro-swimmer can generate a force dipole with opposite directions. For pushers, we will model it by applying two opposite forces. The support of the forces form a dumbbell, which is composed of a large sphere at the head and a small sphere at the tail. The center of the spheres are chosen to be the foci of the ellipse as a natural choice. Forces are applied to be a constant function over the support. The force on the head has the same direction as the ellipse where the force on the tail has the opposite direction.

The repulsive forces are also considered on the foci of the ellipse. We can treat the 2 foci as particles and apply a repulsive force when particles from different ellipses come close to each other. The potential is chosen in the same way as the modified Vicsek model.

For each particle i, we denote its center of mass by x_i, velocity by v_i and orientation by the angle θ_i. The angular velocity is denoted by ω_i which is $\frac{d\theta_i}{dt}$. For the density and force on the particle i, we denote by ρ_i and f_i whereas for the density and external force on the fluid, we denote by ρ_f and f_f.

Firstly we consider the fluid region. In 2D, we want to find the velocity $u = (u1, u2)$ and the pressure field p in $\Omega\backslash\bar{P}$. The governing equations are

$$\rho_f \frac{Du}{Dt} - \mu\Delta u + \nabla p = f_f \text{ in } \Omega\backslash\bar{P}, \tag{3.3.1}$$

$$\nabla \cdot u = 0 \text{ in } \Omega\backslash\bar{P}, \tag{3.3.2}$$

$$u = 0 \text{ on } \partial\Omega. \tag{3.3.3}$$

The motion of each particle is then governed by

$$u = v_i + \omega_i \times (x - x_i)^{\perp} \text{ on } \partial P_i, \quad \forall i. \tag{3.3.4}$$

Next we consider the forces on the particles. Using Newton's second law, we can write

$$m_i \frac{dv_i}{dt} = \int_{P_i} f_i dx - \int_{\partial P_i} \sigma n ds, \quad \forall i, \tag{3.3.5}$$

$$J_i \frac{d\omega_i}{dt} = \int_{P_i} (x - x_i)^{\perp} \cdot f_i dx - \int_{\partial P_i} (x - x_i)^{\perp} \cdot \sigma n ds, \quad \forall i. \tag{3.3.6}$$

In summary, the model describes rigid ellipses immersed in fluid. The fluid domain satisfies the Navier-Stokes equation. The velocity of the particle and the fluid agrees on the surface of the particle.

3.3.1 Numerical methods

Now we discuss the numerical methods to solve the systems (3.3.1) to (3.3.6). Standard approaches would require a mesh discretization in the fluid region and leave the polymers as rigid bodies. This would cause remeshing every time step. In order to avoid remeshing, as developed in [69], we can introduce the following functional spaces to deal with the rigid constraints.

$$K_\nabla = \{u \in H_0^1(\Omega), \nabla \cdot u = 0\}, \tag{3.3.7}$$

$$K_P = \{u \in H_0^1(\Omega), \forall i \quad \exists (v_i, \omega_i) \in \mathbb{R}^2 \times \mathbb{R}, u = v_i + \omega_i (x - x_i)^\perp \quad a.e. \text{ in } P_i\}. \tag{3.3.8}$$

Basically, K_∇ is the space of divergence free functions on Ω and K_P is the space of functions on Ω that will not deform P. We can also write K_P in the following form

$$K_P = \{u \in H_0^1(\Omega), D(u) = 0 \quad a.e. \text{ in } P\}. \tag{3.3.9}$$

Note that the space K_P is dependent on P and can vary over time. Now we will find a solution u which is in $K_\nabla \cap K_P$. this solution is defined over the whole domain while the rigidity is guaranteed. Choosing $\tilde{u}$ as a test function and multiply the Navier Stokes equation by $\tilde{u}$ over the fluid domain $\Omega \backslash \bar{P}$. We can have

$$\int_{\Omega\backslash\bar{P}} \rho_f \frac{Du}{Dt} \cdot \tilde{u} dx - \mu \int_{\Omega\backslash\bar{P}} \Delta u \cdot \tilde{u} dx + \int_{\Omega\backslash\bar{P}} \nabla p \cdot \tilde{u} dx = \int_{\Omega\backslash\bar{P}} f_f \cdot \tilde{u} dx. \tag{3.3.10}$$

Using integration by parts,

$$\begin{aligned} &\int_{\Omega\backslash\bar{P}} \rho_f \frac{Du}{Dt} \cdot \tilde{u} dx + 2\mu \int_{\Omega\backslash\bar{P}} D(u) : D(\tilde{u}) dx \\ &- \int_{\Omega\backslash\bar{P}} p \nabla \cdot \tilde{u} dx - \int_{\partial(\Omega\backslash\bar{P})} \sigma n \cdot \tilde{u} dx = \int_{\Omega\backslash\bar{P}} f_f \cdot \tilde{u} dx. \end{aligned} \tag{3.3.11}$$

We can extend the result from $\Omega\backslash\bar{P}$ to Ω since we have $D(\tilde{u}) = 0$ in P, which implies that $\nabla \cdot \tilde{u} = 0$ in P. Therefore the second and third terms can be extended to Ω directly. The fourth term can be dealt with the fact $\tilde{u}$ is in K_P. Therefore we have

$$\forall i, \exists \tilde{v}_i, \omega_i \text{ such that } \tilde{u}(x) = \tilde{v}_i + \tilde{\omega}_i (x - x_i)^\perp \text{ in } P_i. \tag{3.3.12}$$

The relation as well as the boundary condition leads to

$$\begin{aligned}&\int_{\Omega\backslash\bar{P}} \rho_f \frac{Du}{Dt}\cdot\tilde{u}dx + \sum_{i=1}^{N} m_i \frac{dv_i}{dt}\tilde{v}_i dx + \sum_{i=1}^{N} J_i \frac{d\omega_i}{dt}\tilde{\omega_i} dx \\ &+ 2\mu\int_{\Omega} D(u) : D(\tilde{u})dx - \int_{\Omega} p\nabla\cdot\tilde{u}dx = \int_{\Omega} f\cdot\tilde{u}dx,\end{aligned} \tag{3.3.13}$$

where $f = f_f\chi_{\Omega\backslash\bar{P}} + \sum\limits_{i=1}^{N} f_i\chi_{P_i}$ Here χ is the indicator function for the domain.

Finally we have the following identity:

$$\forall i, m_i\frac{dv_i}{dt}\tilde{v}_i + J_i\frac{d\omega_i}{dt}\tilde{\omega}_i = \int_{P_i}\rho_i\frac{Du}{Dt}\cdot\tilde{u}. \tag{3.3.14}$$

In the end, we have the complete variational formulation:

$$\int_{\Omega}\rho\frac{Du}{Dt}\cdot\tilde{u}dx + 2\mu\int_{\Omega} D(u) : D(\tilde{u})dx - \int_{\Omega} p\nabla\cdot\tilde{u}dx = \int_{\Omega} f\cdot\tilde{u}dx, \forall\tilde{u}\in K_P, \tag{3.3.15}$$

$$\int_{\Omega} q\nabla\cdot udx = 0, \forall q\in L^2(\Omega). \tag{3.3.16}$$

Here $\rho = \rho_f\chi_{\Omega\backslash\bar{P}} + \sum\limits_{i=1}^{N}\rho_i\chi_{P_i}$.

To discretize the variational form, we use the method of characteristics to discretize the total derivative $\frac{Du}{Dt}$. Note that this material derivative is from the Lagrangian point of view, and we can define the characteristic trajectory X passing x at time t as

$$\frac{\partial X}{\partial\tau}(x,t,\tau) = u(X(x,t,\tau),\tau), X(x,t,t) = x, \tag{3.3.17}$$

and $\frac{Du}{Dt}$ can be discretized as

$$(\frac{Du}{Dt})^{n+1}(x) \approx \frac{u^{n+1}(x) - u^n(X(x,t_{n+1},t_n))}{\Delta t}. \tag{3.3.18}$$

Up to now, the formulation requires that the test functions are in the constrained functional space K_P, to ensure the rigid motion of the particles. In order to get rid of the constraint, we adopt the idea in [69]. The constraint can be relaxed by introducing a penalty term in the minimized functional, which is

$$\frac{1}{\epsilon}\int_{P^{n+1}} D(u^{n+1}) : D(u^{n+1})dx. \tag{3.3.19}$$

it is shown in [69], that as $\epsilon \to 0$, we will have $D(u^{n+1})|_{P^{n+1}} \to 0$ therefore u^{n+1} would tends to a rigid motion in P^{n+1}.

Put everything together, we can get the following scheme: For every time step, the velocity of a particle i is computed as

$$V_i^n = \frac{\int_{P_i^n} u^n dx}{\int_{P_i^n} 1dx} \tag{3.3.20}$$

The position of the center of mass is then

$$x_i^{n+1} = x_i^n + \Delta V_i^n, \tag{3.3.21}$$

and the density ρ is computed as

$$\rho^{n+1} = \rho_f \chi_{\Omega \bar{P}^{n+1}} + \sum_{i=1}^{N} \rho_i \chi_{P_i^{n+1}}. \tag{3.3.22}$$

Finally we solve the dicretized weak formulation, that is, find $u^{n+1} \in H_0^1(\Omega)$ and $p^{n+1} \in L^2(\Omega)$ such that

$$\begin{aligned} &\frac{1}{\Delta t}\int_\Omega \rho^{n+1}u^{n+1} \cdot \tilde{u}dx + (2\mu + \frac{2}{\epsilon})D(u^{n+1}) : D(\tilde{u})dx - \int_\Omega p^{n+1}\nabla \cdot \tilde{u}dx \\ &= \frac{1}{\Delta t}\int_\Omega (\rho^n u^n) \circ X^n \cdot \tilde{u}dx + \int_\Omega f^{n+1} \cdot \tilde{u}dx, \forall \tilde{u} \in H_0^1(\Omega), \end{aligned} \tag{3.3.23}$$

$$\int q\nabla \cdot u^{n+1}dx = 0, \forall q \in L^2(\Omega). \tag{3.3.24}$$

Here $(\rho^n u^n) \circ X^n = \rho^n u^n$ at $X^n(x)$ where $X^n(x)$ is an approximation of $X(x, t_{n+1}, t_n)$, on the characteristic trajectory. This term takes convection into consideration.

With this weak formulation, any finite element solvers can be applied to solve the problem. We use Freefem to solve this problem. The domain Ω is triangulated into T^h using a built-in Delaunay-Voronoi algorithm. We use $P2$ finite element space for velocities and $P1$ finite element space for pressures.

3.3.2 Numerical results

We show the numerical result for few particles within a unit square domain with zero Dirichlet boundary condition. We choose $\mu = 0.01, \epsilon = 1 \times 10^{-8}$, $dt = 0.05$,

$\rho_f = 1$ and $\rho_p = 5$. The shape of the ellipse is determined as the sum of the distance to the 2 foci is 0.25 while the foci are separated by 0.2.

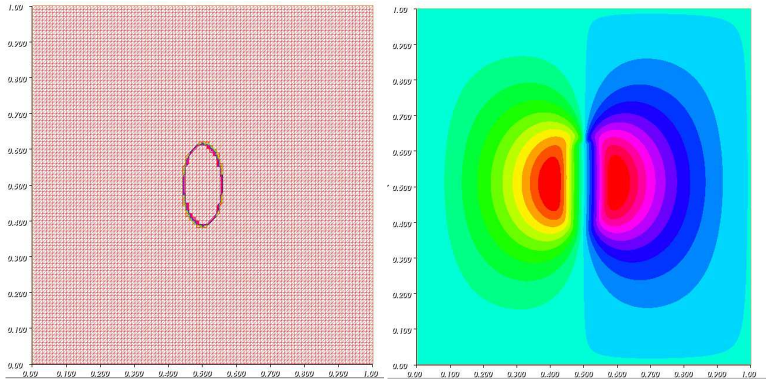


Figure 3.3: The mesh and streamlines for a ellipsoid particle.

In Figure (3.3), the ellipse is placed at the center of the domain pointing upward.

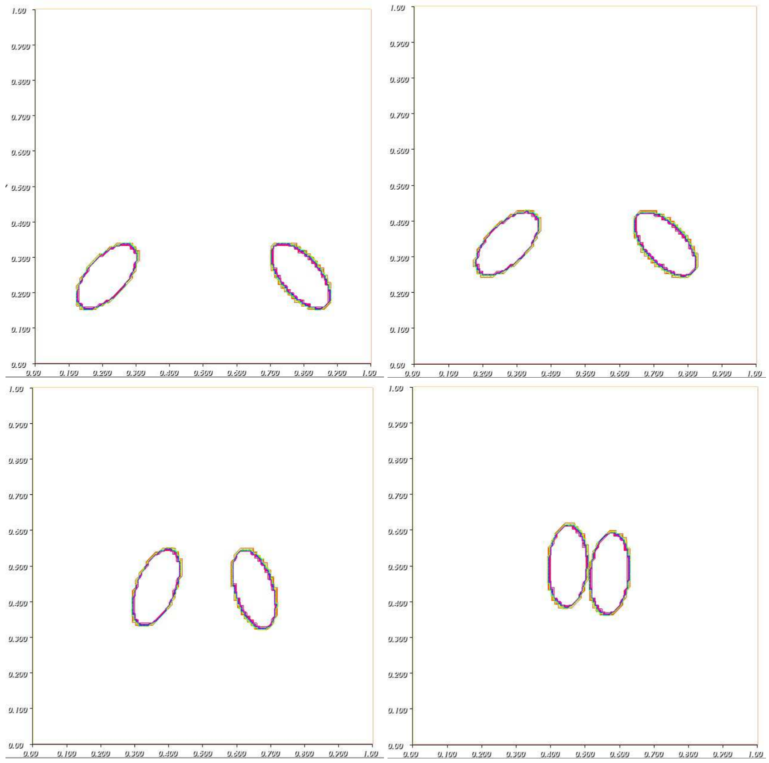


Figure 3.4: The interaction for 2 particles. They approach each other and align.

In Figure (3.4), the center of the two ellipses are placed at $(0.2, 0.2), (0.8, 0.2)$ with orientations $\theta = \frac{\pi}{4}, \frac{3\pi}{4}$ respectively.

Chapter 4

Single Polymer Statistics

In this chapter, we focus on the single polymer statistics. Because of their broad range of properties, polymers play an essential and ubiquitous role in everyday life. In particular, the mechanics of polymer is very important in nanomotors from biology and nanotechnology. For example, the molecular motors which walk along a linear track make use of polymers. Virtually all the track-walking biomotors [109] and the man-made counterparts reported to date use polymer linkers to connect the leg-like track-binding parts. The polymer linkers are often semi-flexible oligomers that consist of 10100 monomer units and exhibit a rich interplay between bending energy and conformational freedom (i.e., entropy). The polymer linkers can be peptides in the form of soft random coils or relatively rigid alpha helices, or be nucleotides in the form of single strands or double-stranded helices. The length and intra-chain force of a polymer linker largely decide how fast a diffusing leg binds to a binding site of the track and how much force can be transmitted into the motor's forward motion against a load. After an event of leg binding, the free energy of the polymer linker affects the overall stability of a multi-leg bound state of the motor and thereby the ensuing leg dissociation for the motor's continual motion. Therefore, controlling the polymer's length, force and free energy is a rather common mechanism by which biological and artificial nanomotors control their leg binding/dissociation, force transmission and even motional direction. Indeed, biological motors exhibit a wide variety of sophisticated polymer control. As a prominent example, a bipedal biological motor called kinesin aligns a part of its soft peptide linker at one end in a zipper-like manner along the track to throw forward the diffusing leg at the linker's other end. This polymer 'zippering' generates a power stroke that not only provides a forward bias for leg binding but also transmits an intra-chain force forward against a load. This force is largely associated with the entropy change of the linker. Another biological bipedal motor called myosin V generates a similar power stroke but by forward tilting its rigid helical linker at one end. Interestingly, tilting the linker at its one end can also transmit a force to selectively dissociate a leg at the other end from the track even though the leg binds the track stronger than the leg at the tilted end, as found in another biological bipedal motor. For both kinesin and myosin V, the free energy associated with the polymer linker is comparable to the leg-track binding energy in magnitude. This makes possible a symmetry breaking by which a local asymmetry in the control of one end of the polymer linker may be amplified into much larger directional steps of the motor as a whole. This polymer mechanics-mediated symmetry breaking has been identified in biological nanomotors and later implemented in artificial DNA nanomotors. Indeed, the mechanism for single polymer control offers a mechanistic basis for nanoscale motors and machines.

Understanding the rich single-polymer control effects is important not only for the study of biological nanomotors, but also for development of artificial counterparts for nanotechnology. While biological nanomotors are largely limited to peptides, artificial nanomotors may be made of diverse building blocks ranging from peptides, nucleotides to synthetic polymers. Besides, any effective polymer control may be attributed to a fine interplay among the conformational entropy and bending energy of the polymer and the energy associated with its binding with other molecular partners. Therefore, the study of motor-relevant polymer controls calls for a general and rigorous polymer-mechanical framework that counts the conformational entropy and bending energy accurately and allows study of polymer controls in a generic, conceptually clear way. Such a general framework will be particularly useful in providing guidelines for developing artificial nanomotors from scratch.

In this chapter, we firstly describe some common models in the field of polymer study. We will then focus on the worm like chain model, which is a good candidate for semi-flexible polymers and also a realistic working model for many polymeric building blocks of nanomotors. Numerical methods are discussed to get the exact end to end distribution of the worm like chain model.

4.1 Existing models

We will start with a brief description of fundamental models for polymer chain.

In the discrete level, a single chain is composed of $n+1$ identical main atoms, which are labelled by the position vectors x_j, $j = 0, 1, 2..n$. The atoms are joined by single bonds from one end to the other, and labelled by $l_j = x_j - x_{j-1}$. The vector l_j is named bond vector and the magnitude of l_j is the bond length. The angle θ_j between l_j and l_{j+1} is the supplement of the jth bond angle. The angle between the two planes which contain l_{j-1}, l_j and l_j, l_{j+1} respectively defines the internal rotation angle ϕ_j.

One very important quantity is the end to end vector $r = x_n - x_0$ which is the resultant of all the bond vectors $r = \sum\limits_{j=1}^{n} l_j$. Other related quantities include the mean-square end to end distance $\langle r^2 \rangle$ as a measure of the average chain dimension which is given by

$$\begin{aligned} \langle r^2 \rangle &= \sum_{i=1}^{n} \sum_{j=1}^{n} \langle l_i \cdot l_j \rangle \\ &= nl^2 + 2 \sum_{1 \le i < j \le n} \sum \langle l_i \cdot l_j \rangle . \end{aligned} \tag{4.1.1}$$

Here we assume all the bond length $l_i = l$. Further, if S_i is the vector distance between the ith atom from the center of mass, the radius of gyration S is defined by

$$S^2 = \frac{1}{n+1}\sum_{i=1}^{n} S_i^2, \tag{4.1.2}$$

and the well known formula for the mean square radius of gyration $\langle S^2 \rangle$ is given by

$$\langle S^2 \rangle = \frac{1}{(n+1)^2}\sum\sum_{0\leq i<j\leq n}\langle (S_j - S_i)^2 \rangle. \tag{4.1.3}$$

Now suppose the starting end of the polymer chain is fixed at the origin of a Cartesian coordinate system, we can define the partition function of the system to be

$$Z = \int \exp\left[-\frac{U(\{x_n\})}{k_B T}\right] d\{x_n\}. \tag{4.1.4}$$

Here $\{x_n\}$ is the notation for the set $x_1, x_2..$ and each set represents a unique configuration. U is the potential energy and is defined as

$$U(x_n) = \sum_{j=1}^{n} u_j(x_{j-1}, x_j) + W(x_n). \tag{4.1.5}$$

where $u_j(x_{j-1}, x_j) = u_j(l_j)$ is due to the fact that the bond between bead j and $j-1$ are connected by covalent bonds. W is the potential for all the other types of potential including bond angle restrictions.

With these, the instantaneous distribution is given as

$$P(\{x_n\}) = Z^{-1}\exp\left[-\frac{U(\{x_n\})}{k_B T}\right]. \tag{4.1.6}$$

To compute the end to end distribution, we can just integrate $P(\{x_n\})$ over all the configurations with the constraint $\sum_{j=1}^{n} l_j = r$.

The simplest model is the random flight chains. For this model, we have $\langle l_i \cdot l_j \rangle = 0$, for $i \neq j$ which means that there is no correlation between any two bonds. It is also called the freely joined chain or the random coil model. Note that $\langle r^2 \rangle = nl^2$ and $\langle r^2 \rangle$ is proportional to n. In the large n limit, the distribution function $P(r)$ is found to be [129]

$$P(r) = (\frac{3}{2\pi n l^2})^{3/2}\exp(-\frac{3r^2}{2nl^2}). \tag{4.1.7}$$

In this limit, the random-flight chain becomes the Gaussian chain. Also, for the Gaussian chain, the following relationship is well-known:

$$\langle S^2 \rangle = \frac{1}{6}\langle r^2 \rangle. \tag{4.1.8}$$

Another discrete model is the freely rotating chain. In this case both bond length l and bond angle $\pi - \theta$ are fixed, while the internal rotation angle ϕ_j is uniformly distributed in its range from $-\pi$ to π. It is shown that $\langle l_i \cdot l_{i+1} \rangle = l^2 \cos\theta$ and generally $\langle l_i \cdot l_j \rangle = l^2 \cos^{j-i}\theta (i < j)$. By (4.1.1), we can obtain

$$\langle r^2 \rangle = nl^2 \frac{1+\cos\theta}{1-\cos\theta} - 2l^2 \cos\theta \frac{1-\cos^n\theta}{(1-\cos\theta)^2}. \tag{4.1.9}$$

For this case, in the limit $n \to \infty$, assuming that $0 < \theta < \pi/2$. we can see that

$$\langle r^2 \rangle \to nl^2 \frac{1+\cos\theta}{1-\cos\theta}. \tag{4.1.10}$$

Similarly we can compute the average $\langle r \cdot u_0 \rangle$ with $u_0 = l_1/l$ being the unit vector in the direction of the first bond.

$$\langle r \cdot u_0 \rangle = l^{-1} \sum_{i=1}^{n} \langle r_i \cdot r_i \rangle = l \frac{1-\cos^n\theta}{1-\cos\theta}. \tag{4.1.11}$$

Note that $r \cdot u_0$ is the projection of r in the direction of l_1. Again if we assume $0 < \theta < \pi/2$, as n goes to infinity, we have

$$\langle r \cdot u_0 \rangle \to \frac{l}{1-\cos\theta}. \tag{4.1.12}$$

This quality in the right hand side is defined as the persistence length l_p. Note that $l_p > l$ for freely rotating chain and $l_p = l$ for random flight chain. Thus this l_p is usually used as a measure of the stiffness of the polymer.

The chains discussed so far are discrete models. For a discrete chain with n bonds, with each bond of length l, we can define the total contour length l_c to be $l_c = nl$. Also the contour distance s of the ith atom from the initial (0th atom) is $s = li$. Now we can take the limit $n \to \infty$ and $l \to 0$ to get a continuous model. In this way, the discrete chain contour becomes a continuous and differentiable space curve. The unit tangent vector is then defined by

$$u(s) = \frac{dx(s)}{ds}. \tag{4.1.13}$$

Here $x_i = x(s)$ is the radius vector. Note that $|u(s)^2| = 1$ and the end to end distribution can then be expressed as

$$r = x(l_c) - x(0) = \int\limits_0^{l_c} u(s)ds. \tag{4.1.14}$$

Also, in the continuous limit, we have

$$\begin{aligned} \langle r^2 \rangle &= \int_0^{l_c} \int_0^{l_c} \langle u(s_1) \cdot u(s_2) \rangle ds_1 ds_2, \\ \langle S^2 \rangle &= \frac{1}{{l_c}^2} \int_0^{l_c} ds_1 \int_{s_1}^{l_c} ds_2 \langle R^2(s_1, s_2) \rangle. \end{aligned} \tag{4.1.15}$$

Here $r(s_1, s_2)$ is the vector distance between contour points s_1 and s_2. For the continuous models, we define the Kuhn segment length A_K and the persistence length l_p by

$$\begin{aligned} A_k &= \lim_{l_c \to \infty} (\langle r^2 \rangle / l_c), \\ l_p &= \lim_{l_c \to \infty} \langle r \cdot u_0 \rangle. \end{aligned} \tag{4.1.16}$$

Recall that the Gaussian chain is the limit case for the random flight chain when N is large. We consider a continuous Gaussian chain, since $\langle r^2 \rangle = ll_c$, this becomes the defining equation for "bond length" l in the continuous case. Regarding l_c merely as a continuous variable, (4.1.7) can be rewritten as

$$P(r; l_c) = (\frac{3}{2\pi ll_c})^{3/2} \exp(-\frac{3r^2}{2ll_c^2}). \tag{4.1.17}$$

This $P(r; l_c)$ turns to be the solution of the following differential equation

$$(\frac{\partial}{\partial l_c} - \frac{l}{6} \nabla_r^2) P(r; l_c) = 0. \tag{4.1.18}$$

Note that if l_c is regarded as "time", the above equation simply describes the diffusion associated with the random process $r(s)$ of a Brownian particle with diffusion coefficient $l/6$ at a long time. This analogue would also be used when we study the worm like chain model in the following sections.

4.2 Worm Like Chain model

Previous theoretical studies of motor-relevant polymer effects focus on specific molecular systems (mostly biological motors and associated peptide systems) or are based on approximate solution of polymer mechanics. A good candidate for developing the general framework is the worm like chain (WLC) model for semi-flexible polymers. On one hand, the exact solution to the WLC model becomes available recently to allow a precise counting of the conformational entropy, which remains difficult for other methods like atomic simulations. On the other hand, the WLC model has only two parameters: contour length as a measure of the maximum stretch of a polymer and persistence length as a measure of the polymers bending rigidity. Both parameters can be extracted by fitting experimental data, potentially making the WLC model a realistic working model for any polymeric building blocks of nanomotors.

Basically, the worm like chain model is a continuous model from the freely rotating chain. Consider a freely rotating chain with n bonds, bond length l and bond angle $\pi - \theta$. We take its continuous limit $n \to \infty$ and $\theta \to 0$. When taking this limit, we should also require $l_c = nl$ and the persistence length is kept as a constant. Here we set the the persistence length to be $(2\lambda)^{-1}$

$$l_p = \frac{1}{2\lambda} = \frac{l}{1 - \cos\theta}. \tag{4.2.1}$$

and $\theta \to 0$. so that

$$\cos\theta = 1 - 2\lambda l = 1 - \frac{2\lambda l_c}{n}. \tag{4.2.2}$$

Denote this limit taking process to be $\lim\limits_{\text{worm}}$, we have

$$\lim_{\text{worm}} \cos^n\theta = \lim_{n\to\infty} (1 - \frac{2\lambda l_c}{n})^n = e^{-2\lambda l_c}. \tag{4.2.3}$$

Using the results in the freely rotating chain, we have

$$\begin{aligned} \langle R \cdot u_0 \rangle &= \frac{1}{2\lambda}(1 - e^{-2\lambda l_c}), \\ \langle R^2 \rangle &= \frac{l_c}{\lambda} - \frac{1}{2\lambda^2}(1 - e^{-2\lambda l_c}). \end{aligned} \tag{4.2.4}$$

In the limit of $\lambda l_c \to 0$ which is the rigid rod limit, we can see that $\langle R^2 \rangle = l_c^2$ and if $\lambda l_c \to \infty$ which is the random coil limit, we have $\langle R^2 \rangle = \frac{l_c}{\lambda}$. As a result, the worm like chain is an interpolation from both extremes. Therefore, it is a good model for most typical semi-flexible polymers.

Now we study the end to end distribution. Recall that the worm like chain is a continuous space curve $x(s)$ by taking the limit from the freely rotating chain and the persistence length being fixed, here s is the arc-length parameter, which ranges from 0 to l_c, the contour length of the polymer. The tangent vector $t(s)$ to the curve is $t(s) = dx(s)/ds$ and it is a unit vector ($|t^2(s)| = 1$). Furthermore, the curvature of the polymer is given by $\kappa = |dt(s)/ds|$. We define the conditional distribution function $G(r, t|t_0; l_c)$, such that at the terminal end, $x(l_c) = r$,$t(l_c) = t$ and at the initial end, $x(0) = 0$, $t(0) = t_0$. $G(r, t|t_0; l_c)$ is called the Green function and is normalized such that

$$\int G(r, t|t_0; l_c) dr dt = 1. \tag{4.2.5}$$

The characteristic function $I(k, t|t_0; l_c)$ is defined to be the Fourier transform of G with respect to r

$$I(k, t|t_0; l_c) = \int G(r, t|t_0; l_c) \mathrm{exp}(ik \cdot r) dr. \tag{4.2.6}$$

Other distribution functions can be obtained in the follwing manner:

$$G(t|t_0; l_c) = \int G(r, t|t_0; l_c) dr = I(0, t|t_0; l_c), \tag{4.2.7}$$

$$G(r|t_0; l_c) = \int G(r, t|t_0; l_c) dt, \tag{4.2.8}$$

$$G(r, t; l_c) = (4\pi)^{-1} \int G(r, t|t_0; l_c) dt_0, \tag{4.2.9}$$

$$G(r; l_c) = (4\pi)^{-1} \int G(r, t|t_0; l_c) dt dt_0. \tag{4.2.10}$$

Note that the generation of the curve may be described by a simple Markov process and the distribution function follows a Fokker-Planck equation. As shown in [36], the Fokker-Planck equation reads

$$(\frac{\partial}{\partial l_c} - \lambda \nabla_t^2 + t \cdot \nabla_r) G(r, t|t_0; l_c) = \delta(l_c)\delta(r)\delta(t - t_0) \tag{4.2.11}$$

Taking the Fourier transform of 4.2.11, we obtain

$$(\frac{\partial}{\partial l_c} - \lambda\nabla_t^2 - ik\cdot t)I(k,t|t_0;l_c) = \delta(l_c)\delta(t-t_0). \tag{4.2.12}$$

These are the governing equations in the statistical mechanics of the worm like chain.

4.3 Methods for the WLC end to end distribution

For simplicity, we denote $G(r,u|u_0;l_c)$ by $Q(r)$. To compute $Q(r)$, we need to sum over all the configuration C, such that the chain ends at r. For each particular configuration, the energy associated is given as $E(C) = \frac{1}{2}A\int\limits_0^{l_c}\kappa^2 ds$. Here A is the bending modulus, given as $A = l_p k_B T$. Using this energy, we can give the worm like chain a statistical interpretation. $Q(\vec{r})$ can be expressed as the sum of Boltzmann factor $\exp(-E(C)/k_BT)$ over all the possible configurations. This is a standard counting problem in statistical mechanics thus it can be addressed in the language of path integral. It has been addressed that the difficulty is presented by the inextensibility constraint. The key to solve the problem is to consider Brownian motion in the space of the tangent vectors $u(s)$ instead of the position vectors $x(s)$. The tangent vectors form a unit sphere and the problem reduces to studying Brownian motion on the unit sphere. For now we will suppose the initial tangent vector is $t_A = dx/ds|_{s=0}$ and the final tangent vector is $t_B = dx/ds|_{s=l_c}$. Then $Q(\vec{r})$ would have the following path integral representation:

$$Q(\vec{r}) = N\int\limits_{t_A}^{t_B} D[t(s)]\exp(-\frac{l_p}{2}\int\limits_0^{l_c}(\frac{dt}{ds})^2 ds). \tag{4.3.1}$$

Here N is a normalizing constant, and the delta function takes out all the configurations that end at position r.

4.3.1 1D case

As a starting point, we will firstly consider the problem in one dimension. Instead of considering $Q(r)$, we can focus on $P(z) = \int Q(\vec{r})\delta(r_3 - z)d\vec{r}$ where r_3 is the third component of $\vec{r}$, which is $Q(\vec{r})$ integrated over a plane of constant z. Note that

$$\begin{aligned} P(z) &= \int Q(\vec{r})\delta(r_3 - z)d\vec{r} \\ &= N\int_{t_A}^{t_B} D[t(s)]\exp(-\frac{l_p}{2}\int_0^{l_c}(\frac{dt}{ds})^2 ds) \times \delta(z - \int_0^{l_c} t_3 ds). \end{aligned} \tag{4.3.2}$$

To find $P(z)$, we will consider the following transform

$$\begin{aligned} \tilde{P}(f) &= \int_{-l_c}^{l_c} \exp(\frac{fz}{l_p})P(z)dz \\ &= N\int_{t_A}^{t_B} D[t(s)]\exp(-\frac{l_p}{2}\int_0^{l_c}(\frac{dt}{ds})^2 ds) \times \exp(\frac{f}{l_p}\int_0^{l_c} t_3 ds) \\ &= N\int_{t_A}^{t_B} D[t(\tau')]\exp(-\int_0^{\beta}\frac{1}{2}(\frac{dt}{d\tau'})^2 - ft_3 d\tau') \quad \text{Here} \quad \tau' = \frac{s}{l_p}, \beta = \frac{l_c}{l_p} \\ &= N\int_{t_A}^{t_B} D[t(\tau)]\exp(i\int_0^{-i\beta}\frac{1}{2}(\frac{dt}{d\tau})^2 + ft_3 d\tau) \quad \text{Here} \quad \tau = -i\tau' \\ &= N\int_{t_A}^{t_B} D[t(\tau)]\exp(i\int_0^{-i\beta} L d\tau). \end{aligned} \tag{4.3.3}$$

Here $L = \frac{1}{2}(\frac{dt}{d\tau})^2 + ft_3$ is the Lagrangian of the system. If we can find $\tilde{P}(f)$, then we can take the inverse transform to get the result. To do the inverse transform, note that $P(z)$ vanishes for $z < -l_c$ and $z > l_c$, therefore, we can see that the transform can be written as

$$\begin{aligned} \tilde{P}(f) &= \int_{-\infty}^{\infty} \exp(\frac{fz}{l_p})P(z)dz \\ &= \int_{-\infty}^{\infty} \exp(\beta f\tilde{z})p(\tilde{z})d\tilde{z} \quad \text{Here} \quad \tilde{z} = \frac{z}{l_c}, p(\tilde{z}) = l_c P(z), \beta = \frac{l_c}{l_p} \\ &= \int_{-\infty}^{\infty} \exp(-iu\tilde{z})p(\tilde{z})d\tilde{z} \quad \text{Here} \quad -iu = \beta f. \end{aligned} \tag{4.3.4}$$

Realizing this is in the form of Fourier transform, we can write down the inverse transform for $p(\tilde{z})$ as

$$p(\tilde{z}) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \tilde{P}(-\frac{iu}{\beta})\exp(iu\tilde{z})du. \tag{4.3.5}$$

Finally $P(z)$ can be obtained as $\frac{p(\tilde{z})}{l_c}$.

Now we focus on $\tilde{P}(f) = N\int_{t_A}^{t_B} D[t(\tau)]\exp(i\int_0^{-i\beta} L d\tau)$ with $L = \frac{1}{2}(\frac{dt}{d\tau})^2 + ft_3$.

To compute this quantity, we shall note that there is a relationship between the path integral and the Schrödinger's equation. The path integral approach to quantum mechanics was developed by Feynmann in 1942. In its original form, it applies to a point particle moving in a Cartisian cooridinate system and yields the transition amplitude. We will sketch the basic idea of path integral representation of time evolution amplitudes.

In terms of the bra-ket notation, the Schrödinger equation can be expressed in the following way:

$$ih\frac{d}{dt}|\psi\rangle = H|\psi\rangle. \tag{4.3.6}$$

Here H is the Hamiltonian operator. We shall illustrate the one dimensional case for simplicity. The Hamiltonian operator can be written as

$$H = \frac{p^2}{2m} + V(q), \tag{4.3.7}$$

with $V(q)$ being the potential energy. The above Schrödinger equation can be integrated to find the wave function $|\psi(t)\rangle$ at any time t from the state at another time 0 as

$$|\psi(t)\rangle = \exp(-\frac{i}{h}Ht)|0\rangle. \tag{4.3.8}$$

The transition amplitude for a transition from an initial state $|0\rangle$ to a final state $|F\rangle$ at time T is given by

$$\langle F|\psi(t)\rangle = \langle F|\exp(-\frac{i}{h}HT)|0\rangle. \tag{4.3.9}$$

Feynman realized that we can divide the interval from 0 to T into N infinitesimal time steps, each of length $\Delta t = \frac{T}{N}$, then transition amplitude can then be written as

$$\langle F|\exp(-\frac{i}{h})|0\rangle = \langle F|\exp(-\frac{i}{h}H\Delta t)\exp(-\frac{i}{h}H\Delta t)...\exp(-\frac{i}{h}H\Delta t)|0\rangle. \tag{4.3.10}$$

We can insert the identity $I = \int dq|q\rangle\langle q|$ between each pair of the exponentials to obtain

$$\langle F|\exp(-\frac{i}{h}HT)|0\rangle = (\prod_{j=1}^{N-1}\int dq_j)\langle F|\exp(-\frac{i}{h}H\Delta t)|q_{N-1}\rangle...\langle q_1|\exp(-\frac{i}{h}H\Delta t)|0\rangle.$$

(4.3.11)

Using the Baker-Campbell-Hausdorff formula, where H=T+V being the sum of a kinetic and a potential energy:

$$e^{-i\Delta t(T+V)/h} = e^{-i\Delta tV/h} e^{-i\Delta tT/h} e^{-i\Delta tX/h^2}, \tag{4.3.12}$$

and dropping order ϵ^2 terms, we can write each transition amplitude as

$$\langle q_{j+1}|\exp(-\frac{i}{h}H\Delta t)|q_j\rangle = \langle q_{j+1}|\exp(-\frac{i}{h}\frac{p^2}{2m}\Delta t)\exp(-\frac{i}{h}V(q_j)\Delta t)|q_j\rangle \tag{4.3.13}$$

Again we insert the identity $I = \int \frac{dp}{2\pi}|p\rangle\langle p|$ into the above expression to yield

$$\begin{aligned}\langle q_{j+1}|\exp(-\frac{i}{h}H\Delta t)|q_j\rangle &= \exp(-\frac{i}{h}V(q_j)\Delta t)\int \frac{dp}{2\pi}\langle q_{j+1}|\exp(-\frac{i}{h}\frac{p^2}{2m}\Delta t)|p\rangle\langle p|q_j\rangle \\ &= \exp(-\frac{i}{h}V(q_j)\Delta t)\int \frac{dp}{2\pi}\exp(-\frac{i}{h}\frac{p^2}{2m}\Delta t)\langle q_{j+1}|p\rangle\langle p|q_j\rangle \\ &= \exp(-\frac{i}{h}V(q_j)\Delta t)\int \frac{dp}{2\pi}\exp(-\frac{i}{h}\frac{p^2}{2m}\Delta t - \frac{i}{h}p(q_{j+1}-q_j)).\end{aligned} \tag{4.3.14}$$

In the last line, we used the identity $\langle p|q_j\rangle = \frac{\exp(\frac{i}{h}pq_j)}{\sqrt{h}}$. This integral over p can be done exactly to obtain

$$\langle q_{j+1}|\exp(-\frac{i}{h}H\Delta t)|q_j\rangle = (\frac{-im}{2\pi\Delta th})^{\frac{1}{2}}\exp[\frac{i}{h}\Delta t(\frac{1}{2}m(\frac{q_{j+1}-q_j}{\Delta t})^2 - V(q_j))] \tag{4.3.15}$$

Combining the result, the transition ampltude for the entire period is

$$\langle F|\exp(-\frac{i}{h}HT)|0\rangle = (\frac{-im}{2\pi\Delta th})^{\frac{N}{2}}\exp[\frac{i}{h}\sum_{j=0}^{N-1}\Delta t(\frac{1}{2}m(\frac{q_{j+1}-q_j}{\Delta t})^2 - V(q_j))] \tag{4.3.16}$$

Taking the limit $N \to \infty$,the transition amplitude can be expressed in the path integral formulation

$$\langle F|\exp(-\frac{i}{h}HT)|0\rangle = \int Dq(t)\exp[\frac{i}{h}S]. \tag{4.3.17}$$

with S being the classical action as the integral of the Lagrangian $S = \int_0^T Ldt$ and $L = \frac{1}{2}m\dot{q}^2 - V(q)$. Note that in classical mechanics, the classical path which is the only possible path is determined by the minimization of the action integral S, and in quantum motions,various paths other than the classical path can be realized. The path integral sums all the paths which contribute to the quantum amplitude.

Now consider our $\tilde{P}(f) = N\int_{t_A}^{t_B} D[t(\tau)]\exp(i\int_0^{-i\beta} Ld\tau)$. with $L = \frac{1}{2}(\frac{dt}{d\tau})^2 + ft_3$. This can be interpreted as the path integral representation for the kernal of a quantum particle on the surface of a unit sphere at inverse temperature β. Thus we can express $\tilde{P}(f)$ as the quantum amplitude to go from an initial tangent vector t_A to a final tangent vector t_B in an imaginary time in the presence of an external potential $-f\cos\theta$.

Note that the Lagrangian is $L = \frac{1}{2}(\frac{dt}{d\tau})^2 + ft_3$, therefore the corresponding Hamiltonian is $H_f = -\frac{\nabla^2}{2} - f\cos\theta$. The gradient operator acts on the unit sphere and we have the following expression in a spherical coordinate:

$$\nabla^2 = \frac{1}{\sin\theta}\frac{\partial}{\partial\theta}(\sin\theta\frac{\partial}{\partial\theta}) + \frac{1}{\sin^2\theta}\frac{\partial^2}{\partial\phi^2}. \tag{4.3.18}$$

As a result, we can write

$$\begin{aligned}\int_{t_A}^{t_B} D[t(\tau)]\exp(i\int_0^{-i\beta} Ld\tau) &= \langle t_A|e^{-iH_f(-i\beta)}|t_B\rangle \\ &= \langle t_A|e^{-\beta H_f}|t_B\rangle.\end{aligned} \tag{4.3.19}$$

Here $\langle\cdot\rangle = \int_0^{2\pi}\int_0^{\pi}\cdot\sin\theta d\theta d\phi$ Note that $\tilde{P}(f)$ can be expressed as

$$\tilde{P}(f) = \sum_n e^{-\beta E_n}\psi_n^*(t_A)\psi_n(t_B). \tag{4.3.20}$$

Here $\psi_n(t)$ is a complete set of normalized eigenstates of the Hamiltonian $H = -\frac{\nabla^2}{2} - f\cos\theta$ and E_n are the corresponding eigenvalues. The key to the numerical evaluation of $\tilde{P}(f)$ is a convenient choice of basis. We shall choose the basis as $\hat{P}_l(x) = \sqrt{\frac{2l+1}{4\pi}}P_l(x)$. Here $x = \cos\theta$, $l = 0..\infty$ and $P_l(x)$ are the Legendre functions defined to be

$$P_l(x) = \frac{1}{2^l l!}\frac{d^l}{dx^l}(x^2-1)^l. \tag{4.3.21}$$

With this basis chosen, we will firstly truncate the initial and tangent vector up to size N. For a free end, which means an equal probability in all directions, the state is a constant function over the unit sphere which is normalized as $\langle t|t\rangle = 1$, therefore we have $|t\rangle = \sqrt{1}4\pi$ for a free end. In our set of basis chosen, this is a column vector which the first entry is 1 and the rest are 0.

For a fixed end along the z direction, the state is a delta function given as

$$|t\rangle = \frac{1}{2\pi}\delta(\cos\theta - \cos 0) = \frac{1}{2\pi}\delta(x-1). \tag{4.3.22}$$

The expansion coefficient is computed as

$$\int_0^{2\pi}\int_{-1}^{1}\frac{1}{2\pi}\delta(x-1)\sqrt{\frac{2l+1}{4\pi}}P_l(x)dxd\phi = \sqrt{\frac{2l+1}{4\pi}}P_l(1) = \sqrt{\frac{2l+1}{4\pi}}. \tag{4.3.23}$$

Thus the state $|t\rangle$ is a column vector with entries $\sqrt{\frac{2l+1}{4\pi}}$ for each corresponding l. Now to get a matrix representation for $\exp(-\beta H_f)$, we can write H_f as $H_f = H_0 - fH_1$, with $H_0 = -\frac{\nabla^2}{2}$ and $H_1 = \cos\theta$. Firstly, we get the matrix representation for H_0. Note that $\hat{P}_l(x)$ are the eigenfunctions of the operator H_0, with eigenvalues $\frac{l(l+1)}{2}$. Since the basis chosen is a complete set of orthonormal basis, H_0 is then a diagonal matrix with diagonal entries $\langle\hat{P}_l|H_0|\hat{P}_l\rangle = \frac{l(l+1)}{2}$.

Now we consider H_1, where $H_1\hat{P}_l = x\hat{P}_l$. Using the well known three term recurrence relation known as Bonnets recursion formula:

$$xP_l = \frac{l+1}{2l+1}P_{l+1} + \frac{l}{2l+1}P_{l-1}, \tag{4.3.24}$$

we can compute

$$x\hat{P}_l = \frac{l+1}{\sqrt{(2l+1)(2l+3)}}\hat{P}_{l+1} + \frac{l}{\sqrt{(2l-1)(2l+1)}}\hat{P}_{l-1}. \tag{4.3.25}$$

As a result we can get the element for the matrix representation of H_1: the $(l+1), l$ entry is $\langle\hat{P}_{l+1}|H_1|\hat{P}_l\rangle = \frac{l+1}{\sqrt{(2l+1)(2l+3)}}$ and the $l-1, l$ entry is $\langle\hat{P}_{l-1}|H_1|\hat{P}_l\rangle = \frac{l}{\sqrt{(2l-1)(2l+1)}}$.

Finally, the matrix H_f is an infinite tri-diagonal matrix with the diagonal entries as the elements in $H_{ll} = l(l+1)/2$ and the super-diagonal entries $H_{ll+1} = -f(l+1)\sqrt{1/[(2l+1)(2l+3)]}$. To evaluate $\tilde{P}(f)$, we truncate the infinite matrix to size N by N.

Once we have the matrix representation of the matrix H_f, the matrix exponential is evaluated using Padé's approximation. For the scalar case, the Padé's approximation of a function $f(x)$ is the quotient of two polynomials $N_{p/q}(x)$ and $D_{p/q}(x)$ of degree p and q respectively. And for a matrix A, the quotient, which is called the (p,q)-degree type Padé' approximation is defined as

$$R_{p/q}(A) = [D_{p/q}(A)]^{-1} N_{p/q}(A), \tag{4.3.26}$$

where $N_{p/q}(A) = n_) I + n_1 A + n_2 A^2 + ... + n_p A^p$ and $D_{p/q}(A) = I + d_1 A + d_2 A^2 + ... + d_q A^q$. Note that when $q = 0$, this reduce to the Taylor expansion for $f(A)$. We assume $f(A)$ has the Maclaurin expansion

$$f(A) = a_0 I + a_1 A + a_2 A^2 + ... + a_k A^k + ... \quad . \tag{4.3.27}$$

Since the highest possible order of non zero derivative of $R_{p/q}(A)$ is $p+q$, we have

$$R_{p/q}(A) - f(A) = \sum_{k=p+q+1}^{\infty} c_k' A^k. \tag{4.3.28}$$

Multiply $D_{p/q}$ on both sides,

$$D_{p/q}(A) f(A) - N_{p/q}(A) = D_{p/q} \sum_{k=p+q+1}^{\infty} c_k' A^k = \sum_{k=p+q+1}^{\infty} c_k A^k. \tag{4.3.29}$$

Therfore we have

$$\sum_{i=0}^{p} n_i A^i - (I + \sum_{i=1}^{q} d_i A^i)(\sum_{i=0}^{\infty} a_i A^i) = \sum_{k=p+q+1}^{\infty} c_k A^k. \tag{4.3.30}$$

When the left hand side of the above is multiplied out and setting the coefficient of the powers of A^i to be 0 for $i = 0, 1, ..., p+q$, we have a system of $p+q+1$ linear equations. There are q equations involves only the unknowns $d_1, d_2 ... d_q$ which can be solved first. Another $p+1$ equations involving $n_0, n_1 ... n_p$ can be solved after. Now we are interested in computing the matrix exponential. Therefore setting $f(A)$ as e^A, we have $a_n = \frac{1}{n!}$ in the Taylor series of f. Soving the system of equations, we finally have

$$N_{p/q} = \sum_{i=0}^{p} \frac{(p+1-i)!p!}{(p+q)!i!(p-i)!} A^i. \tag{4.3.31}$$

and

$$D_{p/q} = \sum_{i=0}^{q} \frac{(p+q-i)!q!}{(p+q)!i!(q-i)!}(-A)^i. \tag{4.3.32}$$

It is shown in [86] that the diagonal approximants $p = q$ are preferred. And in this case, we have $n_0 = 1, n_i = n_{i-1}\frac{p+1-i}{(2p+1-i)i}$, and $d_i = (-1)^i n_i$. Finally our approximation for e^A which is $R_{p/p}$ is given as

$$R_{p/p} = \begin{cases} 1 + 2(\sum\limits_{i=0}^{p/2} n_{2i}A^{2i} - A\sum\limits_{i=0}^{p/2-1} n_{2i+1}A^{2i})^{-1}(A\sum\limits_{i=0}^{p/2-1} n_{2i+1}A^{2i}), & \text{if } p \text{ is even,} \\ -1 - 2(A\sum\limits_{i=0}^{(p-1)/2} n_{2i+1}A^{2i} - \sum\limits_{i=0}^{(p-1)/2} n_{2i}A^{21})^{-1}(\sum\limits_{i=0}^{(p-1)/2} n_{2i}A^{2i}), & \text{if } p \text{ is odd.} \end{cases} \tag{4.3.33}$$

The drawback of Padé approximation is that it is only accurate near the origin and it is not valid, when $|A|$ is too large. This problem can be solved using the scaling and squaring method [86]. Using the identity

$$e^A = (e^{2^{-s}A})^{2^s} \approx [R_{p/p}(2^{-s}A)]^{2^s}. \tag{4.3.34}$$

We can choose $||2^{-s}A|| \leq 1/2$ and obtain the result by repeated squaring. Choosing $p = 6$ yields desired results.

In this way, we can get $\tilde{P}(f)$. To perform the inverse transform, we consider $k = -\frac{N}{2} : \frac{N}{2} - 1$ and compute an array of $\tilde{P}_k$ defined to be $\tilde{P}_k = \tilde{P}(-\frac{iu}{\beta})$ with $u = \pi k$. We want to compute $p(\tilde{z})$. Numerically we consider $\tilde{z} = j \cdot \frac{2}{N}$ and compute an array of P_j defined to be $P_j = p(\frac{2j}{N})$. We have

$$\begin{aligned} \tilde{P}_k = \tilde{P}(f) &= \int_{-1}^{1} e^{-iu\tilde{z}} p(\tilde{z}) d\tilde{z} \\ &\approx \frac{2}{N} \sum_{j=-\frac{N}{2}}^{\frac{N}{2}-1} e^{-iu(j\cdot\frac{2}{N})P_j} = \frac{2}{N} \sum_{j=-\frac{N}{2}}^{\frac{N}{2}-1} e^{-\frac{2\pi i}{N}\cdot j \cdot k} P_j. \end{aligned} \tag{4.3.35}$$

This is in the form of the discrete Fourier transform, and we perform the inverse fast Fourier transform to get $p(\tilde{z})$ and finally $P(z) = \frac{p(\tilde{z})}{l_c}$.

4.3.2 3D case

Note that $P(z)$ is the $1D$ distribution, or the end to end probability such that the final end lies on a particular plane. To compute the end to end distribution in dimension three, the interested quantity is $Q(\vec{r})$, which is the probability that the end position lies at position $\vec{r}$. The simplest case is that when both ends are free. Due to symmetry, we can see that $Q(r)$ only depends on the distance between $\vec{r}$ and the origin, thus $Q(\vec{r}) = Q(r)$, with $r = |\vec{r}|$. Note that $P(z)$ is obtained by integrate all the possibilities on plane z, we have

$$P(z) = \int_0^{\sqrt{l_c^2 - z^2}} 2\pi x Q(\sqrt{z^2 + x^2}) dx = -\int_{l_c}^{z} 2\pi r Q(r) dr. \tag{4.3.36}$$

Differentiating both sides yields $P'(z) = -2\pi z Q(z)$, therefore we have the 3D distribution. The case where the ends are fixed becomes more complicated. To find $Q(\vec{r})$, we again consider the transform:

$$\begin{aligned}\tilde{P}(\vec{f}) &= \int \exp(\frac{\vec{f} \cdot \vec{r}}{l_p}) Q(\vec{r}) d\vec{r} \\ &= N \int_{t_A}^{t_B} D[t(\tau')] \exp(-\int_0^{\beta} \frac{1}{2} (\frac{dt}{d\tau'})^2 - \vec{f} \cdot t d\tau') \quad \text{Here} \quad \tau' = \frac{s}{l_p}, \beta = \frac{l_c}{l_p} \\ &= N \int_{t_A}^{t_B} D[t(\tau)] \exp(i \int_0^{-i\beta} \frac{1}{2} (\frac{dt}{d\tau})^2 + f \cdot \tau d\tau) \quad \text{Here} \quad \tau = -i\tau' \\ &= N \int_{t_A}^{t_B} D[t(\tau)] \exp(i \int_0^{-i\beta} L d\tau),\end{aligned} \tag{4.3.37}$$

where the Lagrangian is given by $L = \frac{1}{2}(\frac{dt}{d\tau})^2 + f_1 t_1 + f_2 t_2 + f_3 t_3$.

To find the inverse transform, we see that

$$\begin{aligned}\tilde{P}(\vec{f}) &= \int \exp(\frac{\vec{f} \cdot \vec{r}}{l_p}) Q(\vec{r}) d\vec{r} \\ &= \int \exp(\beta \vec{f} \cdot \vec{\tilde{r}}) q(\vec{\tilde{r}}) d\vec{\tilde{r}} \quad \text{Here} \quad \vec{\tilde{r}} = \frac{\tilde{r}}{l_c}, q(\vec{\tilde{r}}) = l_c^3 Q(\vec{\tilde{r}}), \beta = \frac{l_c}{l_p} \\ &= \int \exp(-i\vec{u} \cdot \vec{\tilde{r}}) q(\vec{\tilde{r}}) d\vec{\tilde{r}} \quad \text{Here} \quad -i\vec{u} = \beta \vec{f}.\end{aligned} \tag{4.3.38}$$

Again this is in the form of Fourier transform, we have

$$q(\vec{\tilde{r}}) = \frac{1}{(2\pi)^3} \int \tilde{P}(-\frac{i\vec{u}}{\beta}) \exp(i\vec{u} \cdot \vec{\tilde{r}}) d\vec{u}, \tag{4.3.39}$$

And $Q(\vec{r}) = \frac{q(\tilde{\vec{r}})}{l_c^3}$ in the inverse transform if we can obtain $\tilde{P}(\vec{f})$.

To find $\tilde{P}(\vec{f})$, Again we realize it is written in the path integral form $\tilde{P}(\vec{f}) = N\int_{t_A}^{t_B} D[t(\tau)]\exp(i\int_0^{-i\beta} Ld\tau)$ In 3D we have $L = \frac{1}{2}(\frac{dt}{d\tau})^2 + f_1t_1 + f_2t_2 + f_3t_3$. Therefore the corresponding Hamiltonian becomes

$$H_f = -\frac{\nabla^2}{2} - f_1\sin\theta\cos\phi - f_2\sin\theta\sin\phi - f_3\cos\theta = H_0 - f_1H_1 - f_2H_2 - f_3H_3. \quad (4.3.40)$$

Again we shall choose a convenient set of basis. In 3D, due to the presence of the angle ϕ in H_f, the Legendre functions are not enough and our choice is the normalized spherical harmonics defined as

$$Y_l^m(x,\phi) = \sqrt{\frac{2l+1}{4\pi}\frac{(l-m)!}{(l+m)!}}P_l^m(x)e^{im\phi}, \quad (4.3.41)$$

with $P_l^m(x)$ being the associated Legendre functions defined in terms of the derivatives of the Legendre functions as

$$P_l^m(x) = (1-x^2)^{m/2}P_l^{[m]}(x). \quad (4.3.42)$$

Here $l = 0...\infty$ and $m = -l...l$. And the normalization is due to

$$\int_0^\pi P_l^m(x)P_l^m(x)\sin\theta d\theta = \frac{(l+m)!}{(l-m)!}\frac{2}{2l+1} \quad (4.3.43)$$

In order to get the matrix representation of H_f, we will get the the matrix representation for each of H_0,H_1,H_2 and H_3. The $lm, l'm'$ entry of the matrix elements is given by $\langle Y_l^m|H|Y_{l'}^{m'}\rangle$ for each H. We choose a cutoff L such that $l = 0...L$ and for each l, $m = -l...l$.Note the for m negative, we have

$$P_l^{-|m|}(x) = \frac{(l-|m|)!}{(l+|m|)!}P_l^{|m|}(x). \quad (4.3.44)$$

Firstly, $H_0 = -\frac{\nabla^2}{2}$, we have Y_l^m being the eigenfunctions of H_0: $H_0Y_l^m = \frac{l(l+1)}{2}Y_l^m$, therefore we have:

$$\langle Y_{l'}^{m'}|H_0|Y_l^m\rangle = \begin{cases} \frac{l(l+1)}{2}, & l' = l \;\&\; m' = m, \\ 0, & \text{otherwise}, \end{cases} \quad (4.3.45)$$

Next, we deal with $H_3 = \cos\theta$, the integral on angle ϕ can be done directly to obtain $\int_0^{2\pi}(e^{im\phi})^* e^{im\phi'}d\phi = 2\pi$ for $\phi = \phi'$ and is 0 otherwise. To deal with the integral on $x(=\cos\theta)$, we make use of the recursion relationship: $xP_l^m = \frac{l+m}{2l+1}P_{l-1}^m + \frac{l-m+1}{2l+1}P_{l+1}^m$ for $l \geq 1$. Combining the result, we have the matrix elements for H_3:

$$\langle Y_{l'}^{m'}|H_3|Y_l^m\rangle = \begin{cases} \frac{1}{\sqrt{3}}, & l = m = m' = 0 \ \&\ l' = 1, \\ \sqrt{\frac{(l+1+m)(l+1-m)}{(2l+1)(2l+3)}}, & l' = l+1 \ \&\ m' = m \ \&\ l \geq 1, \\ \sqrt{\frac{(l-m)(l+m)}{(2l-1)(2l+1)}}, & l' = l-1 \ \&\ m' = m \ \&\ l \geq 1, \\ 0, & \text{otherwise.} \end{cases} \tag{4.3.46}$$

Next we consider $H_1 = \sin\theta\cos\phi$, we will only write the non-zero entries for now. For the integral on ϕ, we have

$$\int_0^{2\pi}(e^{i(m+1)\phi})^* \cos\phi e^{im\phi} = \pi, \tag{4.3.47}$$

$$\int_0^{2\pi}(e^{i(m-1)\phi})^* \cos\phi e^{im\phi} = \pi. \tag{4.3.48}$$

To deal with the integral for θ, we seek for recurrence relation for $\sin\theta P_l^m$, and we only consider the decompositions with $m+1$ and $m-1$. As in [1], we have for $l \geq 1$,

$$\sin\theta P_l^m = \frac{1}{2l+1}P_{l+1}^{m+1} - \frac{1}{2l+1}P_{l-1}^{m+1}. \tag{4.3.49}$$

Note that the associated Lengendre functions are obtained by differentiating the Lengendre functions, and the other recurrence relation can be found based on the generating function for the Legendre Polynomials, which is

$$\frac{1}{\sqrt{1-2rx+r^2}} = \sum_{l=0}^{\infty} r^l P_l(x). \tag{4.3.50}$$

By differentiating the generating function with respect to r and compare r^l coefficient, followed by differentiating it with respect to x for $m-1$ times, we have

$$(l+1)P_{l+1}^{[m-1]} - (2l+1)xP_l^{[m-1]} - (m-1)(2l+1)P_l^{[m-2]} + lP_{l-1}^{[m-1]} = 0. \tag{4.3.51}$$

In similar manner, we can get

$$(x^2-1)P_l^{[m]} + (2m-2-l)xP_l^{[m-1]} + (m-1)(m-2-l)P_l^{[m-2]} + lP_{l-1}^{[m-1]} = 0. \tag{4.3.52}$$

and

$$(2l+1)P_l^{[m-2]} = P_{l+1}^{[m-1]} - P_{l-1}^{[m-1]}. \tag{4.3.53}$$

Using (2l+1) × (4.3.52) + (2m-2-l) × (4.3.51) and subs into (4.3.53), after some manipulation, we get the desired result

$$(2l+1)(1-x^2)^{\frac{1}{2}}P_l^m = (l+m)(l+m-1)P_{l-1}^{m-1} - (l-m+2)(l-m+1)P_{l+1}^{m-1}. \tag{4.3.54}$$

Note $x = \cos\theta$, this can be written as

$$\sin\theta P_l^m = \frac{(l+m)(l+m-1)}{(2l+1)}P_{l-1}^{m-1} - \frac{(l-m+2)(l-m+1)}{2l+1}P_{l+1}^{m-1}. \tag{4.3.55}$$

Combining the results for the integrals for θ and ϕ, we have the elements in the matrix representation of H_1 as below:

$$\langle Y_{l'}^{m'}|H_1|Y_l^m\rangle = \begin{cases} \frac{1}{\sqrt{6}}, & l = m = 0 \ \& \ l' = m' = 1, \\ \sqrt{\frac{(l+m+2)(l+m+1)}{4(2l+1)(2l+3)}}, & l' = l+1 \ \& \ m' = m+1 \ \& \ l \geq 1, \\ -\sqrt{\frac{(l-m)(l-m-1)}{4(2l-1)(2l+1)}}, & l' = l-1 \ \& \ m' = m+1 \ \& \ l \geq 1, \\ -\sqrt{\frac{(l-m+2)(l-m+1)}{4(2l+1)(2l+3)}}, & l' = l+1 \ \& \ m' = m-1 \ \& \ l \geq 1, \\ \sqrt{\frac{(l+m)(l+m-1)}{4(2l-1)(2l+1)}}, & l' = l-1 \ \& \ m' = m-1 \ \& \ l \geq 1, \\ 0, & \text{otherwise}, \end{cases} \tag{4.3.56}$$

The treatment for $H_2 = \sin\theta\sin\phi$ is very similar to H_1. For the integral of ϕ, we have

$$\int_0^{2\pi} (e^{i(m+1)\phi})^* \sin\phi e^{im\phi} = -i\pi, \tag{4.3.57}$$

$$\int_0^{2\pi} (e^{i(m-1)\phi})^* \sin\phi e^{im\phi} = i\pi. \tag{4.3.58}$$

Together with the previous recurrence formula, we have the matrix representation for H_2 as

$$\langle Y_{l'}^{m'}|H_2|Y_l^m\rangle = \begin{cases} -\frac{i}{\sqrt{6}}, & l=m=0 \ \& \ l'=m'=1, \\ -i\sqrt{\frac{(l+m+2)(l+m+1)}{4(2l+1)(2l+3)}}, & l'=l+1 \ \& \ m'=m+1 \ \& \ l\geq 1, \\ i\sqrt{\frac{(l-m)(l-m-1)}{4(2l-1)(2l+1)}}, & l'=l-1 \ \& \ m'=m+1 \ \& \ l\geq 1, \\ -i\sqrt{\frac{(l-m+2)(l-m+1)}{4(2l+1)(2l+3)}}, & l'=l+1 \ \& \ m'=m-1 \ \& \ l\geq 1, \\ i\sqrt{\frac{(l+m)(l+m-1)}{4(2l-1)(2l+1)}}, & l'=l-1 \ \& \ m'=m-1 \ \& \ l\geq 1, \\ 0, & \text{otherwise,} \end{cases} \tag{4.3.59}$$

Finally, we have the full matrix representation of $H_f = H_0 - f_1H_1 - f_2H_2 - f_3H_3$. The matrix exponential can be computed in the same pattern as the previous section for each $\vec{f}$. Numerically, $\tilde{P}(f)$ is a three dimensional array with size $N \times N \times N$. $\tilde{P}_{k_1,k_2,k_3}$ is an approximation for $\tilde{P}(\vec{f})$ with $f_m = -\frac{i\pi k_m}{\beta}$ and $k_m = -\frac{N}{2}...\frac{N}{2}-1$ for $m = 1,2,3$. After computing each element in this three dimensional array, we apply a three dimensional inverse fast Fourier transform to obtain the resultant three dimensional array Q_{j_1,j_2,j_3}, which is a representation for $Q(\vec{R})$, with $j_m = -\frac{N}{2}...\frac{N}{2}-1$ and $R_m = \frac{2j_m}{N}$ in each dimension for $m = 1,2,3$.

Now we consider different configurations of the initial and final ends. For a free end or a fixed end, the state is the same as in the 1D case. Note that the basis in 1D corresponds to the spherical harmonics Y_l^m for $m = 0$. Therefore the entries corresponding to the coefficients of Y_l^0 is the same as the entries corresponding to the coefficients of $\hat{P}_l(x)$.

Next we deal with the case that the starting end makes an angle Θ with the z axis while the final end is free. In order to find the probability $Q(0,0,z)$ along the z axis, we can rotate the whole system by an angle Θ. The problem becomes finding the probability distribution along a line which makes an angle Θ with the z axis while the starting vector is fixed along the z direction, and the probability $Q(0,0,z)$ in our original problem is $Q(0, z\cos\Theta, z\sin\Theta)$ in the new system. The data can be drawn from the probability distribution in the $y - z$ plane. We can compute the probability distribution $Q(0, y, z)$ first. Numerically this corresponds to the plane $Q_{\frac{N}{2}+1,i,j}$ in our three dimensional array. On the 2D grid, we use bilinear interpolation to get the interpolated value.

Based on the distribution function $Q(\vec{r})$, thermodynamic quantities can be derived. The free energy of the ensemble in which the free end of the polymer is at position $\vec{r}$ is

$$F(\vec{r}) = -k_B T \ln Q(\vec{r}). \tag{4.3.60}$$

To keep the free end staying at the position $\vec{r}$, a force must be applied. The force is

$$\vec{f}(\vec{r}) = \nabla F(\vec{r}). \tag{4.3.61}$$

These quantities can be easily retrieved using finite difference methods.

4.4 Numerical results and applications

In this section, we present some numerical results for the exact WLC model to study several single-polymer control effects concerning remote positioning and force transmission, which are essential to biological and artificial nanomotors. Figure 4.1 shows the end to end distribution in 1D.

We then present the 3D end to end probability for different chain rigidity, or β. Results are taken on different y planes in Figure 4.2. With the exactly distribution at hand, we can discuss their applications.

4.4.1 Single-polymer 'flyfishing' by a local alignment at one end

In the study of biological nano-motors, it is long believed that a zippering-like control of a soft polymer linker at one end biases a motor leg at the other end of the linker to bind the track forward towards the zippering direction over opposite direction. As we shall show below, the exact WLC dynamics suggests that more effects pertinent to a motor's position-selective binding can occur generally. Figure 4.3 presents the probability distribution of one free end of a WLC polymer along the z axis when the other end is located at the origin and has different orientations for its tangent: the end at the origin either has free orientation or is aligned towards an angle $\theta = 90, 45, 0$ degree with reference to the positive z direction to mimic the zippering effect. For the free orientation and the vertical orientation ($\theta = 90$ degree), the free end has two symmetric most probable positions at a positive and a negative z value. When the fixed end is tilted from 90 to 45 degree and further 0 degree, the probability peak at negative z values virtually vanishes but the peak at the positive z value rises. This is in consistence with the well-known zippering-induced bias. However, an examination of the extent of accuracy by which aligning the end can position the other free end to a certain location reveals several new, distinct patterns.

Firstly, the full width at half maximum (FWHM) of the probability peaks becomes as small as about 2nm for the zero-degree alignment (Figure 4.4). This suggests the possibility of positioning the free end to the most probable location, or the critical z_c within an error of about 1nm for a polymer with a contour length of 10nm.

Secondly, such a precise site-selective positioning may occur for rather flexible polymers with $l_p/l_c = 0.3$ to 0.4, it does not necessarily require a rigid molecular rod ($l_p/l_c > 1$). Indeed the probability for locations around z_c peaks at intermediate values of persistence length (l_p) for a certain contour length (l_c) (Figure 4.5). The FWHM also becomes largely flat when l_p/l_c changes from 0.3 to 1. Hence a flexible polymer of l_p/l_c about 0.3 to 0.4 already accesses the regime of precise positioning.

Lastly, the precise positioning can be modulated by adjusting the end alignment and polymer rigidity. Changing the end alignment from 90 degree to 0 degree shrinks the FWHM of the probability peaks monotonically (Figure 4.4), hence improves the positioning accuracy. The most probable location of positioning z_c may be modulated too by the alignment, or by changing the polymer rigidity, which may be done by adjusting the solution conditions like ion concentrations, pH values, temperature etc..

Hence aligning a flexible polymer of $l_p/l_c = 0.3$ to 0.4 and $l_c = 10$nm at one end positions the other free end to a most probable position of z_c about 8nm within an error of about 1nm along the path parallel to the alignment. Such a surprisingly precise positioning to a unique, remote location (relative to the length of the polymer) is achieved by controlling the flexible polymer at one end, essentially resembling the art of flyfishing but at molecular level.

The direction and speed of a bipedal nanomotor is largely decided by the position-selective binding of its legs, which is in turn affected by the mechanics of the inter-leg polymer linker. Fine-tuning a nanomotor into the regime of molecular flyfishing may improve the motors directional fidelity, which is important for making motors of high energy efficiency. The flyfishing effect also helps improve a motors speed by accelerating forward binding of its legs: From the free or vertical orientation to zero-degree orientation, the FWHM drops by a factor of about 2nm (Figure 4.4) and two most-probable positions merge into one. This amounts to a drop of a factor of about 4nm for the search volume of a diffusing leg at the linkers free end. A rough estimation for the first-passage time suggests more than two-fold increase in the legs forward binding rate by the flyfishing. Hence the single-polymer flyfishing likely plays a role in biological nanomotors that possess high directional fidelity, energy efficiency or high speed. This is consistent with the observation of zippering in biomotor kinesin, which is the smallest bipedal motor found in biology but has a directional fidelity of more than 90%, an efficiency of 60% - 80% and a speed of a few micrometers per second.

4.4.2 Single-polymer power stroke and intra-chain force transmission

The polymer zippering effect found in biomotors is often likened to the power stroke of a piston in macroscopic engines. However, validity of the notation of power stroke for a polymer linker is questioned since any effective transmission of force along a soft polymer is hard to imagine. The molecular flyfishing effect lends a support to the single-polymer power stroke. Consider again the flexible polymer of $lp/lc = 0.3$ to 0.4 and $l_c = 10$nm. Keeping an alignment at one end requires application of a force, which may be regarded as an input. The outcome at the other end is a combination of two effects along the direction of the alignment. First, the two most probable positions before the alignment converge into one, resulting in a forward displacement of $2 \times z_c \approx 16$nm for the free end. Second, the alignment causes a free-energy difference of $\sim 11k_BT$ over the displacement, amounting to a force transmission of 2.75 pN (for room temperature of $25°C$) from the controlled end to the free end. The directional displacement and the force transmission of a few pN combine to suggest a piston-like power stroke in real sense.

Furthermore, a finite force transmission is possible by the flysfishing effect even for very soft polymers. The force transmission becomes zero only for zero displacement, which is made clear below using the Gaussian chain model and the freely joint chain model, and is confirmed by the exact solution. (Figure 4.7) For the same displacement of 16nm, the alignment causes a free-energy difference of $\sim 7k_BT$ for a polymer even with $l_p/l_c \to 0$. This implies a minimum force transmission of $1.75pN$ by the flyfishing for arbitrary polymers. Interestingly, the minimum free-energy difference of $\sim 7k_BT$ is close to the experimentally observed entropy difference ($\sim 6k_BT$) associated with the directional bias of biomotor kinesin, which possesses an inter-leg peptide linker of $l_c \approx 10$nm and $l_p < 1$nm and makes a forward leg binding over a regular displacement of ~ 16nm.

The end alignment-induced free-energy change may be derived analytically for

Gaussian chains and freely joint chains. Note that the Gaussian chain here is the discrete Gaussian chain which is made up of Kuhn bonds that are assumed Gaussian distributed, and is not the continuous limit of the freely joint chain.

The end-to-end distance distribution of a Gaussian chain under the approximation of one-dimensional random walk is:

$$P(N,z) = \frac{1}{\sqrt{2\pi N}} e^{-\frac{z^2}{2N}}. \tag{4.4.1}$$

N is number of the total steps. Each step is either in forward or backward direction and z is the position after making N steps, so $-N \leq z \leq N$. If the first step is toward the direction of positive z, the end-to-end distance distribution function becomes P' as

$$P'(N+1,z) = P(N,z-1), \quad P'(N+1,-z) = P(N,-z-1) \tag{4.4.2}$$

Therefore the free energy difference between z and $-z$ is .

$$\delta F(z) = k_B T \ln \frac{P'(N+1,z)}{P'(N+1,-z)} = k_B T \ln \frac{P(N,z-1)}{P(N,-z-1)} = 2qk_BT, \quad q = z/N. \tag{4.4.3}$$

For two or three dimensional random walks, the factor 2 becomes 4 and 6, respectively.

For a freely joint chain without the Gaussian approximation, the end-to-end distance distribution is:

$$P(N,z) = Ce^{-\sum\limits_{s=1}^{z/2} \ln(\frac{N/2+s}{N/2+1-s})}. \tag{4.4.4}$$

Here C is the normalization coefficient. Following a derivation similar to the previous one for the Gaussian approximation, the free energy difference can still be written in a simple analytical form:

$$\delta F(z) = k_B T \ln \frac{N+1+z}{N+1-z} = k_B T \ln \frac{1+q}{1-q} = 2qk_BT(1+\sum_{i=1} \frac{q^{2i}}{2i+1}), \quad q = \frac{z}{N+1}. \tag{4.4.5}$$

The free energy difference is higher than the Gaussian chain result but decays to the latter for $q \to 0$. Both the Gaussian chain and freely joint chain models predict a finite $\delta F(z)$ for non-zero z values.

For the worm like chain, as shown in Figure 4.6, Fixing $l_c = 10$nm, for small l_p, the probability $Q(z)$ can be well described by the Gaussian function as $Q(z) = \frac{1}{10^3 \sigma\sqrt{2\pi}} \exp(-\frac{(z/10-\mu)^2}{2\sigma^2})$, by taking the mean and variance of the worm like chain data, we can do a linear interpolation and get $\mu = 0.1021 l_p$ and $\sigma^2 = 0.0670 l_p$. Taking $l_p \to 0$, we can compute $\Delta F_{Gauss}(z) = 0.30478z$

4.4.3 Site-selective dissociation by intra-chain force

Unlike the force transmission through a free diffusing end, the force transmission along a polymer with both ends restricted is well known. This type of force transmission occurs in the inter-leg linker of a bipedal nanomotor when its two legs are bound to the track, and has been suggested to be a cause for leg dissociation. This is supported by the exact WLC solution. Consider again the example of a flexible polymer of $l_p/l_c = 0.3$ to 0.4 and $l_c = 10$nm for the inter-leg linker of a bipedal nanomotor. Imagine the nanomotor on a linear track along the z axis with one leg bound at the origin $z = 0$ and the other leg near the most probable positions allowed by the inter-leg linker, namely $z = +z_c$ and z_c. Note that z_c remains close to 8nm regardless of end orientation as predicted by the exact WLC solution. When the second leg is bound near $z = z_c$, the intra-linker force is increased when the linkers end at the origin is changed from free or vertical orientation to a forward orientation parallel to the track (i.e., $\theta = 0$) (Figure 4.8). The force becomes more than 12 pN at $\theta = 0$, which is rather sufficient to dissociate a leg off the track for many biological nanomotors. This is like the normal fishing in which the fishing rod is tilted to pull a fish out of water. Moreover, the intra-chain forces on the two ends have opposite direction, which leads to dissociation of one leg but not the other depending on their position if the leg-track binding is responds to forward or backward force in an symmetric way. The site-selective leg dissociation and the asymmetric leg-track binding are common features in biological nanomotors. For an alignment with an intermediate angle between 0 and 90 degree, the intra-linker force may be distributed on the two ends in two dimensions in a highly asymmetric way. As a consequence, a strong leg may be dissociated by the linker alignment at a weak leg as has been observed for biological motors.

4.4.4 New force-extension formula

The intra-chain force dissociating a leg depends on the end-to-end distance as well as a polymers persistence length and contour length. To facilitate nanomotor studies, we obtain an fitting formula capturing these dependences by fitting the force-extension results from the exact WLC solution.

$$\begin{aligned}\frac{f(z)}{k_B T} = \frac{\beta}{l_c}\left(\frac{z}{l_c} + \frac{1}{4(1 - z/l_c)^2} - \frac{1}{4}e^{\frac{5.1}{\beta^{0.3}}\frac{z}{l_c}}\right) \\ + \frac{1}{l_c\, e^{0.94\beta}}\left(52.4\left(\frac{z}{l_c}\right)^2 - 35.8\right)\left(1 - e^{-\frac{10}{\beta}\frac{z}{l_c}}\right).\end{aligned} \tag{4.4.6}$$

This new formula differs from previous proposed force-extension formulas in two aspects. Firstly, at the long chain limit $\beta = l_c/l_p \to \infty$, the new fitting formula recovers the empirical Marko-Siggia formula,

$$\frac{f(z)}{k_B T} = \beta/lc(z/l_c + (1z/l_c)^{-2}/4 - 1/4). \tag{4.4.7}$$

The new formula also captures the exact WLC solution better than the latter for short chains with low β values. We note that short chains with $\beta < 30$ are most relevant to nanomotors. As a matter of fact, the mechanics of such short biopolymers are much less known to date than long ones due to difficulty of single-molecule measurement. Applicable to both long and short polymers, the new formula thus useful not only for nanomotor studies but also for interpreting single-molecule mechanical experiments on short polymers. Secondly, the Marko-Siggia formula and some later improvements apply only to stretched polymers which have an end-to-end distance larger than the thermodynamically most probable extension. The new formula applies not only to stretched polymers but also compressed polymers which have an end-to-end distance smaller than the most probable extension. Different combinations of the end-to-end distance, contour length and persistence length may switch a polymer from the stretch regime to the compression regime reverses direction of the intra-chain force, which is captured by the exact solution and the new formula (Figure 4.9). Figure 4.10 shows a comparison for the infomation at $z = 5$nm and $z = 8$nm for different l_p. We can see that in the case that both ends are free, the fitting formula provides an excellent approximation for the force near these sites.

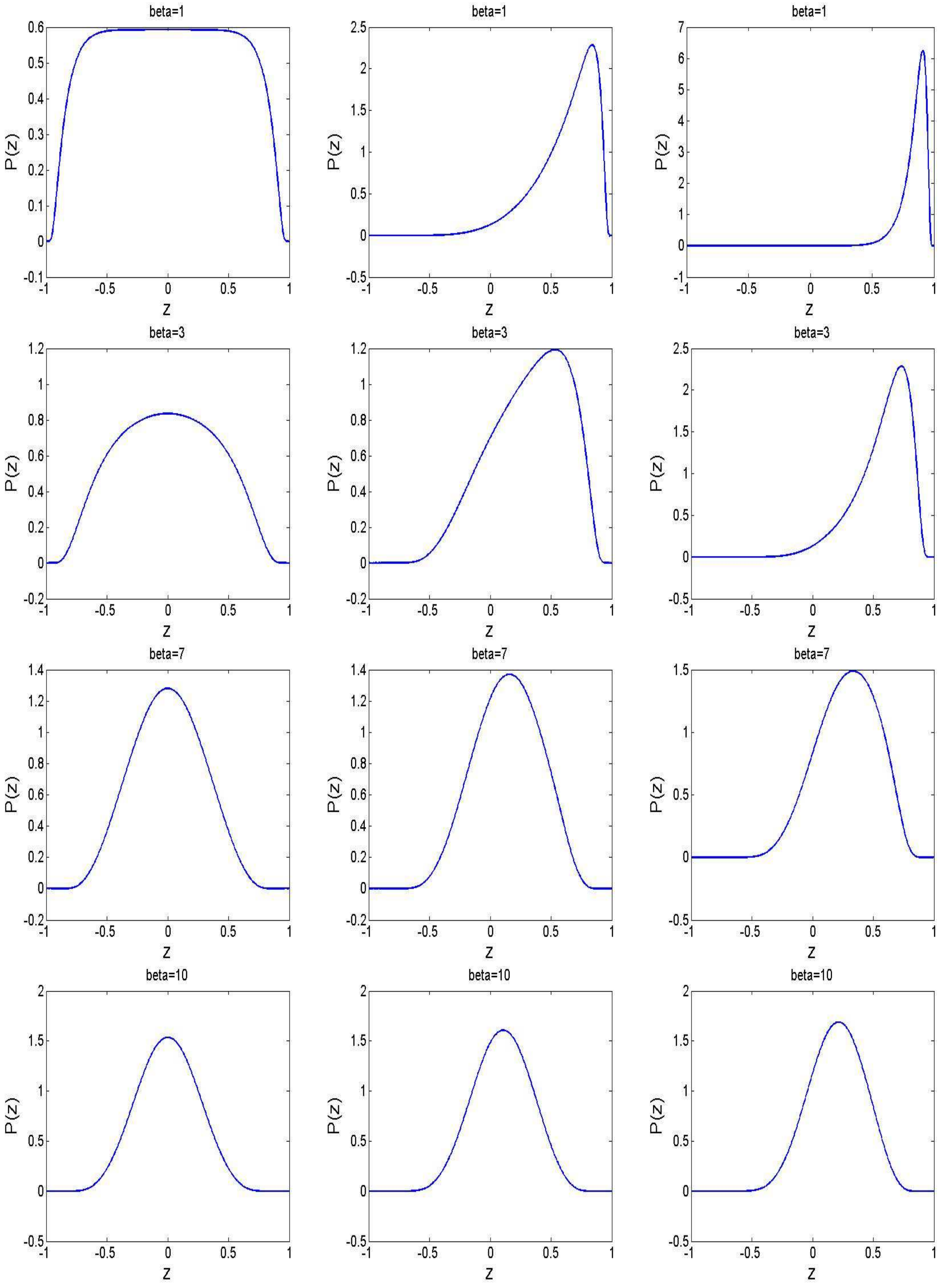


Figure 4.1: The probability distribution integrated over a plane $P(z)$. Left column: both ends are free. Middle column: the initial end is fixed along the z axis. Right column: both ends are fixed along the z axis.

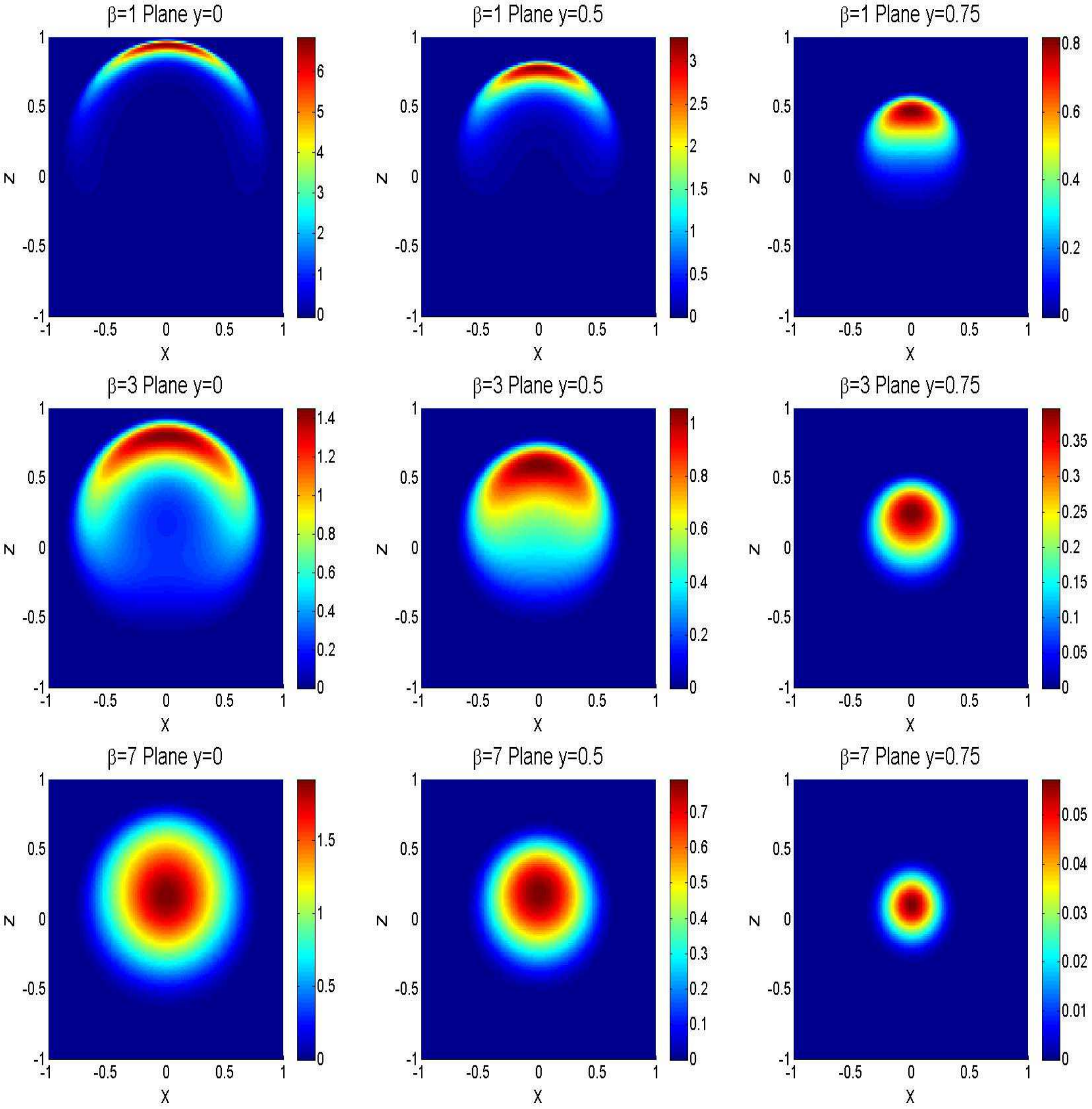


Figure 4.2: The probability distribution $P(x, y, z)$ taken at different y planes for different values of β, fixing $l_c = 1$

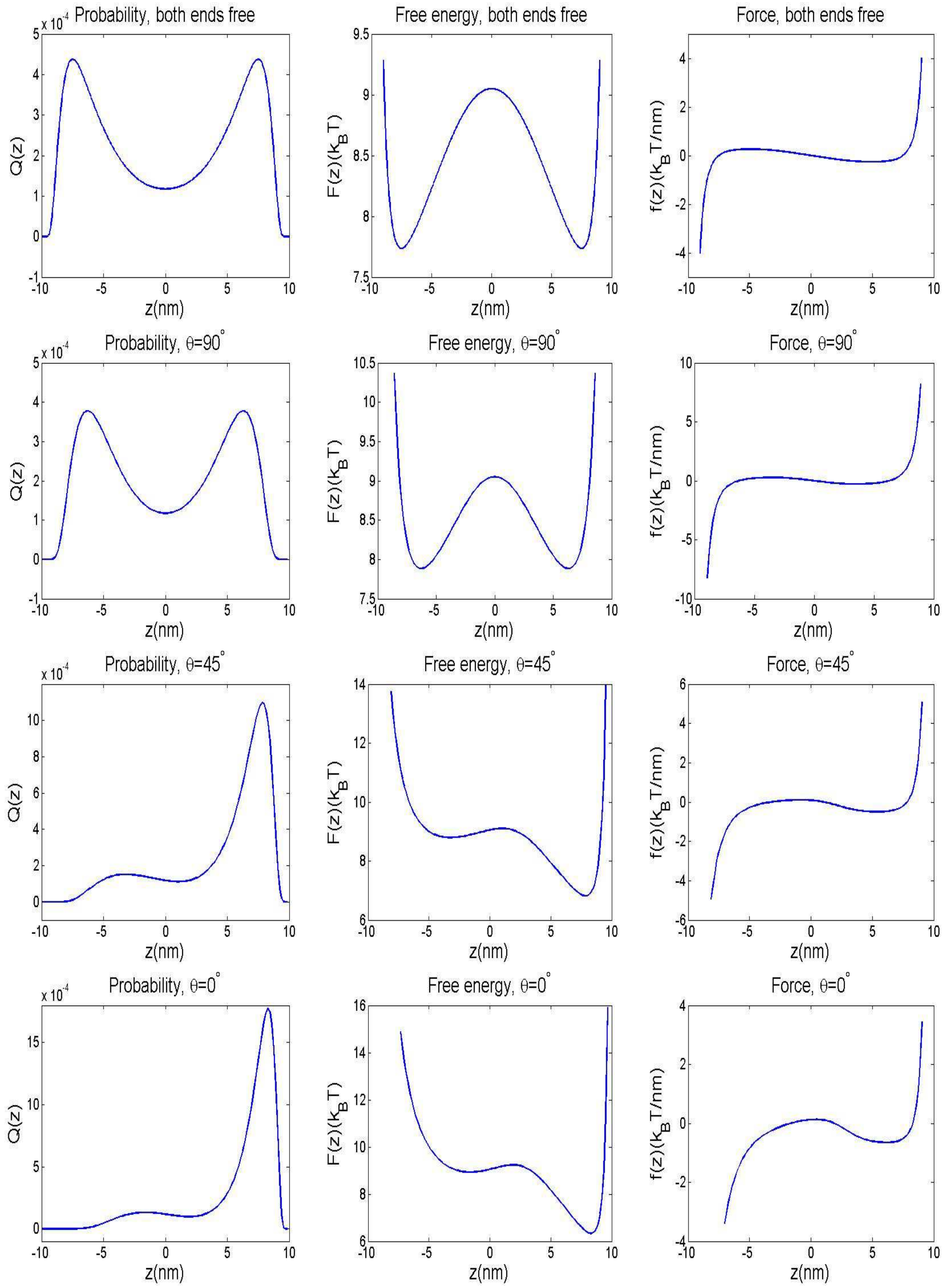


Figure 4.3: The probability distribution $Q(z)$, energy $F(z)$ and force $f(z)$ for the case $l_p = 4\text{nm}$ and $l_c = 10\text{nm}$, with the starting direction being free or make an angle θ with the z axis.

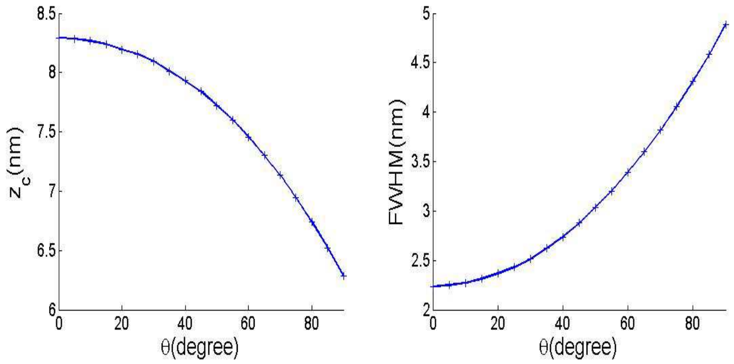


Figure 4.4: The critical z where $Q(z)$ reaches its maximum and the full width half maximum for different θ where the starting vector makes an angle θ with the z-axis. Here $l_c = 10$nm and $l_p = 4$nm.

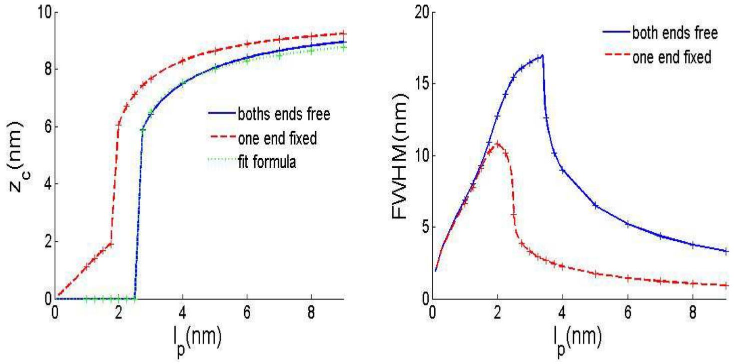


Figure 4.5: The critical z where $Q(z)$ reaches its maximum and the full width half maximum for different l_p

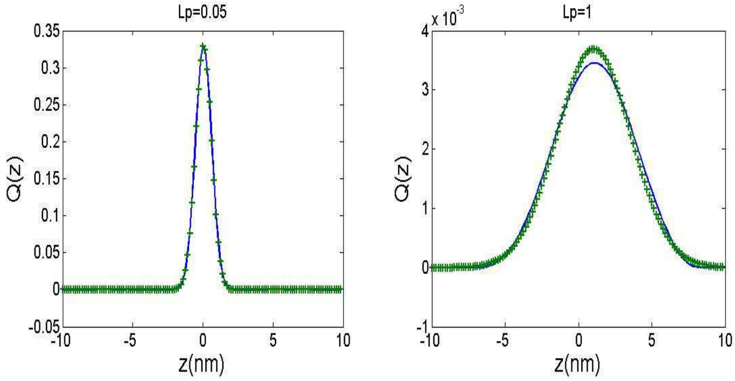


Figure 4.6: The probability distribution Q(z) (blue curve) together with the Gauss approximation for different l_p

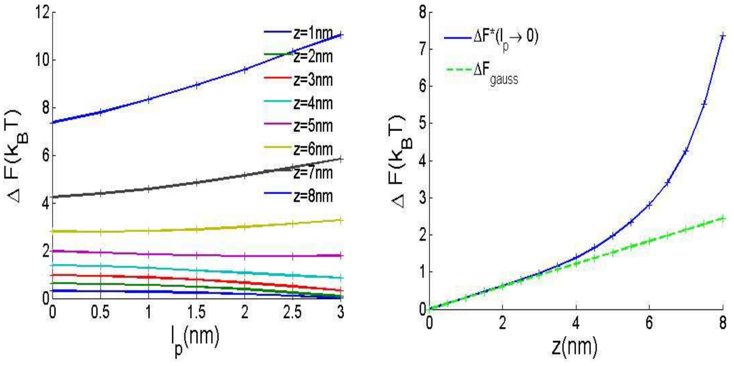


Figure 4.7: The energy difference ΔF for a chain with one end fixed along the z axis. In the left figure, for each value of z, $\Delta F \to \Delta F^*$ as $l_p \to 0$. The dependence of ΔF^* on z is plotted in the right figure(blue curve). The probability $Q(z)$ can be well described by the Gaussian function. Taking $l_p \to 0$, we can compute $\Delta F_{Gauss}(z) = 0.30478z$(green curve). The energy difference agrees near $z = 0$nm.

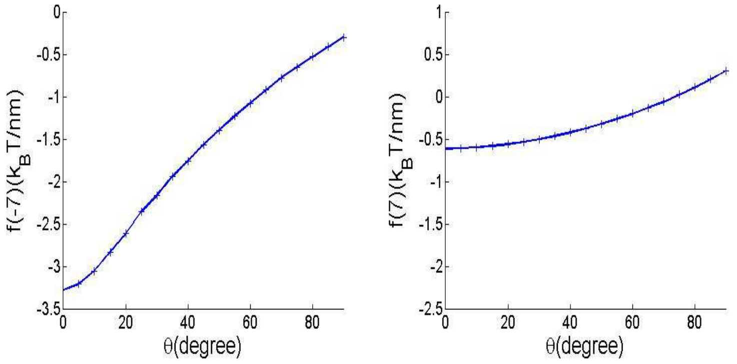


Figure 4.8: The force at $z = -7$nm and $z = 7$nm, for different θ

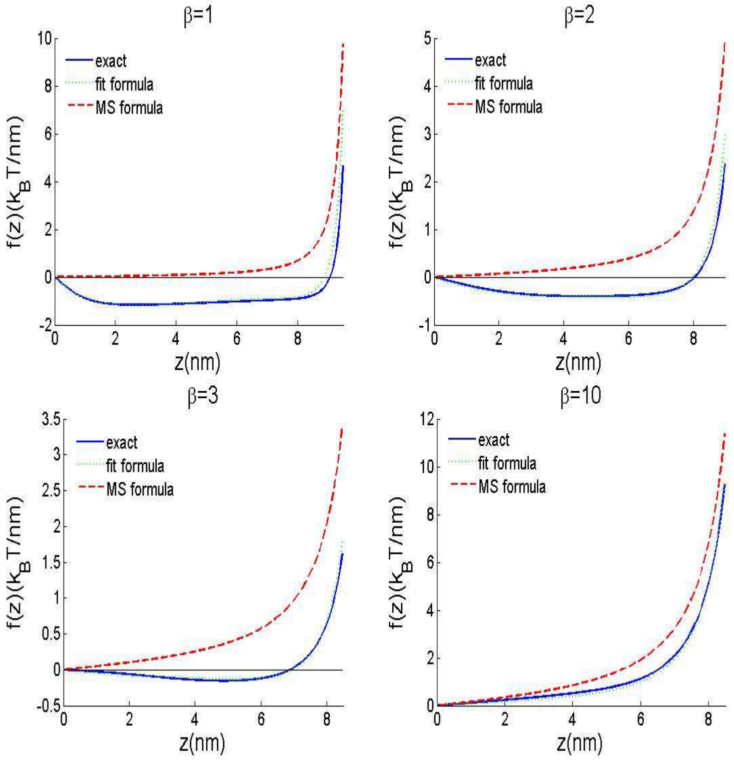


Figure 4.9: The comparison between the new fit formula and the force $f(z)$ obtained by numerical methods for different β.

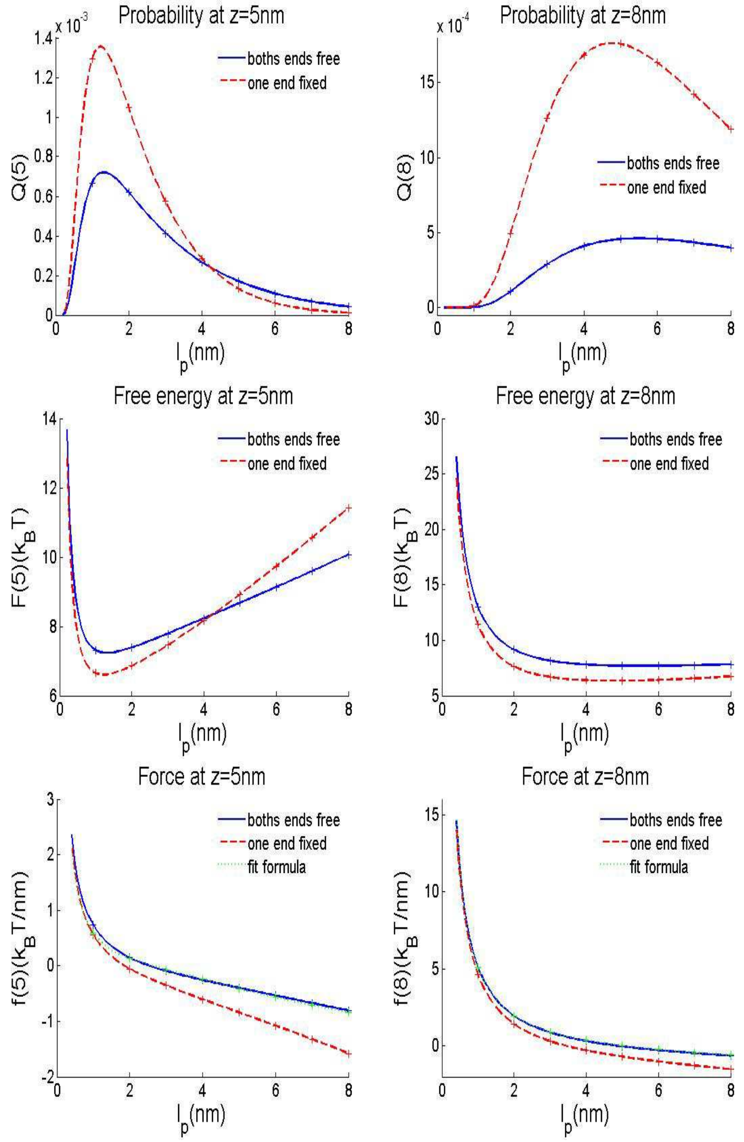


Figure 4.10: The probability distribution, energy, force at position $z = 5\text{nm}$ and $z = 8\text{nm}$, for different l_p

Chapter 5

Conclusion and Future Perspectives

This thesis is devoted to numerical simulations and application for some biological relevant problems, namely the collective motion and single polymer statistics. The results obtained for the subjects will be summarized and possible topics for future work will also be discussed.

In chapter 2, we use simple particle to model spermatozoon and try to understand their collective motion. By applying a repulsive force among individuals and apply the alignment rule in the Vicsek model, we used GPU acceleration to simulate a large system of particles. Simulations are done in both 2D and 3D. Without the repulsion, we can see the formations of clusters especially at the boundaries. As we increase the radius of repulsion and the repulsive force, we can see that a more homogeneous fluid structure can be formed. To study the global macroscopic pattern in 2D, we defined the empirical function and try to construct a differential equation for the density and orientation. The construction is done using a mean field limit as the first step followed by the hydrodynamic limit making use of a concept called the generalized collision invariant which is a variant to the concept in gas dynamics. Numerical methods are proposed to solve the macroscopic model. Again GPU acceleration is used in the finite difference method. Comparison between the two models are made and excellent agreement validates our derivation. In the future, better hardware and parallel algorithm can be used to simulate up to billions of particles and the 3D macroscopic model together with its simulation is still missing.

For a better particle model, we also used the global structure as the starting point, which is an active self propelling fluid. Using the concept of smoothed particle hydrodynamics, which is a particle methods for the fluid, we can discretize the Navier-Stokes equation and yields the interaction rules for individual particles. The forces consists of an alignment term, a repulsive term and a self propulsion term which agrees with our previous model. Numerical results are provided showing that this model would produce a more homogeneous structure than the previous model. Future studies may include finding proper discretizations for a divergence free flow. We may also start with a macroscopic model which can take account for the vortices and waves appears in the spermatozoon suspension pattern.

In chapter 3, we consider individuals as polymers and take the steric interaction into consideration. Firstly we model the polymer as rigid rods. GPU acceleration help us to simulate a system of rigid bodies. While the only mechanism affecting the self propelling rods is the volume exclusion effect, we still can observe different patterns including collective motion. Results with different boundary conditions in 2D show good agreement with experiments with bristle-bot, while 3D simulations are also provided. By changing the aspect ratio, the repulsive force or boundary conditions, various patterns can be observed. It would be more realistic to model spermatozoon as a chain with a corresponding rigidity. To better understand the global structure, it is also helpful to derive a macroscopic model from it for further studies.

We then modelled the particles by including hydrodynamic forces into the system and use a finite element method to study the flow in 2D. A good parallel algorithm is required for future studies in order to simulate a large system of particles. In summary, we studied different models with different complexity, hoping to understand the individual interactions that would produce a global collective pattern. The number of particles that can be simulated decrease while we increase the complexity of the model. Future studies will be conducted to find a model of the spermatozoon suspension with an appropriate level of accuracy, such that understanding the individual interaction rules can help us capture the global behaviour of collective motion with circular waves and whirlpools. Since the collective motion of spermatozoon, or massal motility is the only parameter of semen sample that shows a goods agreement with male fertility, Understanding it would help us to produce an automated assessment process of semen fertility.

In chapter 4, we focus on the single polymer statistics instead a collection of polymers. Semi-flexible polymers are modelled using the worm like chain model. The statistics of the worm like chain model play an important role in the field of nanomotors. By mapping the end to end distribution to a quantum rotor on a unit sphere, we use a path integral approach to get the 3D end to end distribution. The results suggest feasibility of these single-polymer controls up to a surprising accuracy even for a rather soft polymer, which rationalizes high optimality previously found for some biological nanomotors and reveals new mechanistic regimes to improve performance of artificial nanomotors. This study demonstrates the capacity of the exact WLC model to serve as a general working framework to study motor-relevant polymer effects.

Also, a new force-extension formula is obtained from the exact solution of the WLC model. The new formula relation has an improved accuracy over the widely used approximate formula for stretched polymers, and also is applicable to compressed polymers.

Future studies include understanding the chain statistics with different start and end orientations, also semi-flexible polymers can be modelled with more sophisticated models such as the helical worm like chain model, or models with kinks.

Bibliography

[1] Abramowitz M, Stegun I A. Handbook of Mathematical Functions. New York: Dover, 1965.

[2] Agueh M, Illner R, Richardson A. Analysis and simulations of a refined flocking and swarming model of Cucker-Smale type. Kinetic and Related Models, 2011, 4(1): 1-16.

[3] Aldana M, Dossetti V, Huepe C, et al. Phase transitions in systems of self-propelled agents and related network models. Physical Review Letters, 2007, 98(9): 095702.

[4] Anderson J A, Lorenz C D, Travesset A. General purpose molecular dynamics simulations fully implemented on graphics processing units. Journal of Computational Physics, 2008, 227(10): 5342-5359.

[5] Aoki I, A simulation study on the schooling mechanism in fish. Bulletin of the Japan Society of Scientific Fisheries, 1982, 48: 1081-1088.

[6] Barbaro A B T, Degond P. Phase transition and diffusion among socially interacting self-propelled agents. arXiv preprint arXiv:1207.1926, 2012.

[7] Baglietto G, Albano E V. Nature of the order-disorder transition in the Vicsek model for the collective motion of self-propelled particles. Physical Review E, 2009, 80(5): 050103.

[8] Baskaran A, Marchetti M C. Nonequilibrium statistical mechanics of self-propelled hard rods. Journal of Statistical Mechanics: Theory and Experiment, 2010 ,04: P04019.

[9] Batchelor G K. Slender-body theory for particles of arbitrary cross-section in Stokes flow. Journal of Fluid Mechanics, 1970, 44(03): 419-440.

[10] Bertin E, Droz M, Gregoire G. Boltzmann and hydrodynamic description for self-propelled particles. Physical Review E, 2006, 74(2): 022101.

[11] Bertin E, Droz M, Gregoire G. Hydrodynamic equations for self-propelled particles: microscopic derivation and stability analysis. Journal of Physics A: Mathematical and Theoretical, 2009, 42(44): 445001.

[12] Ben-Jacob E, Levine H. Self-engineering capabilities of bacteria. Journal of the Royal Society Interface, 2006, 3(6): 197-214.

[13] Ben-Jacob E, Levine H. The artistry of nature. Nature, 2001, 409(6823): 985-986.

[14] Birnir B. An ODE model of the motion of pelagic fish. Journal of Statistical Physics, 2007, 128(1-2): 535-568.

[15] Bolley F, Cañizo J A, Carrillo J A, Mean-field limit for the stochastic Vicsek model, Appl. Math. Lett, 2011(25): 339-343.

[16] Bolley F, Caizo J A, Carrillo J A. Mean-field limit for the stochastic Vicsek model. Applied Mathematics Letters, 2012, 25(3): 339-343.

[17] Buhl J, Sumpter D J T, Couzin I D, et al. From disorder to order in marching locusts. Science, 2006, 312(5778): 1402-1406.

[18] Bostan M, Carrillo J A. Asymptotic fixed-speed reduced dynamics for kinetic equations in swarming. arXiv preprint arXiv:1202.6557, 2012.

[19] Carrillo J A, Fornasier M, Toscani G, et al. Particle, kinetic, and hydrodynamic models of swarming. Mathematical modeling of collective behavior in socio-economic and life sciences, 2010: 297-336.

[20] Carrillo J A, Fornasier M, Rosado J, et al. Asymptotic flocking dynamics for the kinetic Cucker-Smale model. SIAM Journal on Mathematical Analysis, 2010, 42(1): 218-236.

[21] Camazine S, Scott, ed. Self-Organization in Biological Systems. Princeton University Press, 2003.

[22] Chaté H, Ginelli F, Grégoire G, Raynaud F, Collective motion of self-propelled particles interacting without cohesion. Phys. Rev. E, 2008, 77: 046113.

[23] Chaté H, Ginelli F, Grégoire G, et al. Modeling collective motion: variations on the Vicsek model. The European Physical Journal B, 2008, 64(3-4): 451-456.

[24] Chepizhko A A, Kulinskii V L, Holovatch Y, et al. The kinetic regime of the Vicsek model. Aip Conference Proceedings. 2009, 1198(1): 25.

[25] Chuang Y L, D'Orsogna M R, Marthaler D, A. L. Bertozzi and L. S. Chayes, State transitions and the continuum limit for a 2D interacting, self-propelled particle system. Physica D, 2007, 232: 33-47.

[26] Cisneros L H, Cortez R, Dombrowski C, et al. Fluid dynamics of self-propelled microorganisms, from individuals to concentrated populations. Experiments in Fluids, 2007, 43(5): 737-753.

[27] Cottet G H, Koumoutsakos P D. Vortex Methods: Theory and Practice. Cambridge university press, 2000.

[28] Colin F, Egli R, Lin F Y. Computing a null divergence velocity field using smoothed particle hydrodynamics. Journal of Computational Physics, 2006, 217(2): 680-692.

[29] Couzin I D, Krause J, James R, Ruxton G D and Franks N R, Collective Memory and Spatial Sorting in Animal Groups, J. theor. Biol, 2002, 218: 1-11.

[30] Czirók A, Ben-Jacob E, Cohen I, et al. Formation of complex bacterial colonies via self-generated vortices. Physical Review E, 1996, 54(2): 1791.

[31] Czirók A, Vicsek M, Vicsek T. Collective motion of organisms in three dimensions. Physica A, 1999,264(1) : 299-304.

[32] Cucker F, Smale S, Emergent behavior in flocks, IEEE Transactions on Automatic Control, 2007, 52: 852-862.

[33] D'Orsogna M R, Chuang Y L, Bertozzi A L, et al. Self-propelled particles with soft-core interactions: patterns, stability, and collapse. Physical review letters, 2006, 96(10): 104302.

[34] Dalmao F, Mordecki E. Cucker-smale flocking under hierarchical leadership and random interactions. SIAM Journal on Applied Mathematics, 2011, 71(4): 1307-1316.

[35] Dalrymple R A, Rogers B D. Numerical modeling of water waves with the SPH method. Coastal engineering, 2006, 53(2): 141-147.

[36] Daniels H E. XXI.The Statistical Theory of Stiff Chains. Proceedings of the Royal Society of Edinburgh. Section A. Mathematical and Physical Sciences, 1952, 63(03): 290-311.

[37] Degond P, Dimarco G, Mac T B N, Wang N. Macroscopic models of collective motion with repulsion. arXiv preprint arXiv:1404.4886, 2014.

[38] Degond P, Frouvelle A, Liu J G, Macroscopic limits and phase transition in a system of self-propelled particles. J. Nonlinear Sci. 2013, 23: 427-456.

[39] Degond P, Frouvelle A, Liu J G, Phase transitions, hysteresis, and hyperbolicity for self-organized alignment dynamics, preprint. arXiv:1304.2929.

[40] Degond P, Liu J G, Motsch S, Panferov V, Hydrodynamic models of self-organized dynamics: derivation and existence theory. Methods Appl. Anal. 2013, 20: 089-114.

[41] Degond P, Motsch S, Continuum limit of self-driven particles with orientation interaction, Math. Models Methods Appl. Sci., 18 Suppl. 2008, 1193-1215.

[42] Degond P, Peyrard P F, Russo G, et al., Polynomial upwind schemes for hyperbolic systems, C. R. Acad. Sci. Paris, Ser I, 1999, 328: 479-483

[43] Desbrun M, Gascuel M P. Smoothed particles: A new paradigm for animating highly deformable bodies. Computer Animation and Simulation96. Springer Vienna, 1996: 61-76.

[44] Dusenbery D B. Minimum size limit for useful locomotion by free-swimming microbes. Proceedings of the National Academy of Sciences, 1997, 94(20): 10949-10954.

[45] Dworkin M, Kaiser D. Cell interactions in myxobacterial growth and development. Science, 1985, 230(4721): 18-24.

[46] Fehske H, Schneider R and Weisse A, Computational Many-Particle Physics, Springer Verlag, 2007.

[47] Frouvelle A, A continuum model for alignment of self-propelled particles with anisotropy and density-dependent parameters. Math. Mod. Meth. Appl. Sci, 2012, 22(07)

[48] Glowinski R, Pan T W, Hesla T I, et al. A distributed Lagrange multiplier/fictitious domain method for particulate flows. International Journal of Multiphase Flow, 1999, 25(5): 755-794.

[49] Gautrais J, Jost C, Theraulaz G. Key Behavioural Factors in a Self-Organised Fish School Model. Annales Zoologici Fennici. Finnish Zoological and Botanical Publishing, 2008, 45(5): 415-428.

[50] Ghosh K, Carri G A, Muthukumar M. Configurational properties of a single semiflexible polyelectrolyte. The Journal of Chemical Physics, 2001, 115(9): 4367-4375.

[51] Gingold R A, Monaghan J J. Smoothed particle hydrodynamics-theory and application to non-spherical stars. Monthly notices of the royal astronomical society, 1977, 181: 375-389.

[52] Grégoire G, Chaté H, Onset of collective and cohesive motion. Phys. Rev. Lett, 2004, 92: 025702.

[53] Grünbaum D, Okubo A. Modelling social animal aggregations. Frontiers in mathematical biology. Springer Berlin Heidelberg, 1994: 296-325. MLA

[54] Gueron S, Levin S A, Rubenstein D I. The dynamics of herds: from individuals to aggregations. Journal of Theoretical Biology, 1996, 182(1): 85-98

[55] Ha S Y, Levy D. Particle, kinetic and fluid models for phototaxis. Discret. Contin. Dyn. Syst. B, 2009, 12: 77-108.

[56] Haines B M, Aranson I S, Berlyand L, et al. Effective viscosity of dilute bacterial suspensions: a two-dimensional model. Physical biology, 2008, 5(4): 046003.

[57] Helbing D, Keltsch J, Molnar P. Modelling the evolution of human trail systems. Nature, 1997, 388(6637): 47-50.

[58] Helbing D, Farkas I, Vicsek T. Simulating dynamical features of escape panic. Nature, 2000, 407(6803): 487-490.

[59] Henkes S, Fily Y, Marchetti M C, Active jamming: Self-propelled soft particles at high density. Phys. Rev. E, 2011, 84: 040301.

[60] Hernandez-Ortiz J P, Stoltz C G, Graham M D. Transport and collective dynamics in suspensions of confined swimming particles. Physical Review Letters, 2005, 95(20): 204501.

[61] Hernandez-Ortiz J P, Underhill P T, Graham M D, Dynamics of confined suspensions of swimming particles. J. Phys. Condens. Matter, 2009, 21: 204107.

[62] Hockney R W and Eastwood J W, Computer Simulation Using Particles. Institute of Physics Publishing, 1988.

[63] Huepe C, Aldana M. Intermittency and clustering in a system of self-driven particles. Physical review letters, 2004, 92(16): 168701.

[64] Hoover W G. Isomorphism linking smooth particles and embedded atoms. Physica A: Statistical Mechanics and its Applications, 1998, 260(3): 244-254.

[65] Hu H H, Joseph D D, Crochet M J. Direct simulation of fluid particle motions. Theoretical and Computational Fluid Dynamics, 1992, 3(5): 285-306.

[66] Hu H H. Direct simulation of flows of solid-liquid mixtures. International Journal of Multiphase Flow, 1996, 22(2): 335-352.

[67] Huth A, Wissel C. The simulation of the movement of fish schools. Journal of theoretical biology, 1992, 156(3): 365-385.

[68] Jadbabaie A, Lin J, Morse A S. Coordination of groups of mobile autonomous agents using nearest neighbor rules. Automatic Control, IEEE Transactions on, 2003, 48(6): 988-1001.

[69] Janela J, Lefebvre A, Maury B. A penalty method for the simulation of fluid-rigid body interaction. ESAIM: Proceedings. EDP Sciences, 2005, 14: 115-123.

[70] Kansa E J. Multiquadricsa scattered data approximation scheme with applications to computational fluid-dynamicsI surface approximations and partial derivative estimates. Computers and Mathematics with applications, 1990, 19(8): 127-145.

[71] Kansa E J. MultiquadricsA scattered data approximation scheme with applications to computational fluid-dynamicsII solutions to parabolic, hyperbolic and elliptic partial differential equations. Computers and mathematics with applications, 1990, 19(8): 147-161.

[72] Koch D L, Subramanian G, Collective hydrodynamics of swimming microorganisms: Living fluids. Annu. Rev. Fluid Mech, 2011, 43: 637-659.

[73] Kratky O, Porod G. Röntgenuntersuchung gelöster fadenmoleküle. Recueil des Travaux Chimiques des Pays-Bas, 1949, 68(12): 1106-1122.

[74] Lattanzio J C, Monaghan J J, Pongracic H, et al. Interstellar cloud collisions. Monthly Notices of the Royal Astronomical Society, 1985, 215(1): 125-147.

[75] Lauga E, Powers T R. The hydrodynamics of swimming microorganisms. Reports on Progress in Physics, 2009, 72(9): 096601.

[76] Lee E S, Moulinec C, Xu R, et al. Comparisons of weakly compressible and truly incompressible algorithms for the SPH mesh free particle method. Journal of Computational Physics, 2008, 227(18): 8417-8436.

[77] Lindholm E, Nickolls J, Oberman S, et al. NVIDIA Tesla: A unified graphics and computing architecture. IEEE micro, 2008, 28(2): 39-55.

[78] Lighthill M J. On the squirming motion of nearly spherical deformable bodies through liquids at very small Reynolds numbers. Communications on Pure and Applied Mathematics, 1952, 5(2): 109-118.

[79] Lefebvre A. Fluid-particle simulations with FreeFem++. ESAIM: Proceedings. EDP Sciences, 2007, 18: 120-132.

[80] Liu Z, Guo L. Connectivity and synchronization of Vicsek model. Science in China Series F: Information Sciences, 2008, 51(7): 848-858.

[81] Lucy, L.B. A numerical approach to the testing of fusion process. Astronomical Journal, 1977, 88: 1013-1024.

[82] Major P F, Dill L M. The three-dimensional structure of airborne bird flocks. Behavioral Ecology and Sociobiology, 1978, 4(2): 111-122.

[83] Mendelson N H, Bourque A, Wilkening K, et al. Organized cell swimming motions in Bacillus subtilis colonies: patterns of short-lived whirls and jets. Journal of bacteriology, 1999, 181(2): 600-609.

[84] Micikevicius P. 3D finite difference computation on GPUs using CUDA. Proceedings of 2nd Workshop on General Purpose Processing on Graphics Processing Units. ACM, 2009: 79-84.

[85] Mogilner A, Edelstein-Keshet L, Bent L and Spiros A, Mutual interactions, potentials, and individual distance in a social aggregation. J. Math. Biol, 2003, 47: 353-389.

[86] Moler C, Van Loan C. Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later. SIAM review, 2003, 45(1): 3-49.

[87] Molina J J, Nakayama Y, Yamamoto R. Hydrodynamic interactions of self-propelled swimmers. Soft Matter, 2013, 9(19): 4923-4936.

[88] Monaghan J J, Lattanzio J C. A refined particle method for astrophysical problems. Astronomy and astrophysics, 1985, 149: 135-143.

[89] Monaghan J J. Smoothed particle hydrodynamics. Annual review of astronomy and astrophysics, 1992, 30: 543-574.

[90] Monaghan J J. Simulating free surface flows with SPH. Journal of computational physics, 1994, 110(2): 399-406.

[91] Motsch S, Navoret L, Numerical simulations of a non-conservative hyperbolic system with geometric constraints describing swarming behavior. Multiscale Model. Simul, 2011, 9: 1253-1275.

[92] Motsch S, Tadmor E, A new model for self-organized dynamics and its flocking behavior. J. Stat. Phys, 2011, 144: 923-947.

[93] Muller M, Charypar D, Gross M. Particle-based fluid simulation for interactive applications. Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation. Eurographics Association, 2003: 154-159.

[94] Najafi A, Golestanian R. A simplest swimmer at low Reynolds number: Three linked spheres. arXiv preprint cond-mat/0402070, 2004.

[95] Nestor R M, Basa M, Lastiwka M, et al. Extension of the finite volume particle method to viscous flow. Journal of Computational Physics, 2009, 228(5): 1733-1749.

[96] Owens J D, Houston M, Luebke D, et al. GPU computing. Proceedings of the IEEE, 2008, 96(5): 879-899.

[97] Patankar N A, Singh P, Joseph D D, et al. A new formulation of the distributed Lagrange multiplier/fictitious domain method for particulate flows. International Journal of Multiphase Flow, 2000, 26(9): 1509-1524.

[98] Pedley T J, Hill N A, Kessler J O, The growth of bioconvection patterns in a uniform suspension of gyrotactic micro-organisms. J. Fluid. Mech, 1988, 195: 223-237.

[99] Perea L, Elosegui P, Gmez G. Extension of the Cucker-Smale control law to space flight formations. Journal of guidance, control, and dynamics, 2009, 32(2): 527-537.

[100] Peruani F, Deutsch A, Bär M, Nonequilibrium clustering of self-propelled rods. Phys. Rev. E, 2006, 74: 030904(R).

[101] Peruani F, Starrub J, Jakovljevic V, et al. Collective motion and nonequilibrium cluster formation in colonies of gliding bacteria. Physical review letters, 2012, 108(9): 098102.

[102] Ramaswamy S. "The mechanics and statistics of active matter." The Mechanics and Statistics of Active Matter 1 (2010): 323-345.

[103] Ratushnaya V I, Bedeaux D, Kulinskii V L, Zvelindovsky A V, Collective behavior of self propelling particles with kinematic constraints: the relations between the discrete and the continuous description. Phys. A, 2007, 381: 39-46.

[104] Reif J H, Tate S R. The Complexity of N-body Simulation, Automata, Languages and Programming. Springer Berlin Heidelberg, 1993: 162-176.

[105] Saintillan D, Shelley M J. Emergence of coherent structures and large-scale flows in motile suspensions. Journal of the Royal Society Interface, 2012, 9(68): 571-585.

[106] Saintillan D, Shelley M J, Instabilities, pattern formation and mixing in active suspensions, Phys.Fluids, 2008, 20: 123304.

[107] Saintillan D, Shelley M J. Emergence of coherent structures and large-scale flows in motile suspensions. Journal of the Royal Society Interface, 2012, 9(68): 571-585.

[108] Samuel J, Sinha S. Elasticity of semiflexible polymers. Physical Review E, 2002, 66(5): 050801.

[109] Schliwa M. Molecular Motors. Springer Berlin Heidelberg, 2006.

[110] Schoonderwoerd R, Holland O E, Bruten J L, et al. Ant-based load balancing in telecommunications networks. Adaptive behavior, 1997, 5(2): 169-207.

[111] Shen J. Cucker-Smale flocking under hierarchical leadership. SIAM Journal on Applied Mathematics, 2007, 68(3): 694-719.

[112] Sokolov A, Aranson I S, Kessler J O, et al. Concentration dependence of the collective dynamics of swimming bacteria. Physical Review Letters, 2007, 98(15): 158102.

[113] Spakowitz A J, Wang Z G. End-to-end distance vector distribution with fixed end orientations for the wormlike chain model. Physical Review E, 2005, 72(4): 041802.

[114] Sutmann G, Stegailov V. Optimization of neighbor list techniques in liquid matter simulations. Journal of Molecular Liquids, 2006, 125(2): 197-203.

[115] Swift J, Hohenberg P C. Hydrodynamic fluctuations at the convective instability. Physical Review A, 1977, 15(1): 319.

[116] Szabó B, J Szöllösi G, Gönci B, Zs. Jurányi, Selmeczi D, and Vicsek T Phase transition in the collective migration of tissue cells: Experiment and model. Phys. Rev. Lett, 2006, 74: 061908.

[117] Temperley H N V, Lieb E H. Relations between the'percolation'and'colouring'problem and other graph-theoretical problems associated with regular planar lattices: some exact results for the'percolation'problem. Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences, 1971, 322(1549): 251-280.

[118] Tiwari S, Kuhnert J. Finite Pointset Method Based on the Projection Method for Simulations of the Incompressible Navier-Stokes Equations. Springer Berlin Heidelberg, 2003.

[119] Toner J, Tu Y. Flocks, herds, and schools: A quantitative theory of flocking. Physical review E, 1998, 58(4): 4828.

[120] Toner J, Tu Y and Ramaswamy S, Hydrodynamics and phases of flocks. Annals of Physics, 2005, 318: 170-244

[121] Vabo R, Nottestad L. An individual based model of fish school reactions: predicting antipredator behaviour as observed in nature. Fisheries Oceanography, 1997, 6(3): 155-171.

[122] Vicsek T, Czirók A, Ben-Jacob E, Cohen I, Shochet O, Novel type of phase transition in a system of self-driven particles, Phys. Rev. Lett., 75 (1995) 1226-1229.

[123] Vicsek T, Zafeiris A, Collective motion. Phys. Rep, 2.12, 517: 71-140.

[124] Viscido S V, Miller M, Wethey D S. The response of a selfish herd to an attack from outside the group perimeter. Journal of Theoretical Biology, 2001, 208(3): 315-328.

[125] Wang X J, Warner M. Theory of nematic backbone polymer phases and conformations. Journal of Physics A: Mathematical and General, 1986, 19(11): 2215.

[126] Wang N, Hou R Z, Bao W Z, Wang Z S. Single-polymer 'flyfishing' effect for nanoscale motors and machines: an exact worm-like-chain model study. Manuscript, 2014.

[127] Woodhouse F G, Goldstein R E, Spontaneous Circulation of Confined Active Suspensions. Phys. Rev. Lett, 2012, 109: 168105.

[128] Wu E H. State of the art and future challenge on general purpose computation by graphics processing unit. Journal of Software, 2004, 15(10): 1493-1504.

[129] Yamakawa H. Modern Theory of Polymer Solutions. 1971.

[130] Yamakawa H, Fujii M. Translational friction coefficient of wormlike chains. Macromolecules, 1973, 6(3): 407-415.

[131] Yamao M, Naoki H, Ishii S. Multi-cellular logistics of collective cell migration. PloS one, 2011, 6(12): e27950.

[132] Yao Z, Wang J S, Liu G R, et al. Improved neighbor list algorithm in molecular simulations using cell decomposition and data sorting method. Computer physics communications, 2004, 161(1): 27-35.

[133] Yang J, Wang Y, Chen Y. GPU accelerated molecular dynamics simulation of thermal conductivities. Journal of Computational Physics, 2007, 221(2): 799-804.

[134] Yang Z, Zhu Y, Pu Y. Parallel image processing based on CUDA. Computer Science and Software Engineering. 2008 International Conference on. IEEE, 2008, 3: 198-201.