# Influence of particle smoothness and substrate mechanics on the clustering of self-propelled rods

Submitted in partial fulfillment of the requirements

of the degree of

Doctor of Philosophy

of the

Indian Institute of Technology, Bombay, India

and

Monash University, Australia

by

**Md Imaran**

Supervisors:

Prof. Raghunath Chelakkot (IIT Bombay)

Prof. Mandar M Inamdar (IIT Bombay)

Dr. Prabhakar Ranganathan (Monash University)



*The course of study for this award was developed jointly by the Indian Institute of Technology, Bombay and Monash University, Australia and was given academic recognition by each of them. The programme was administered by The IITB-Monash Research Academy*

Year 2022

**Declaration**

This thesis is an original work of my research and contains no material which has been accepted for the award of any other degree or diploma at any university or equivalent institution and that, to the best of my knowledge and belief, this thesis contains no material previously published or written by another person, except where due reference is made in the text of the thesis.

Signature:

Print Name: Md Imaran

Date: 22 February, 2022

## Abstract

Motile rod-shaped cells are common in biology. It is now well established that dry systems of motile rods show a rich variety of dynamic phases varying from polar clusters, moving bands, complex vortices, swirls to lane formation. The relative importance of mechanical and chemical properties of self-propelled rods in determining their collective behavior has received wide attention. It is reported that the shape-induced effects could play an important role in microbial processes such as collective sperm swimming or algal colony formation.

Furthermore, the colonization of a soft, passive material by these motile cells such as bacteria or cancer cells is a widely observed phenomenon. In this case, the resulting colonies of the invading cells are often observed to exhibit intricate patterns whose morphology and dynamics can depend on the mechanical properties of the substrate in addition to the motility of the individual cells. It has been suggested that this may be a form of stigmergy, wherein cells collectively coordinate over large length and time scales through interactions with their mechanical environment to colonize it optimally.

This thesis presents minimal models of a motile rod, made of a rigid linear array of overlapping beads and a passive, compliant medium consisting of particles that offer elastic resistance before being plastically displaced from their equilibrium positions. Simulations of motile rods in the absence of a compliant medium show that inter-bead spacing in a rod plays a crucial role in determining the collective behavior of a dense system. In the case of rough rods, bubbles of low-density regions emerge. It is observed that, at a given rod's smoothness, the size of bubbles increases with an increase in motility of rods. However, the size of bubbles decreases with an increase in rod smoothness at given motility. As we increase the smoothness of rods, coherently moving clusters are formed at intermediate smoothness, while at very high smoothness, the jammed phase is observed. The correlated motion in the system seems to depend on the collision and clustering phenomena of rough and smooth rods. These studies show that rod smoothness in a dense system might be a relevant parameter for studying non-equilibrium phenomena such as the collective behavior of active systems.

We further conduct simulations of rough rods in the presence of compliant substrate. It is found that the tendency of motile (self-propelled) rods to form clusters of different kinds is crucial for understanding the morphodynamics of colonization

of soft, passive substrate. Interestingly, it was found that the rate at which the colony edge advances depends non-monotonically on substrate stiffness. At any given particle activity, a distinct maximum in the colonization rate is obtained at a particular value of stiffness. We find that this non-monotonicity is due to the dynamics of vanguard clusters that form at the leading edge. The speed of these clusters depends on their shape and size, which in turn depends on substrate stiffness and particle activity. These suggest that, in biological systems, particle motility may be tuned to maximize the colonization rate, given the stiffness of the surrounding substrate. We also find that colonies can expand faster than the case if individual rods move through the substrate. This is yet another example of the fascinating collective behavior that active matter systems exhibit.

Further, as the particles move through the plastic substrate, they create furrow networks, just as in real bacterial colonies. We show that these networks have a fractal-like structure whose dimension varies systematically with substrate stiffness but is less sensitive to particle activity. The power-law growth exponent of the furrowed area is smaller than unity, suggesting that, to sustain such extensive furrow networks, colonies must regulate their overall growth rate.

## Publications

The following is a list of publications based on the research presented in this dissertation:

Journal articles:

1) Imaran, M., Inamdar, M., Prabhakar, R. and Chelakkot, R., "Cluster and conquer: the morphodynamics of invasion of a compliant substrate by active rods", Soft Matter, 17:7459–7465, 2021, 2021, doi: 10.1039/d1sm00860a.

2) Imaran, M., Inamdar, M., Prabhakar, R. and Chelakkot, R., "Effect of roughness on collective behavior in a dense system of self-propelled rods". [In progress]

Conference proceedings/talks/posters:

1) Imaran, M., Inamdar, M., Prabhakar, R. and Chelakkot, R., "Stigmergic furrowing by an active suspension in a soft substrate", 22nd Australasian Fluid Mechanics Conference (AFMC), 2020, doi: 10.14264/cac8c23, Dec., 2020, Brisbane, Australia. (Conference proceeding)

2) Imaran, M., Inamdar, M., Prabhakar, R. and Chelakkot, R., "Substrate stiffness shapes stigmergy in active furrowers", International Congress on Rheology, Dec., 2020. (Conference talk)

3) Imaran, M., Inamdar, M., Prabhakar, R. and Chelakkot, R., "Cluster and conquer: the morphodynamics of invasion of a compliant substrate by active rods" COMPFLU 2020. (Conference poster)

4) Imaran, M., Inamdar, M., Prabhakar, R. and Chelakkot, R., "Stigmergy exploits mechanics !", COMPFLU, Dec., 2019, IISER Bhopal, India. (Conference poster)

## Acknowledgement

This research was financially supported by IIT Bombay – Monash University Research Academy, Mumbai, India.

I would like to express my sincere gratitude to my research supervisors and mentors Dr. Prabhakar Ranganathan, Prof. Raghunath Chelakkot, and Prof. Mandar M Inamdar, whose stellar guidance, encouragement and personal support right from the grass root level enabled me to successfully conduct this thesis work. It is indeed an honor for me to have worked under their supervision and become a part of their research group. I acknowledge my deep sense of gratitude to the members of my doctoral research committee Prof. Amitabha Nandi, and Prof. Murray Rudman for their valuable and timely suggestions on my research topic.

I am indebted to all my lab mates at IIT Bombay, Supriya, Suchismita, Akhil, Monika, Padmalochini, and Dr. Bappaditya Roy for providing a warm research atmosphere, sharing of knowledge and personal support. I am also thankful to memebers of the Complex Fluids group at Monash University of which I was a part, for being excellent and supportive co-workers. Finally, I must thank all the wonderful friends I have made, who have made this journey truly enjoyable and helped me get fresh perspective in challenging times.

I owe my deepest gratitude to my parents and siblings for their love, affection, care and constant encouragement throughout the research work. Last but not the least, I wish to express a heartfelt gratitude to my wife, Tabia for her consistent and selfless sacrifice, encouragement, and support throughout these years.

Finally, I am grateful to God for blessing me with everything that I didn't even ask. All thanks and praises are due to Him alone. Only the mistakes have been mine.

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

## 1.1 Collective motion

Collective motion is the phenomenon wherein independent agents or particles come together and move coherently over length and time scales significantly larger than the scales governing the individual components. In nature, such behavior is frequently observed in systems where the individuals are "active". Active particles capture energy either internally or from the surroundings and use it to propel themselves along a preferred direction [107]. Collective motion is observed in a broad range of biological and synthetic systems, including humans, bird flocks, herds of quadrupeds, insect swarms, groups of bacteria, and self-propelled Janus particles [6, 96, 107, 118, 40, 84, 67, 20, 123, 78, 31, 39, 116]. Various forms of collective motion such as flocking, swarming and laning are observed in these active systems [83, 113]. In biological systems, these large-scale collective motion patterns have important functional consequences. Large-scale collective pattern formation by self-motile elements is a widely studied phenomenon in physics and biology [34, 6, 67, 78, 107]. Microbial systems have, in particular, received wide attention in this context. Microbes show collective behavior in many processes, such as attacking predators [25], hunting of micromolecules released during lysis of other microbial cells by *Myxococcus xanthus* [10], defending themselves against


FIGURE 1.1: Example of collective behavior in microbes [121]. (a) *M. xanthus* cells engaging in coordinated motility. (b) Sketch of swarming and individual cells of *C. aurantiacus*. (c) An *M. xanthus* swarm (movement left to right) consuming a colony of *Escherichia coli*.

predators such as protozoan species [61], surviving harsh environments [30] by superbugs such as *Staphylococcus aureus*, *Pseudomonas aeruginosa*, and formation of multicellular structures like biofilms [89]. It has been observed that, to kill biofilm of *Staphylococcus aureus* cells, requirement for concentration of antibacterial agents such as chlorine increases more than 600-folds compare to planktonic cells of the same species [75].

Collective behavior of microbes is also important for their survival and evolution [54]. Survivability and evolution depend on the ability of microbes to migrate towards nutrient-rich regions. It has been observed that collective motion of microbes has a direct impact on human life by being resistant to antibiotics. It is also responsible for acute and chronic infection and degradation of medical implants [128, 114, 76, 55, 26, 42, 71, 94]. Bio-film formation is also associated with the industrial problem of bio-fouling in water distribution systems [125].

Thus, in the context of collective motion, the important questions addressed by most of the studies present in literature include - how does collective motion arise in systems consisting of mobile particles? What are the different kinds of patterns that can be generated and what are the conditions that lead to these different patterns? However, another important feature of agents or particles in many active systems is the inherent stochasticity in particle behavior. Thermal fluctuations present in the environment of microbial systems result in significant Brownian forces on the particles. Internal biological processes within cells may

further lead to non-Brownian stochastic effects in the particle's response to its environment [2]. When these random influences are dominant, one expects an isotropic structure with no long-term spatio-temporal correlations in motion *i.e.* no collective motion. It is of interest, therefore, to understand the conditions under which inherent stochasticity is overcome and collective motion is observed.

## 1.2 Computer simulations of collective motion in active systems

The above questions have been widely studied in recent years [8, 107, 78, 51, 9, 135]. Many artificial systems have been synthesized as experimental analogs or with a view of exploiting this pattern forming behavior for human applications [126, 1, 24]. Experiments mimicking natural systems are difficult to control while it is equally difficult to design artificial systems. Computer simulations of large numbers of mobile particles have therefore become a powerful tool in exploring collective motion and have yielded many important insights. The seminal work of Vicsek *et al.* [122] considered a system of point particles, each of which followed a simple rule that oriented its instantaneous velocity along the average velocity of its neighbors. In such rule-based models, collective motion is observed to set in above a critical local density and below a critical fluctuation level. In such models, effect of mechanical interactions is neglected, and hence these models were unable to explain the emergent effects in system of active particles that occupy space. In order to understand the connections behind collective motion and particle characteristics/interactions, simulations with mechanical interactions between the individual particles have been extensively used.

It is important to consider the effect of mechanical interactions as it has been shown that even that alone can cause collective behaviour [22]. For instance, motility induced phase separation observed in systems of repulsive self-propelled 2-d disks, particles slow down due to crowding and dilute and dense phases emerge [108]. In these systems of mass-less particles, phase separation depends on a number

of factors such as mechanical interaction, density of the system, and propulsion strength of particles.

When particles are anisotropic in nature, they are known to show phase separation. Even a system of non-motile anisotropic particles, at high density shows isotropic-nematic transition [22]. This anisotrpoic nature further enriches the behavior of active systems. In many simulations, anisotropic particles are modeled as self-propelled rods (SPRs). When the inertia of the rods is negligible, they are comparable to systems of motile cells and can thus yield valuable insights into pattern formation in these biological systems.

SPRs show a number of non-equilibrium phases due to their anisotropy and interactions, for example, polar clusters, moving bands, complex vortices, swirls, and lane formation [100, 2, 105, 91, 120, 13, 7, 69]. In these systems, contact-mediated anisotropic repulsion and motility are the dominant forces and drive the collective motion.

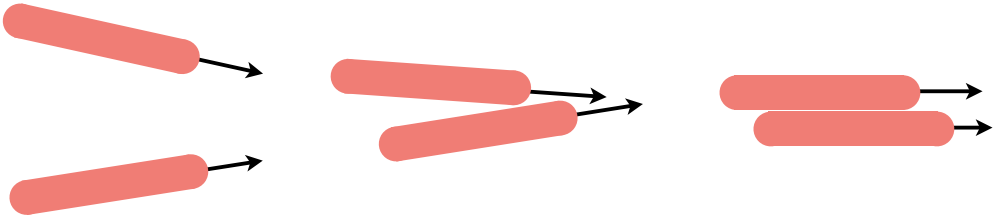

FIGURE 1.2: Collision of two self-propelled rods. Rods align due to anistropic repulsion.

From, Fig. 1.2, it can be inferred that two SPRs (moving in a random direction) come in contact, they align themselves in the parallel direction by the induced torque on the rods due to their anisotropic shapes.

Several studies show that the formation of different non-equilibrium phases in a system of SPRs, strongly depends on the density and aspect ratio [100, 82, 2, 8, 129, 105, 120, 69, 7, 11, 120, 124, 13] similar to their passive counterpart. The first attempt to computationally model the SPRs [100] shows that the non-equilibrium clustering occurs above a critical rod density at a given aspect ratio. This critical

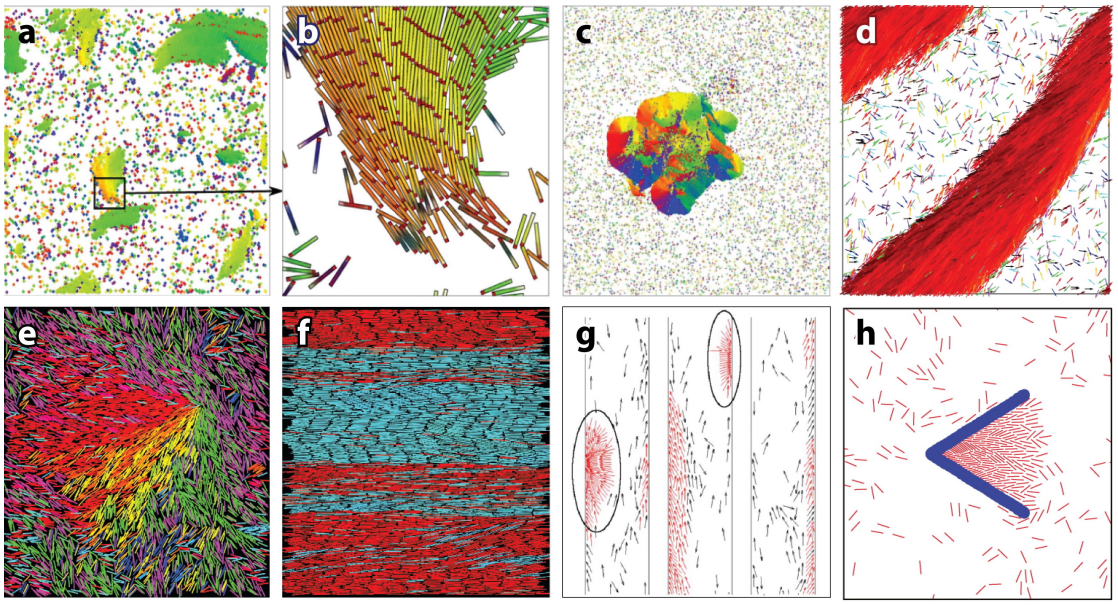


FIGURE 1.3: Non-equilibrium dynamic phases in systems of dry self-propelled rods. In these system short range interaction are dominant. (a) Polar clusters with local smectic order as blown-up figure in (b); (c) giant aggregates; (d) polar bands; (e) nematic chaos, showing a nematic defect; (f) laning ; (g) accumulation of self-propelled rods at walls; (h) trapping for self-propelled rods [7].

density is lower than the density observed for isotropic-nematic transition in two dimensional systems of passive rod.

In the dilute system of SPRs, polar clusters are formed due to the inherent property of moving of SPRs in parallel or anti-parallel direction after collision [Fig. 1.3(a)]. Mostly these clusters are motile and have smectic (layered) arrangements of constituent rods. When rods of the polar clusters rearrange themselves, polar bands form Fig. 1.3(d). The stability of these polar bands depend on the system size. Thus, at above a critical system size, these polar bands break up into dilute and dense phases by forming a disordered high-density large aggregate [Fig. 1.3(c)]. The aggregate shows a fission and fusion mechanism by losing and absorbing smaller polar clusters at the boundary. These collective behavior mechanisms are different from the density induced slowdown, motility induced phase separation (MIPS) observed in self-propelled 2-D discs. Many complex mechanism such as quorum sensing also has been studied in the context of self-propelled rods [120].

At high density, SPRs are known to show a plethora of non-equilibrium dynamic

phases. These dynamic phases include formation of polar clusters [129], polar bands [2], percolating turbulent, laning, and jammed states [133]. The stability of large polar clusters and lanes are dependent upon the system size. As system size increases, cluster breaks up, while lanes of rods moving in opposite directions (Fig. 1.3(f)), destabilize and result in nematic chaos (Fig. 1.3(e)).

When interaction between the SPRs becomes softer, in the limit where self-propulsion is stronger than repulsion, SPRs can easily cross each other and mimic the 2-D realization of elongated rod experiments [2]. At an intermediate noise level, these systems form high-density polar bands with nematic order. The size of these bands decreases with a decrease in noise level.

SPRs show a specific interaction and accumulating behavior in confined boundaries [132]. In contrast to system of passive particles placed in confinement, in which particles accumulate at the boundary, SPRs continuously push through the confining wall [Fig. 1.3(g)]. Initially, SPRs accumulate at the wall and at later time scale, due to the anisotropic shape of aggregate, they slip off collectively [70]. The tendency of rods to form clusters and accumulate at walls enables the construction of traps as shown in Fig. 1.3(h) [65, 63, 62, 64].

Another important factor that influences the motion of SPRs is shape. In nature, active particular occur in different shapes. The behavior of L-shaped rods and crescent, convex shapes filaments are explored in literature. Crescent and convex shaped rigid filaments form aligned fronts and rotating clusters [131]. The dynamics of L-shaped rods are significantly different from straight SPRs. L-shaped rods form short-lived, small clusters. The collective motion of these system can be explained with the distinct collision and oscillation mechanism [74].

The placement of passive particles in a system of SPRs often results in the segregation of SPRs and passive particles. SPRs and passive particles collide with different frequencies, leading to segregation which is supported by the inherent property of active rods of forming a cluster after collision [82]. When flexible chains are put into a system with passive spherical obstacles, stiff chains quickly pass through obstacles while flexible chains form spiral around it and get stuck. The flexible

chains show random motion by unspiraling and hopping through obstacles [85]. The effect of flexibility has been also studied in context of self-propelled chain-bead filament model. In the limit of bending rigidity of these filaments when rigidity becomes infinite, these chain-bead filaments behave as SPRs. These flexible filaments, made of spherical beads, tend to show the formation of flowing melts, coherently moving aggregates, swirls, and spirals at sufficiently large aspect ratios [37, 105, 86]. The phase boundaries for these non-equilibrium states are often specified by the flexibility and propulsion of the chain [105, 60].

## 1.3 Kinetic theory for clustering of self-propelled rods

In literature many theories and description for the kinetics and clustering of SPRs have been discussed [68, 100, 101, 102, 99]. A kinetic theory exploiting the idea of fragmentation and coagulation of particles was proposed [100]. In the presented description, clusters lose rod by fission at the boundary and collide with other clusters and particles to form bigger aggregates. The fusion and fission mechanism for clusters can be written as follows ,

$$\begin{aligned} \mathrm{C}_k + \mathrm{C}_j &\xrightarrow{\mathrm{c}_{j,k}} \mathrm{C}_k, \\ \mathrm{C}_k &\xrightarrow{\mathrm{f}_k} \mathrm{C}_{k-1} + \mathrm{C}_1 \end{aligned}$$

where $C_k$ is the cluster containing $k$ rods, $f_k$ and $c_{j,k}$ are fragmentation and coagulation rates respectively.

Assuming the total number of rods, $N$ is constant in the system and only binary cluster collision occurs, the rate of change of mean number, $\dot{n}_j(t)$ for cluster of size $C_k$ at time, $t$ can be written as,

$$\dot{n}_j(t) = \begin{cases} 2f_2 n_2 + \sum_{k=3}^{N} f_k n_k - \sum_{k=1}^{N-1} c_{k,1} n_k n_1 \text{ if } j = 1 \\ -f_n n_N + \frac{1}{2} \sum_{k=1}^{N-1} c_{k,N-k} n_k n_{N-k} \text{ if } j = N \\ f_{j+1} n_{j+1} - f_j n_j - \sum_{k=1}^{N-j} c_{k,j} n_k n_j + \frac{1}{2} \sum_{k=1}^{j-1} c_{k,j-k} n_k n_{j-k} \text{ else} \end{cases} \tag{1.1}$$

The cluster size distribution is given as,

$$P(k) = n_k k / N \tag{1.2}$$

This fragmentation-coagulation theory quantitatively reproduces the observations for the soil bacteria *Myxococcus xanthus* [103].

## 1.4 Effect of the environment

The surrounding medium often plays a significant part in mediating the interaction between the individual agents. In some cases surrounding medium modifies the behaviour of motile agents. For example, in wet systems, at low Reynolds numbers (whereby hydrodynamic interactions are nearly instantaneous), surrounding medium transfers momentum between the agents without getting modified. Inhomogeneous and complex environments qualitatively change the individual, as well as collective dynamics of such systems [9, 39, 106, 87, 28, 97, 116, 29]. In contrast to this, in dry systems, agents may leave behind chemical or mechanical cues in surrounding medium, thereby increasing the level of medium-induced complexity and thus dynamics of surrounding medium as well as agents get modified. This phenomenon is known as stigmergy. The current study focuses on the dry systems of self-propelled rods.

Stigmergy was first introduced in 1959 by a French scientist Grassé, in context

of social behaviours of insect like ants and termites [53]. It is described as self-organization shown by agents, by exchanging information through the environment. Stigmergy is a mechanism of indirect coordination, through the environment, between agents or actions. The principle is that the trace left in the environment by an action stimulates the performance of a next action, by the same or a different agent. In that way, subsequent actions tend to reinforce and build on each other, leading to the spontaneous emergence of collective behaviours and complex structures. The phenomena of stigmergy is observed abundantly in nature; some of which include the building of nests by termites, migration of animals in herds [56], formation of pheromone trails by ant colony [53, 4, 36, 12, 35, 58], and so on.

As it has been observed that when environment is involved in more complex interaction with agents, systems tend to show complex emergent behaviours. Individual cells of bacteria such as *Pseudomonas aeruginosa* are known to drag themselves forward by throwing out grappling hooks called "Type IV pili" which then bind to the substrate at their tips and are reeled in by motors inside the cell [15, 98]. Effectively, the substrate acts as a source of momentum to each cell and the net propulsive force is on average found to be directed along the principal axis of the rod-like cell body [16]. In this respect, therefore, these bacteria crawling across a surface are no different from cattle on the Serengeti plain, and indeed, several studies have already modeled the motion of large surface colonies of motile bacteria as dry systems [140].

The mechanical and chemical form of stigmergy have been observed in motile bacteria *Pseudomonas aeruginosa* and *Myxococcus xanthus* when cultured on soft hydrogel substrates [47, 49, 48]. Under favorable conditions, these bacteria form extensive networks of permanent furrows as they move collectively as a monolayer across the soft surface in the initial stages of biofilm formation. The rate of expansion of the colony is further observed to be intimately related to the morphology of the network and the cellular traffic within the furrows [47, 49, 48].

## 1.5 Furrow-networks in bacterial colonies

In a recent study [47], it was observed that cells of the bacterium *Pseudomonas aeruginosa* work collectively to build a trench network to colonize an agar surface. These bacteria are motile and furrow through the soft semi-solid surface to form an interconnecting network trail. The network was also observed to have a distinct complex morphology.

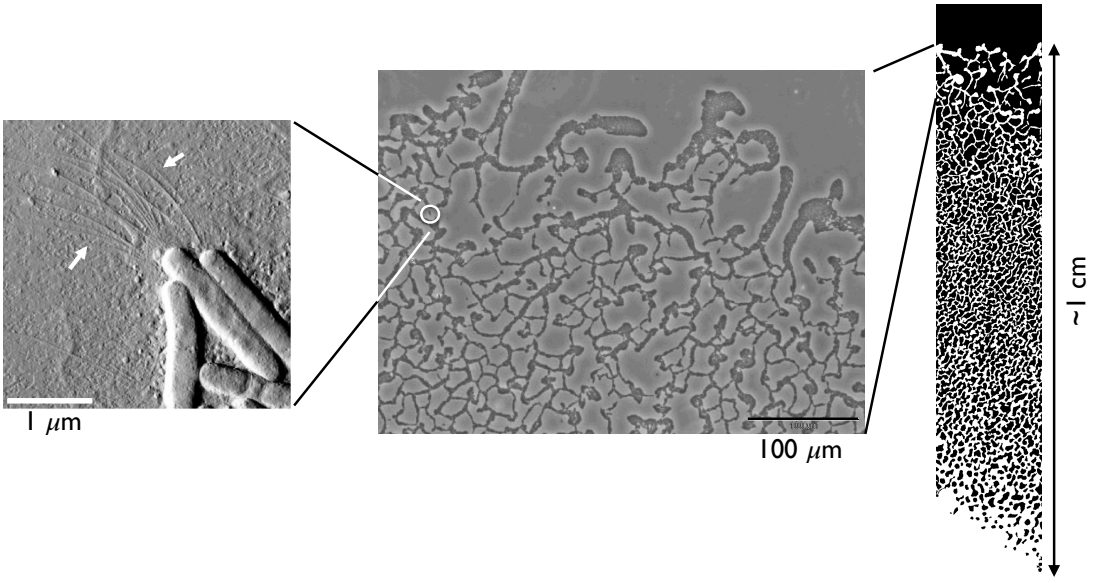


FIGURE 1.4: Left: *Pseudomonas aeruginosa* cells deploying Type IV pili grappling hooks, Middle: Region close to the advancing edge of an actively expanding monolayer of a *P. aeruginosa* colony on agar. Right: Overall lattice-like morphology of colony [47].

The constant motion of tightly packed rod-shaped cells within these trenches was observed to be regulated by the secretion of a chemical agent which appeared to be essential to cells exhibiting lane discipline, following one another. Destroying that agent resulted in traffic chaos within the trenches, ultimately leading to slower colony expansion. This kind of self-organized behaviour has also been observed in other bacteria, with studies showing a strong dependence of the morphology and dynamics on the motility of the particles and the mechanical properties of the substrate [103].

Although such spatio-temporal patterns are well known across the microbial world [137, 142], the mechanisms behind their formation and regulation are still poorly understood. Understanding these mechanisms can have far-reaching implications. For instance, *Pseudomonas aeruginosa* is a well known antibiotic-resistant "superbug" that is adept at first colonizing soft exposed tissue surfaces and then building complex biofilms that are practically impossible to get rid off even with the most potent antibiotics available [14]. The formation of structures such as the trench network by a monolayer of cells is the first stage in the establishment of biofilms. Understanding this process may enable the design of therapeutic strategies to reduce or eliminate the spread of superbug infections. It may further be possible to artificially create such micron-scale network structures on soft media using synthetic active particles such as catalytically-driven Janus micromotors. The structure could then be used as a mould template to synthesize electrically-conducting or heat-conducting micronetworks.

## 1.6 Open questions

### 1.6.1 Effect of rod smoothness on collective behavior

In literature, most of the studies involving self-propelled rod particles are carried out by considering the rod as a chain of spherical beads. In the dense limit, when contact mediated interaction dominates, a subtle change in mechanical properties of th rod can change the collective behavior of the system. However, change in shape may induce shear friction or blocking of rods by grooves in chain-bead rods, which would not be present if rods are made of overlapping beads, thus minimising the effect of the inter-locking . In this case, rods could slide along each other during a collision[7, 23].

The effect of inter-particle interaction and mechanical properties of SPRs i.e., aspect ratio, interaction, and propulsion strength on emergence of non-equilibrium phases is well reported [7]. However, the effect of roughness in the context of

dense suspension is not well explored. These studies are crucial for understanding whether surface topology is relevant for non-equilibrium phenomena, i.e., collective behaviour, vortex formation in the dense system of SPRs.

### 1.6.2 Effect of mechanics of surrounding substrate on collective behavior

In an attempt to understand the behaviour observed in experiments in colonies of motile, rod-shaped cells of *P. aeruginosa* growing on agar [47], Zachreson *et al.* [140, 141] used simulations of self-propelled spherocylinders interacting with a continuum model of a deformable substrate. While their results demonstrate that substrate stiffness and its viscoelastic relaxation time can strongly influence the morphodynamics of active furrowing, their parameters were chosen specifically for the experimental system at hand. Moreover, the simulations also included cell division and population growth. The parameters in the simulations of Zachreson *et al.* were explicitly chosen for *P. aeruginosa*. Moreover, the simulations simultaneously included cell growth and division along with motility and interactions between particles and of particles with a continuum substrate. In a dense colony of growing but immotile cells, excluded-volume interactions will lead to the build-up of pressure within the colony that pushes the edge of the colony outward. Fingering instabilities could then initiate the development of furrows [79, 90]. However, it is not evident whether such growth induced pressure is a necessity for furrow formation. This thesis focuses on isolating the role specifically played by cell motility in the formation of extensive furrow networks. Therefore, simulations of motile but non-growing rods of fixed size and aspect ratio are carried out.

## 1.7 Emerging insight: Motility-induced clustering

In the above context, this thesis uses 2D simulations to study behavior SPRs in absence and presence of elasto-palstic substrate. A dense system of pure rods shows formation of coherently moving clusters and bubbles. It is observed that the formation clusters and bubbles depend on the roughness (inter-bead spacing the rod) of the rod. It is observed that the clustering is a key mechanism in the formation of different phases in the system.

In the presence of substrate, a system of rods shows formation of a furrow structure. The morphology of such a furrow structure depends on the clustering of rods. It was found that *Motility-induced clustering* is the generic cause behind the emergence of furrow networks. Also, the fractal dimension of these structures depends on substrate plasticity. Clustering further enhances the rate at which the colony edge advances, and this speed gain again, depends on cluster morphology. Our results also suggest that colonies must regulate their overall cell growth rate to sustain extended furrow networks.

## 1.8 Quantifying collective motion and clustering

In the systems of SPRs in the absence and presence of substrate, collective motion is observed. The collective motion observed in the dense system of SPRs is quantified by calculating the density distribution and spacial velocity auto-correlation functions. The local number density distribution gives an insight into the distribution of rods in the system. For a homogeneous system a single peak is observed. However, when system shows phase separation (formation of bubbles and dense region), a plateau behavior is reported. We further use clustering algorithm to calculate the bubble statistics in the system. The extend of collective motion for the motile cluster in the system, is determined by the spatial correlations of quantities such as velocity and position[19, 18].

SPRs in the presence of a compliant substrate, form clusters of different geometry. We further report that the colonization of the compliant substrate and furrow formation depends on the clustering. Statistics of clusters is estimated by using *train-raft* and cluster tracking algorithms developed in this study.

## 1.9 Thesis structure

Chapter 2 focuses on the development of model for simulation of pure SPRs and SPRs invading through elasto-plastic substrate. It also includes the non-dimentionalization of model equations and calculation of systems parameters for the study. It also describes the implementation details of additional in-house packages developed for the study in LAMMPS.

Chapter 3 focuses on quantification techniques developed for the study. Techniques are mostly developed to determine collective behavior and cluster statistics in the system. For quantification of collective behavior, algorithms for local density calculation, spatial correlation are used. Different methodologies such as cluster tracking and train raft algorithm were developed during the study to get the statistics of clusters in the system.

In Chapter 4, effect of rod surface roughness on the collective behavior of dense system is discussed. The surface roughness of the rods is tuned by changing the inter-bead distance. The inter-bead distance affects the inter-particle interaction and surface roughness. The simulations show that surface roughness play a crucial role in determining the collective behavior of the system. The formation of bubbles is observed for rough rods while for the smooth rods correlated behavior is observed. It is further shown that the correlated behavior depends non-monotonically on the smoothness of the rod at given motility.

Chapter 5 enumerates the core results of the study which offer key insights into the colonization of an elastic substrate by motile rods. The simulations predict

complex network structures near the advancing edge that visually resemble experiments. These structures are created by clusters of rods whose collective trajectories have distinctive curvatures. This causes new furrows to emerge and re-merge, leading to a network. Finally, Chapter 6 summarizes the conclusion and future outlook of the thesis.

# Chapter 2

# Model and simulations

As introduced in Chapter 1, the primary objective of this work is to study the collective behavior of SPRs. For that purpose appropriate numerical models have been developed. In this chapter, the model for a system of SPRs in two dimensions is presented. A new minimal model for an elasto-plastic substrate is also described. Key dimensionless parameters are identified for the systems of SPRs in absence and presence of an elasto-plastic substrate. The model equations are implemented on the open-source Large-scale Atomic/Molecular Massively Parallel Simulator (LAMMPS) platform [104].

## 2.1 Central concepts

In this study there are mainly two concepts that we introduce. The first one is the smoothness of SPRs, which depends on the number of overlapping beads in a rod of given length, and second is the elasto-plastic nature of substrate particles. These concepts are described in details in subsections (2.1.1) and (2.1.2).

### 2.1.1 Rod smoothness

Each SPR is modeled as a *rigid* linear array of $N_\mathrm{b}$ "beads" and of mass, $m_\mathrm{r}$. Each bead has a nominal diameter, $\sigma_\mathrm{b}$. Adjacent beads belonging to the same rod are allowed to overlap to achieve partially smooth rods.


FIGURE 2.1: Schematic of rods at different values of $\chi$.

The smoothness of a rod can be defined in terms of a smoothness factor,

$$\chi \;=\; 1 - \frac{L-1}{(N_\mathrm{b}-1)\,\sigma_\mathrm{b}} \;=\; 1 - \frac{N_{\mathrm{b,\,min}}-1}{N_\mathrm{b}-1}\,, \tag{2.1}$$

where $L$ is the total end-to-end distance of the rod, including its end-caps. If $\chi$ is zero, the rod is "rough" *i.e.* one in which beads just touch. In such a rod, the number of beads is $N_{\mathrm{b,\,min}} = L/\sigma_\mathrm{b}$, the minimum number of beads needed to make a rod of length $L$. As $N_\mathrm{b}$ is increased keeping $L$ fixed, and the spacing between adjacent bead centers is reduced and $\chi$ increases to unity.

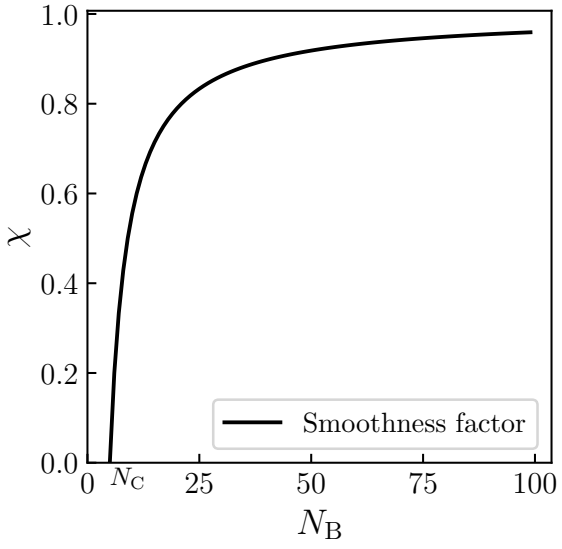


FIGURE 2.2: Smoothness factor as function of number of beads in a rod of length $5\sigma_b$.

### 2.1.2 Substrate plasticity

In experiments, bacteria and other cells are often cultivated on agar, which is a soft hydrogel. As discussed earlier in Chapter 1, as cells furrow over such surfaces, the surface deforms locally [47]. The deformation can be elastic *i.e.* the surface at a point can fully recover to its undeformed state once a rod-shaped cell has passed that location and a force is no longer exerted. This recovery can occur over a characteristic time scale. Alternatively, the deformation can be permanent such that once furrows are formed, they remain even when all rods are removed from the system. We consider this type of deformation as plastic deformation.

Our goal is to implement a minimal particle-based model of substrate elasticity and plasticity in our numerical simulations to understand the role the elastic or plastic character of the substrate plays in the morphology of a colony.

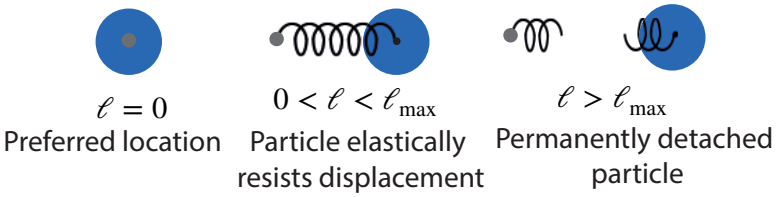


FIGURE 2.3: Three states of a substrate particle in the minimal model of a plastic substrate: (i) Relaxed (ii) Elastic (iii) Plastic.

Although the furrows formed in experiments [47, 49] have a mechanical depth and are 3-D in nature. Here, in our minimal model, we map the furrow formation mechanism into a 2-D system. A semi-solid substrate could, in principle, be modeled as 2D array of particles anchored using linear harmonic potential. To realize faster simulations, a plastic substrate is minimally modeled here as follows (Fig. 2.3). Fig. 2.3 shows the three states of the substrate particle. In the figure, $\ell$ is the displacement of the substrate particle and $l_{\max}$ is the cut-off of the linear harmonic potential. We assume that the position of the substrate particle in its initial, undisturbed, state is its equilibrium position. On being disturbed to a different position the particle experiences a linear elastic restoring force, as long as the particle is not disturbed beyond the length, $l_{\max}$. As soon as a substrate particle moves beyond this distance, the elastic stiffness is set to zero *i.e.* that

particle breaks off permanently, and no longer experiences a restoring force. This mimics the creation of a furrow by cells by depressing the soft agar permanently. This behaviour of the elasto-plastic substrate is quantified by a non-dimensional number $P$, plasticity number. Plasticity number is defined as the ratio of force required to permanently displace a substrate particle and the propulsive force on a rod. The free substrate particle is still retained in the simulation and offer frictional resistance while being passively advected by the active rods. If it were to be removed from the simulation, the frictional resistance from the substrate would continually decrease as furrows form and will eventually vanish completely.

## 2.2 Dynamical equations

The two-dimensional system consists of $N_\mathrm{r}$ SPRs. The total mass of a rod is $m_\mathrm{r} = N_b\, m_\mathrm{b}$. Since each rod is completely rigid, the centre-of-mass of the $i$-th rod, $\mathbf{r}_i$ and the instantaneous unit vector defining its orientation, $\hat{\mathbf{p}}_i$ evolve according to the following equations of motion:

$$\frac{d\,\mathbf{r}_i}{d\,t} = \mathbf{v}_i\,, \tag{2.2}$$

$$\frac{d\,\hat{\mathbf{p}}_i}{d\,t} = \omega_i\,, \tag{2.3}$$

$$m_\mathrm{r}\,\frac{d\,\mathbf{v}_i}{d\,t} = \mathbf{F}_i^\mathrm{tot}\,, \tag{2.4}$$

$$I_\mathrm{r}\,\frac{d\,\omega_i}{d\,t} = \mathbf{T}_i^\mathrm{tot}\,, \tag{2.5}$$

where $I_\mathrm{r}$ is the moment of inertia of a rod about its center, and $\mathbf{v}_i$ and $\omega_i$ are its center-of-mass velocity and the angular rotation rate about its center, respectively. The total force on the each rod, $\mathbf{F}_i^\mathrm{tot}$ is the sum of forces acting on the beads and can be given as, $\mathbf{F}_i^\mathrm{tot} = \sum_\nu \mathbf{F}_{\nu,i}$.

There are five different forces that act on the $\nu$-th bead in the $i$-th rod: the active propulsive force, excluded-volume repulsion from beads of neighboring rods, excluded-volume repulsion from substrate particles, frictional resistance from the

substrate, and random noise. The total force on the $\nu$-th bead of the $i$-th rod is,

$$\mathbf{F}_{\nu,i} = \mathbf{F}^{\mathrm{a}}_{\nu,i} + \mathbf{F}^{\mathrm{f}}_{\nu,i} + \sum_{j\neq i}\sum_{\mu\neq\nu}\mathbf{F}^{\mathrm{bb}}_{\nu,i;\mu,j} + \sum_{p}\mathbf{F}^{\mathrm{bs}}_{\nu,i;p} + \mathbf{F}^{\mathrm{r}}_{\nu,i}\,. \tag{2.6}$$

Where, $\mathbf{F}^{\mathrm{a}}_{\nu,i} = (F^{\mathrm{a}}/N_{\mathrm{b}})\,\hat{\mathbf{p}}_i$ is the propulsion force on any bead of rod, acting along the instantaneous rod axis, $\hat{\mathbf{p}}_i$. The magnitude of the propulsion force on the rod $F^{\mathrm{a}}$, is distributed equally on all beads of each rod. $\mathbf{F}^{\mathrm{bb}}_{\nu,i;\mu,j}$ and $\mathbf{F}^{\mathrm{bs}}_{\nu,i;p}$ are the excluded volume force on the $\nu$-th bead in the $i$-th road due to the bead-bead interaction with the $\mu$-th bead in the $j$-th rod, and the bead-substrate interaction with the $p$-th substrate particle, respectively. Both of these interactions are derived from the pairwise Separation shifted Lennard-Jones (SSLJ) potential [2], which is described further below (2.2.1).

In the context of rod-shaped cells moving through furrows in a plastic substrate, such as agar, the furrows contain an aqueous fluid. The bare frictional force on a bead when the rod moves through such "empty" furrows is $\mathbf{F}^{\mathrm{f}}_{\nu,i} = -\gamma_{\mathrm{b}}\mathbf{v}_{\nu,i}$, where $\gamma_{\mathrm{b}} = \gamma_{\mathrm{r}}/N_{\mathrm{b}}$ is a constant friction coefficient. Frictional interactions between beads of rods is neglected.

The components of the random force on each bead, $\mathbf{F}^{\mathrm{r}}_{\nu,i}$, are independent Gaussian random numbers such that:

$$\left\langle \mathbf{F}^{\mathrm{r}}_{\nu,i}(t)\right\rangle = \mathbf{0}\,; \tag{2.7}$$

$$\left\langle \mathbf{F}^{\mathrm{r}}_{\nu,i}(t)\,\mathbf{F}_{\mu,j}(t')\right\rangle = 2\,\gamma_b\,k_B\,T\,\Delta t\,\delta(t-t')\,\delta_{ij}\,\delta_{\mu\nu}\,, \tag{2.8}$$

where $\delta$ is the Dirac delta-function, $\delta_{pq}$ is the Kronecker-delta functions, $k_{\mathrm{B}}$ is the Boltzmann constant and $T$ is the absolute temperature.

Although the noise in biological systems do not obey the fluctuation-dissipation theorem. However, in our case, $T$ represents the thermodynamic temperature or an effective temperature characterizing the strength of the athermal internal biological noise that obeys the fluctuation-dissipation theorem.

These pair-wise forces are calculated in LAMMPS using standard neighbour-list algorithms. Moments of total force about the rod center yield the torque, $\mathbf{T}^{\nu,i}$. The total torque on each rod is $\mathbf{T}_i^{\text{tot}} = \sum_\nu \mathbf{T}_{\nu,i}$.

When a substrate is included in the simulation, the substrate domain is discretised into $N_\text{s}$ closely-packed substrate beads, which we shall refer to as particles to distinguish them from the beads in the rods. Substrate particles exist on the same two-dimensional plane as the SPRs. The equations of motion for the $p$-th substrate particle are:

$$\frac{d\,\mathbf{r}_p}{dt} = \mathbf{v}_p\,, \tag{2.9}$$

$$m_\text{s}\frac{d\,\mathbf{v}_p}{dt} = \mathbf{F}_p^{\text{tot}}\,, \tag{2.10}$$

where $m_\text{s}$ is the mass of each substrate particle. The net force $\mathbf{F}_p^{\text{tot}}$ on the particle arise due to excluded-volume, elastic and frictional interactions. The total force on the $p$-th substrate particle is given as,

$$\mathbf{F}_p^{\text{tot}} = \mathbf{F}_p^{\text{f}} + \mathbf{F}_p^{\text{e}} - \sum_i \sum_\nu \mathbf{F}_{\nu,i;p}^{\text{bs}}\,, \tag{2.11}$$

where, $\mathbf{F}_p^{\text{e}}$ is the linear elastic restoring force on the substrate particle having equilibrium position $\mathbf{r}_{p,0}$, and is given as,

$$\mathbf{F}_p^{\text{e}} = \begin{cases} -\kappa\,(\mathbf{r}_p - \mathbf{r}_{p,0})\ , \text{ if } \mid (\mathbf{r}_p - \mathbf{r}_{p,0}) \mid < l_{\max}\,, \\ \mathbf{0}\,, \text{ if } \mid (\mathbf{r}_p - \mathbf{r}_{p,0}) \mid > \geq l_{\max}\,. \end{cases} \tag{2.12}$$

The interaction between substrate particles and rods $\mathbf{F}_{\nu,i;p}^{\text{bs}}$ is implemented by the SSLJ potential. The direct substrate-substrate excluded-volume interactions is not included. Instead, it is assumed that excluded-volume interactions of substrate particles with each other eventually result in energy being dissipated away over the time-scale over which the semi-solid substrate responds. This effective frictional force on the $p$-th particle is $\mathbf{F}_p^{\text{f}} = -\gamma_\text{s}\,\mathbf{v}_p$ where $\gamma_\text{s}$ is a large substrate friction coefficient. It is assumed that the substrate particles offer high friction, the

diffusivity of substrate particles is low. Random thermal forces on the substrate particles are hence neglected.

### 2.2.1 Excluded-volume interaction

Excluded-volume (EV) forces between the beads of rods, and between beads and substrate particles are derived using the Separation-Shifted Lennard-Jones (SSLJ) potential [2, 120]. For any two interacting beads, if $r$ is the distance between their centres, then the SSLJ potential is,

$$\phi(r) = \begin{cases} 4\,\epsilon\left[\left(\dfrac{\sigma^2}{r^2+\alpha^2}\right)^6 - \left(\dfrac{\sigma^2}{r^2+\alpha^2}\right)^3\right] - \phi_0, \;, & r \le r_c\,; \\ = 0, \quad r > r_c\,. \end{cases} \tag{2.13}$$

where $\sigma$ is the sum of the nominal radii of a pair of interacting particles and $r_c$ is the cut-off distance. We set $r_c = \sigma$. The regularization parameter $\alpha = \sqrt{2^{1/3}\sigma^2 - r_c^2}$, ensures that the force is finite when $r = 0$ and, given $\sigma$ and $r_c$, it is chosen such that $\phi$ has a minimum at $r = r_c$. The constant shift $\phi_0$ ensures the potential is continuous at $r = r_c$. The softness of the potential is controlled by setting the value of the potential energy, $E = \phi(0) - \phi(r_c)$ when two interacting particles overlap. The energy scale $\epsilon$ is then calculated as $\epsilon = \alpha^{12}\,E/(\alpha^{12} - 4\,\sigma^6\,\alpha^6 + 4\sigma^{12})$. The derivation of $\epsilon$ is discussed in Appendix A.We choose $E$ such that the propulsion force on a single rod is insufficient to penetrate into a bead or substrate particle. The excluded-volume force on the interacting particles is calculated from the spatial gradient of the potential.

To simulate the SPRs in absence of substrate particles, interaction between the beads of a rod is set be zero, while beads of the separate rods interact via SSLJ potential. The term, $\sum_p \mathbf{F}^{\mathrm{bs}}_{\nu,\,i;p}$ corresponding to the rod bead and substrate particle in Eq. (2.6) is set to zero.

In the simulation of SPRs in presence of substrate, inter rod interaction and rod bead-substrate interaction are governed by SSLJ potential. Two values of $r_c$s,

which depend on the sizes of the rod-bead and substrate particle are used for SLLJ potential. These set of simulations are performed for *rough rods* in presence of substrate. Therefore the smoothness factor, $\chi$ of rod is set to be zero.

### 2.2.2 Initial and boundary conditions

For the simulation of SPRs without substrate particles, SPRs are arranged randomly in a square box of size, $L_{\text{box}}$ with periodic boundary conditions. To achieve the "random"[1] packing of rods, initially, rods are randomly packed in a much bigger box compared to the simulation box used for the study. The size of the box is then reduced to the desired system s size using `fix deform` in LAMMPS and position of the rods are mapped accordingly with image flags[2]. The initial velocities of the rods are set to zero.

To simulation the system of SPRs invading through elasto-plastic substrate, rods are initially arranged as a row at the bottom and top ends of a periodic simulation box of width $W$ and height $H$. Rods are also initially aligned with the vertical axis. Random packing of 2D spheres are generated using the packages PACKMOL[80] and MOLTEMPLATE [59] to fill the remaining space in the simulation box with substrate particles. The initial velocities of all rods and substrate particles are set to be zero. The polarity vector of each rod is set to point into the simulation box.

## 2.3 Dimensionless parameters

Most of the simulation studies related to active systems are conducted in non-dimensional forms. Therefore, it is recommended to perform computational studies in dimensionless units to compare results present in the literature. The choice

[1] The random packing fraction for 2D systems depends on the size and aspect ratio of particles. For the random packing of rod-shaped particles, rods are not entirely random above the critical density.

[2] Indicates the number of times a particle crosses the simulation box boundary in a particular direction.

of characteristic scales for non-dimensionalization depends on the quantities of interest involved in the study.

The present study emphasizes the influence of the mechanical properties of active systems on their emergent properties. Any physical dimension of a particle (either length or diameter) can be taken as a length scale. The time scale of the system is chosen depending on the fast and slow dynamics and quantity of interest, while the strength of fluctuation energy is taken as the characteristic energy scale. Such a choice of energy scale assumes that all the forces are scaled in comparison to the noise present in the system; hence noise level is an important quantity of interest for stochastic systems such as presented here.

### 2.3.1 Simulation of self-propelled rods

#### 2.3.1.1 Non-dimensionalization

The evolution of position of a bead of a rod, in terms of rod parameters can be written as the force balance [Eq. 2.6],

$$\frac{m_{\mathrm{r}}}{N_{\mathrm{b}}}\ddot{\mathbf{r}}_{\nu,\mathbf{i}} = -\frac{\gamma_{\mathrm{r}}}{N_{\mathrm{b}}}\dot{\mathbf{r}}_{\nu,\mathbf{i}} + \frac{F^{\mathrm{a}}}{N_{\mathrm{b}}}\cdot\hat{\mathbf{p}}_i + F^{\mathrm{bb}}\cdot\hat{\mathbf{u}}_{\mathrm{bb}} + \sqrt{\frac{2\gamma_{\mathrm{r}}k_{\mathrm{B}}T}{N_{\mathrm{b}}}}\frac{\eta_{\mathrm{r}}}{N_{\mathrm{b}}}. \tag{2.14}$$

Where, $\gamma_{\mathrm{r}}$ the friction coefficient of the rod, is $N_{\mathrm{b}}$ times the friction coefficient $\gamma_{\mathrm{b}}$ of a bead, $F^{\mathrm{bb}}$ is the interaction force on the bead and $\hat{\mathbf{u}}_{\mathrm{bb}}$ is the unit vector in the direction of bead-bead interaction. Eq. (2.14) in the dimensionless form can be written by choosing fundamental length scale, time scale and force/energy scales in the model. In the presented model of SPRs without substrate, for non-dimensionalization of the equations, diameter of the bead, $\sigma_{\mathrm{b}}$ as the length scale, rotational diffusion time-scale of a single bead, $\gamma_{\mathrm{b}}\sigma_{\mathrm{b}}^2/k_{\mathrm{B}}T$ as the time scale, and $k_{\mathrm{B}}T$ as the energy scale are chosen as the characteristics scales of the system.

Dimensionless quantities that emerge after scaling all the terms as multiple of characteristic scales are expressed with " $*$ " in superscript.

Eq. (2.14) can be written in dimensionless form as,

$$I_{\mathrm{r}}\ddot{\mathbf{r}}^{*}_{\nu,\,i} = -\dot{\mathbf{r}}^{*}_{\nu,\,i} + \mathrm{Pe}\cdot\hat{\mathbf{p}}_i + F^{\mathrm{bb}*}\cdot\hat{\mathbf{u}}_{\mathrm{bb}} + \frac{2}{N_{\mathrm{b}}}\eta^{*}_{\mathrm{r}}, \tag{2.15}$$

where, $I_{\mathrm{r}} = \frac{m_{\mathrm{r}}}{\gamma_{\mathrm{r}}\tau}$ is the inertia number of rod, $\eta^{*}_{\mathrm{r}}$ is non-dimensional noise, and $F^{\mathrm{bb}*}$ is the dimensionless EV force. In the over-damped limit $I_{\mathrm{r}} \ll 1$.

Other important non-dimensional numbers Pe and penetrability factor of rod, $Q^{\mathrm{bb}}$ . The general definition of the Pe is the ratio of convective time-scale and diffusive time-scale, which can be re-written in terms of rod propulsion strength and strength of the noise (see Appendix A).

$$\mathrm{Pe} = \frac{F^{\mathrm{a}}L}{k_{\mathrm{B}}T}, \tag{2.16}$$

$$Q^{\mathrm{bb}} = \frac{\mathrm{Pe}}{F^{\mathrm{bb}*}} = \frac{F^{\mathrm{a}}L}{E}, \tag{2.17}$$

The penetrability factor which quantifies the penetrability of rods is the ratio of propulsion strength to energy barrier.

#### 2.3.1.2 Choice of time-step size

Time-step for the simulation is estimated by comparing the characteristic time-scale with other time-scales in the system. To capture all the possible phenomena in the system simulation time step should be smaller than the other time scales. For a non-dimensional system, other time scales are scaled in terms of characteristic time scale, $\tau$.

$$\Delta t \ll \Big(\frac{m_{\mathrm{r}}}{\gamma_{\mathrm{r}}}, \frac{m_{\mathrm{b}}}{\gamma_{\mathrm{b}}}\Big) \ll \Big(\frac{\gamma_{\mathrm{b}}\sigma_{\mathrm{b}}^2}{k_{\mathrm{B}}T} = \tau\Big),$$

$$\Delta t^{*} \ll \Big(\frac{m_{\mathrm{r}}}{\gamma_{\mathrm{r}}\tau}, \frac{m_{\mathrm{b}}}{\gamma_{\mathrm{b}}\tau}\Big) \ll 1.$$

The non-dimensional numbers are summarized in Table 2.1.

TABLE 2.1: Simulation parameters

| **Parameter name** | **Parameter symbol** | **Value** |
|---|---|---|
| Rod-bead diameter | $\sigma_{\mathrm{b}}$ | 1.0 |
| Time-scale | $\tau$ | 1.0 |
| Translational noise strength | $k_{\mathrm{B}}T$ | 1.0 |
| Rod length | $L$ | 5.0 |
| System size | $L_{\mathrm{box}}$ | 100.0 |
| Inertia number for bead | $I_{\mathrm{b}}$ | $10^{-4}$ |
| Smoothness factor | $\chi$ | $0.0, 0.2, 0.33, 0.5, 0.7$ |
| Péclet number | Pe | $10 - 100$ |
| Bead-bead penetrability factor | $Q^{\mathrm{bb}}$ | 0.05 |
| Time-step | $\Delta t$ | $10^{-5}$ |

### 2.3.2 Simulation of self-propelled rod in elasto-plastic substrate.

#### 2.3.2.1 Non-dimensionalization

For the study of self-propelled rod invading through elasto-plastic substrate, evolution of a bead of rod can be given by adding a term corresponding to bead-substrate interaction, $F^{\mathrm{bs}} \cdot \hat{\mathbf{u}}_{\mathrm{bs}}$ in Eq. (2.14). $F^{\mathrm{bs}}$ is the interaction force between the rod bead and substrate particle, and $\hat{\mathbf{u}}_{\mathrm{bs}}$ is the unit vector in the direction of bead-substrate interaction

To non-dimensionalize the evolution equation, diameter of the rod, $\sigma_{\mathrm{b}}$ as the length scale, translational-diffusion time-scale of a bead, $\gamma_{\mathrm{b}}\sigma_{\mathrm{b}}^2/k_{\mathrm{B}}T$ as the time scale, and $k_{\mathrm{B}}T$ as the energy scale are chosen as the characteristics scales of the system. The dimensionless equation can be written as,

$$I_{\mathrm{r}}\ddot{\mathbf{r}}^{*}_{\nu,\,i} = -\dot{\mathbf{r}}^{*}_{\nu,\,i} + \mathrm{Pe}\cdot\hat{\mathbf{p}}_i + F^{\mathrm{bb}*}\cdot\hat{\mathbf{u}}_{\mathrm{bb}} + F^{\mathrm{bs}*}\hat{\mathbf{u}}_{\mathrm{bs}} + \frac{2}{N_{\mathrm{b}}}\eta^{*}_{\mathrm{r}}, \tag{2.18}$$

where, $F^{\mathrm{bb}*}$ and $F^{\mathrm{bs}*}$ are the dimensionless EV force. After non-dimensionalization, another dimensionless parameter, penetrability factor for bead-substrate emerges.

$$Q^{\mathrm{bs}} = \frac{\mathrm{Pe}}{F^{\mathrm{bs}*}} = \frac{F^{\mathrm{a}}L}{E}. \tag{2.19}$$

Similarly, equation for the evolution of substrate particles is also non-dimensionalized using above mentioned characteristic scales. Equation for dynamics (Eq. 2.11) of substrate particle can be given as,

$$m_{\mathrm{s}}\ddot{\mathbf{r}}_p = -\gamma_{\mathrm{s}}\dot{\mathbf{r}}_p + F^{\mathrm{sb}}\cdot\hat{\mathbf{u}}_{\mathrm{sb}} + F^{\mathrm{e}}_p\cdot\hat{\mathbf{u}}_{\mathrm{s}}. \tag{2.20}$$

Where, $F^{\mathrm{sb}}$ is the interaction force on the bead, $\hat{\mathbf{u}}_{\mathrm{sb}}$ is the unit vector in the direction of bead-substrate interaction, and $F^{\mathrm{e}}_p$ is the magnitude and $\hat{\mathbf{u}}_{\mathrm{s}}$ is the direction of elastic-restoring force. Eq. (2.20) can be recast in dimensionless form as follows,

$$I_{\mathrm{s}}\ddot{\mathbf{r}}^{*}_p = -\dot{\mathbf{r}}^{*}_p + F^{\mathrm{sb}*}\cdot\hat{\mathbf{u}}_{\mathrm{sb}} + S\cdot\hat{\mathbf{u}}_{\mathrm{s}}, \tag{2.21}$$

where, $S$ characterizes the elasticity of substrate particles .

A dimensionless number, relative plasticity number, $P$ that characterizes the elasto-plastic nature of substrate particles is given as,

$$P = \frac{\kappa l_{\max}}{F^{\mathrm{a}}} \tag{2.22}$$

which is a ratio of binding force of substrate and propulsion force.

#### 2.3.2.2 Choice of time-step size

As discussed in section 2.3.1.2, time-step for the system including substrate particles is calculated as following.

$$\Delta t \ll \left(\frac{m_{\mathrm{r}}}{\gamma_{\mathrm{r}}}, \frac{m_{\mathrm{b}}}{\gamma_{\mathrm{b}}}, \frac{m_{\mathrm{s}}}{\gamma_{\mathrm{s}}}\right) \ll \left(\frac{\sigma_{\mathrm{b}}^2 \gamma_{\mathrm{b}}}{k_{\mathrm{B}}\, T} = \tau\right)$$

$$\Delta t^* \ll \left(\frac{m_{\mathrm{r}}}{\gamma_{\mathrm{r}}\tau}, \frac{m_{\mathrm{b}}}{\gamma_{\mathrm{b}}\tau}, \frac{m_{\mathrm{s}}}{\gamma_{\mathrm{s}}\tau}\right) \ll 1$$

The non-dimensional numbers are summarized in the Table 2.2.

TABLE 2.2: Simulation parameters

| **Parameter name** | **Parameter symbol** | **Value** |
|---|---|---|
| Rod-bead diameter | $\sigma_{\mathrm{b}}$ | 1.0 |
| Time-scale | $\tau_{\mathrm{b}}$ | 1.0 |
| Translational noise strength | $k_{\mathrm{B}}T$ | 1.0 |
| Simulation box width | $W$ | 200.0 |
| Simulation box height | $H$ | 400.0 |
| Rod length | $L$ | 5.0 |
| Substrate particle diameter | $\sigma_{\mathrm{s}}$ | 0.6 |
| Number of beads in a rod | $N_{\mathrm{b}}$ | 5.0 |
| Number of rods | $N_{\mathrm{r}}$ | 400 |
| Number of substrate particles | $N_{\mathrm{s}}$ | 218542 |
| Bead inertia number | $I_{\mathrm{b}}$ | $10^{-4}$ |
| Rod inertia number | $I_{\mathrm{r}}$ | $10^{-4}$ |
| Substrate inertia number | $I_{\mathrm{s}}$ | $10^{-4}$ |
| Péclet number | Pe | 80,100,150,200 |
| Plasticity number | $P$ | 0.0025 – 0.075 |
| Harmonic potential cutoff | $l_{\mathrm{max}}$ | $0.6\sigma_{\mathrm{b}}$ |
| Bead-bead penetrability factor | $Q^{\mathrm{bb}}$ | 0.05 |
| Bead-substrate penetrability factor | $Q^{\mathrm{bs}}$ | 0.05 |
| Time-step | $\Delta t$ | $10^{-5}$ |

In the simulations of both the systems, SPRs in absence and presence of elasto-plastic substrate, inertia number for the rods, $I_{\mathrm{r}}$ and substrate, $I_{\mathrm{s}}$ are kept low ($\sim 10^{-4}$) to simulate an over-damped Langevin dynamics. In the simulation, a softer version of WCA, SSLJ potential, is used. The softness of the potential gives an inherent property of penetrability, $Q$ to the rods. If $Q > 1$, particles can cross each other, while for lower values of $Q$, particles are non-penetrable . Therefore, to simulate a system of non-penetrable rods, $Q$ must be kept low.

## 2.4 Model implementation in LAMMPS

LAMMPS is an open-source software, distributed under the terms of the GNU General Public License. It uses the Message Passing Interface (MPI) for parallel computing. LAMMPS is used for number of studies ranging the field of solid-state materials (metals, semiconductors) and soft matter (biomolecules, polymers)[88, 105, 37, 77] and coarse-grained or mesoscopic systems.

The differential Eq. (2.15) for free rods , Eq.( 2.18) for rods and substrate particles (Eq. (2.20) ), are integrated forward by the LAMMPS' `fix rigid/nve` package. Given the positions, orientations and velocities of the rods and particles at any time, the forces and torques on the right-hand sides of the dynamical equations are determined. The positions of the individual beads comprising the rigid rods and their velocities are then updated using rigid-body kinematics with a time-step of size $\Delta t$.

The following modifications of the LAMMPS source code were implemented in this study.

1. In the LAMMPS, point particles such as particle style `atomic` do not have the polarity attribute. To implement the self-propulsion force, the polarity/orientation attribute is added to the above atom style. The dynamics of disk-shaped/point particles is modelled by a modified `fix Langevin` command. The modified command takes an extra argument as the magnitude of self-propulsion force whose direction is evolved according to the following equation,

$$\dot{\theta}_i = \sqrt{2D_\mathrm{r}}\eta_i^R \tag{2.23}$$

Here, $\dot{\theta}_i$ is rate of change of polarity, $D_\mathrm{r}$ is rotation diffusion coefficient, and $\eta_i^R$ is Gaussian white noise.

2. The anisotropic self-propelled particles such as rods have been studied by using the direction of the principal axis as the polarity direction. To implement the self-propulsion force a new LAMMPS command, inspired by `bond harmonic`, `bond active` was developed. The command `bond active` uses the neighbour-list algorithm and inter-bead unit vector to apply a constant active force in the direction of the principal axis of the particle.

   `bond_style active`

   `bond_coeff N` $F_i^{\mathrm{a}}$ `r_active`

   where `N` is the bond type, $F_i^{\mathrm{a}}$ is the magnitude of the active force on each bead, and `r_active` is the inter-bead distance in a molecule.

3. The SSLJ potential (A.1) was implemented by modifying the source code of the LJ potential. The modified pair-wise interaction command takes an extra argument, $\alpha$, the capping parameter, as input.

   `pair_style lj/cut cutoff`

   `pair_coeff I J` $\epsilon$ $\sigma$ $\alpha$ `cutoff`

   where, `I J` are the bead types, `cutoff` is the cut-off of the interaction.

4. Substrate particles have been modelled as an elastoplastic material. The self-spring package was modified to implement the substrate dynamics. We put a cut-off and spring constant as the input variable in the LAMMPS input script. The modified command can be used as follows,

   `fix ID group-ID spring/self K cutoff`

   where, `ID` is the `fix` id, `group-ID` is the group on which fix is being implemented, `K` is the elastic stiffness and, `cutoff` is the cut-off of the elastic force.

5. To study the effect of continuous introduction of new rod, the source code of `fix pour` of LAMMPS was modified. In the original `fix pour` command, particles are introduced with random orientation, while in the modified version, orientation can be controlled.

The equations of motion are solved numerically using the Velocity-Verlet algorithm for all the beads and substrate particles. Position and velocities are updated at every time step. The equations are solved for the over-damped regime, where the effect of inertia is negligible. Therefore, inertia number, $I_{\mathrm{r}}$ or $I_{\mathrm{b}}$ of the system is kept much lower than 1.. The above numerical method is explicit; therefore, the chosen time step must be sufficiently small for the system to remain stable. As discussed earlier, for the convergent solutions of the equations of motion, the time-step size is chosen to be significantly smaller than the shortest physical time scale in the system.

# Chapter 3

# Characterization of spatio-temporal patterns

In this thesis, two types of systems are studied, (i) SPRs in absence of substrate and (ii) SPRs in presence of elasto-plastic substrate. A large number of static and dynamical morphological patterns are observed in these systems. In this Chapter, algorithms for identifying the morphological patterns and measuring their static and dynamic properties are discussed.

## 3.1 Characterization of dynamics phases and collective motion

In this section, algorithms developed for the system of collective motion of SPRs in absence of substrate is discussed.

### 3.1.1 Density distribution

In the simulation of SPRs without substrate, phase separation is observed similar to motility induced separation observed in a system of active Brownian particles.

The system is separated in low density and high density regions. The low density region is called bubble.

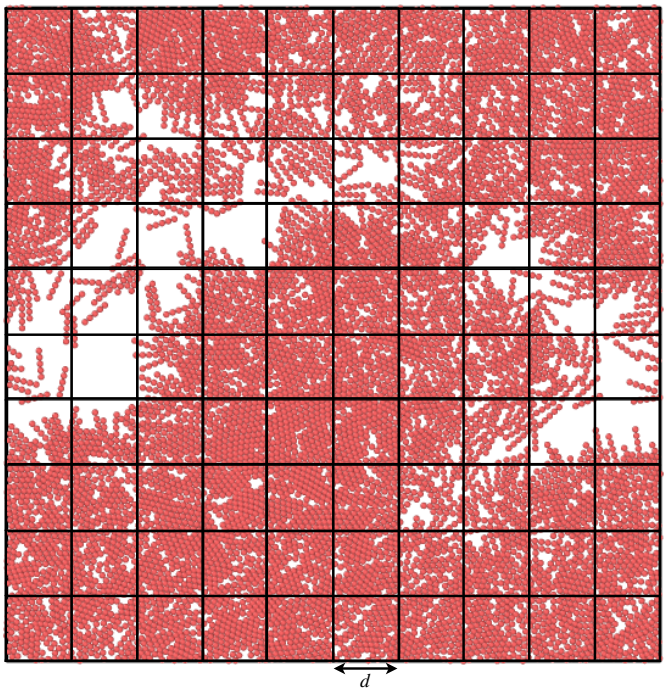


FIGURE 3.1: Discretization of simulation box into square cells of size $d$.

One way to quantify the phases such as bubble and dense, is to calculate the local density of the system. The local density is calculated by dividing the whole box into $N \times N$ bins of cell size of $d$ [Fig . 3.1], and counting the number of center of mass of rods in these bins. Once the number of center of mass of rods are obtained in each bin, its probability distribution is plotted. A system with multiple phases exhibits biomodal or tail behavior in density distribution function whereas, for a homogeneous system, unimodal distribution is observed.

### 3.1.2 Bubble identification and bubble size statistics

In order to quantify the bubble statistics, simulation box was divided into grids of square bin of size $d$. In each cell, number of rod beads is counted. If a bin contains any rod beads then it is labeled as 1 and empty bins are labeled as 0. Once the center of mass of all the empty bins are calculated, then cluster algorithm

is applied to get the bubble sizes. The size of cluster (made of connected empty bins) gives the area of bubble.

### 3.1.3 Spatial velocity correlation

In order to investigate the collective behavior of the system, spatial velocity correlation is calculated. Spatial velocity correlation reveals the coherent motion of the rods in the system. A higher values of correlation indicates a significant fraction of rods are moving with the same velocity whereas, a low value of correlation indicates the random motion. Spatial velocity correlation is calculated by,

$$C(r') = \frac{\langle \mathbf{v}(r).\mathbf{v}(r+r') \rangle}{\langle \mathbf{v}(r)^2 \rangle}$$

To estimate the size of a domain, exhibiting the collective motion, correlation length is calculated. It is obtained by fitting a decreasing exponential function to the spatial velocity correlation data. The exponential function is given as,

$$f(r') = a \exp^{-\frac{r'}{\lambda}} . \tag{3.1}$$

Here $a$ is the fitting parameter and $\lambda$ is the correlation length. Correlation length gives the size of the velocity domain for the particles showing the correlated velocities.

### 3.1.4 Streamline calculation

In the dense system of self-propelled rods, many non-equilibrium phases such as turbulence and vortex formation are observed. Streamline are calculated to visualize the flow of the instantaneous velocity field. The equation of the streamline parameterized by a parameter '$s'$ ("distance" along the streamline), for the '$i'$ the

component is

$$\frac{dr_i}{ds} = v_i \tag{3.2}$$

The length of the streamlines provide a snapshot of flow field behavior. For streamline calculation, simulation box is divided into $N \times N$ bins and instantaneous velocities are interpolated on the grid points.

## 3.2 Characterization of colonization and furrow network

In the simulation of colonization of elasto-plastic substrate, rods make furrow network by displacing the substrate particle from their equilibrium position. During the colonization process a complex furrow network is formed. To quantify such behavior a number of quantities need to be estimated such as colonization speed, cluster statistics and furrow network properties. In this section, methodology to calculate the colonization speed and furrow characteristics are discussed in details.

### 3.2.1 Cluster analysis

Cluster analysis is used for the calculation of colonization speed of elasto-plastic substrate. Cluster analysis involves mainly three steps; (i) Cluster size estimation (ii) Cluster shape identification and (iii) Cluster tracking

Clusters tracking commences only after the colonized area $C \geq 8\,L\,W$. Thereafter, we define a moving *frontier region* as the region with a distance of $8L$ behind $y_{\text{leader}}$ (the y-position of the leader rod). Cluster analysis is carried out for every time step in the frontier region until the simulation ends. At any instant of time, individual clusters are identified by scanning through the inter-bead distance matrix and identifying pairs of beads separated by $\sigma_{\text{b}}$ or less as belonging to the

same cluster. Since this assigns all beads belonging to any single rod to the same cluster, the cluster size – defined as the number of rods in a cluster – is the number of beads in a cluster divided by $N_\mathrm{b}$.

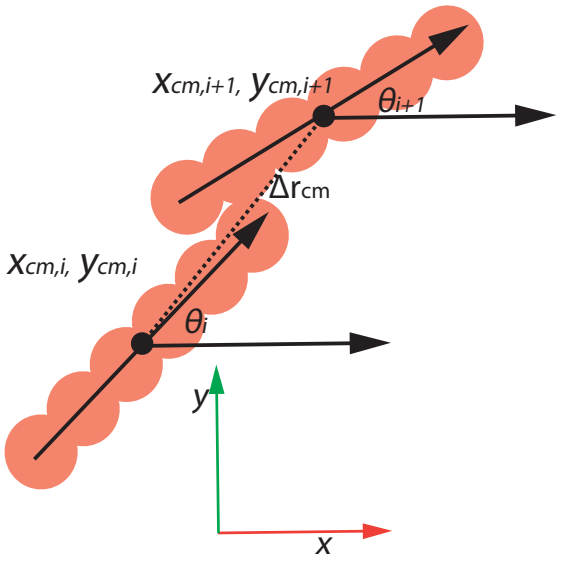


FIGURE 3.2: Cluster classification based on the orientation and centre of mass.

A contact-neighbour list of rods is generated for each cluster. For each contact neighbour pair, the separation between the centres of mass and relative orientation of contact neighbours, $R_{ij} = |\mathbf{r}_i - \mathbf{r}_j|$ and $Y_{ij} = |\theta_i - \theta_j|$ are calculated, where $\theta_i \in [-\pi, \pi]$ is the polar angle of the orientation vector, $\hat{\mathbf{p}}_i$. If $\mathcal{R}$ and $\mathcal{Y}$ are the sets of contact-neighbour separations and relative orientations, then a cluster is a *train* with rods arranged end-to-end as a line if

$$\min(\mathcal{R}) \geq L \quad \text{AND} \quad \max(\mathcal{Y}) \leq 7^\circ\,. \tag{3.3}$$

Any cluster that is not a train is classified as a *raft*. This includes rods with $Y_{ij} \approx \pi$, since such pairs tend to separate from each other. Isolated rods are counted as rafts of unit size.

We calculate the total number of train and raft clusters at each time step and obtain their time-averages, $J^\mathrm{T}$ and $J^\mathrm{R}$, in each simulation run. The fraction of train clusters for that run is, therefore, $f^\mathrm{T} = J^\mathrm{T}/(J^\mathrm{T} + J^\mathrm{R})$.

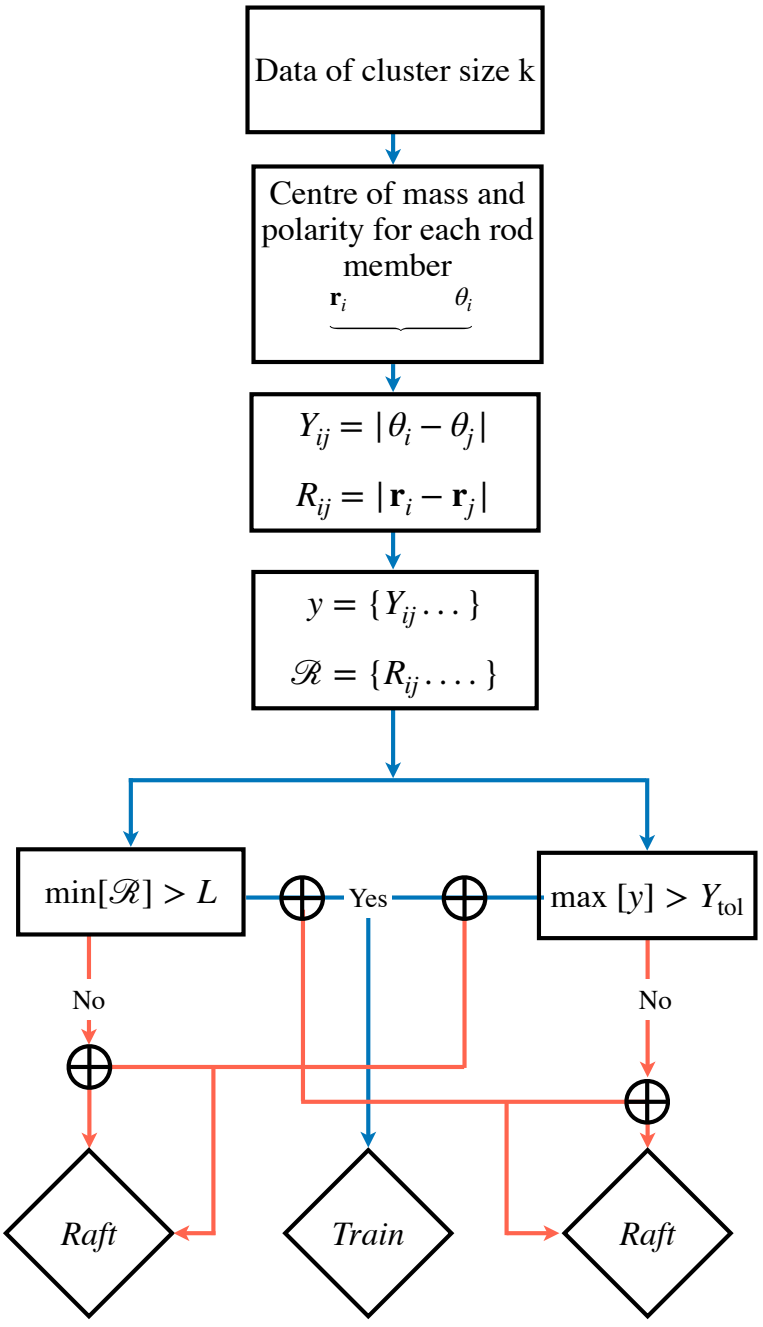


FIGURE 3.3: Algorithm for train and raft classification.

At any time, the orientation angle $\phi$ of a cluster is the average of the angle made with the vertical axis by member rods in that cluster; that is, the average of $\phi_i = (\pi/2) - \theta_i$. The orientational distributions $p(\phi)$ for trains and rafts in the frontier region is obtained by pooling together cluster data over all time-steps and simulation runs, for a given set of simulation parameters [Fig. 3.3].

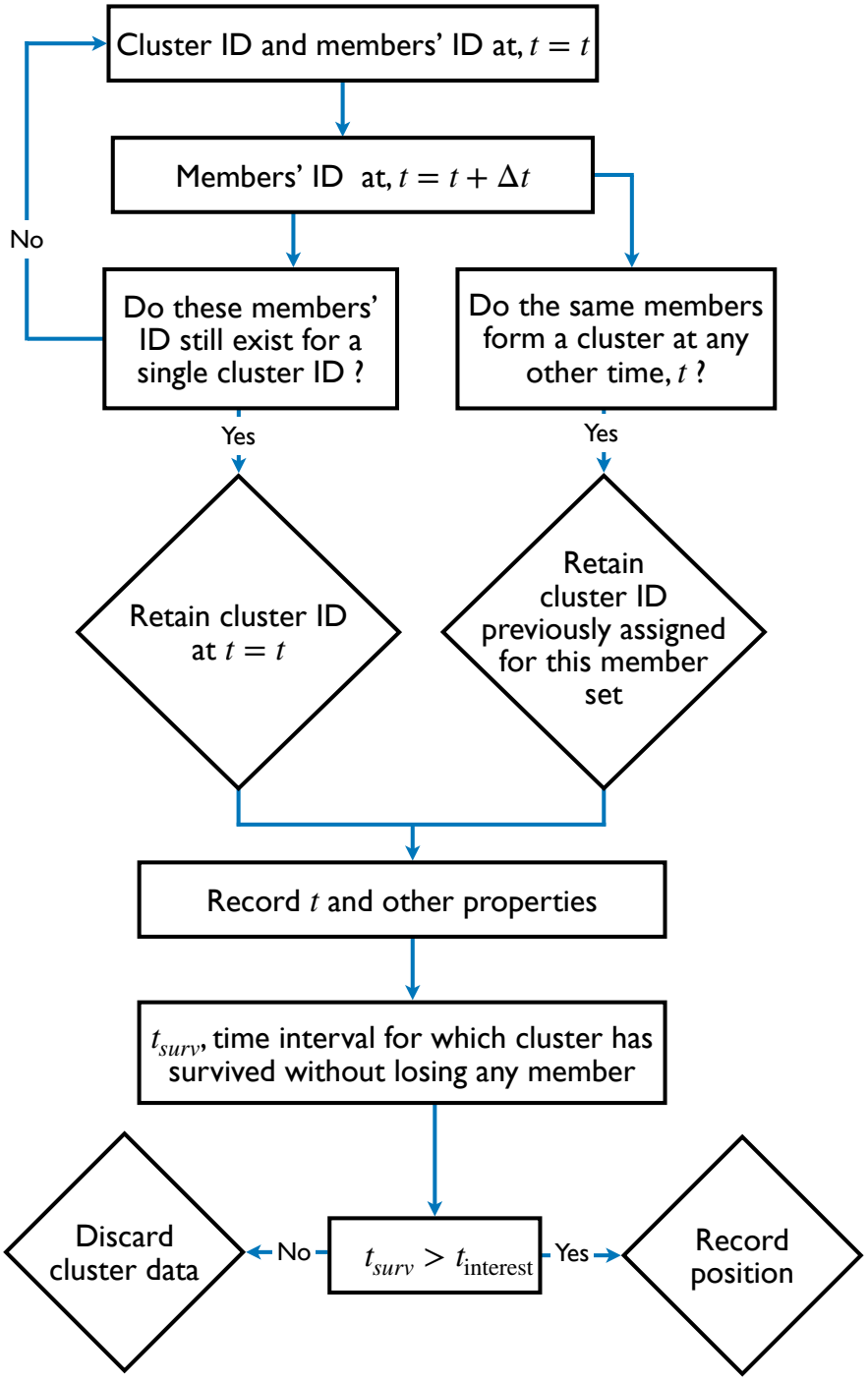


FIGURE 3.4: Cluster tracking algorithm

To analyze the speeds of clusters, individual clusters are tracked for the duration of their survival time. Once formed, a cluster may either gain new members or lose its original members. A cluster's identity remains unchanged for all the time steps during which its members remain unchanged. Every time a particle joins or leaves a cluster, the cluster's identity is changed and a new cluster is considered to have been formed Each cluster of a unique set of member rods is assigned a unique identifier. The average speed $v$ of a cluster over its survival time is calculated as the distance traveled by its center-of-mass divided by its survival time; the average $y$-component of its velocity, $v_y$, is its net $y$-displacement divided by the survival time. The velocity and speed data for all unique clusters across all time-steps and simulation runs are pooled together for trains and rafts [Fig. 3.4].

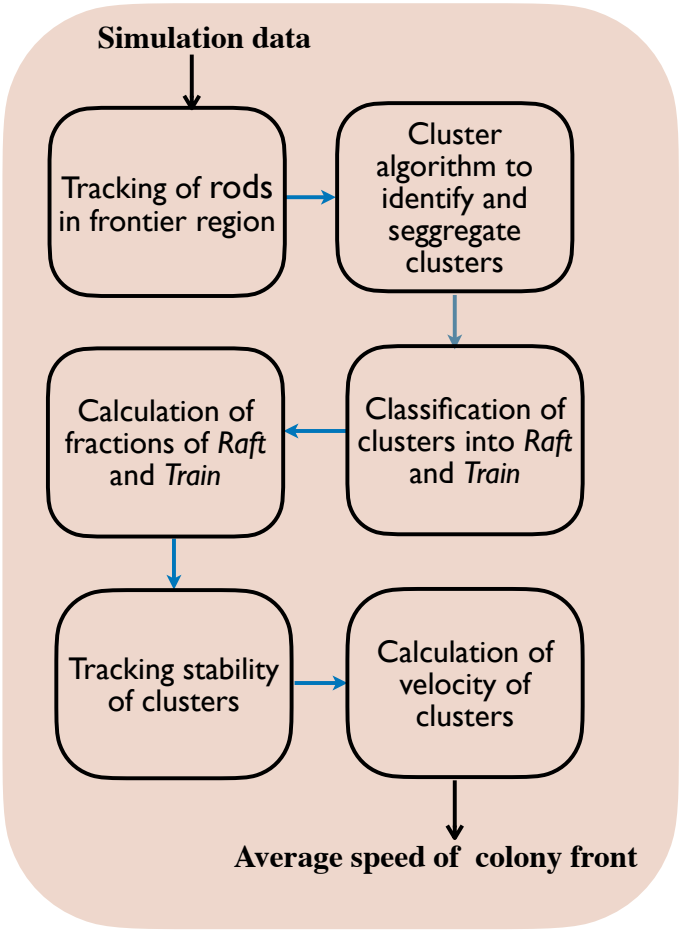


FIGURE 3.5: Schematic showing the steps in calculation of average speed of colony front.

Fig. 3.5 shows the steps involved in the estimation of average colonization speed.

### 3.2.2 Box counting method of fractal calculation

During the colonization, formation of different shapes and sizes of cluster leads to creation of complex furrow network. The furrow network is similar to the many self-similar objects such as coast line. For the characterization self-similar structure such as furrow network observed in the study, box counting is method. It is similar to the measuring perimeter of a structure using different sizes of scales.

It uses an algorithm in which fractal image is covered with a $N$ number of boxes of side $\epsilon_b$. As size of the box is varied, number of boxes that cover the structure changes. Box counting dimension (Fractal dimension) can be given as,

$$D = \lim_{\epsilon_b \to +0} \frac{\log N}{\log 1/\epsilon_b} \tag{3.4}$$

A code was developed which takes an image of furrow structure as input. Before apply the box counting algorithm it pre-treats the image into grey-scale image and then images is binarized into 0 and 1 according to the pixel counts. A matrix with elements 0 and 1 is obtained. Matrix containing non zero elements is counted in $N$. Then using the least square fitting, slope (Fractal number) can be calculated [Fig. 3.6].

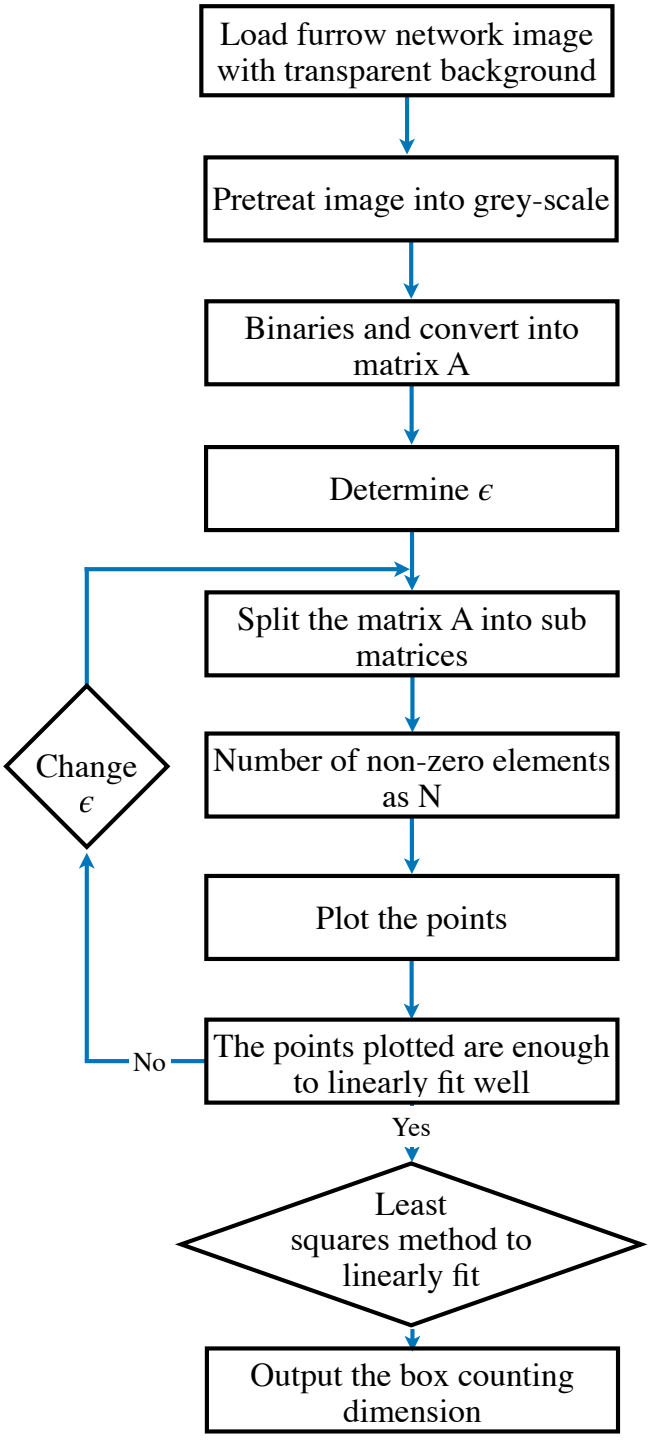


FIGURE 3.6: Algorithm for calculation of box counting fractal number.

# Chapter 4

# Effect of roughness on collective behavior in a dense system of self-propelled rods

The study of anisotropic particles has been of interest recently because of their wide applicability [93, 8, 138, 139, 109, 57, 7] . Many micro-organisms such as *P. aeruginosa*[47], *Bacillus subtilis* [38], *Myxococcus xanthus* [103], *Escherichia coli* [92], and *tobacco mesoporous* [81], have anisotropic rod-like shape. To study the behavior of such biological systems in a complex environment, one first needs to understand the behavior of these micro-organisms in a simplified setting by modeling them as self-propelled rods (SPRs) in an inert environment. A system of SPRs exhibits a large number of dynamical phases like polar clusters, moving bands, complex vortices, nematic chaos, and lane formation [100, 91, 2, 50, 69, 105, 120, 13, 7]. In the polar cluster, rods arranged in smectic order also have a prominent polar order due to the inherent nature of rods to move in parallel direction after collision. In an infinite system, these polar clusters rearrange themselves into moving bands showing a global orientational order [129]. The system in dense limit shows the formation of vortices and nematic chaos in which defects move and reorganize continuously [133]. SPRs with a high aspect ratio are known to show laning in which streams of rods move in opposite directions. In a system of penetrable SPRs,

phases such as small clusters, large clusters, and lanes are observed, depending on the system parameters such as propulsion force, density, and penetrability of rods [2].

Although these studies offer useful insights about the non-equilibrium phases in a system of SPRs, they do not consider the effect of varying rods' roughness on these phases. Modeling rods as a chain of spheres is popular and simple model [129, 2]; however, this substructure may induce shear friction or blocking of close-by, aligned rods that would not be present if rods could slide along each other during a collision in case of spherocylinder. In our model, roughness of a rod is controlled by changing the inter-bead distance. The inter-bead distance in the rod, in turn, changes the effective interaction for smooth and rough rods[23]. The interaction between the rods is governed by a short range repulsive potential. The details of the separation shifted Lennard-Jones potential (SSLJ) is discussed in Chapter 2. Thus the present study systematically explores the combined effects of excluded volume interaction and roughness on the collective behavior of SPRs in a dense system. The effect of roughness on static properties such as phase separation, bubble formation as well as on dynamic properties like streamlines and spatial velocity correlation is presented. This is crucial for understanding whether rod roughness in a dense system is a relevant parameter for studying non-equilibrium phenomena such as collective behavior.

## 4.1 The bubbly regime

To explore the effect of roughness of the rod on the dynamical behavior of the system, the smoothness factor, $\chi$, and the Peclet number, Pe are chosen as the control parameters of the system of constant number density. The detailed description of $\chi$ is given in Chapter 2. In the present study, $\chi$ ranges from zero (rough rod) to 0.7 ($\sim$ spherocylinder). Initially, rods are arranged in the simulation box with random orientation and random position without any overlap. When rods start moving, they collide and form clusters. During the collision, large clusters are

formed due to the merging of smaller clusters, while at the same time, smaller ones break up into even smaller-sized clusters due to noise present in the system. The formation of clusters thus leads to the emergence of dense and dilute phases in the system.

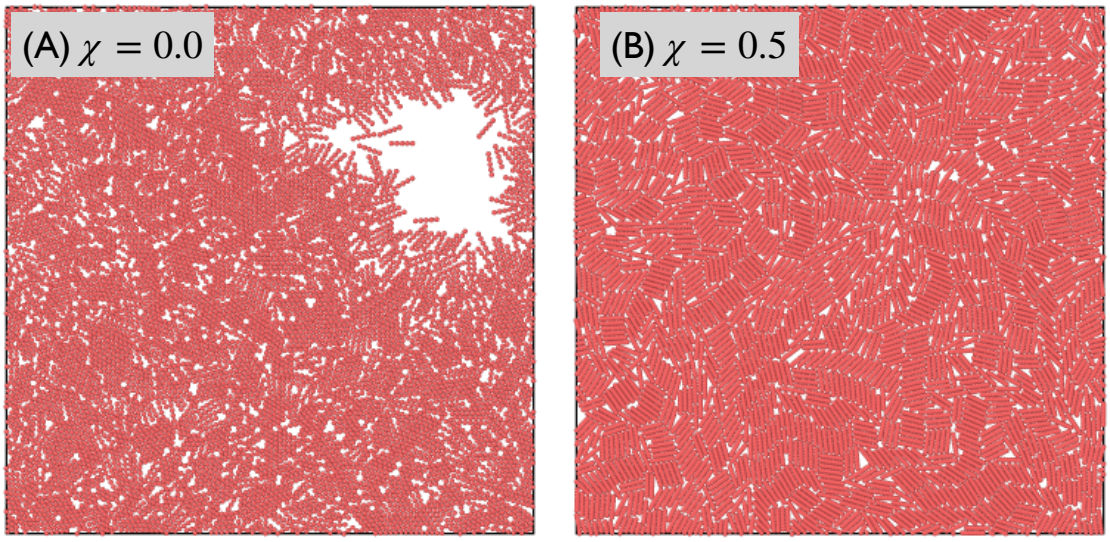


FIGURE 4.1: (A) Formation of dense and dilute regions (white space) in system at Pe = 100 for the case of chain-bead rod, $\chi = 0.0$. (B) For smoother rods, $\chi = 0.5$, system at Pe = 100 does not show phase separation.

In a system of repulsive active particles, phase separation into low density and high-density regimes are observed under certain parametric conditions [100, 108]. For a given Pe and number density, Figure 4.1 (A) illustrates the separation of a system of rough rods ($\chi = 0.0$), into dilute and dense regions, whereas the system remains in a homogeneous state for smoother rods ($\chi = 0.5$) as shown in Fig. 4.1 (B).

The quantification of phase separation behavior is done by calculating the local density distribution of the system. The procedure to calculate the density distribution has been discussed in Chapter 3. In what follows, dilute regions (where the rod density is small) are referred to as "bubbles".

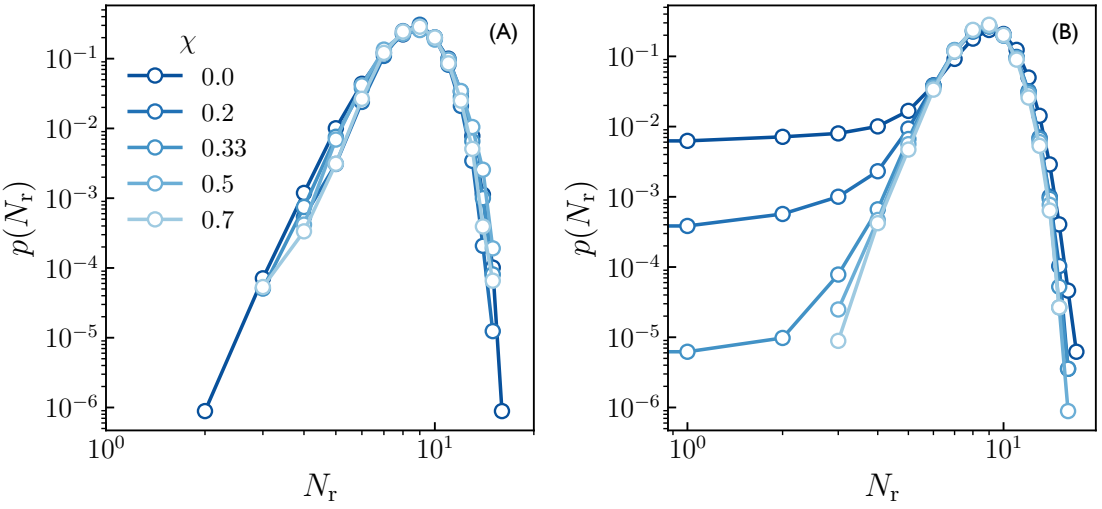


FIGURE 4.2: Local number density distribution of rod particles at (A) Pe = 10, (B) Pe = 100 for different values of $\chi$. The legend guide for (B) is the same as (A).

The density distribution at Pe = 10, is shown in Fig. 4.2 (A). This indicates that at low Peclet numbers, density distributions remain unimodal for all the values of $\chi$, indicating a single dense phase. At high Peclet numbers, the density distribution shows a pronounced plateau at low densities indicating the dilute phase in the system, for rough rods (low values of $\chi$), while for smooth rods (high values of $\chi$), the distribution remains unimodal [Fig. 4.2 (A)].

The formation of dense and dilute phases is similar to the phenomena of motility induced phase separation (MIPS) observed in a system of self-propelled Brownian particles[108]. During the collision of rods, smaller bubbles are continuously formed in the bulk of the system and are either ejected into the dilute phase or merge with an existing bubble [4.3 B]. This leads to the formation of stable bubbles in the system. Figure 4.3 shows the growth of bubbles with time, at $\chi = 0.0$ and Pe = 100.

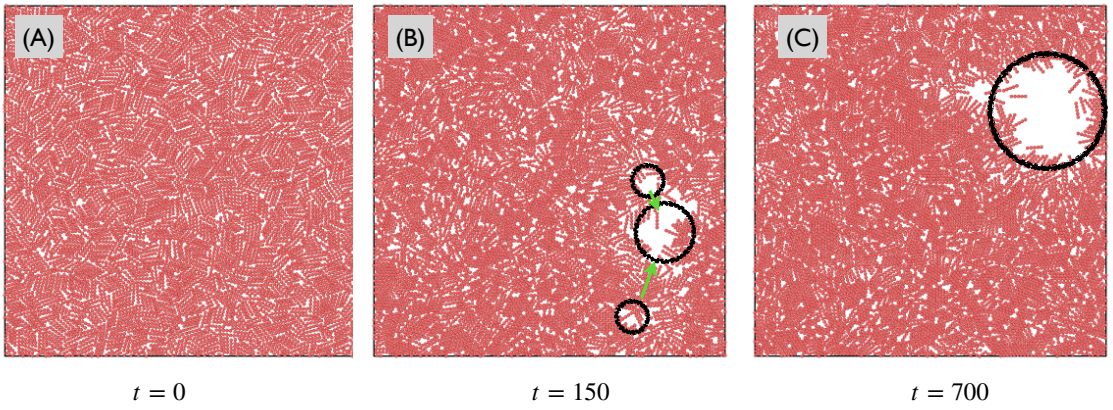


FIGURE 4.3: Bubble formation (white space) in system at Pe = 100 and $\chi = 0.0$ at different times:. (A) Initially, rods are arranged randomly, and the system is in a homogeneous state. (B) As the system evolves, the formation of smaller bubbles (black circles) takes place. The arrows show the direction of bubble movement during the merging. (C) The formation of a large bubble in the system at the steady-state.

Further quantitative evidence for phase separation is obtained from the bubble size distribution (BSD) when the system is in the steady-state. The definition of a bubble is as follows. An empty site (area of a cell after discretization of simulation box) in the system that does not contain any rod or rod-bead is called a bubble of size one (single-sized bubble). A bubble can also be a set of single-sized bubbles that are connected to their neighbors either directly or through other bubbles at a given time. The size of a bubble is then defined as the number of constituent single-sized bubbles. The details of bubble identification and the clustering algorithms discussed in Chapter 3 were applied to obtain BSDs in the cases where two-phase behavior is observed in the system.

The time-averaged BSD is calculated by pooling the data from all the time steps and different runs. At low smoothness factor, $\chi = 0.0$ and at low Pe, the distribution shows bubble formation, but these bubbles are smaller in size. As Pe increases, size of the largest bubble increases many-fold [Fig. 4.4(A)]. The distribution decays approximately as a power law, whose exponent increases with Pe for rough rods. In the case of a smooth rod, $\chi = 0.7$, bubbles formed in the system are significantly smaller in size, and the BSDs are almost independent of Pe [Fig. 4.4(B)].

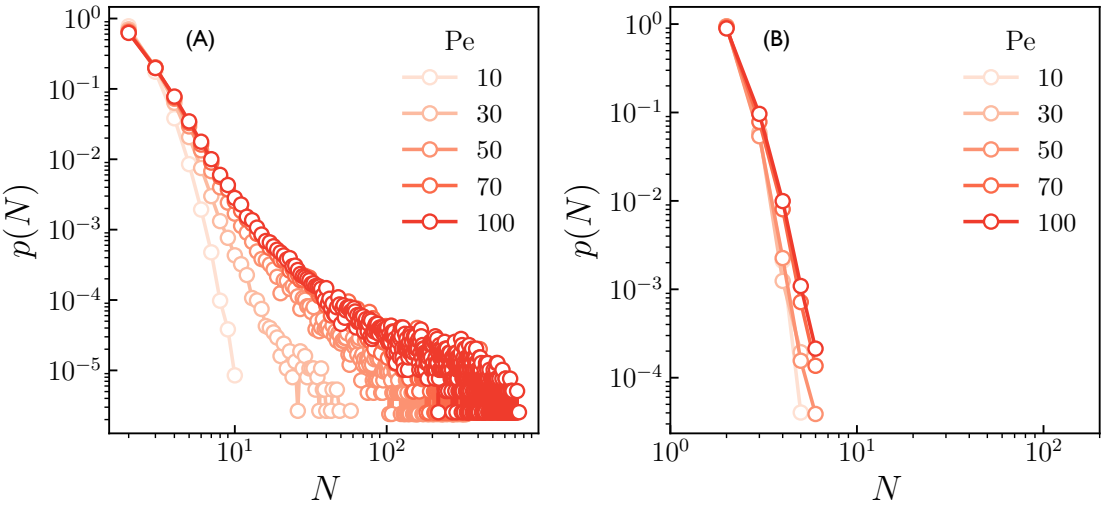


FIGURE 4.4: Bubble size distribution for rods having a smoothing factor of(A)$\chi = 0.0$ and (B) $\chi = 0.7$ at a range of Pe.

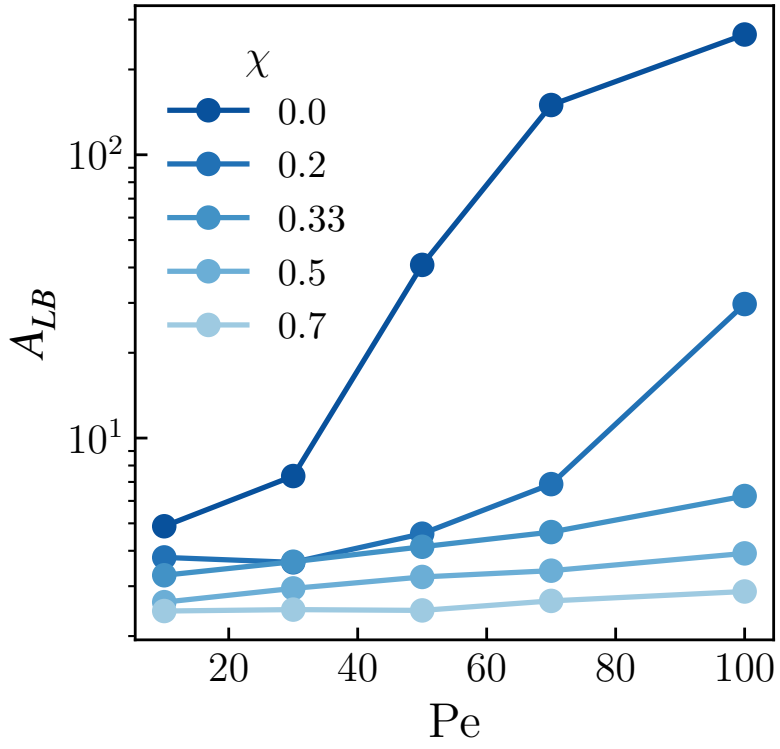


FIGURE 4.5: Variation of average $A_{LB}$, the size of largest bubble with Pe for of different smoothness.

The time-averaged size of the largest bubble, $A_{LB}$, formed in the system is calculated. Figure 4.5 shows that, for rods having lower values of smoothness factor ($\chi = 0.0, 0.2$), with increasing Pe, $A_{LB}$ increases abruptly at a certain Pe, indicating a transition from a homogeneous dense phase to a bubbly phase. However,

for smoother rods ($\chi > 0.2$), no significant bubbles are formed in the range of Pe explored in this study. A heat map constructed in Fig. 4.6 showing the variation of $A_{LB}$ across different Pe and $\chi$, identifies the bubbly region (high $A_{LB}$).

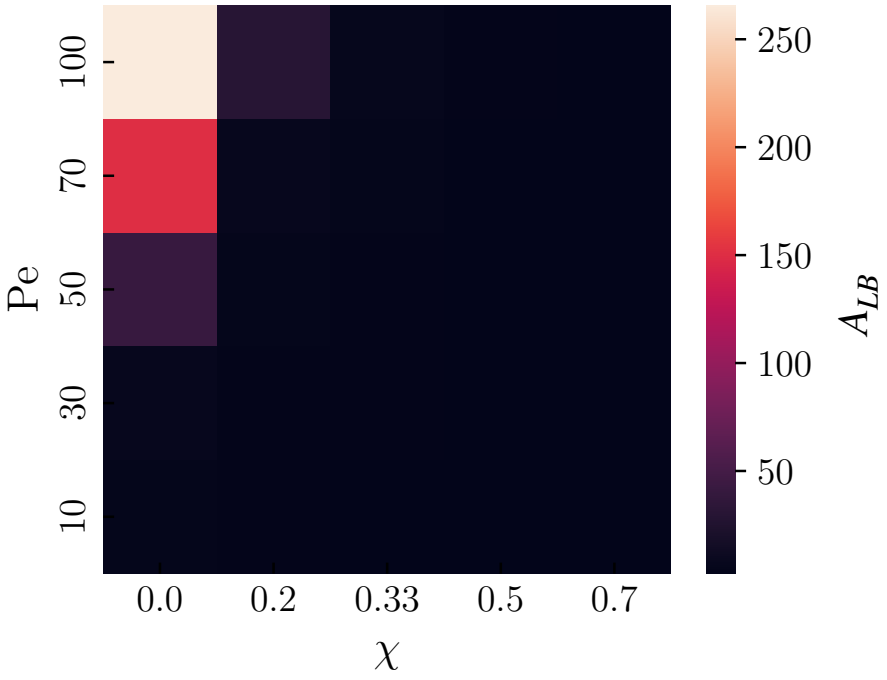


FIGURE 4.6: Heat-map of largest bubble size as a function of Pe and rod smoothness, $\chi$

## 4.2 Collective streaming

It is observed that within the dense phase of the system, rods either move collectively or exhibit random motion. Coherently moving clusters are formed during collective motion of the rods. Most of the clusters are stable and move across the system boundaries without breaking up. In this section, the influence of smoothness on the collective motion of SPRs is discussed by analyzing the velocity field of the system. This was investigated by calculating streamlines using the procedure discussed in Chapter 3.

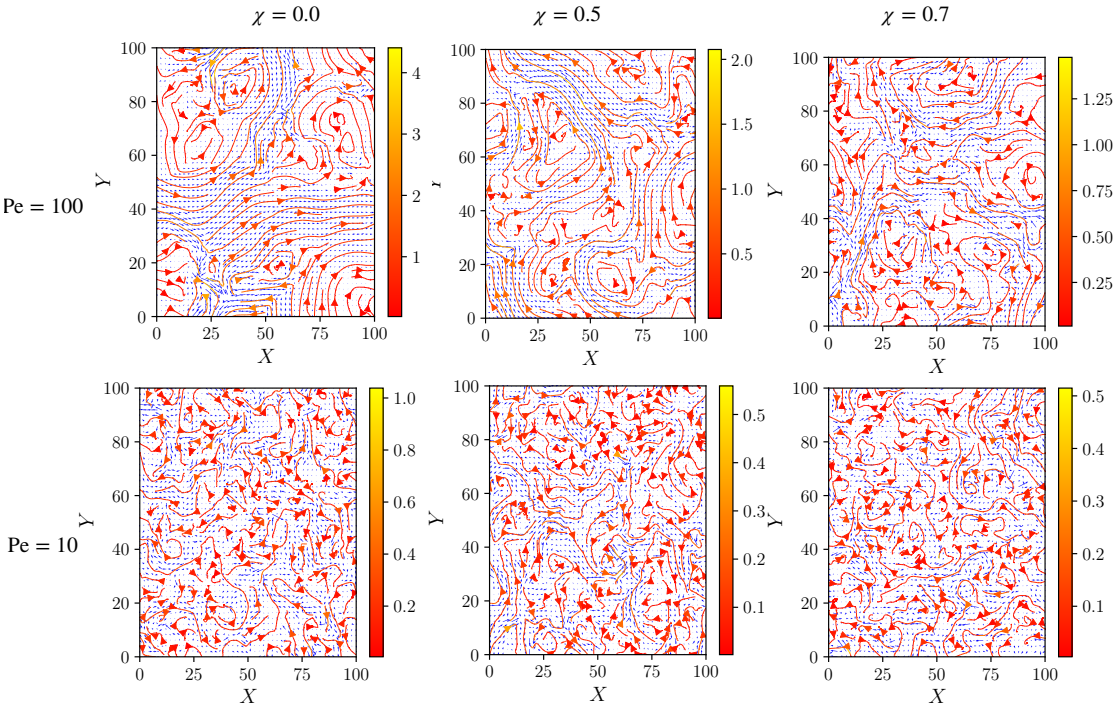


FIGURE 4.7: Snapshots of streamlines for the velocity field of SPRs at $\chi = 0.0$, 0.5, and $\chi = 0.7$, and at Pe = 100 and 10. Colorbar shows the speed of the rods.

Snapshots of the instantaneous streamlines for different values of $\chi$ and Pe are shown in Fig. 4.7. At low Pe (Fig. 4.7, bottom row), motion in the dense system is dominated by random motion as indicated by the broken, non-continuous streamlines. However, at high Pe (Fig. 4.7, top row), the smoother streamlines extend over regions comparable to the box size, indicating that streaming collective motion occurs. Surprisingly, at high Pe, the streamlines are smooth over shorter regions for smooth rods ($\chi = 0.7$) than for rough rods.

To further estimate the extent of coherent motion, the spatial velocity correlation,

$$C(r') = \frac{\langle \mathbf{v}(r).\mathbf{v}(r+r') \rangle}{\langle \mathbf{v}(r)^2 \rangle},$$

was calculated using the algorithm discussed in Chapter 3. The spacial velocity correlation, $C(r')$ takes a value of 1 for perfectly aligned moving rods, $-1$ for anti-aligned rod and 0 in the absence of any form of alignment.

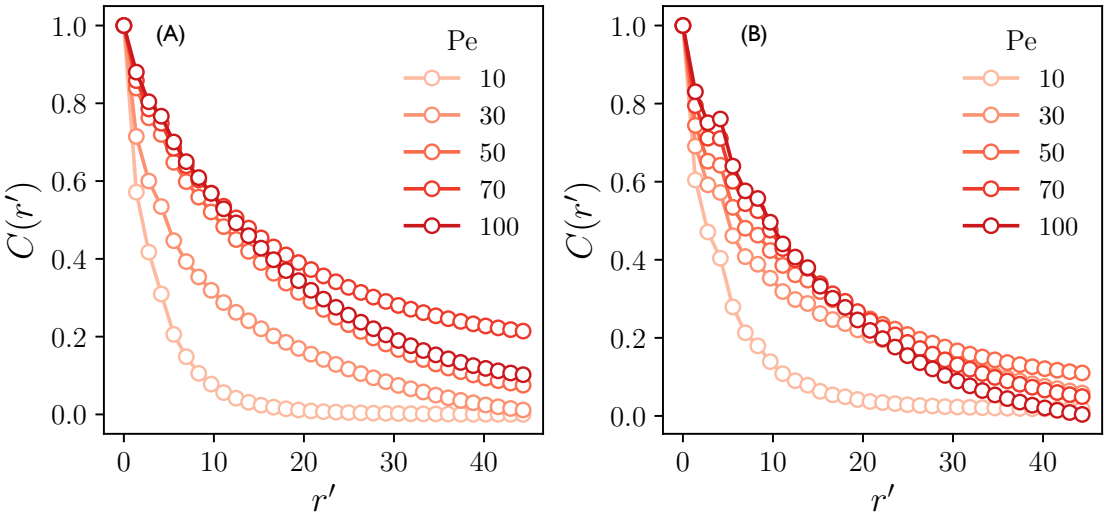


FIGURE 4.8: Spatial velocity correlation of rods at (A) $\chi = 0.0$ and (B) 0.5 at different values of Pe

In general the spatial velocity correlation, $C(r')$ is a decreasing function of $r'$. As shown in Fig. 4.8, the velocity correlation shows a steep decay at low values of Pe (around Pe = 10), for both smooth and rough rods, indicating that rod velocities are correlated only over length scales equal to a single rod length. This suggests no or very weak alignment of rods in the system. Coherent motion begins to emerge with increasing Pe, as indicated by the slower decay of $C(r')$ with $r'$. This indicates that the degree of alignment increases [19, 18]. It is worthwhile to note that spatial velocity correlation shows non-monotonic behavior with an increase in Pe. For rough rods, the correlation increases with an increase in Pe until Pe = 70.0 and then correlation decreases. Similar behavior is observed for smooth rods where the maximum correlation occurs at an even smaller Pe than the rough rods.

In Fig. 4.9 (A) and (B) the variation in correlation across different rod smoothness for a low and high Pe, respectively, is shown. The correlation remains constant for all $\chi$ at a low Pe, while it decreases with increasing $\chi$ at a high Pe. This trend is consistent with the observation in streamlines snapshots (Fig. 4.7 top row).

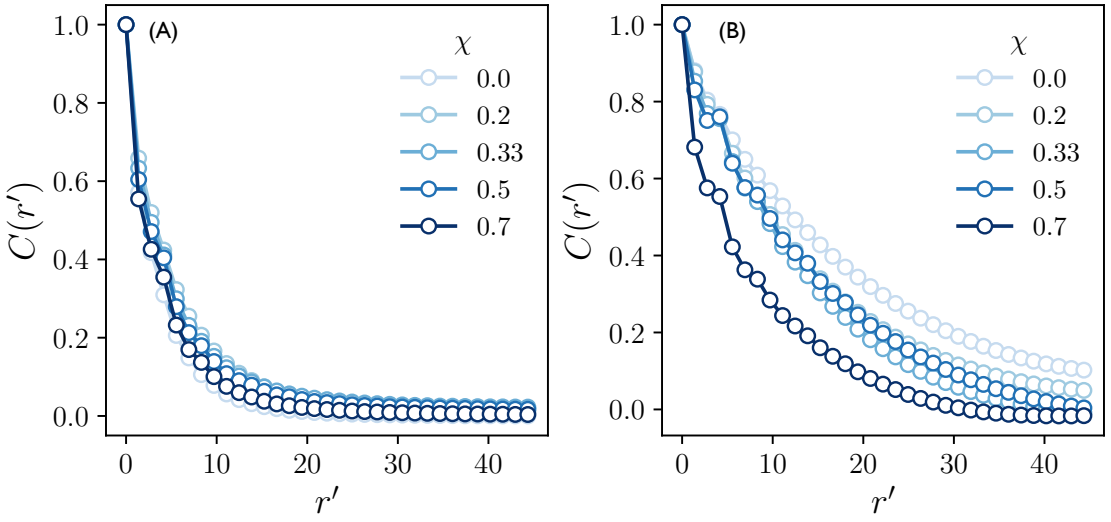


FIGURE 4.9: Spatial velocity correlation of rods at (A)Pe = 10 and (B)Pe = 100 at different values of $\chi$.

The correlation length $\lambda$ gives the size of domain within the simulation box, moving with correlated velocity. It is obtained by fitting a decreasing exponential function, $f(r') = e^{-r'/\lambda}$, to spatial velocity correlation data. Figure 4.10 shows the heat-map of $\lambda$ in the parameter-space of $\chi$ and Pe. At the lowest value of Pe, $\lambda$ is low. At larger Pe, $\lambda$ *decreases* with an increase in $\chi$. This indicates that rod roughness may cause rods to become locally jammed, causing their velocities to remain correlated for longer distances. For a fixed $\chi$, $\lambda$ shows a non-monotonic behavior with Pe.

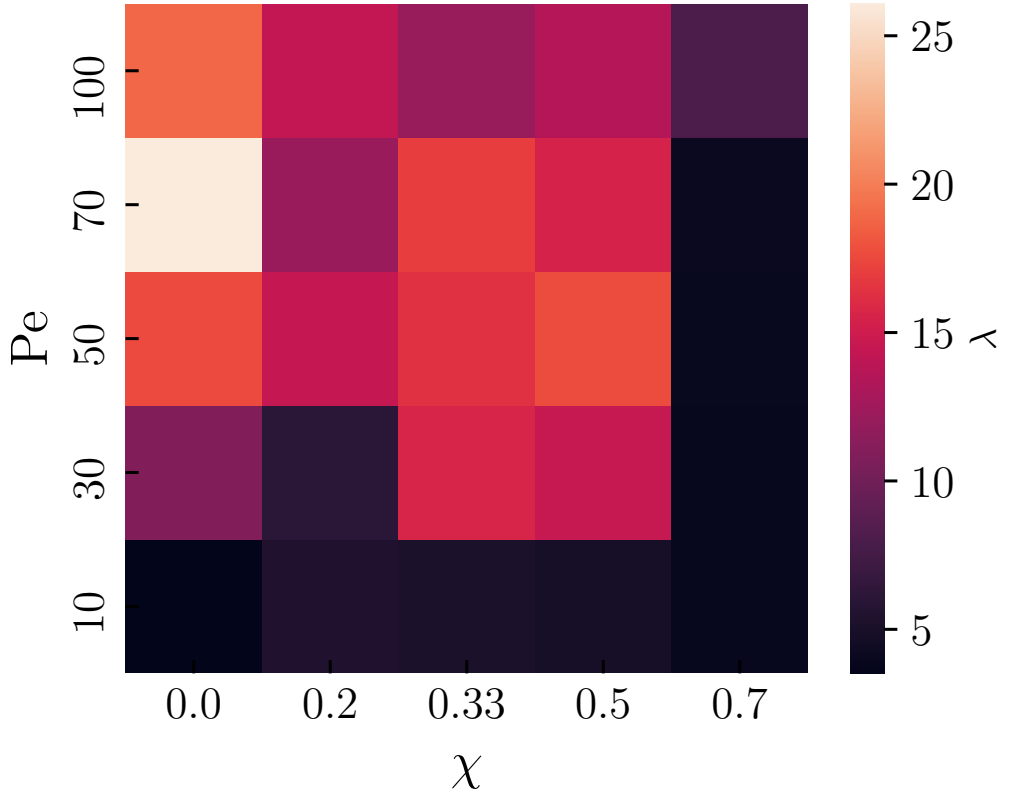


FIGURE 4.10: Heat-map for the velocity correlation length as function of Pe and $\chi$

The non-monotonic variation of correlation length with Pe at a given $\chi$ suggests the role of roughness in collective motion. It is observed that two rods ($\chi < 1.0$), which are not perfect spherocylinders, form an aggregate on collision. We speculate that when noise dominates at low Pe, the system exhibits random motion, while at moderate Pe, correlated motion is observed. Once the Pe is increased beyond a critical value (at which maximum correlation is observed), the aggregates break up, and correlated motion decreases. This seems to be consistent for a range of $\chi$. This also suggests that with an increase in $\chi$, the critical value of Pe should shift to a lower value, which is consistent with the observations. Further investigations are needed to verify the above claim.

## 4.3 Speed and velocity statistics

While simulating smooth rods, it is observed that rods do not move until a significantly high Pe, and the system exhibits a jammed-like state. To quantify such behavior, the speed statistics of the system are investigated. The average speed of the system is given as

$$|v| = \langle \mathbf{v}_i \rangle. \tag{4.1}$$

Here, $\mathbf{v}_i$ is the velocity of rod "$i$" and $\langle\ \rangle$ indicates the averaging over the number of rods in the system.

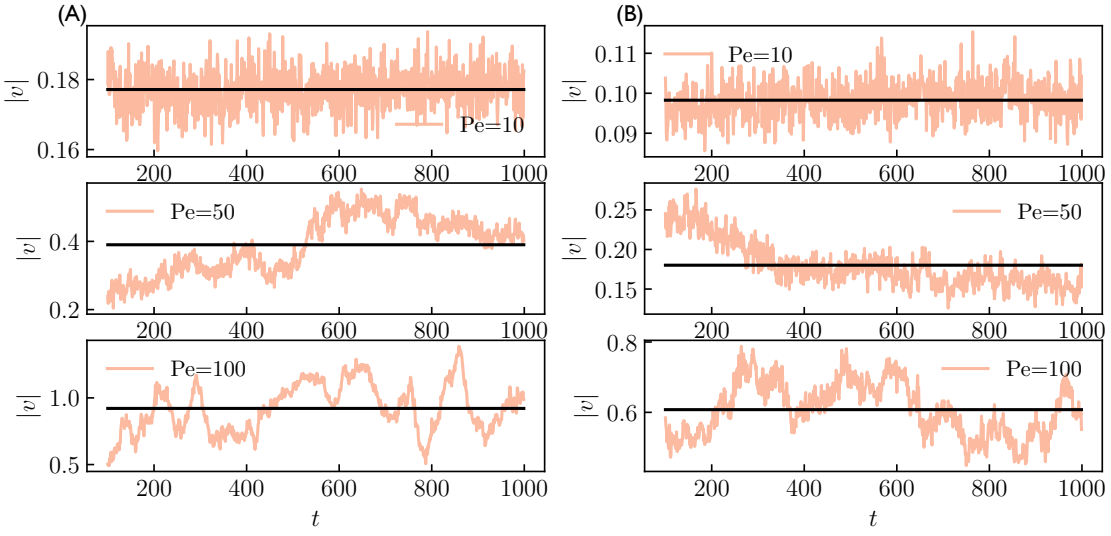


FIGURE 4.11: Speed as a function of time for a range of Pe at smoothness (A) $\chi = 0.0$. and (B) $\chi = 0.5$.

Fig. 4.11 shows the comparison of the average speed of system at different $\chi$ and Pe. At a given $\chi$, large fluctuations in the speed of the system are observed as Pe increases. Surprisingly, Fig. 4.11 shows that the average speed of the system decreases with an increase in $\chi$ at a fixed Pe. With an increase in smoothness factor, although the number density of the system remains constant, the effective packing fraction of the system increases. This leads to lower free space for the rods to move without interacting with others. With increase in the smoothness factor, a smoother rods offer more beads (interaction sites) for inter rod interaction

and surface area for contact. This leads to increase in number of interactions and effective friction of the system. This results in a lower speed of the rod and lower effective Peclet Number for the system.

Figure 4.12 shows the probability distribution of x-component of velocity for a range of Pe and smoothness factor (A) $\chi = 0.0$, and (B) $\chi = 0.5$. At lower (Pe = 10), distribution is Gaussian, however as Pe increases, a non-Gaussian tail behavior is observed. This shows that for a high Pe, there are a significant number of rods having high $v_x$.

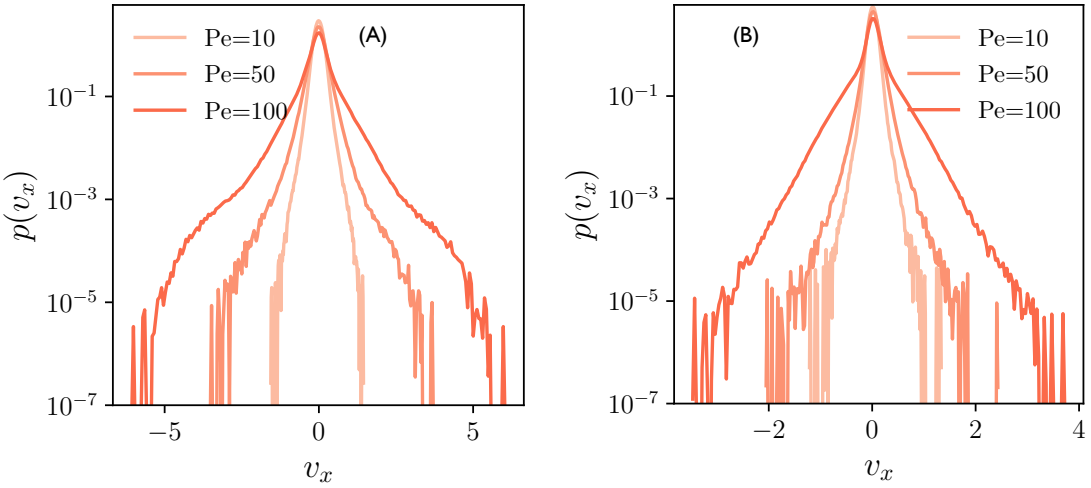


FIGURE 4.12: Distribution of x-component of the velocity of rods at (A) $\chi = 0.0$ and (B) $\chi = 0.5$ at different values of Pe.

We further investigate the effect of roughness on the direction of movement of rods, by calculating the projection of unit velocity vector along the unit polarity vector, $\hat{v} \cdot \hat{p}$ . If this quantity is $-1$, rods move in the opposite direction of their polarity direction and $+1$ if rods move in the direction of their polarity direction.

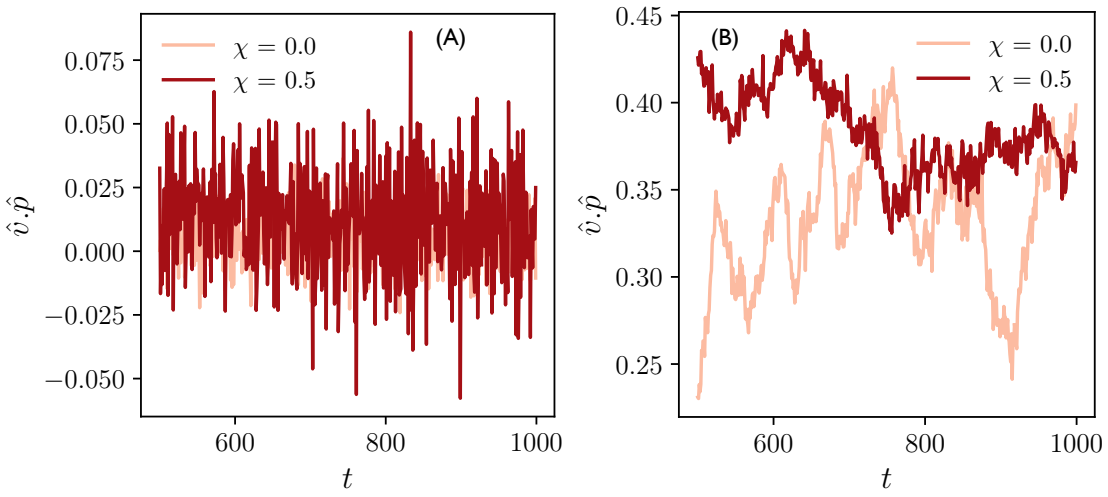


FIGURE 4.13: $\hat{v}\cdot\hat{p}$ for rough rod, $\chi = 0.0$ and smooth rod, $\chi = 0.5$ at (A)Pe = 10 and (B) Pe = 100

Fig. 4.13 (A), shows that at Pe = 10, $\hat{v}\cdot\hat{p}$ is almost zero. This suggests that rods move randomly with respect to their polarity direction. However at Pe = 100, the average value of $\hat{v}\cdot\hat{p}$ is slightly higher for smoother rods, ($\chi = 0.5$) compare to the rough rods. This suggests higher number of smooth rods move in the direction of their orientations compare to rough rod [Fig. 4.13 (B)]. This observed behaviour further needs to be confirmed by running simulations for higher time-scales.

## 4.4 Phase diagram

Based on the values of largest bubble size, $A_{LB}$ and correlation length, $\lambda$ patterns observed in the system are divided into different phases. In order to characterize these patterns, a phase diagram is constructed in $\chi$ and Pe space.

To distinguish different phases in the phase diagram, the following criteria are used.

$$\begin{cases} \text{Jammed Phase,} & \text{if } \lambda \leq 1.5L \& A_{LB} \leq L^2 \ . \\ \text{Bubbly Phase,} & \text{if } \lambda > 1.5L \& A_{LB} \geq L^2 \ . \\ \text{Liquid Phase,} & \text{if } \lambda > 1.5L \& A_{LB} \leq L^2 \ . \end{cases}$$

The region with a lower correlation length and larger size of the bubbles does not appear in the simulations. This also suggests that bubble formation and collective motion of rods are coexisting phenomena.

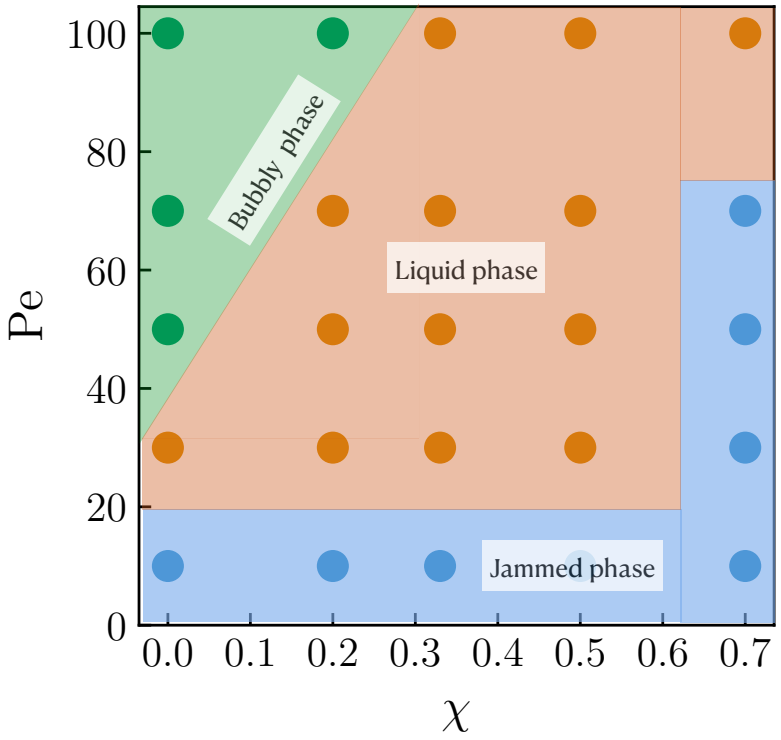


FIGURE 4.14: A phase diagram of the system in $\chi$ and Pe parameter space.

Fig. 4.14 shows the different phases observed in Pe and $\chi$ space. In the jammed phase, rods neither exhibit coherent motion nor the formation of bubbles. In this phase, rods rarely move from their initial position due to the lower effective Pe. However, in the liquid phase, collective motion of rods is observed without bubble formation, and the system shows large values of the correlation length. In this phase, motile clusters with continuous break-up and formation are observed. In the bubbly phase, the formation of bubbles takes place. The size of the bubble increases with an increase in Pe. However, it decreases with an increase in smoothness factor $\chi$.

The formation of bubbles in active systems has been predicted in literature using a continuum model of MIPS [117]. The emergence of bubble phase for the particle system of self-propelled Brownian particles has also been reported recently [17, 111]. However, the emergence of such phases in a system of rods has not been

studied yet. In the continuum model of bubble formation, bubbles are formed in a particular system parameter space. However, such parameter space can not be directly translated in terms of number density, Pe or smoothness of a rod for the particle-based system. In the present study, contrary to the mean-field calculations [117], bubble formation is observed as long as the system shows phase separation. The present study also reports that bubble formation strongly depends on the surface smoothness $\chi$ and Pe for a dense system of rods.

## 4.5 Simulation at constant packing fraction

In this section, the cluster formation phenomena is studied for a system of constant packing fraction, $\varphi$, with a varying number density. The packing fraction of the system is given as,

$$\varphi = \frac{N_{\text{rod}} A_{\text{rod}}}{A_{\text{box}}}. \tag{4.2}$$

Here, $N_{\text{rod}}$ is the number of rods in the system, $A_{\text{rod}}$ is the area of a rod, and $A_{\text{box}}$ is the area of the simulation box. As discussed in Appendix A, the area of a rod increases with an increase in rod smoothness. The area of a rod is given as,

$$A_{\text{rod}} = \frac{N_{\text{b}}\pi\sigma^2}{4} - \frac{N_{\text{b}}-1}{2}\Big[\sigma_{\text{b}}^2 \cos^{-1}\Big(\frac{1-\chi}{\sigma_{\text{b}}}\Big) - (1-\chi)\sqrt{\sigma_{\text{b}}^2 - (1-\chi)^2}\Big]. \tag{4.3}$$

Here, $N_{\text{b}}$ is the number of beads in the rod, and $\sigma_{\text{b}}$ is the diameter of the bead.

Simulations are carried out for rough rods, $\chi = 0.0$ and smoother rods, $\chi = 0.5$ at fixed packing fraction, $\varphi = 0.786$. The configuration and speed of rods for different smoothness factors, $\chi$ is shown in Fig. 4.15.

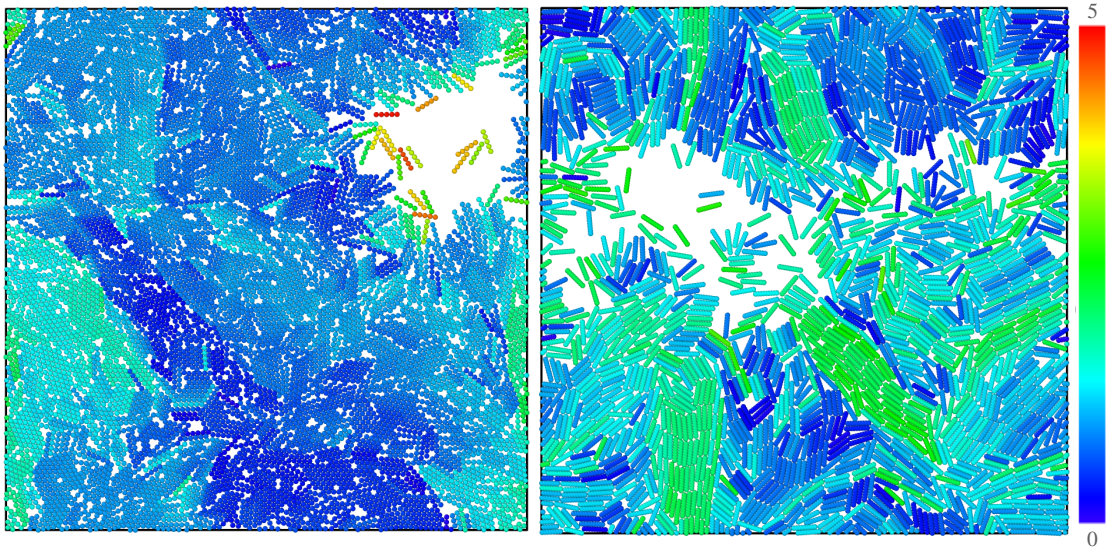


FIGURE 4.15: The configuration and speed of rods at Pe = 100 at two different smoothness factors (A) $\chi = 0.0$, and (B) $\chi = 0.5$. The colors indicate the speed.

It is observed that for both high and low values of $\chi$, clusters are formed in the system. For the rough rods, clusters are stable and move with a similar velocity [Fig. 4.15 (A)]. The collision of clusters in the system leads to the formation of aggregate that spans the size of the simulation box. In the aggregate, rods have different orientation and try to move in their respective polarity directions. This effects lower the overall speed of the aggregate as observed in (Fig. 4.15 (A)). While in the case of smooth rods, clusters are more fluid, and the rate of break-up and joining the clusters is more frequent. It is also observed that in the case of smooth rods, many small lanes are formed in the system [Fig. 4.15 (B)]. These lanes are observe as a subset of a bigger cluster. The colorbar in Fig. 4.15 indicates the speed of rods in the system. Contrary to the previous cases (constant number density) it is also observed that phase separation and bubble formation take place for both high as well as low values of $\chi$.

The above observations indicate that in a system of rods at constant packing fraction, smoothness plays a crucial role in the collective motion of rods. For rough rods, stable clusters are formed due to the fact that inter-bead spacing of the rod helps in the trapping of other rods in a cluster. This also leads to the formation of stable clusters, as observed in the simulation. The inter-bead spacing

in smooth rods is small, the trapping effect is minimal, and thus upon collision, clusters quickly break up, and small polar clusters are observed.

## 4.6 Summary

The model of SPRs (made of overlapping beads) presented in this chapter shows that the smoothness factor is an important parameter to study their collective behavior. The results indicate that a dense system of SPRs exhibits phase separation when SPRs have a lower smoothness factor. As a result, bubbles are formed in the system, whose sizes depend on the $\chi$ and Pe. While for the smooth rods, no significant phase separation and bubble formation are observed. The spatial velocity correlation varies non-monotonically with Pe at a given $\chi$. The correlation is maximum around the critical value of Pe, beyond which the effect of inter-bead separation on the trapping of rods in a cluster is minimal. This value of Pe appears to depend on $\chi$ and decreases with an increase in $\chi$. Interestingly, it is also shown that the average speed of the system decreases with an increase of the smoothness factor. The decrease in speed of the system is due to the fact that the effective area of a rod increases as rods become smoother. The increase in the area of rods reflects an increase in the packing fraction of rods in the system. This leads to an increased interaction and frictional dissipation between the rods, which lowers the system's average speed. The above-reported phases, such as bubble formation, are new to the particle-based system of SPRs and a key contribution of this work. This chapter conducts several studies to examine the role of parameters like rod smoothness and Pe on the emergence of different phases. We also conclude that, in a dense system, the role of smoothness in collective motion of a system of SPRs is significant.

# Chapter 5

# Morphodynamics of invasion of a soft substrate by active rods

Experiments with motile bacteria growing on the surface of an agar hydrogel have shown that the cells generate an interconnected network by furrowing through the soft substrate [47, 49, 46, 52]. It has further been shown through simulation studies that the colony morphology depends on the mechanical properties of the substrate [140, 141]. In this Chapter, this behavior is explained using simulations of active rods and the minimal model of a plastic substrate described in Chapter 2. The effect of mechanical properties such as plasticity of substrate, rod motility and their aspect ratio on the dynamics of the colony expansion and the morphology of the furrow network are presented.

## 5.1 Formation of furrow networks

To explore the effect of mechanical properties of the substrate on the morphological behavior of the system, the Peclet number, $\mathrm{Pe} = F^{\mathrm{a}} L/k_{\mathrm{B}} T$ and the plasticity ratio, $P = k\, l_{\max}/F^{\mathrm{a}}$ are chosen as control parameters for the system. In the present study, limit of the plasticity is such that, rods at moderate Pe make permanent furrows in the system. As a results, at all the values of Pe, and $P$ that

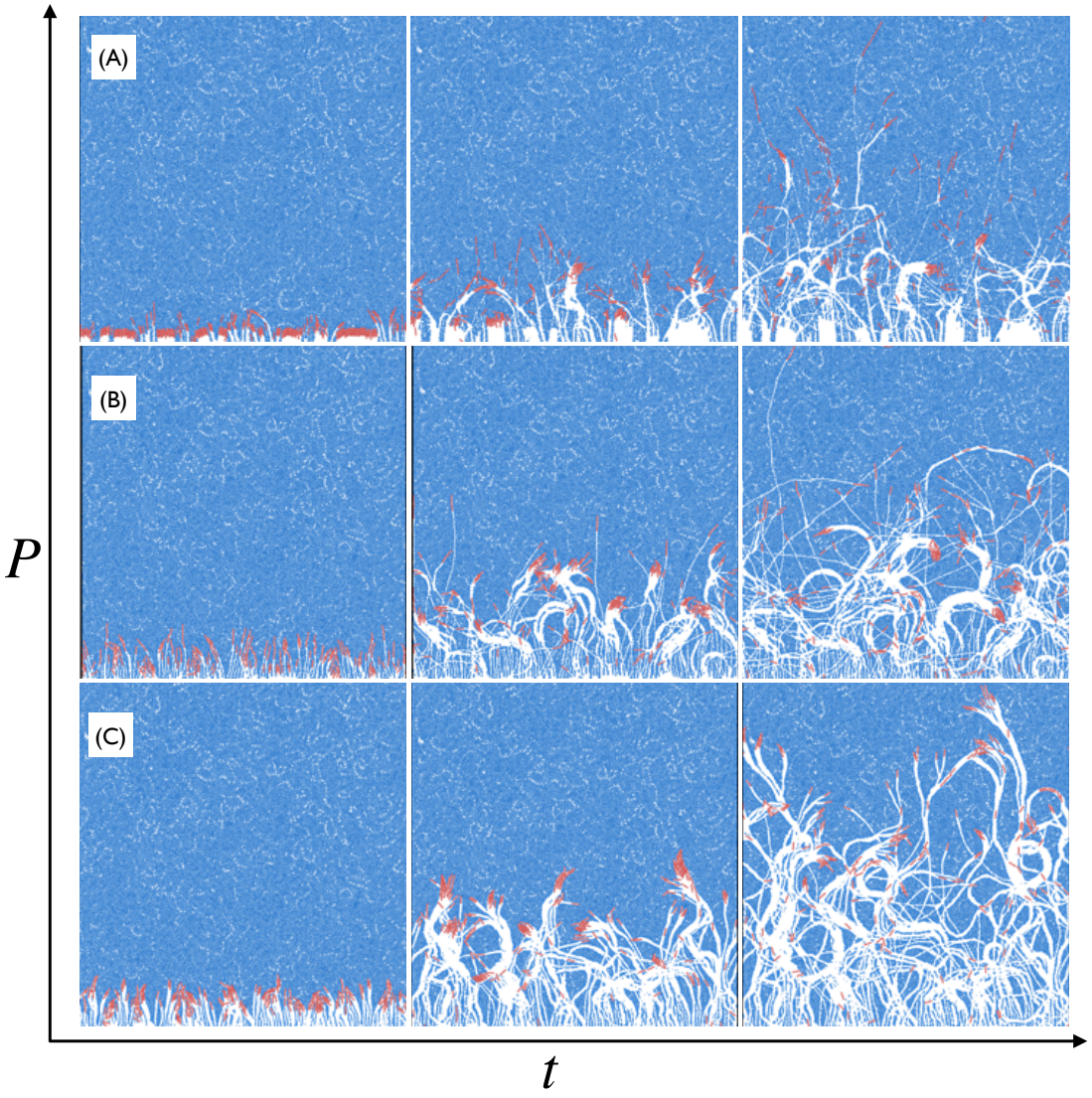


FIGURE 5.1: Evolution of furrow networks (white space) by active rods advancing at Pe = 100 into a soft substrate of plasticity, $P$ = (A) 0.075, (B) 0.035, and (C) 0.0025.

were investigated, a single rod can displace substrate particles sufficiently to break them off and create permanent furrows *i.e.* the substrate is plastic. A row of active rods released at the bottom of the simulation box thus rods creates permanent furrows as they move ahead. The minimal model of the substrate does not have substrate-substrate excluded-volume interactions. The plastically-displaced substrate particles are freely moved by the rods.

Figure 5.1 shows the evolution of the furrow network morphology with time for different values of the substrate plasticity, $P$. Rods are initially vertically oriented and begin individually creating relatively straight and narrow furrows. Small

orientational fluctuations cause the rods to start colliding with neighboring rods to begin forming dynamic clusters. In simulations with random initial orientations, rods begin forming dynamic clusters from the beginning. As the clusters propel through the substrate, they permanently displace substrate particles, forming a complex network of permanent furrows.


FIGURE 5.2: Furrow network (in white) formation in a substrate of plasticity, $P = 0.0025$, at $\mathrm{Pe} = 100$. Active rods are colored red and substrate particles are blue. Adjacent figures show close-up images of two distinct types of motility-induced clusters: *rafts* are arrow-head shaped clusters, while *trains* consist of rods arranged end-to-end.

It is observed that when the substrate plasticity is low, the rods begin to form small, bullet-shaped or pointy clusters which push into the substrate, forming furrows (Fig. 5.3). These clusters are henceforth referred to as *rafts*. These rafts are further not perfectly symmetric. As a result, the net force exerted on them by the substrate particles has a component that is transverse to the net velocity of the cluster, causing rafts to veer and create arc-shaped furrows (Fig. 5.3 A) The repeated formation and intersection of these arc-shaped furrows with neighboring furrows leads to the formation of a highly ramified furrow network.

In stiffer substrates with high values of $P$, although rafts form initially, they are unstable and break up through a variety of mechanisms (Fig. 5.3 B - D). At high $P$, there is another kind of cluster that appears more stable: *trains* of rods (Fig. 5.1 (a)). The furrows formed by these trains are thinner and straighter than

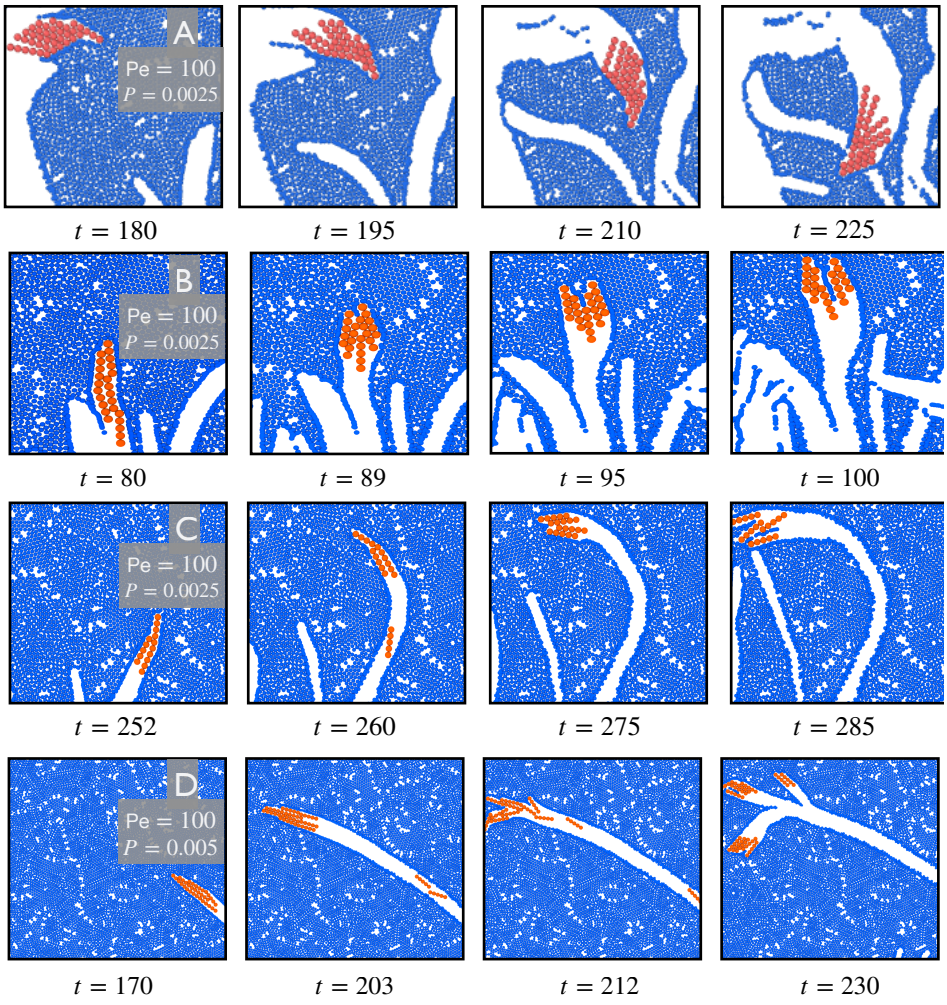


FIGURE 5.3: Dynamics of pointy raft clusters. (A)A large pointy raft creating a wide furrow (Pe $= 80, P = 0.0025$). (B) Rods within a pointy raft slide past each other to catch up with the head of the raft, creating an unstable flat cluster head (Pe $= 100, P = 0.0025$). (C) A free rod moving up through an empty furrow collides with a pointy raft to create an unstable flat cluster head (Pe $= 100, P = 0.0025$). (D) Collisions with multiple free rods from the rear perturbs orientation of rods in the pointy raft to break it up into smaller rafts (Pe $= 100, P = 0.005$).

those formed at lower plasticity. These distinctive morphologies are robust and are repeatedly obtained in different ensembles at the same values of the dimensionless parameters.

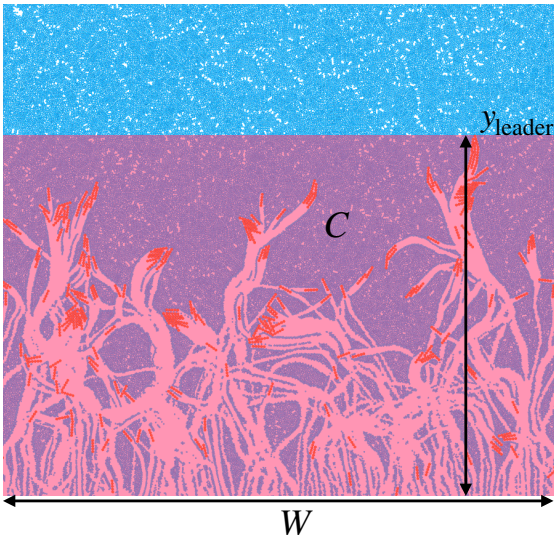


FIGURE 5.4: Definition of colonized area

## 5.2 Colonization dynamics

The area of the substrate that has been "colonized" by the advancing active rods is defined as the area behind the outermost leader rod at any instant of time. Thus, this area, $C = y_{\text{leader}}W$, where $y_{\text{leader}}$ is the $y$-coordinate of the outermost leader rod at any instant and $W$ is the width of the simulation box as shown ( Fig. 5.4).

For any given plasticity, $P$, the colonized area $C$ is observed to grow nearly linearly with time in all simulations until they are terminated (Fig. 5.5). The average colonization speed, $V_c$, is estimated by a linear least-squares fit through the $C$-vs-$t$ data. Figure 5.6 compares with $V_c$ with the speed of an *isolated* rod through the same substrate, $V_1$. These speeds are normalized by speed $V_1^0$ observed for a single rod traveling at the same Pe in a substrate of zero stiffness *i.e.* when $P = 0$. In this case, the speed of an active rod is entirely determined by the frictional resistance offered by *unbound* substrate particles. When thermal fluctuations are weak, the average speed of an isolated rod in a fluid-like medium consisting of fully plasticized substrate is

$$V_1^0 \approx \frac{F^{\text{a}}}{\gamma_{\text{s}} + \gamma_{\text{r}}} = \frac{\text{Pe}}{N_{\text{b}}^2 \left(\gamma_{\text{s}}/\gamma_{\text{r}} + 1\right)} . \tag{5.1}$$

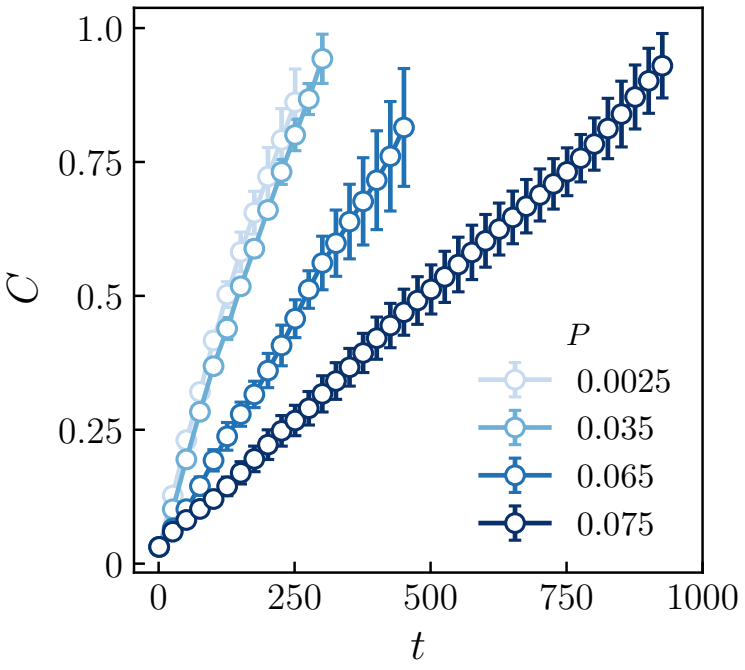


FIGURE 5.5: Growth of colonized area, $C$, at Pe = 100. The colonized area increases nearly linearly over the duration of the simulations.

It is observed in Fig. 5.6 that, at any given Pe and $P$, the colonization $V_c$ is systematically *larger* than $V_1$, the speed at which an isolated rod moves through the substrate. Further, while $V_1$ decreases nearly linearly with $P$ at a fixed Pe, $V_c$ varies non-monotonically with $P$, displaying a peak colonization rate at a non-zero value of $P$. The same behavior is obtained even with simulations in which the rods are initially oriented randomly (grey symbols in Fig. 5.6), rather than all in the $y$-direction.

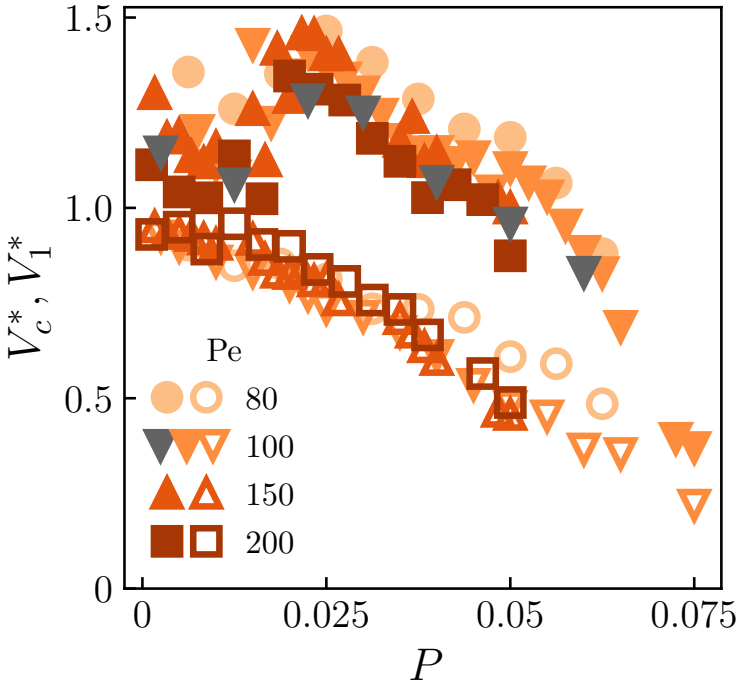


FIGURE 5.6: Comparison of normalized colonization speed, $V_c^*$ (filled symbols), and the normalized speed of isolated rods, $V_1^*$ (open symbols): the speeds are normalized by $V_1^0$, the speed of an isolated rod in a substrate of zero plasticity offering only frictional resistance. The colored symbols represent results obtained with rods initially oriented vertically, whereas the gray symbols are results with rods with random initial orientations.

These interesting differences in colonization speed from isolated-rod speed clearly arise from collective effects and appear to be related to the behavior of clusters of active rods that drive the formation of furrows. The speed gain ratio,

$$G = \frac{|V_c|}{|V_1|}, \tag{5.2}$$

is plotted in Fig. 5.7. It is observed that, for the range of Pe values explored here, increasing substrate stiffness $P$ largely enhances the gain in the colonization speed due to collective motion, relative to the motion of an isolated rod. Colonies of rods can expand much faster than the speed at which individual rods can move through the substrate. This trend of collective enhancement of colony expansion remains at values of $P$ greater than the maxima observed in Fig. 5.6. However, it appears that the collective gain $G$ reduces with increasing Pe.

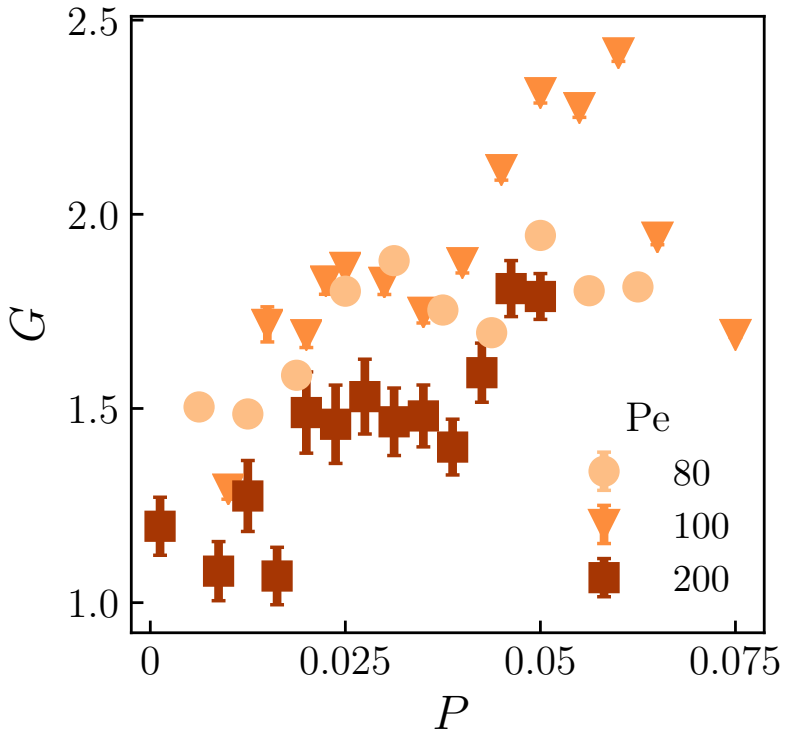


FIGURE 5.7: Variation of speed gain with Pe and plasticity.

## 5.3 Cluster dynamics

The speed gain and the morphology of the furrow network appear to be related to the relative proportion of rafts and trains. The clustering algorithm described in Chapter 3 was used to identify separate clusters of beads. The motion of individual clusters were tracked in a frontier region of depth $8\,L$ from the leading edge of the colony until they broke up or were joined by new members. Frequency distributions of quantities such as the number of rods in a cluster (*i.e.* the cluster size), average cluster speed and vertical velocity component, and orientation angle relative to the vertical axis were determined. Based on the relative configurations of the rods in a cluster, each cluster was classified as a train or a non-train raft [Fig. 5.2]. A cluster is identified as a train if the angles between neighboring pairs of rods in the cluster are *all* below 7° and if the distances between pairs of consecutive rods is approximately equal to $L$. Any cluster that is not a train was classified as a raft.

Figure 5.8 shows that, at low substrate plasticity, most clusters in the frontier

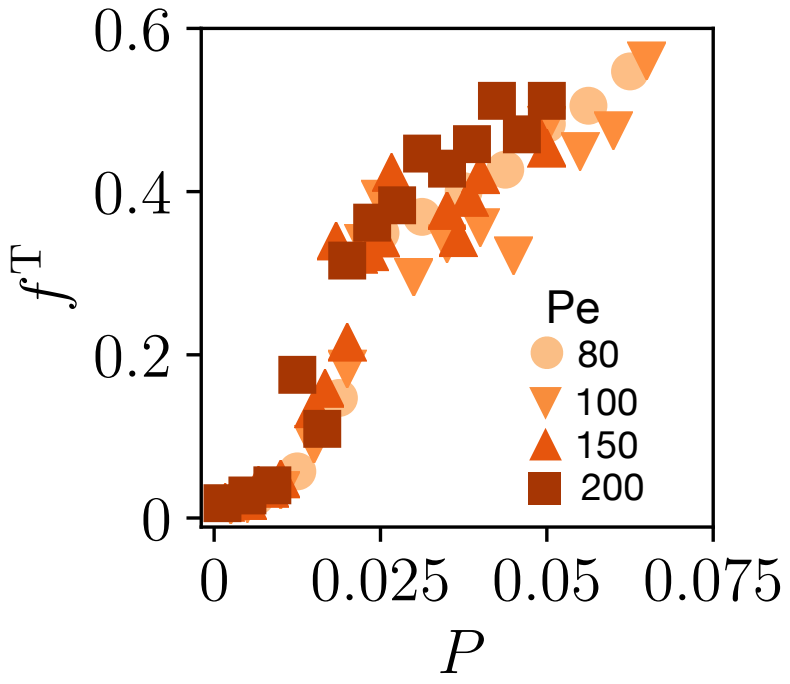


FIGURE 5.8: Fraction of train clusters, $f_T$ , out of all the clusters (the fraction of rafts, $f_T = 1 - f_R$).

region are of raft type. With increasing plasticity, trains begin to dominate. It is seen that the average speed of the train fraction of the population is always greater than that of rafts. Looking at this more closely, trains of any give cluster size consistently move faster than rafts of the same size. The speeds of either rafts or trains increases with cluster size, with isolated rods moving the slowest. This provides further evidence that the gain in colonization speed observed earlier is due to clustering [Fig. 5.9].

These observations suggest that cluster speed is determined by the frictional resistance offered by the substrate particles that are broken off and pushed by the cluster. Clusters expose less surface area per rod to the substrate. Further, there is only a single substrate typically standing in the way a whole train, and the resistance of the substrate per rod is thus smaller for a train, than for a raft.

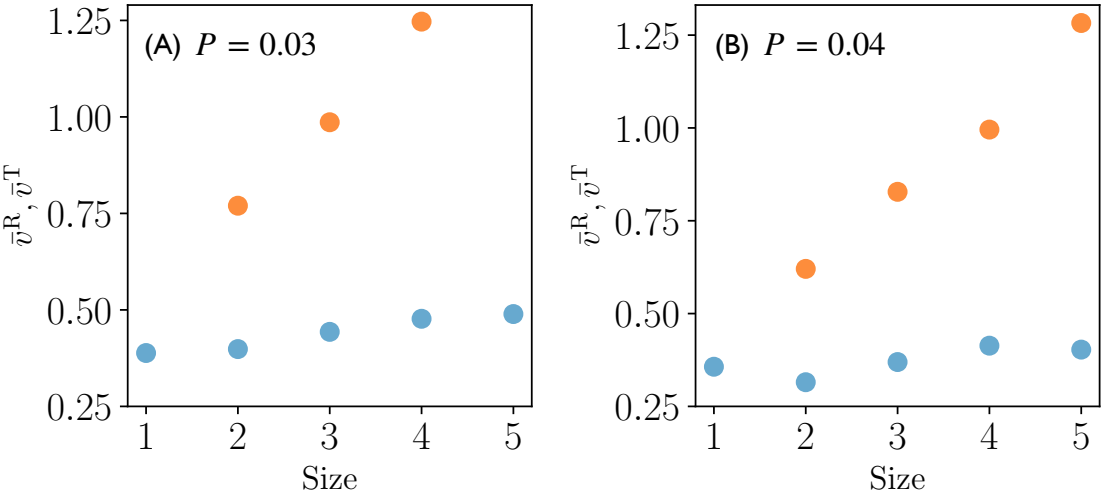


FIGURE 5.9: Speeds of trains (orange symbol) and rafts (blue symbols) of different sizes at fixed Pe = 100 and at (A) $P = 0.03$ and (B) $P = 0.04$.

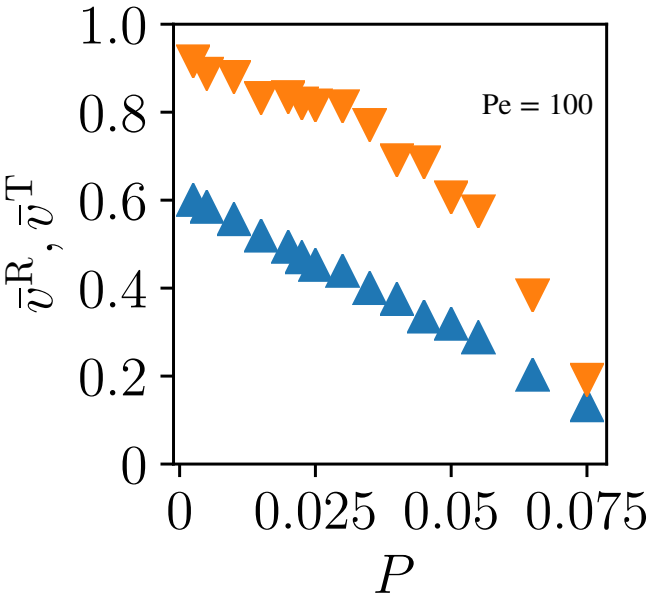


FIGURE 5.10: Average speeds of trains (orange symbol) and rafts (blue symbols) of different sizes at fixed Pe = 100.

The mean $y$-component of the velocity ($v_y$) of clusters within the frontier region is computed as $\overline{v}_y = \overline{v}_y^{\mathrm{R}}(1 - f^{\mathrm{T}}) + \overline{v}_y^{\mathrm{T}} f^{\mathrm{T}}$, where $f^{\mathrm{T}}$ is the fraction of trains in the clusters.

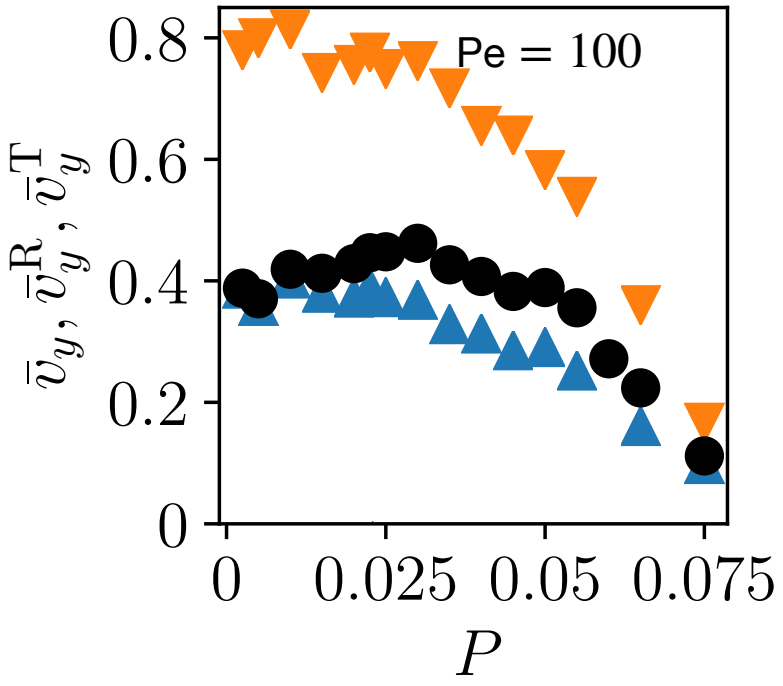


FIGURE 5.11: Averages of the $y$-component of the cluster velocity of rafts and trains, and the overall population average, for Pe = 100. Orange symbols are for the $y$-component of train clusters blue symbols are for raft clusters and black symbols are for average speed.

In Fig. 5.11 , the individual means, $\overline{v}_y^{\mathrm{R}}$ and $\overline{v}_y^{\mathrm{T}}$, decrease non-monotonically with $P$ , the overall average $\overline{v}_y$ shows a maximum around the same $P$ at which the peak colonization speed occurs in Fig. 5.6. The orientational distributions of clusters in Fig. 5.12 further show that trains in the frontier region tend to be strongly aligned in the outward direction than rafts, which have more diverse orientations relative to the outward direction. As noted previously, trains are faster since the propulsive force on train is almost entirely directed along the train axis.

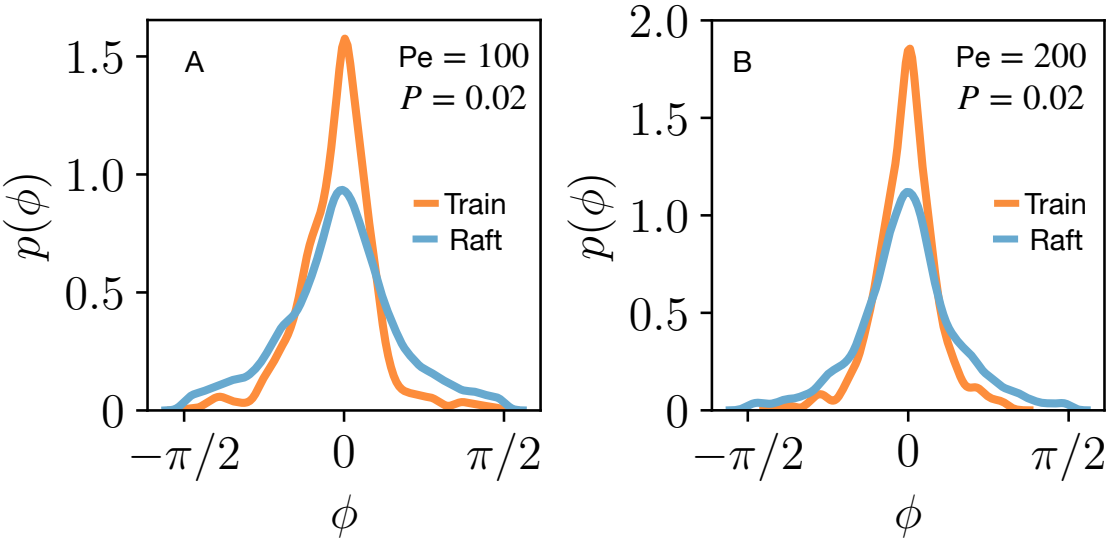


FIGURE 5.12: Distribution of rod orientation angle $p(\phi)$ with respect to the vertical axis ($y$ axis) for rafts (blue curve) and trains (orange curve) for Pe = 100 and $P = 0.02$.

These observations provides an explanation of the peak in colonization speed observed in Fig. 5.6. At lower plasticity, $P$ there is a larger number of rafts in the system, and they control the overall colonization direction. The orientation of the rafts is more disperse; therefore overall colonization rate/cluster movement towards other end is lower despite having a large number of the clusters. As substrate plasticity increases, the overall fraction of rafts decreases and trains increases (Fig. 5.8). Trains are aligned towards the other end of the box, and this enhances the colonization rate. The initial increase of the colonisation speed with $P$ observed in that Figure at small values of $P$ appears to be because of the growth in the fraction of faster-moving trains that are also more aligned along the outward direction. When the fraction of trains begins to saturate at high values of $P$ (Fig. 5.8), the decrease of their speeds with stiffness takes over begins to dominate the picture, and the overall colonization speed decreases with stiffness. Thus, the analysis of clusters shows that the peak in colonization speed is due to a competition between two effects. On the one hand, the increase in substrate plasticity slows down both rafts and trains. On the other hand, as plasticity increases, the population changes from relatively slower rafts to faster, more oriented trains.

## 5.4 Furrow network morphology

As observed in Fig. 5.1, the morphology of the furrow network depends significantly on $P$. At lower plasticity, when rafts dominate, more curved furrows are observed, whereas when trains dominate at higher plasticity, straighter furrows are obtained. Interestingly, across a wide range of Pe and $P$ values, it is observed that the furrowed area displays a non-trivial power-law growth with time (Fig. 5.13).

The dynamics of raft clusters plays an important role in determining the morphology of the furrow networks. We observe visually that arrow-head shaped "pointy" rafts are stable for longer times than broad-headed rafts. The rods at the head of pointy rafts have convergent orientations. This leads to a rectification of their propulsion forces along the longitudinal axis of the cluster. Any remaining transverse component of the net propulsion force causes pointy rafts to swerve and take curved trajectories. Large rafts can thus create wide and long furrows that serve as arterial conduits in the network. In contrast, broad-headed rafts with rods of divergent orientations quickly break up into multiple smaller rafts, which create their own furrows.

The network morphology is generated by clusters at the head of furrows ploughing through empty furrows already created previously. If a previous furrow that a cluster encounters is narrow, the cluster can plough through and continue unhindered. On the other hand, when the previous furrow has a width comparable to $L$ or greater, the cluster quickly breaks up, as the small orientational noise present causes rods at the edge of the cluster to escape the cluster easily before the cluster passes through the furrow. These free rods experience lower (bare) friction in empty furrows and move through the network very quickly. They sometimes encounter a furrow wall head-on and push through to create thin furrows on their own. However, more typically, on colliding with furrow walls, they reorient along the furrow axis and travel along it to catch up with a raft at the head of a newly forming furrow (or exit the periodic boundary at the back of their colony to enter the colony on the other side and catch up with a cluster at a furrow head on that side). A free rod approaching a raft or a train from behind cannot, however,

overtake that cluster. They must either collide with the cluster from behind, or squeeze through the side of the furrow to align with other rods at the head. These collisions of pointy clusters with free rods, or with other clusters, destabilize them by disturbing the alignment of the rods in the clusters, leading to their break-up.

This power-law scaling suggests that a self-similar process operates in the furrow formation and that network structures may have a fractal character.

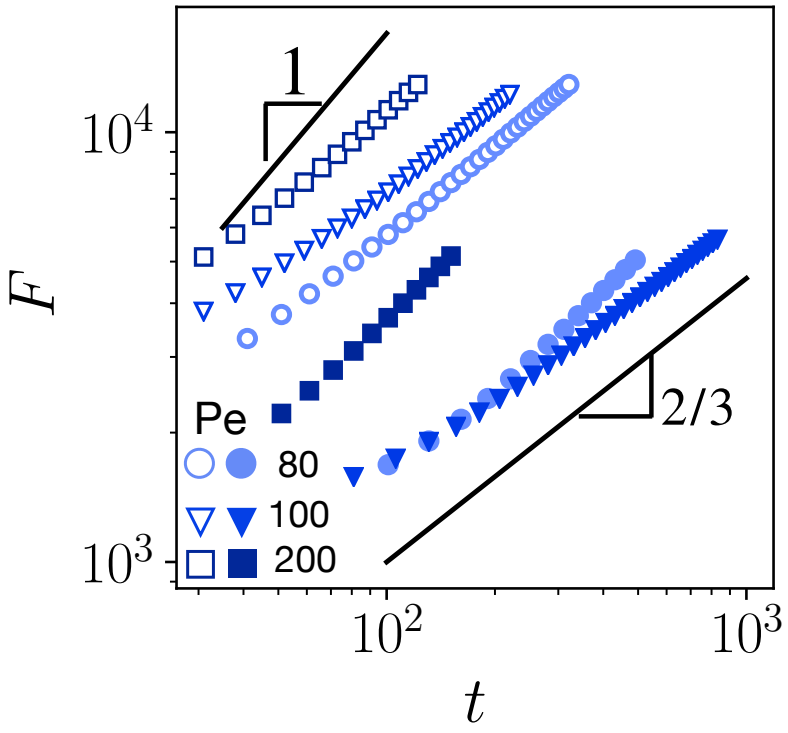


FIGURE 5.13: The growth of total furrowed area with time, for different $Pe$ and $P$. furrowed area growth shows a non-trivial power law behavior. The symbols are as follows: $P = 0.00125$( □) and $P = 0.05$(■) at Pe = 200; $P = 0.0025$(▽) and $P = 0.075$ (▼) at Pe = 100; $P = 0.00625$ (◯) and $P = 0.0625$(●) at Pe = 80.

The box-counting method is commonly used to calculate the fractal dimension of self-similar structures [73, 110, 127]. The algorithm is to find the number, $N_{\text{box}}$, of boxes of side $\epsilon_b$ that cover the object. Then, the fractal dimension,

$$D = \lim_{\epsilon_b \to +0} \frac{\log N\text{box}}{\log 1/\epsilon_b}. \tag{5.3}$$

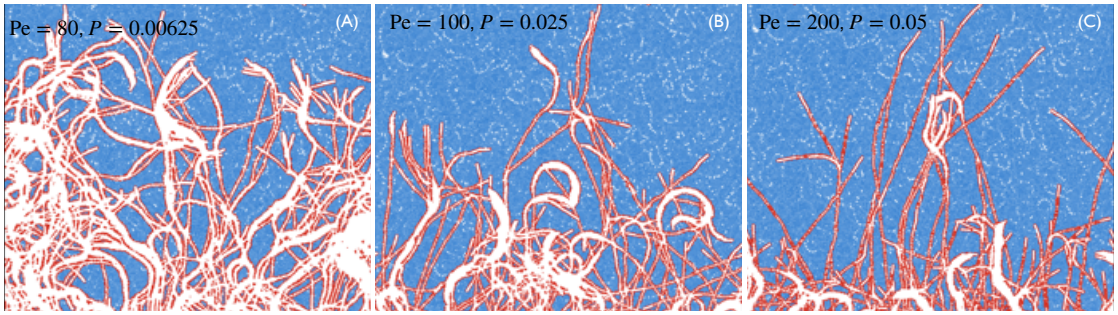


FIGURE 5.14: Furrow networks at different values of $P$ and Pe.

A code was developed to apply the box-counting algorithm on binarized images of the furrow network [41]. Figure 5.15 shows that the fractal dimension of the furrow networks lies between 1 and 2, as is expected for partially space-filling 1D trajectories embedded in a 2D space.

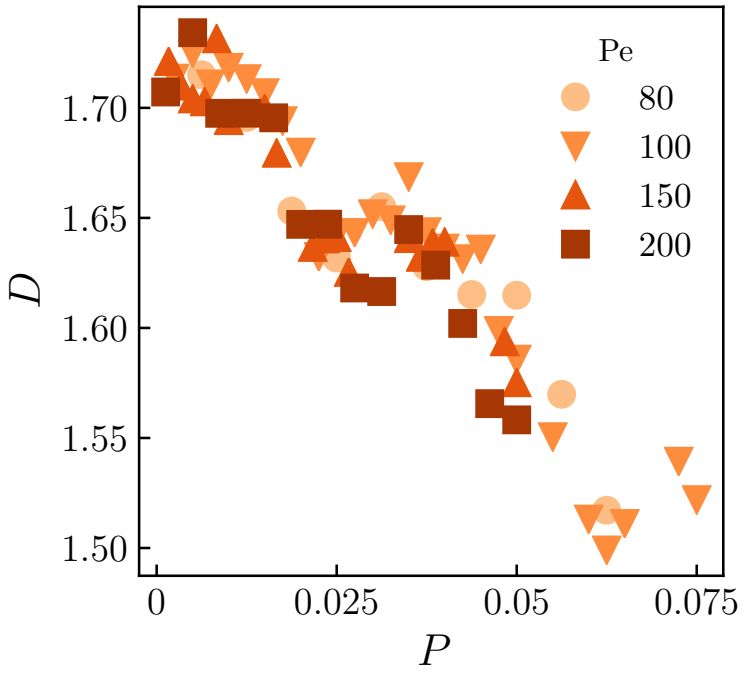


FIGURE 5.15: Box-counting fractal dimension as a function of plasticity number $P$, measured at termination of simulation.

The dimension $D$ decreases systematically from round 1.75 for $P = 0$, to around 1.5 for high values of $P$. Data for different Pe values appear to nearly collapse onto a master curve, showing that the fractal morphology depends only weakly depend on Pe. These results are consistent with the observations that, at low $P$

values, the network is the result of repeated intersections the swerving trajectories of raft clusters, whereas at high $P$ values, the network is dominated by the nearly linear furrows created by trains.

To understand the origin of the fractal dimension better, we analyzed the colonization process using a chemical reaction analogy.

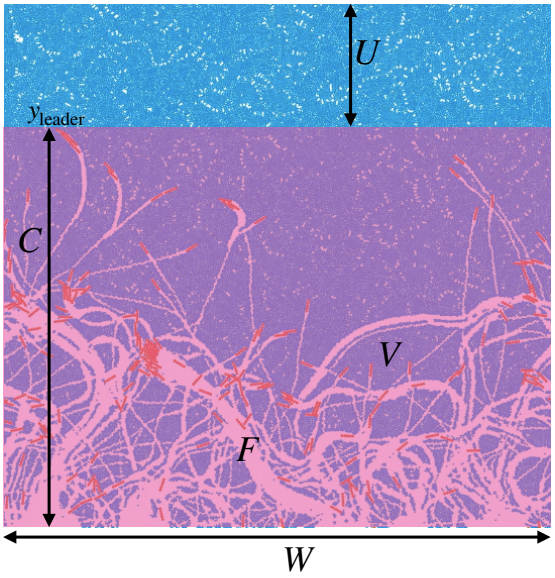


FIGURE 5.16: Formation of different regions during the invasion of the substrate by SPRs: Colonized area, $C$ the area behind the $y_{\text{leader}}$ (shadowed region), Uncolonized substrate, $U$, the space occupied by substrate particle beyond the position of $y_{\text{leader}}$ (blue region), Furrowed area, $F$, the area of the furrow in the box (white regions) and void, $V$ area of the un-visited substrate within the colonized region.

The"reactor" space is initially full of the uncolonized substrate. As explained earlier, all the space behind the outermost leader rod is treated as having been colonized (Fig. 5.4). The colonized space consists of the furrows and substrate pockets left behind the leading edge, which are referred to as voids. This simple picture allows us to write rate equations for the formation of different regions. In the description below, the chemical "species" involved in the reactor are the uncolonized substrate, U; the colonized area, C; the furrows created, F; the void regions, V. The reactions are "catalyzed" by the active rods, R. These rods can be further split into the few leader rods, $R_l$ that advance the colony front line, and all the other rods in the interior, $R_i$. The rods do not fill the furrows completely,

and a significant part of the furrows are empty. These empty furrow regions are denoted as E. The areas occupied by these species are denoted as $U$, $C$, $F$, $V$, $R$, $R_\mathrm{l}$, $R_\mathrm{i}$, and $E$, respectively.

We firstly assume that the formation of the colonised area is a zero-order process. This process is abstractly represented by the reaction:

$$\mathrm{U} \xrightarrow{\mathrm{R}\mathit{l}} \mathrm{C}\,, \tag{R1}$$

The total colonized area,

$$C = F + V$$

*Any* rod within the furrows can potentially furrow into void regions. All the results presented thus far are for the case where substrate particles are permanently plasticized as they are displaced by more than the length scale for plastic deformation, $\ell$. More generally, however, if this length scale is large, it is possible that substrate particles remain elastically deformed as rods move past them, and if furrows remain empty without any rods present, those empty regions can convert back into void pockets as the substrate particles return to their equilibrium positions. Thus, the formation of furrows and voids is represented as a pair of reactions:

$$\mathrm{V} \underset{\mathrm{E}}{\overset{\mathrm{R}}{\rightleftharpoons}} \mathrm{F}\,, \tag{R2}$$

where the area of empty furrow regions, $E = F - R$. The forward reaction is catalyzed by all the rods in the system, where $R = R_\mathrm{l} + R_\mathrm{i}$. The forward and reverse reactions are referred to as *infilling* and *ex-filling*. Since the reverse ex-filling reaction only occurs in empty regions of the furrow, it is considered as being catalyzed by E.

The expansion of the colony boundary line is achieved by vanguard clusters. To estimate the rate of colonization, we note that, although there can be multiple leader rods in a vanguard cluster, the colony expansion can happen with just

one leader rod at the outermost point. Therefore, the colonization rate is not expected to depend strongly on the number of leader rods. The orientation of a rod in a vanguard cluster with respect to the vertical axis is a random variable in the range of $[-\pi/2, \pi/2]$. We ignore the fact that this distribution is peaked, and instead, for the crude model here, assume that the distribution is uniform with a probability density, $p(\theta) = 1/\pi$. Then, if $v_1$ is the speed of an average vanguard cluster size, then, the $y$- component of a vanguard cluster velocity is on average $\int_{-\pi/2}^{\pi/2} v_1 \cos\theta p(\theta)\, d\theta = 2v_1/\pi$. The average rate of area expansion is thus proportional to $(2W/\pi)\, v_1$. For any other orientational distribution, we expect that the rate to be proportional to $W\, v_1$.

Furrowed regions and voids are part of the colonised area. The rate of the infilling reaction can be expected to be proportional to the total length $\mathcal{L}$, of the void-furrow interface (VFI) within the colony. The VFI is the perimeter of the furrowed area. As the results show, a self-similar furrow structure exists. Then, $\mathcal{L} \sim F^{D/2}$. In addition, the rate at which infilling at the VFI occurs will depend on whether or not there are rods at the interface. If the furrows are largely empty, then the rate will depend on encounters of rods moving within the furrow with VFI. These encounters are largely random when the furrows are mostly empty. Then, the infilling rate will be proportional to $R/F$. When the furrows are full, then the rate will be independent of $R$. However, then $R = F$ or $R/F = 1$. Combining these, the infilling rate is expected to be proportional to,

$$\mathcal{L}\,(R/F) \;\sim\; F^{(D/2)}\,(R/F) \;=\; F^{\nu}\,R\,. \tag{5.4}$$

Here, $\nu = D/2 - 1$. Since the fractal dimension, $1 < D < 2$, we expect $-1/2 < \nu < 0$. As mentioned earlier, the exfilling rate is expected to be proportional to $E = F - R$.

Conservation equations for different regions in the simulation box can now be written as follows:

$$\frac{dC}{dt} = k_1 W v_1 ; \tag{5.5}$$

$$\frac{dF}{dt} = k_{2,f} F^{\nu} R - k_{2,r} (F - R) , \tag{5.6}$$

where $k_1$, $k_{2,f}$ and $k_{2,r}$ are constants that may depend on Pe or $P$, but are independent of the reacting species themselves. Note that $U+F+V$, the instantaneous total reactor area, is a constant. Further, $V = C-F$ until $V = 0$. In rate equation 5.6, first term corresponds to the furrow formation whereas second term is for the relaxation of furrowed area due to empty furrow.

When the plasticization length is small, elastic relaxation is unimportant *i.e.* $k_{2,r} = 0$, and:

$$\frac{dF}{dt} = k_{2,f} F^{\nu} R . \tag{5.7}$$

In this case, the rate equation, Eqn. 5.7, can be integrated:

$$\int_{F_0}^{F} \frac{dF}{F^{\nu}} = \int_0^t k_{2,f} R \, dt ,$$

giving, in the case where the number of rods on average is a constant at $R_0$,

$$F^{1-\nu} - F_0^{1-\nu} = (1-\nu) R_0 k_{2,f} t , \tag{5.8}$$

where $F_0$ is the initial furrow area occupied by the rods at $t = 0$. When $F \gg F_0$,

$$F^{1-\nu} \approx (1-\nu) R_0 k_{2,f} t ,$$

giving,

$$F = ((1-\nu) R_0 k_{2,f})^{1/(1-\nu)} t^{1/(1-\nu)} \tag{5.9}$$

The equation above shows that $F \sim t^{1/(1-\nu)}$. Since the fractal dimension $1 < D <$

2, the slope of $F$ against $t$ curve pn a log-log plot is in the range $2/3 < 1/(1-\nu) < 1$. This is in good agreement with the results in Figs. 5.13 and 5.15.

## 5.5 Effect of injection flux on furrow formation

In this section, we explore the effect of a continuous injection of new rods at the bottom boundary. If the injection flux, $j \neq 0$, the area occupied by the rods at any time $t$,

$$R = R_0 + j a_R W t \,, \tag{5.10}$$

where, $R_0$ is the initial number of rods, and $a_R$ is area of one rod. A sparse furrow network is only possible if the colony expansion rate outstrips the rate at with rod area increases due to injection flux *i.e.* $dR/dt < dC/dt$. This implies that sparse networks are formed when

$$j < k_1 v_1 / a_R \,. \tag{5.11}$$

In this case, we have a simple analytical solution of Eqn. (5.6) for plastic substrates:

$$F^{1-\nu} - F_0^{1-\nu} = (1-\nu)\, k_{2,\,f} \left( R_0\, t + j a_R W \frac{t^2}{2} \right) . \tag{5.12}$$

Once the rod population grows to be significantly larger than the number of rods *i.e.* $R \gg R_0$, and $F \gg F_0$, we again have a power-law regime, where $F \sim t^{2/(1-\nu)}$. The ratio of furrowed area to the colonized area,

$$\frac{F}{C} \sim \frac{t^{2/(1-\nu)}}{t} \sim t^{(1+\nu)/(1-\nu)} \,. \tag{5.13}$$

Thus, this ratio can *increase* as $t^1$ when $\nu = 0$, to $t^{1/3}$ when $\nu = -1/2$.

In a real system consisting of cells, the rods absorb nutrients from the medium to divide and grow. Although this situation is difficult to implement in simulations,

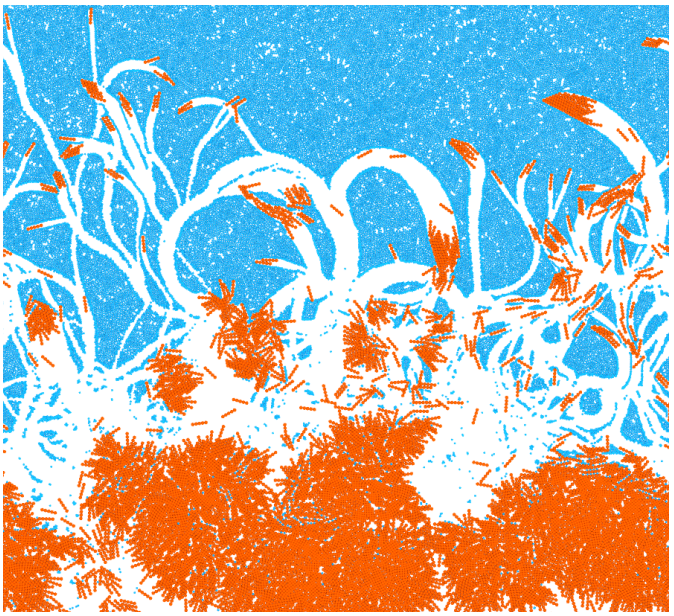

FIGURE 5.17: Colony morphology at Pe = 100 and $P = 0.0025$, with linear rod influx in the system: rods are introduced at a linear rate of 25 rods per $\tau_b$.

the result above shows that an exponential growth in the number of cells will quickly fill up the furrow network, which only grows as a power-law through the processes described above. The colony edge moving at a constant rate cannot outpace the internal creation of new rods. When rods become tightly packed in the furrows, the physical pressure in the furrow grows rapidly due to confinement. Cells then breach furrow walls to rapidly fill into the void pockets and the integrity of the furrow network cannot be maintained.

To test this idea, simulations were performed using a high, but constant (rather than exponential), injection rate of rods at the bottom boundary. It was observed that, even with this linear growth in the number of cells, the network structure created by the vanguard cells is quickly obliterated by the motion of the growing number of new cells in the colony interior (Fig. 5.17).

Therefore, in colonies of bacteria or other species growing exponentially with time, the colony assumes a uniformly dense morphology with no internal substrate voids. The front is densely-packed front and is driven by the pressure in the colony interior. While this may lead to fingering patterns at the front [90, 45], an extensive furrow network cannot be sustained under these conditions.

However, the experiments with *P. aeruginosa* show 2D furrow networks that extend across distances around two orders of magnitude greater than the length of a single cell [47]. This suggests that, in cases where extended networks are sustainably formed, there could be mechanisms that limit the growth rate of cells to not exceed the rate at which the furrows are created. Frequent spontaneous explosive cell lysis events have been observed in the furrow networks of *P. aeruginosa* which could slow down the growth rate within the network sufficiently to keep the furrow network. Interestingly, this lysis releases DNA and other polysaccharides into the furrows [119]. Other experiments have discovered lattices of extracellular DNA and polysaccharides in biofilms of pathogenic bacteria in semi-solid biofluids such as sputum and otorrhea (ear-discharge fluid), and that such lattices are essential for the mechanical integrity of those biofilms [33]. It is possible that these lattices are laid down in the furrow network originally formed as the colonies expand through the semi-solid substrates. In other words, furrow networks could have a functional role in bacterial colonies. Beyond bacterial collectives, substrate interactions also occur during cancer metastasis in which groups of migrating cancer cells exploit the plasticity of the extracellular matrix (ECM) and actively remodel the ECM fibers to enhance their migration speed [72, 136, 134].

## 5.6 Effect of rod aspect ratio

Besides Pe and $P$, the rod aspect ratio, $\mathcal{A} = N_{\mathrm{b}}$, can be expected to determine colonization dynamics and network morphology. The results presented thus far were obtained with $\mathcal{A} = 5$. This section compares those results against results obtained with $\mathcal{A} = 2$ and 3.

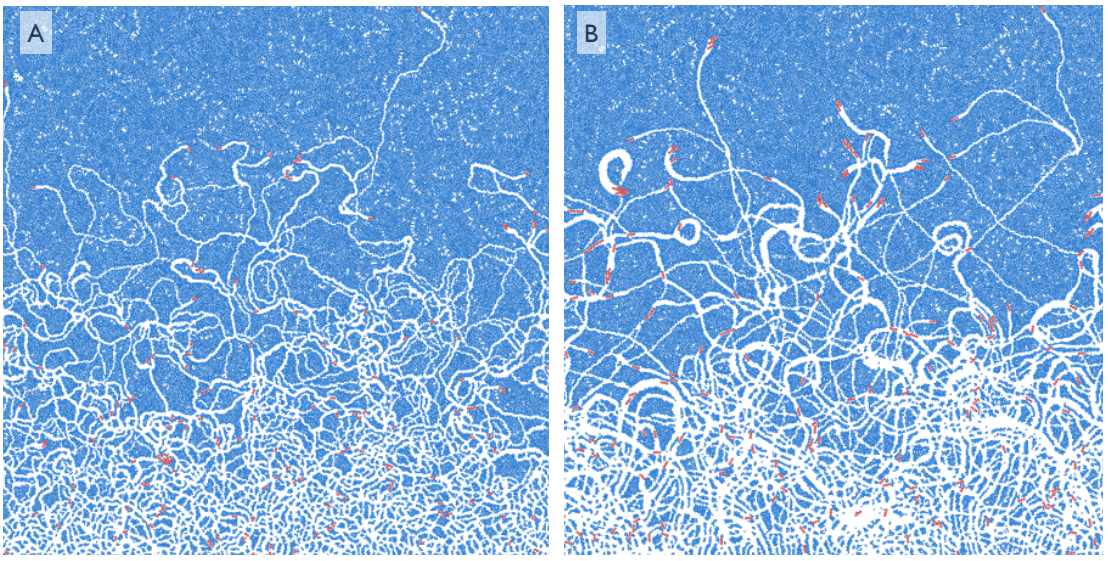


FIGURE 5.18: Colony morphology for rods with different aspect ratio (A) 2, and (B) 3 at Pe = 100.

At an aspect ratio of $\mathcal{A} = 2$, initially rods begin to move as a group. As simulation progresses initial clusters break up. Thereafter, clusters form rarely and rapidly break up. The random motion of individual rods leads to a dense network of thin furrows whose width are comparable to the diameter of the rods (Fig. 5.18 A).

Clustering becomes more prominent as aspect ratio is increased. With ratio $\mathcal{A} = 3$, dynamic clusters similar to the raft clusters at $\mathcal{A} = 5$ in the previous sections are observed. As before, these rafts, due to their asymmetrical shape, make loops that intersect with empty furrows in their neighborhood and break up. The constituent rods either begin to explore the uncolonized substrate or quickly move through the furrow to join a leading cluster. This process leads to wider furrows than obtained with $\mathcal{A} = 2$ (Fig. 5.18 B). These results appear to be consistent with the results of Peruani et al. [100] for free rods, that, at any given Pe, large clusters become rarer as the aspect ratio decrease.

Fig. 5.19 shows the comparison of colonization rates for the three values of $\mathcal{A}$. Also shown are the speeds of isolated rods. It is observed that, for rods of the lowest aspect ratio of 2, The colonization rate, $V_c$, is lower than the isolated-rod *speed* through the substrate, $V_1$. As noted above, at this aspect ratio, there is no

significant clustering. The colonization rate, $V_c$, in this case is thus the average $y$-component of the velocity of a single rod whose orientation is broadly distributed about the vertical direction. Therefore, it is to be expected that $V_c < v_1$ in this case.

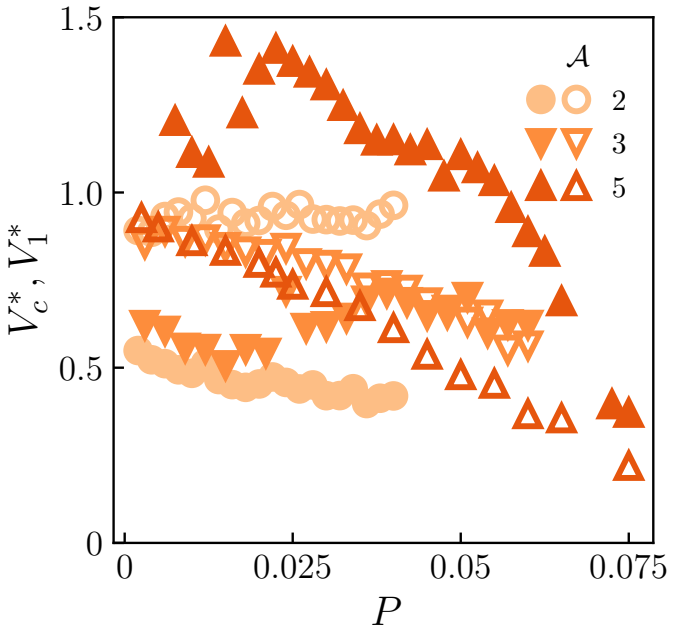


FIGURE 5.19: Effect of aspect ratio $\mathcal{A}$ on normalized colonization speed, $V_c^*$ (filled symbols), and the normalized speed of isolated rods, $V_1^*$ (open symbols

With an aspect ratio of 3, on the other hand, it is observed in Fig. 5.19 that, at low values of $P$, $V_c < V_1$, whereas, at higher values of $P$, $V_c \approx V_1$. This is due to the fact that, although raft do form more often and for longer at this aspect ratio at low $P$ values, the gain in speed due to clustering is not enough to counter the effect on $V_c$ due to random orientations relative to the vertical direction. However, at higher values of $P$, where trains begin to form, the higher speed of trains compensates for the effect of random orientation, and $V_c \approx V_1$. With an aspect ratio of 5, the gain in speed over isolated rods due to clustering as either rafts or trains becomes the dominant effect and overcomes the effect of random orientations at the leading edge.

Figure 5.20 shows the time evolution of the furrowed area for the rods having aspect ratios 2 and 3. As before, power-law growths are obtained. In general,

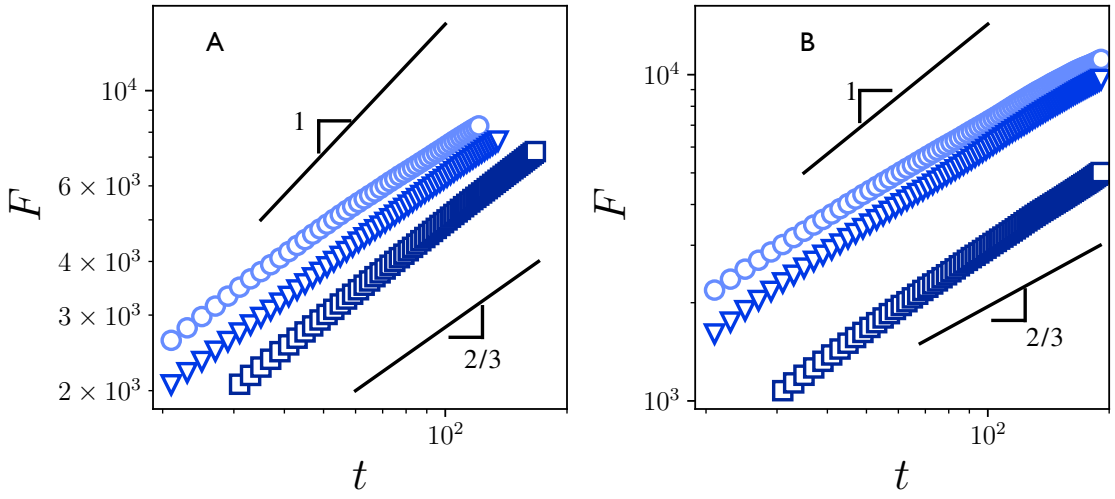


FIGURE 5.20: Furrowed area for rods with different aspect ratio, $\mathcal{A}$ (A) 2, and (B)3.

at any given $P$ and Pe lower rod aspect ratios lead to more space-filling furrow networks with a higher value of the fractal dimension (Fig. 5.21).

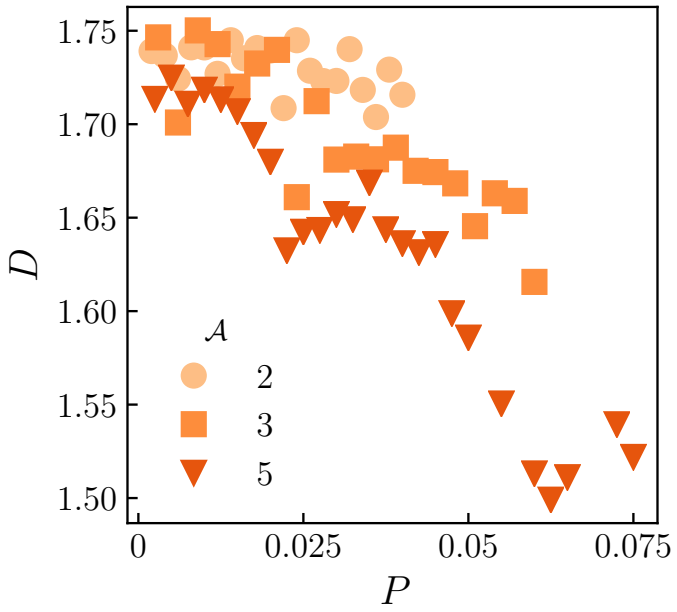


FIGURE 5.21: Box-counting fractal dimension as a function of plasticity number $P$, measured at termination of simulation.

## 5.7 Summary

Our particle-based simulations show that active self-propelled rods can create highly ramified networks of furrows when pushing through a soft substrate. The

results here indicate that when rods are of sufficiently large aspect ratios ($\mathcal{A} \gtrsim 5$), rods form large rafts or train-like clusters which can move faster together through the substrate than isolated rods. As a result, the edge of the colony advances at a considerably faster rate than what would have been expected from the speed of an isolated rod in the substrate. The colonization rate varies a non-monotonically with substrate plasticity. This is shown to be due to a shift in the composition of the clusters in the population from being largely pointy rafts to forming long trains. The colonization rate is highest around the value of the plasticity at which the shift in cluster composition is observed. The morphology of this furrow network, as measured by its fractal dimension, depends strongly on the plasticity of the substrate which is characterized by the dimensionless ratio, $P = \frac{\kappa l_{\max}}{F^{\mathrm{a}}}$. The fractal nature of the network appears to be due to the self-similar nature of the processes of motility-induced clustering, the trajectories those clusters form, and the break-up of clusters once their trajectories intersect pre-existing furrows in their neighborhood. A simple model of furrow formation shows how the fractal dimension of the furrow network is related to the power-law growth of the furrowed area with time. The model and simulations show that, in systems of dividing cells, these fascinating furrow networks are only expected to form when the growth-rate of the cells matches the power-law growth of the furrow network with time. An exponential growth of cells within the furrow network is expected to quickly infill the substrate voids in the network and lead to a more uniform and dense colony with no substrate pockets.

# Chapter 6

# Conclusion and outlook

## 6.1 Conclusion

The simulation of self-propelled particles (SPRs) has proven to be useful in understanding the basic aspects of collective behavior of many active systems such as bacterial suspension [103, 130]. The objective of this work was to explore the role of the roughness of SPRs and the plasticity of the surrounding substrate on the collective behavior of SPRs. A minimal model to study the behavior of SPRs was first developed. In this model a self-propelled rod is modeled as a rigid array of spherical beads. Additionally, the deformable substrate was represented by a 2D array of particles anchored using spring potential.

The collective motion in a dense system of SPRs in the absence of any substrate was first investigated by varying the rod surface roughness. In most of the previous studies, systems such as bacterial suspension are modeled either as a chain of spherical beads, or as smooth spherocylinders, and the effect of surface smoothness was ignored. It was previously suggested that the shape induced effects could play an important role in microbial processes such as collective sperm swimming or algal colony formation [66]. Our simulations indeed show that even subtle changes in surface topology of rods lead to profound changes in the collective behavior of active system. A system of rough SPRs (rigid linear arrays of touching

beads) exhibits the formation of a bubbly phase. It is also observed that the spatial correlation length in rod velocity varies non-monotonically with the activity of SPRs. For a given smoothness, the correlation length first increases with an increase in Pe, and then decreases. This behavior could be explained based on the distinct clustering mechanism of smooth and rough rods.

Secondly, in the presence of a compliant substrate, we report that the colonization of substrate and furrow formation in substrate depends on the clustering of SPRs. Our simulations further show that the colonization of the substrate depends on the position of the leading rod, the speed of the SPRs in the front, and the stiffness of the substrate. The deformable substrate allows rods to displace substrate particles beyond their equilibrium positions. This displacement of substrate particles results in the emergence of permanent furrow networks. The stability of these furrow networks can be controlled by tuning the stiffness of the substrate. To analyze the colonization of the substrate, an algorithm to track the SPRs in the moving front of the system was developed. This allowed us to report the statistics of clusters in the moving front. The geometry of clusters in the moving front of the colony was categorized using the *train-raft* algorithm. We further tracked the stability (the time until clusters do not change their member rods) of these clusters and calculated the average speed of the moving front. These results collectively show that the clustering of SPRs is an important factor for understanding the morphodynamics of surrounding substrate colonization. It is thus observed that the clustering of SPRs enables faster colonization of substrate than it would have been due to an isolated rod. The speed of colonization depends non-monotonically on the stiffness of substrate with a distinct maximum at a non-zero value of substrate stiffness. The observed non-monotonic nature of colonization speed depends on the morphology of the clusters. It is further observed that the morphology of clusters in the system depends on the substrate stiffness. The observed change in cluster morphology results in the formation of a complex furrow network. The furrow networks created by the SPRs have a fractal-like structure whose dimension varies systematically with substrate stiffness but is less sensitive to particle activity.

This work contributes towards explaining the colonization of a deformable substrate by SPRs. In these simulations, we track the behavior of motile rods advancing over a moderate distance. This provides useful insights into the mechanism behind the formation of furrow networks in large colonies of cells moving through soft matrices. The raft-like clusters observed in our simulations are similar to furrow-forming cell rafts observed in experiments with *P. aeruginosa* and *M. xanthus* cultured under confinement on semi-solid agar [47, 48]. Our results further demonstrate that even when motile rods are initially arranged in a neat row, the line of rods never expands out into a substrate of low or moderate stiffness as a uniform front. Instead, initial collisions between the rods trigger cluster formation and thus lead to the formation of a distinct furrow network. This behavior is likely to occur in three dimensions as well.

Clustering is widely observed in active systems, and the underlying mechanisms have just begun to be understood [21]. Even in the presence of a simple dissipative environment and when orientational effects are neglected, density perturbations grow when the speed of active particles decreases with density. This decrease in speed is due to the result of interactions such as steric effects. In systems consisting of a large number of particles, this leads to motility-induced phase separation (MIPS) and the formation of dense liquid-like and rarefied vapor-like phases. However, in our simulations, the number of rods is relatively small, and the system involves a coupling of density and orientation effects. Also, in this system, the interactions with an evolving plastic boundary are complex. Therefore, it is difficult to draw direct parallels with previous studies on MIPS. However, the formation of the vanguard raft cluster could be due to a change in local density. This density coarsening occurs as a result of the trapping and confinement of rods at the concave-shaped furrow head. When clusters merge with an empty furrow, clusters break up into smaller rods. This break-up of clusters that penetrate empty furrows can be regarded as an “evaporation” whereby the local density around the cluster drops when it encounters a vacant furrow. These observations suggest the occurrence of MIPS in the system.

The above results demonstrate the underlying mechanisms in the frontier region of expanding colonies. It is expected that the structure and dynamics of furrow formation would change considerably if the simulation box height $H$ is made very large compared to the rod length. As the area of the furrow network, $F$ increases in simulations with a fixed number of rods, the number density of the rods in the furrows will drop to low enough values at which no clusters form at the front. However, this regime is not of interest in understanding real colonies where there is a continuous supply of new cells from the colony interior. In real colonies, cells in the interior region follow the network structures generated by the clusters in the frontier region. Cells are also observed to actively grow and divide as they move through the furrow network. However, if the number of cells grows exponentially in time while the area of the furrow network produced by the vanguard cells only grows as a power-law, the cells will rapidly fill into the void substrate pockets, and the integrity of the furrow network cannot be maintained. Thus, exponential growth can be expected to lead to a dense substrate-free colony. We tested this in our 2D simulations by introducing new rods at a constant rate at the boundary of the simulation box, at arbitrary orientations. Since the power-law exponent for the growth of $F$ is less than unity in 2D simulations, we observed that, even with this linear growth in the number of cells, the network structure created by the vanguard cells is quickly obliterated by the motion of the growing number of new cells in the colony interior. The colony edge then advances by bulk motion of a densely-packed front that is driven by the pressure in the colony interior. While this may lead to fingering patterns at the front [90, 45], an extensive furrow network cannot be sustained.

## 6.2 Future directions

The work proposed in this thesis opens up many avenues of future research. The developed substrate model in this thesis can be used in many biological systems. The model proposed in this work can be augmented as :

1. In this work, the anisotropic self-propelled particle is modeled as a rigid array of beads. However, it has been observed that many bacterial systems show flexible structures in response to external force [5]. Understanding the collective motion of such flexible filament could mimic this behavior.

2. As in many biological systems, it has been observed that cells constantly go through the birth and death cycle [3, 32, 112]. Implementing such behavior will make the simulation model more realistic.

3. Many biological systems communicate through the means of chemical stigmergy by secreting extracellular polymeric substances (EPS)[27, 95, 47, 43, 44]. Secretion of EPS in the system can be used to implement marker-based stigmergy. An interplay between the secretion and degradation rate of EPS is expected to play an important role in pattern formation in bacterial colonies.

4. The present work reports the role of surface roughness as one of the key parameters in the collective motion of SPRs. It might be worth repeating the studies of furrow formation with smoother rods. It is expected that smoother rods in the presence of substrate are likely to change the clustering behavior.

One limitation of the present model is that it only explores the behavior of early-stage colonies. To study the behavior at a very large time-scale (behavior of matured colonies), one has to make the simulation box quite large compared to the cell length. For such types of colonies, the system size becomes too large and computationally intensive. One possible solution here is to formulate a continuum model for such a system.

# Bibliography


[1] Abdelmohsen, L. K., Peng, F., Tu, Y. and Wilson, D. A. [2014], 'Micro-and nano-motors for biomedical applications', *J. Mater. Chem. B* **2**(17), 2395–2408.

[2] Abkenar, M., Marx, K., Auth, T. and Gompper, G. [2013], 'Collective behavior of penetrable self-propelled rods in two dimensions', *Phys. Rev. E* **88**(6), 062314.

[3] Amir, A. [2014], 'Cell size regulation in bacteria', *Phys. Rev. Lett.* **112**(20), 208102.

[4] Attygalle, A. B. and Morgan, E. D. [1985], Ant trail pheromones, *in* 'Advances in insect physiology', Vol. 18, Elsevier, pp. 1–30.

[5] Balagam, R. and Igoshin, O. A. [2015], 'Mechanism for collective cell alignment in myxococcus xanthus bacteria', *PLoS Comput. Biol.* **11**(8), e1004474.

[6] Ballerini, M., Cabibbo, N., Candelier, R., Cavagna, A., Cisbani, E., Giardina, I., Lecomte, V., Orlandi, A., Parisi, G., Procaccini, A. et al. [2008], 'Interaction ruling animal collective behavior depends on topological rather than metric distance: Evidence from a field study', *P. Natl. Acad. Sci.* **105**(4), 1232–1237.

[7] Bär, M., Großmann, R., Heidenreich, S. and Peruani, F. [2020], 'Self-propelled rods: insights and perspectives for active matter', *Annu. Rev. Condens. Matter Phys.* **11**, 441–466.

[8] Baskaran, A. and Marchetti, M. C. [2008], 'Hydrodynamics of self-propelled hard rods', *Phys. Rev. E* **77**(1), 011920.

[9] Bechinger, C., Di Leonardo, R., Löwen, H., Reichhardt, C., Volpe, G. and Volpe, G. [2016], 'Active particles in complex and crowded environments', *Rev. Mod. Phys.* **88**(4), 045006.

[10] Berleman, J. E. and Kirby, J. R. [2009], 'Deciphering the hunting strategy of a bacterial wolfpack', *FEMS Microbiol. Rev.* **33**(5), 942–957.

[11] Be'er, A., Ilkanaiv, B., Gross, R., Kearns, D. B., Heidenreich, S., Bär, M. and Ariel, G. [2020], 'A phase diagram for bacterial swarming', *Commun. Phys.* **3**(1), 1–8.

[12] Bonabeau, E., Theraulaz, G., Deneubourg, J.-L., Aron, S. and Camazine, S. [1997], 'Self-organization in social insects', *Trends Ecol. Evol. & evolution* **12**(5), 188–193.

[13] Bott, M. C., Winterhalter, F., Marechal, M., Sharma, A., Brader, J. M. and Wittmann, R. [2018], 'Isotropic-nematic transition of self-propelled rods in three dimensions', *Phys. Rev. E* **98**(1), 012601.

[14] Breidenstein, E. B., de la Fuente-Núñez, C. and Hancock, R. E. [2011], 'Pseudomonas aeruginosa: all roads lead to resistance', *Trends Microbiol.* **19**(8), 419–426.

[15] Burrows, L. [2012], 'Pseudomonas aeruginosa twitching motility: Type iv pili in action', *Annu. Rev. Microbiol.* **66**, 493–520.

[16] Cabeen, M. T. and Jacobs-Wagner, C. [2005], 'Bacterial cell shape', *Nat. Rev. Microbiol.* **3**(8), 601.

[17] Caporusso, C. B., Digregorio, P., Levis, D., Cugliandolo, L. F. and Gonnella, G. [2020], 'Motility-induced microphase and macrophase separation in a two-dimensional active brownian particle system', *Phys. Rev. Lett.* **125**(17), 178004.

[18] Caprini, L. and Marconi, U. M. B. [2021], 'Spatial velocity correlations in inertial systems of active brownian particles', *Soft Matter* **17**(15), 4109–4121.

[19] Caprini, L., Marconi, U. M. B. and Puglisi, A. [2020], 'Spontaneous velocity alignment in motility-induced phase separation', *Phys. Rev. Lett.* **124**(7), 078001.

[20] Cates, M. E. [2012], 'Diffusive transport without detailed balance in motile bacteria: does microbiology need statistical physics?', *Rep. Prog. Phys.* **75**(4), 042601.

[21] Cates, M. E. and Tailleur, J. [2015], 'Motility-induced phase separation', *Annu. Rev. Condens. Matter Phys.* **6**(1), 219–244.

[22] Chaté, H., Ginelli, F. and Montagne, R. [2006], 'Simple model for active nematics: Quasi-long-range order and giant fluctuations', *Phys. Rev. Lett.* **96**(18), 180602.

[23] Chelakkot, R., Hagan, M. F. and Gopinath, A. [2021], 'Synchronized oscillations, traveling waves, and jammed clusters induced by steric interactions in active filament arrays', *Soft Matter* **17**(4), 1091–1104.

[24] Cichos, F., Gustavsson, K., Mehlig, B. and Volpe, G. [2020], 'Machine learning for active matter', *Nat. Mach. Intell.* **2**(2), 94–103.

[25] Cordero, O. X., Wildschutte, H., Kirkup, B., Proehl, S., Ngo, L., Hussain, F., Le Roux, F., Mincer, T. and Polz, M. F. [2012], 'Ecological populations of bacteria act as socially cohesive units of antibiotic production and resistance', *Science* **337**(6099), 1228–1231.

[26] Costerton, J., Montanaro, L. and Arciola, C. R. [2005], 'Biofilm in implant infections: its production and regulation', *Artif. Organs* **28**(11), 1062–1068.

[27] Costerton, J. W., Lewandowski, Z., Caldwell, D. E., Korber, D. R. and Lappin-Scott, H. M. [1995], 'Microbial biofilms', *Annu. Rev. Microbiol.* **49**(1), 711–745.

[28] Das, S. and Chelakkot, R. [2020], 'Morphological transitions of active brownian particle aggregates on porous walls', *Soft Matter* **16**, 7250–7255.

[29] Das, S., Ghosh, S. and Chelakkot, R. [2020], 'Aggregate morphology of active brownian particles on porous, circular walls', *Phys. Rev. E* **102**(3), 032619.

[30] Davies, D. [2003], 'Understanding biofilm resistance to antibacterial agents', *Nat. Rev. Drug Discovery* **2**(2), 114–122.

[31] De Magistris, G. and Marenduzzo, D. [2015], 'An introduction to the physics of active matter', *Physica A* **418**, 65–77.

[32] Deforet, M., Van Ditmarsch, D. and Xavier, J. B. [2015], 'Cell-size homeostasis and the incremental rule in a bacterial pathogen', *Biophys. J.* **109**(3), 521–528.

[33] Devaraj, A., Buzzo, J. R., Mashburn-Warren, L., Gloag, E. S., Novotny, L. A., Stoodley, P., Bakaletz, L. O. and Goodman, S. D. [2019], 'The extracellular dna lattice of bacterial biofilms is structurally related to holliday junction recombination intermediates', *P. Natl. Acad. Sci.* **116**, 25068–25077.

[34] Dombrowski, C., Cisneros, L., Chatkaew, S., Goldstein, R. E. and Kessler, J. O. [2004], 'Self-concentration and large-scale coherence in bacterial dynamics', *Phys. Rev. Lett.* **93**(9), 098103.

[35] Dorigo, M., Bonabeau, E. and Theraulaz, G. [2000], 'Ant algorithms and stigmergy', *Future Gener. Comput. Syst.* **16**(8), 851–871.

[36] Downing, H. A. and Jeanne, R. L. [1988], 'Nest construction by the paper wasp, polistes: a test of stigmergy theory', *Anim. Behav.* **36**(6), 1729–1739.

[37] Duman, Ö., Isele-Holder, R. E., Elgeti, J. and Gompper, G. [2018], 'Collective dynamics of self-propelled semiflexible filaments', *Soft Matter* **14**(22), 4483–4494.

[38] Dunkel, J., Heidenreich, S., Drescher, K., Wensink, H. H., Bär, M. and Goldstein, R. E. [2013], 'Fluid dynamics of bacterial turbulence', *Phys. Rev. Lett.* **110**(22), 228102.

[39] Elgeti, J., Winkler, R. G. and Gompper, G. [2015], 'Physics of microswimmers—single particle motion and collective behavior: a review', *Rep. Prog. Phys.* **78**(5), 056601.

[40] Fletcher, D. A. and Geissler, P. L. [2009], 'Active biological materials', *Annu. Rev. Phys. Chem.* **60**, 469–486.

[41] Florio, B. J., Fawell, P. D. and Small, M. [2019], 'The use of the perimeter-area method to calculate the fractal dimension of aggregates', *Powder Technol.* **343**, 551–559.

[42] Fux, C. A., Costerton, J. W., Stewart, P. S. and Stoodley, P. [2005], 'Survival strategies of infectious biofilms', *Trends Microbiol.* **13**(1), 34–40.

[43] Ghosh, P., Ben-Jacob, E. and Levine, H. [2013], 'Modeling cell-death patterning during biofilm formation', *Phys. Biol.* **10**(6), 066006.

[44] Ghosh, P., Mondal, J., Ben-Jacob, E. and Levine, H. [2015], 'Mechanically-driven phase separation in a growing bacterial colony', *P. Natl. Acad. Sci.* **112**(17), E2166–E2173.

[45] Giverso, C., Verani, M. and Ciarletta, P. [2015], 'Branching instability in expanding bacterial colonies', *J. R. Soc. Interface.* **12**(104), 20141290.

[46] Gloag, E. S., Elbadawi, C., Zachreson, C. J., Aharonovich, I., Toth, M., Charles, I. G., Turnbull, L. and Whitchurch, C. B. [2017], 'Micro-patterned surfaces that exploit stigmergy to inhibit biofilm expansion', *Front. Microbiol.* **7**, 2157.

[47] Gloag, E. S., Turnbull, L., Huang, A., Vallotton, P., Wang, H., Nolan, L. M., Mililli, L., Hunt, C., Lu, J., Osvath, S. R. et al. [2013], 'Self-organization of bacterial biofilms is facilitated by extracellular dna', *P. Natl. Acad. Sci.* **110**(28), 11541–11546.

[48] Gloag, E. S., Turnbull, L., Javed, M. A., Wang, H., Gee, M. L., Wade, S. A. and Whitchurch, C. B. [2016], 'Stigmergy co-ordinates multicellular collective behaviours during *myxococcus xanthus* surface migration', *Sci. Rep.* **6**, 26005.

[49] Gloag, E. S., Turnbull, L. and Whitchurch, C. B. [2015], 'Bacterial stigmergy: an organising principle of multicellular collective behaviours of bacteria', *Scientifica* **2015**.

[50] Gompper, G. [2014], 'Collective Motion of Active Particles With and Without Hydrodynamics Microswimmers and Self-Propelled Particles Many biological examples : Interesting phenomena : surface adhesion , vortices ,'.

[51] Gonnella, G., Marenduzzo, D., Suma, A. and Tiribocchi, A. [2015], 'Motility-induced phase separation and coarsening in active matter', *C R Phys* **16**(3), 316–331.

[52] Gopinath, A., Tamayo, J., Zhang, Y., Ardekani, A. M. and Patteson, A. E. [2020], 'Swarming bacterial fronts: Dynamics and morphology of active swarm interfaces propagating through passive frictional domains', *bioRxiv* .

[53] Grass&, P. [1959], '233;. la reconstruction du nid et les coordinations inter-individuelles chez bellicosi-termes natalensis et cubitermes sp. la theorie de la stigmergie', *Insectes Soc.* **6**, 41–83.

[54] Hall-Stoodley, L., Costerton, J. W. and Stoodley, P. [2004], 'Bacterial biofilms: from the natural environment to infectious diseases', *Nat. Rev. Microbiol.* **2**(2), 95–108.

[55] Heithoff, D. M. and Mahan, M. J. [2004], 'Vibrio cholerae biofilms: stuck between a rock and a hard place', *J. Bacteriol.* **186**(15), 4835–4837.

[56] Helbing, D., Schweitzer, F., Keltsch, J. and Molnar, P. [1997], 'Active walker model for the formation of human and animal trail systems', *Phys. Rev. E* **56**(3), 2527.

[57] Hess, H. [2011], 'Engineering applications of biomolecular motors', *Annu. Rev. Biomed. Eng.* **13**, 429–450.

[58] Jackson, D. E., Holcombe, M. and Ratnieks, F. L. [2004], 'Trail geometry gives polarity to ant foraging networks', *Nature* **432**(7019), 907–909.

[59] Jewett, A. I., Stelter, D., Lambert, J., Saladi, S. M., Roscioni, O. M., Ricci, M., Autin, L., Maritan, M., Bashusqeh, S. M., Keyes, T. et al. [2021], 'Moltemplate: A tool for coarse-grained modeling of complex biological matter and soft condensed matter physics', *J. Mol. Biol.* **433**(11), 166841.

[60] Joshi, A., Putzig, E., Baskaran, A. and Hagan, M. F. [2019], 'The interplay between activity and filament flexibility determines the emergent properties of active nematics', *Soft Matter* **15**(1), 94–101.

[61] Jousset, A. [2012], 'Ecological and evolutive implications of bacterial defences against predators', *Environ. Microbiol.* **14**(8), 1830–1843.

[62] Kaiser, A., Peshkov, A., Sokolov, A., Ten Hagen, B., Löwen, H. and Aranson, I. S. [2014], 'Transport powered by bacterial turbulence', *Phys. Rev. Lett.* **112**(15), 158101.

[63] Kaiser, A., Popowa, K., Wensink, H. and Löwen, H. [2013], 'Capturing self-propelled particles in a moving microwedge', *Phys. Rev. E* **88**(2), 022311.

[64] Kaiser, A., Sokolov, A., Aranson, I. S. and Löwen, H. [2015], 'Motion of two micro-wedges in a turbulent bacterial bath', *EPJ-Special Topics* **224**(7), 1275–1286.

[65] Kaiser, A., Wensink, H. and Löwen, H. [2012], 'How to capture active particles', *Phys. Rev. Lett.* **108**(26), 268307.

[66] Kantsler, V., Dunkel, J., Polin, M. and Goldstein, R. E. [2013], 'Ciliary contact interactions dominate surface scattering of swimming eukaryotes', *P. Natl. Acad. Sci.* **110**(4), 1187–1192.

[67] Katz, Y., Tunstrøm, K., Ioannou, C. C., Huepe, C. and Couzin, I. D. [2011], 'Inferring the structure and dynamics of interactions in schooling fish', *P. Natl. Acad. Sci.* **108**(46), 18720–18725.

[68] Kraikivski, P., Lipowsky, R. and Kierfeld, J. [2006], 'Enhanced ordering of interacting filaments by molecular motors', *Phys. Rev. Lett.* **96**(25), 258103.

[69] Kuan, H.-S., Blackwell, R., Hough, L. E., Glaser, M. A. and Betterton, M. D. [2015], 'Hysteresis, reentrance, and glassy dynamics in systems of self-propelled rods', *Phys. Rev. E* **92**(6), 060501.

[70] Kudrolli, A., Lumay, G., Volfson, D. and Tsimring, L. S. [2008], 'Swarming and swirling in self-propelled polar granular rods', *Phys. Rev. Lett.* **100**(5), 058001.

[71] Lee, B., Haagensen, J. A., Ciofu, O., Andersen, J. B., Høiby, N. and Molin, S. [2005], 'Heterogeneity of biofilms formed by nonmucoid pseudomonas aeruginosa isolates from patients with cystic fibrosis', *J. Clin. Microbiol.* **43**(10), 5247–5255.

[72] Lee, B., Konen, J., Wilkinson, S., Marcus, A. I. and Jiang, Y. [2017], 'Local alignment vectors reveal cancer cell-induced ecm fiber remodeling dynamics', *Sci. Rep.* **7**, 39498.

[73] Liebovitch, L. S. and Toth, T. [1989], 'A fast algorithm to determine fractal dimensions by box counting', *Phys. Lett. A* **141**(8-9), 386–390.

[74] Liu, Y., Yang, Y., Li, B. and Feng, X.-Q. [2019], 'Collective oscillation in dense suspension of self-propelled chiral rods', *Soft matter* **15**(14), 2999–3007.

[75] Luppens, S. B., Reij, M. W., van der Heijden, R. W., Rombouts, F. M. and Abee, T. [2002], 'Development of a standard test to assess the resistance of staphylococcus aureus biofilm cells to disinfectants', *Appl. Environ. Microbiol.* **68**(9), 4194–4200.

[76] Mah, T.-F., Pitts, B., Pellock, B., Walker, G. C., Stewart, P. S. and O'Toole, G. A. [2003], 'A genetic basis for pseudomonas aeruginosa biofilm antibiotic resistance', *Nature* **426**(6964), 306–310.

[77] Mandal, S., Liebchen, B. and Löwen, H. [2019], 'Motility-induced temperature difference in coexisting phases', *Phys. Rev. Lett.* **123**(22), 228001.

[78] Marchetti, M. C., Joanny, J.-F., Ramaswamy, S., Liverpool, T. B., Prost, J., Rao, M. and Simha, R. A. [2013], 'Hydrodynamics of soft active matter', *Rev. Mod. Phys.* **85**(3), 1143.

[79] Mark, S., Shlomovitz, R., Gov, N. S., Poujade, M., Grasland-Mongrain, E. and Silberzan, P. [2010], 'Physical model of the dynamic instability in an expanding cell culture', *Biophys. J.* **98**(3), 361–370.

[80] Martínez, L., Andrade, R., Birgin, E. G. and Martínez, J. M. [2009], 'Packmol: a package for building initial configurations for molecular dynamics simulations', *J. Comput. Chem.* **30**(13), 2157–2164.

[81] Marín-Caba, L., Chariou, P. L., Pesquera, C., Correa-Duarte, M. A. and Steinmetz, N. F. [2018], 'Tobacco mosaic virus-functionalized mesoporous silica nanoparticles, a wool-ball-like nanostructure for drug delivery', *Langmuir* **35**(1), 203–211.

[82] McCandlish, S. R., Baskaran, A. and Hagan, M. F. [2012], 'Spontaneous segregation of self-propelled particles with different motilities', *Soft Matter* **8**(8), 2527–2534.

[83] Mendelson, N. H., Bourque, A., Wilkening, K., Anderson, K. R. and Watkins, J. C. [1999], 'Organized cell swimming motions in bacillus subtilis colonies: patterns of short-lived whirls and jets', *J. Bacteriol.* **181**(2), 600–609.

[84] Menon, G. I. [2010], Active matter, *in* 'Rheology of complex Fluids', Springer, pp. 193–218.

[85] Mokhtari, Z. and Zippelius, A. [2019], 'Dynamics of active filaments in porous media', *Phys. Rev. Lett.* **123**(2), 028001.

[86] Moore, J. M., Thompson, T. N., Glaser, M. A. and Betterton, M. D. [2020], 'Collective motion of driven semiflexible filaments tuned by soft repulsion and stiffness', *Soft Matter* **16**(41), 9436–9442.

[87] Mousavi, S. M., Gompper, G. and Winkler, R. G. [2020], 'Wall entrapment of peritrichous bacteria: a mesoscale hydrodynamics simulation study', *Soft Matter* **16**(20), 4866–4875.

[88] Murtola, T., Bunker, A., Vattulainen, I., Deserno, M. and Karttunen, M. [2009], 'Multiscale modeling of emergent materials: biological and soft matter', *Phys. Chem. Chem. Phys.* **11**(12), 1869–1892.

[89] Nadell, C. D., Xavier, J. B. and Foster, K. R. [2008], 'The sociobiology of biofilms', *FEMS Microbiol. Rev.* **33**(1), 206–224.

[90] Nagilla, A., Prabhakar, R. and Jadhav, S. [2018], 'Linear stability of an active fluid interface', *Phys. Fluids* **30**, 022109.

[91] Narayan, V., Menon, N. and Ramaswamy, S. [2006], 'Nonequilibrium steady states in a vibrated-rod monolayer: tetratic, nematic, and smectic correlations', *J. Stat. Mech: Theory Exp.* **2006**(01), P01005.

[92] Nishiguchi, D., Nagai, K. H., Chaté, H. and Sano, M. [2017], 'Long-range nematic order and anomalous fluctuations in suspensions of swimming filamentous bacteria', *Phys. Rev. E* **95**(2), 020601.

[93] Nitta, T., Tanahashi, A., Hirano, M. and Hess, H. [2006], 'Simulating molecular shuttle movements: Towards computer-aided design of nanoscale transport systems', *Lab Chip* **6**(7), 881–885.

[94] Obst, U., Schwartz, T. and Volkmann, H. [2006], 'Antibiotic resistant pathogenic bacteria and their resistance genes in bacterial biofilms', *Artif. Organs* **29**(4), 387–394.

[95] O'Toole, G., Kaplan, H. B. and Kolter, R. [2000], 'Biofilm formation as microbial development', *Annu. Rev. Microbiol.* **54**(1), 49–79.

[96] Parrish, J. K. and Hamner, W. M. [1997], *Animal groups in three dimensions: how species aggregate*, Cambridge University Press.

[97] Patteson, A. E., Gopinath, A. and Arratia, P. E. [2018], 'The propagation of active-passive interfaces in bacterial swarms', *Nat. Commun.* **9**(1), 1–10.

[98] Persat, A., Inclan, Y. F., Engel, J. N., Stone, H. A. and Gitai, Z. [2015], 'Type iv pili mechanochemically regulate virulence factors in pseudomonas aeruginosa', *P. Natl. Acad. Sci.* **112**(24), 7563–7568.

[99] Peruani, F. and Baer, M. [2013], 'A kinetic model and scaling properties of non-equilibrium clustering of self-propelled particles', *New J. Phys.* **15**(6), 065009.

[100] Peruani, F., Deutsch, A. and Bär, M. [2006], 'Nonequilibrium clustering of self-propelled rods', *Phys. Rev. E* **74**(3), 030904.

[101] Peruani, F., Deutsch, A. and Bär, M. [2008], 'A mean-field theory for self-propelled particles interacting by velocity alignment mechanisms', *EPJ-Special Topics* **157**(1), 111–122.

[102] Peruani, F., Schimansky-Geier, L. and Baer, M. [2010], 'Cluster dynamics and cluster size distributions in systems of self-propelled particles', *EPJ-Special Topics* **191**(1), 173–185.

[103] Peruani, F., Starruß, J., Jakovljevic, V., Søgaard-Andersen, L., Deutsch, A. and Bär, M. [2012], 'Collective motion and nonequilibrium cluster formation in colonies of gliding bacteria', *Phys. Rev. Lett.* **108**(9), 098102.

[104] Plimpton, S. [1993], Fast parallel algorithms for short-range molecular dynamics, Technical report, Sandia National Labs., Albuquerque, NM (United States).

[105] Prathyusha, K., Henkes, S. and Sknepnek, R. [2018], 'Dynamically generated patterns in dense suspensions of active filaments', *Phys. Rev. E* **97**(2), 022606.

[106] Qi, K., Westphal, E., Gompper, G. and Winkler, R. G. [2020], 'Enhanced rotational motion of spherical squirmer in polymer solutions', *Phys. Rev. Lett.* **124**(6), 068001.

[107] Ramaswamy, S. [2010], 'The mechanics and statistics of active matter', *Annu. Rev. Condens. Matter Phys.* **1**(1), 323–345.

[108] Redner, G. S., Hagan, M. F. and Baskaran, A. [2013], 'Structure and dynamics of a phase-separating active colloidal fluid', *Phys. Rev. Lett.* **110**(5), 055701.

[109] Ruhnow, F., Zwicker, D. and Diez, S. [2011], 'Tracking single particles and elongated filaments with nanometer precision', *Biophys. J.* **100**(11), 2820–2828.

[110] Sarkar, N. and Chaudhuri, B. B. [1994], 'An efficient differential box-counting approach to compute fractal dimension of image', *IEEE Trans. Syst. Man Cybern. Syst.* **24**(1), 115–120.

[111] Shi, X.-q., Fausti, G., Chaté, H., Nardini, C. and Solon, A. [2020], 'Self-organized critical coexistence phase in repulsive active particles', *Phys. Rev. Lett.* **125**(16), 168001.

[112] Soifer, I., Robert, L. and Amir, A. [2016], 'Single-cell analysis of growth in budding yeast and bacteria reveals a common size regulation strategy', *Curr. Biol.* **26**(3), 356–361.

[113] Sokolov, A., Aranson, I. S., Kessler, J. O. and Goldstein, R. E. [2007], 'Concentration dependence of the collective dynamics of swimming bacteria', *Phys. Rev. Lett.* **98**(15), 158102.

[114] Stewart, P. S. [2002], 'Mechanisms of antibiotic resistance in bacterial biofilms', *Int. J. Med. Microbiol.* **292**(2), 107–113.

[115] ten Hagen, B., van Teeffelen, S. and Löwen, H. [2011], 'Brownian motion of a self-propelled particle', *J. Phys.: Condens. Matter* **23**(19), 194119.

[116] Theeyancheri, L., Chaki, S., Samanta, N., Goswami, R., Chelakkot, R. and Chakrabarti, R. [2020], 'Translational and rotational dynamics of a self-propelled janus probe in crowded environments', *Soft Matter* **16**(36), 8482–8491.

[117] Tjhung, E., Nardini, C. and Cates, M. E. [2018], 'Cluster phases and bubbly phase separation in active fluids: reversal of the ostwald process', *Phys. Rev. X* **8**(3), 031080.

[118] Toner, J., Tu, Y. and Ramaswamy, S. [2005], 'Hydrodynamics and phases of flocks', *Ann. Phys.* **318**(1), 170–244.

[119] Turnbull, L., Toyofuku, M., Hynen, A. L., Kurosawa, M., Pessi, G., Petty, N. K., Osvath, S. R., Cárcamo-Oyarce, G., Gloag, E. S., Shimoni, R., Omasits, U., Ito, S., Yap, X., Monahan, L. G., Cavaliere, R., Ahrens, C. H., Charles, I. G., Nomura, N., Eberl, L. and Whitchurch, C. B. [2016], 'Explosive cell lysis as a mechanism for the biogenesis of bacterial membrane vesicles and biofilms', *Nat. Commun.* **7**, 11220.

[120] Velasco, C. A., Abkenar, M., Gompper, G. and Auth, T. [2018], 'Collective behavior of self-propelled rods with quorum sensing', *Phys. Rev. E* **98**(2), 022605.

[121] Velicer, G. J. and Vos, M. [2009], 'Sociobiology of the myxobacteria', *Annu. Rev. Microbiol.* **63**, 599–623.

[122] Vicsek, T., Czirók, A., Ben-Jacob, E., Cohen, I. and Shochet, O. [1995], 'Novel type of phase transition in a system of self-driven particles', *Phys. Rev. Lett.* **75**(6), 1226.

[123] Vicsek, T. and Zafeiris, A. [2012], 'Collective motion', *Phys. Rep.* **517**(3), 71–140.

[124] Vliegenthart, G. A., Ravichandran, A., Ripoll, M., Auth, T. and Gompper, G. [2020], 'Filamentous active matter: Band formation, bending, buckling, and defects', *Sci. Adv.* **6**(30), eaaw9975.

[125] Walker, J. T., Surman, S. and Jass, J. [2000], *Industrial biofouling: detection, prevention, and control*, Wiley New York.

[126] Wang, J. and Gao, W. [2012], 'Nano/microscale motors: biomedical opportunities and challenges', *ACS nano* **6**(7), 5745–5751.

[127] Wang, W. and Chau, Y. [2009], 'Self-assembled peptide nanorods as building blocks of fractal patterns', *Soft Matter* **5**(24), 4893–4898.

[128] Watnick, P. I. and Kolter, R. [1999], 'Steps in the development of a vibrio cholerae el tor biofilm', *Mol. Microbiol.* **34**(3), 586–595.

[129] Weitz, S., Deutsch, A. and Peruani, F. [2015], 'Self-propelled rods exhibit a phase-separated state characterized by the presence of active stresses and the ejection of polar clusters', *Phys. Rev. E* **92**(1), 012322.

[130] Wensink, H. H., Dunkel, J., Heidenreich, S., Drescher, K., Goldstein, R. E., Löwen, H. and Yeomans, J. M. [2012], 'Meso-scale turbulence in living fluids', *P. Natl. Acad. Sci.* **109**(36), 14308–14313.

[131] Wensink, H., Kantsler, V., Goldstein, R. and Dunkel, J. [2014], 'Controlling active self-assembly through broken particle-shape symmetry', *Phys. Rev. E* **89**(1), 010302.

[132] Wensink, H. and Löwen, H. [2008], 'Aggregation of self-propelled colloidal rods near confining walls', *Phys. Rev. E* **78**(3), 031409.

[133] Wensink, H. and Löwen, H. [2012], 'Emergent states in dense systems of active rods: from swarming to turbulence', *J. Phys.: Condens. Matter* **24**(46), 464130.

[134] Winkler, J., Abisoye-Ogunniyan, A., Metcalf, K. J. and Werb, Z. [2020], 'Concepts of extracellular matrix remodelling in tumour progression and metastasis', *Nat. Commun.* **11**(1), 1–19.

[135] Winkler, R. G. and Gompper, G. [2020], 'The physics of active polymers and filaments', *J. Chem. Phys.* **153**(4), 040901.

[136] Wisdom, K. M., Adebowale, K., Chang, J., Lee, J. Y., Nam, S., Desai, R., Rossen, N. S., Rafat, M., West, R. B., Hodgson, L. et al. [2018], 'Matrix mechanical plasticity regulates cancer cell migration through confining microenvironments', *Nat. Commun.* **9**(1), 1–13.

[137] Woodward, D., Tyson, R., Myerscough, M., Murray, J., Budrene, E. and Berg, H. [1995], 'Spatio-temporal patterns generated by salmonella typhimurium', *Biophys. J.* **68**(5), 2181–2189.

[138] Yang, Y., Elgeti, J. and Gompper, G. [2008], 'Cooperation of sperm in two dimensions: synchronization, attraction, and aggregation through hydrodynamic interactions', *Phys. Rev. E* **78**(6), 061903.

[139] Yang, Y., Marceau, V. and Gompper, G. [2010], 'Swarm behavior of self-propelled rods and swimming flagella', *Phys. Rev. E* **82**(3), 031904.

[140] Zachreson, C., Wolff, C., Whitchurch, C. B. and Toth, M. [2017], 'Emergent pattern formation in an interstitial biofilm', *Phys. Rev. E* **95**(1), 012408.

[141] Zachreson, C., Yap, X., Gloag, E. S., Shimoni, R., Whitchurch, C. B. and Toth, M. [2017], 'Network patterns in exponentially growing two-dimensional biofilms', *Phys. Rev. E* **96**(4), 042401.

[142] Zhou, Y., Kellermann, C. and Griebler, C. [2012], 'Spatio-temporal patterns of microbial communities in a hydrologically dynamic pristine aquifer', *FEMS Microbiol. Ecol.* **81**(1), 230–242.

# Appendix A

## A.1 Derivation of $\epsilon$

The separation shifted, Lennard-Jones (SSLJ) potential is given as

$$\phi(r) = \begin{cases} 4\,\epsilon\left[\left(\dfrac{\sigma^2}{r^2+\alpha^2}\right)^6 - \left(\dfrac{\sigma^2}{r^2+\alpha^2}\right)^3\right] - \phi_0, \,, & r \leq r_c\,; \\ = 0, \quad r > r_c\,. \end{cases} \tag{A.1}$$

where $r_c$ is the cut-off distance. The capping parameter $\alpha$ is given as

$$\alpha = \sqrt{2^{1/3}\sigma^2 - r_c^2} \tag{A.2}$$

The softness of the potential is controlled by setting the value of the potential energy, $E = \phi(0) - \phi(r_c)$ when two interacting particles overlap.

Calculating the $\phi$ at $r = 0, r_c$ by substituting values of $r$ in Eqn . A.1.

$$\phi(0) = 4\,\epsilon\left[\left(\frac{\sigma^2}{\alpha^2}\right)^6 - \left(\frac{\sigma^2}{\alpha^2}\right)^3\right] - \phi_0, \tag{A.3}$$

$$\phi(r_c) = 4\,\epsilon\left[\left(\frac{\sigma^2}{r_c^2+\alpha^2}\right)^6 - \left(\frac{\sigma^2}{r_c^2+\alpha^2}\right)^3\right] - \phi_0, \tag{A.4}$$

From Eqn. A.2,

$$\alpha^2 = 2^{1/3}\sigma^2 - r_c^2 \tag{A.5}$$

$$\alpha^2 + r_c^2 = 2^{1/3}\sigma^2 \tag{A.6}$$

Substituting the value of $\alpha^2 + r_c^2$ from Eqn. A.6 in Eqn. A.4

$$\phi(r_c) = 4\,\epsilon\left[\left(\frac{\sigma^2}{2^{1/3}\sigma^2}\right)^6 - \left(\frac{\sigma^2}{2^{1/3}\sigma^2}\right)^3\right] - \phi_0, \tag{A.7}$$

$$\phi(r_c) = -\epsilon - \phi_0, \tag{A.8}$$

$E$ can be given as

$$E = 4\,\epsilon\left[\left(\frac{\sigma^2}{\alpha^2}\right)^6 - \left(\frac{\sigma^2}{\alpha^2}\right)^3\right] - \phi_0, +\epsilon + \phi_0 \tag{A.9}$$

$$E = 4\,\epsilon\left[\frac{4\sigma^{12} - 4\sigma^6\alpha^6 + \alpha^{12}}{4\alpha^{12}}\right] \tag{A.10}$$

$$\epsilon = \frac{\alpha^{12} E}{\alpha^{12} - 4\sigma^6\alpha^6 + 4\sigma^{12}}. \tag{A.11}$$

## A.2 Calculation of Pe

The general definition of the Peclet Number, Pe is the ratio of diffusion time and convection time. The Pe for rod can be given as,

$$\mathrm{Pe} = \frac{\text{Diffusion time-scale}}{\text{Convection time-scale}} \tag{A.12}$$

$$\mathrm{Pe} = \frac{L^2/D}{L/\mathbf{v}^a} \tag{A.13}$$

This can be re-written as,

$$\mathrm{Pe} = \frac{L(\gamma_\mathrm{r}\mathbf{v}^a)}{k_\mathrm{B}T} = \frac{\text{Length * Propulsion force}}{\text{translational noise strength}} \tag{A.14}$$

The above can also be interpreted as the ratio of propulsion strength and noise strength in the system.

## A.3 Relation between smoothness factor and area of a rod

Smoothness factor, $\chi$ for a rod of given length $L$ can be given as,

$$\chi = 1 - \frac{L-1}{(N_\mathrm{b}-1)\sigma_\mathrm{b}}. \tag{A.15}$$

Here, $N_b$ is the number of beads in a rod of diameter $\sigma_b$.

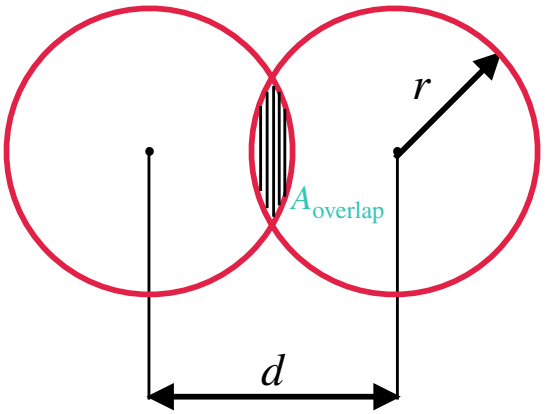


FIGURE A.1: Schematic of overlapped area between two beads.

Overlapping area between two beads of same size (radius, $r$) is given as,

$$A_{overlap} = \left[r^2\alpha + r^2\beta - \frac{1}{2}\left(r^2\sin(2\alpha) + r^2\sin(2\beta)\right)\right] \tag{A.16}$$

Here, $\alpha = \beta = \cos^{-1}\left(\frac{d}{\sigma_b}\right)$ are the angles made by the overlapped region and centers. Center to center distance between the overlapping bead, $d$ is given as

$$d = 1 - \chi = \frac{L-1}{(N_b-1)\sigma_b}. \tag{A.17}$$

Eq. A.16 can be written as

$$A_{\text{overlap}} = 2r^2\alpha - \sin(2\alpha)r^2 \tag{A.18}$$

$$= 2r^2\alpha - 2r^2\sin(\alpha)\cos(\alpha) \tag{A.19}$$

$$= 2r^2\alpha - 2r^2\sin(\cos^{-1}\left(\frac{d}{\sigma_{\text{b}}}\right)\cos(\cos^{-1}\left(\frac{d}{\sigma_{\text{b}}}\right) \tag{A.20}$$

$$= 2r^2\alpha - \left[2r^2\sin\left[\sin^{-1}\left(\frac{\sqrt{(\sigma_{\text{b}}^2 - d^2)}}{\sigma_{\text{b}}}\right)\right]\frac{d}{\sigma_{\text{b}}}\right] \tag{A.21}$$

$$= 2r^2\alpha - 2r^2\frac{\sqrt{(\sigma_{\text{b}}^2 - d^2)}}{\sigma_{\text{b}}}\frac{d}{\sigma_{\text{b}}} \tag{A.22}$$

$$= 2r^2\Big[\alpha - \frac{d\sqrt{(\sigma_{\text{b}}^2 - d^2)}}{\sigma_{\text{b}}^2}\Big] \tag{A.23}$$

$$= \frac{\sigma_{\text{b}}^2}{2}\Big[\alpha - \frac{d\sqrt{(\sigma_{\text{b}}^2 - d^2)}}{\sigma_{\text{b}}^2}\Big] \tag{A.24}$$

$$= \frac{1}{2}\big[\alpha\sigma_{\text{b}}^2 - d\sqrt{(\sigma_{\text{b}}^2 - d^2)}\big] \tag{A.25}$$

$A_{\text{rod}}$ = Area of a bead $\times N_{\text{b}}$ - $(N_{\text{b}} - 1)$ $\times A_{\text{overlap}}$

$$A_{\text{rod}} = \frac{N_{\text{b}}\pi\sigma_{\text{b}}^2}{4} - \frac{N_{\text{b}} - 1}{2}\Big[\sigma_{\text{b}}^2\cos^{-1}\left(\frac{d}{\sigma_{\text{b}}}\right) - d\sqrt{\sigma_{\text{b}}^2 - d^2}\Big] \tag{A.26}$$

Substituting the value of $d$ in Eq. A.26

$$\boxed{A_{\text{rod}} = \frac{N_{\text{b}}\pi\sigma_{\text{b}}^2}{4} - \frac{N_{\text{b}} - 1}{2}\Big[\sigma_{\text{b}}^2\cos^{-1}\left(\frac{1-\chi}{\sigma_{\text{b}}}\right) - (1-\chi)\sqrt{\sigma_{\text{b}}^2 - (1-\chi)^2}\Big]} \tag{A.27}$$

Area of a rod of a given length $L$ and made of $N_{\text{b}}$ beads can be given as,

$$\boxed{A_{\text{rod}} = \frac{N_{\text{b}}\pi\sigma_{\text{b}}^2}{4} - \frac{N_{\text{b}} - 1}{2}\left[\sigma_{\text{b}}^2\cos^{-1}\left(\frac{L-1}{(N_{\text{b}}-1)\sigma_{\text{b}}^2}\right) - \frac{L-1}{(N_{\text{b}}-1)\sigma_{\text{b}}^2}\sqrt{\sigma_{\text{b}}^2 - \frac{(L-1)^2}{(N_{\text{b}}-1)^2\sigma_{\text{b}}^2}}\right]} \tag{A.28}$$

$$\phi_{\text{eff}} = \frac{N_{\text{rod}} A_{\text{rod}}}{A_{\text{box}}} \tag{A.29}$$

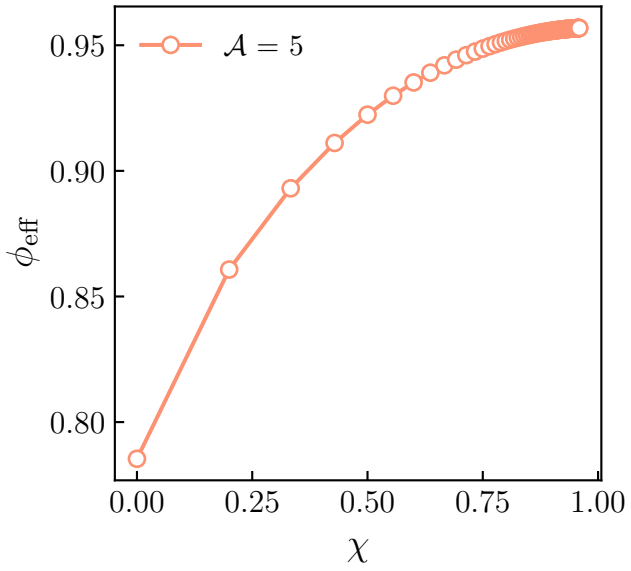


FIGURE A.2: Effective packing fraction of a system as a function of smoothness factor for a length of $L = 5$ at constant scaled number density of 5.

# Appendix B

# Estimation of velocity of isolated self-propelled rod in elasto-plastic substrate

Consider the case where substrate stiffness is zero *i.e.* all substrate particles are unbound. Then, when Brownian forces are negligible, the force balance on a rod is given as,

$$\gamma_{\mathrm{r}}\, v_1 \;=\; F^{\mathrm{a}} \;-\; \gamma_{\mathrm{s}}\, v_1\,, \tag{B.1}$$

where $v_1$ is the single-rod speed. Hence,

$$v_1 \;=\; \frac{F^{\mathrm{a}}}{\gamma_{\mathrm{s}} + \gamma_{\mathrm{r}}}\,. \tag{B.2}$$

Hence, the dimensionless single-rod speed at zero stiffness is:

$$V_1^0 \;=\; \frac{1}{N_{\mathrm{b}}^2}\,\frac{F^{\mathrm{a}}}{\gamma_{\mathrm{s}} + \gamma_{\mathrm{r}}}\,\frac{L\gamma_{\mathrm{r}}}{k_{\mathrm{B}}\,T} \;=\; \frac{\mathrm{Pe}}{N_{\mathrm{b}}^2\,(1 \,+\, \gamma_{\mathrm{s}}/\gamma_{\mathrm{r}})}\,. \tag{B.3}$$

# Appendix C

# Validation studies

To study the behavior of SPRs in systems with and without elasto-plastic substrate, simulations were performed using the LAMMPS [104] package. New modules were developed to study the behavior of active systems. This in-house modification of LAMMPS codes was tested against the well-known results for dilute and dense systems of self-propelled Brownian particles.

For the dilute system, a single self-propelled Brownian particle was simulated, and its dynamic properties, such as mean-square displacement(MSD) was compared against the theory [115, 9]. We further performed simulations of a dense system of active Brownian particles. In this system, Pe was chosen as a control parameter of the system. We compared the MSD and density distributions behaviors of the system with the results of Redner *et al.* [108].

## C.1 Behavior of a dilute system of self-propelled particle

To validate the in-house developed codes, a system consisting of a single self-propelled Brownian particle was simulated. We first compute the MSD of the particle as follows.

The evolution of position of particle of mass, $m_{\mathrm{b}}$, and diameter, $\sigma_{\mathrm{b}}$ can be given by Langevin dynamics,

$$m_{\mathrm{b}}\ddot{\mathbf{r}} = -\gamma\dot{\mathbf{r}} + F_0\hat{\mathbf{p}} + \sqrt{2D_t}\eta_t. \tag{C.1}$$

Here, $F_0$ is the magnitude of a constant self-propelled force with polarity given as an unit vector $\hat{\mathbf{p}} = (\cos\theta + \sin\theta)$, where $\theta$ is the polarity of the particle. The polarity direction evolves as,

$$\dot{\theta} = \sqrt{2D_r}\eta_{\mathbf{r}}. \tag{C.2}$$

$D_t$ and $D_r$ are translational and rotational diffusion coefficients respectively. The $\eta$ are Gaussian white noise which are characterized as

$$\big\langle \eta(t) \big\rangle = 0 \tag{C.3}$$

$$\big\langle \eta(t)\eta(t') \big\rangle = \delta(t - t') \tag{C.4}$$

Eq. C.1 can be recast into non-dimensionless form by taking particle diameter $\sigma_{\mathrm{b}}$ as length scale, rotational diffusion time-scale, $1/D_r$ as time-scale of system, and $k_{\mathrm{B}}\,T$ as energy scale.

$$I\ddot{\mathbf{r}}^* = -\dot{\mathbf{r}}^* + \mathrm{Pe} + E\eta_t^* \tag{C.5}$$

For over-damped dynamics of particle, $I << 1$, which characterizes the inertial effect of the particle. The equation of motion becomes,

$$\dot{\mathbf{r}}^* = \mathrm{Pe} + E\eta_t^*. \tag{C.6}$$

Here, $\mathrm{Pe} = v\frac{\tau}{\sigma_{\mathrm{b}}}$ is Peclet number and $E$ is scaled thermal energy.

The MSD of a single self-propelled particle [9] can be given as,

$$\langle|\Delta r(t)|^2\rangle = \left[4D_t + 2v^2 t_r\right]\tau + 2v^2 t_r^2\left[e^{-\frac{t}{t_r}} - 1\right]. \tag{C.7}$$

Here, $|\Delta r(t)|$ is the displacement of the particle at time $t$, $v$ is the self-propulsion speed, and $t_r$ is the rotational diffusion time scale of the system.

Another expression for MSD is given by Hegan et al (2011) [115].

$$\langle|\Delta r(t)|^2\rangle = \frac{4}{3}\sigma_{\mathrm{b}}^2 D_r t + 2\Big(\frac{1}{3}\beta F\sigma^2\Big)^2\Big[D_r t - 1 + e^{-D_r t}\Big] \tag{C.8}$$

Here, $\beta = (k_{\mathrm{B}}\,T)^{-1}$ is the inverse effective thermal energy and $t_r$ is the rotation diffusion time-scale. Non-dimensionalization of equation C.8 leads to

$$\langle|\Delta r(\tau)|^2\rangle = \left[\tfrac{4}{3} + 2\mathrm{Pe}^2\right]\tau^* + 2\mathrm{Pe}^2(\exp^{-\tau^*} - 1) \tag{C.9}$$

A single particle is initially placed with random position and random polarity in a periodic box of size $1000\sigma_{\mathrm{b}} \times 1000\sigma_{\mathrm{b}}$. The following table summarizes the simulation parameters used for this study.

TABLE C.1: Simulation parameters

| **Parameter** | **Value** |
|---|---|
| $\sigma_{\mathrm{b}}$ | 1.0 |
| $\tau = 1/D_r$ | 1.0 |
| $k_{\mathrm{B}}\,T$ | 1.0 |
| $I$ | $10^{-4}$ |
| $\frac{D_t}{D_r}$ | $\frac{\sigma_{\mathrm{b}}^2}{3}$ |
| $k_{\mathrm{B}}\,T$ | 1.0 |
| $N$ | 1.0 |
| $L_{\mathrm{box}}$ | $1000\sigma$ |
| Pe | 50,80,100 |

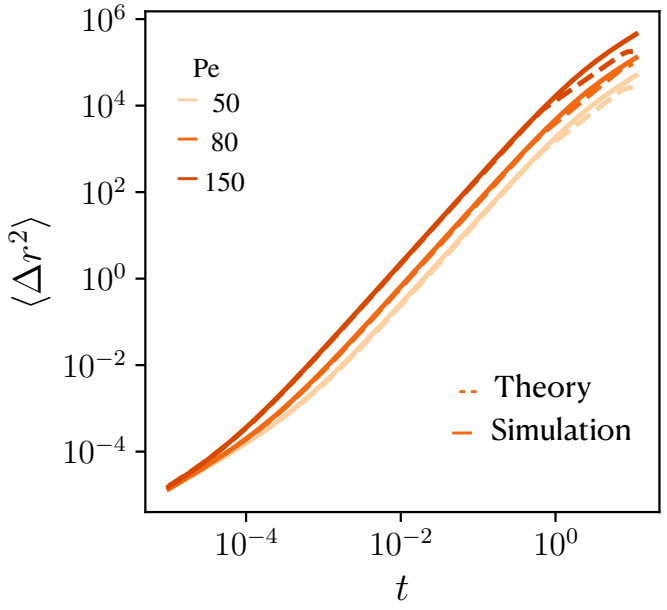


FIGURE C.1: Validation of mean square displacement behavior of active particles against Eq. C.9.

Figure C.1 shows that initially particle moves in ballistic motion. At later time scale particle performs diffusive motion. The MSD behavior obtained from the simulation shows a good agreement with the theory (Fig. C.1)

### C.1.1 Motility induced phase separation

The in-house developed codes were further validated by comparing the results for a dense system of self-propelled Brownian particles. The motility induced phase separation (MIPS) [108], a characteristic feature of the system of self-propelled Brownian particles, was reproduced.

Table C.2 shows the simulation parameters used for the validation study.

TABLE C.2: Simulation parameters

| Parameter | Value |
|---|---|
| $\sigma$ | 1.0 |
| $\tau = 1/D_r$ | 1.0 |
| $k_{\mathrm{B}}\,T$ | 1.0 |
| $I$ | $10^{-4}$ |
| $\frac{D_t}{D_r}$ | $\frac{\sigma_{\mathrm{b}}^2}{3}$ |
| $k_{\mathrm{B}}\,T$ | 1.0 |
| $\varphi$ | 0.65 |
| $L_{\mathrm{box}}$ | $150\sigma_{\mathrm{b}}$ |
| Pe | [10-150] |

Fig. C.2 shows the three distinct morphologies of a dense system ($\varphi = 0.65$) at three Pe [10, 50, 150]. In Fig. C.2 A, at Pe=10, particles move randomly without forming clusters in the system. As Pe increases, particles begin to form small clusters (Fig. C.2 B). At moderate Pe [$50 \sim 80$], phase separation is not observed in the system. On the further increase of Pe, particles from a cluster of almost the size of the system and phase separation is observed. The two distinct phases observed in the systems are dense (cluster) and vapor phases.

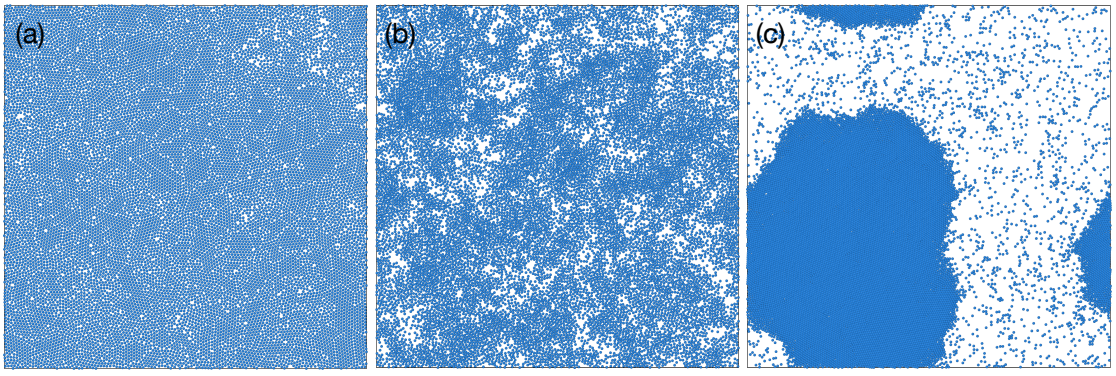


FIGURE C.2: Phase separation in a dense active system (Packing fraction $\varphi = 0.65$) of a periodic system size of $150\sigma_{\mathrm{b}} \times 150\sigma_{\mathrm{b}}$. (A) At low Pe,(Pe = 10) particles move randomly in the simulation box and system shows a homogeneous state. (B) As Pe (Pe = 50) increases the small clusters begin to form. (C) At higher Pe (Pe = 150) system shows phase separation and separates into dilute and dense phases.

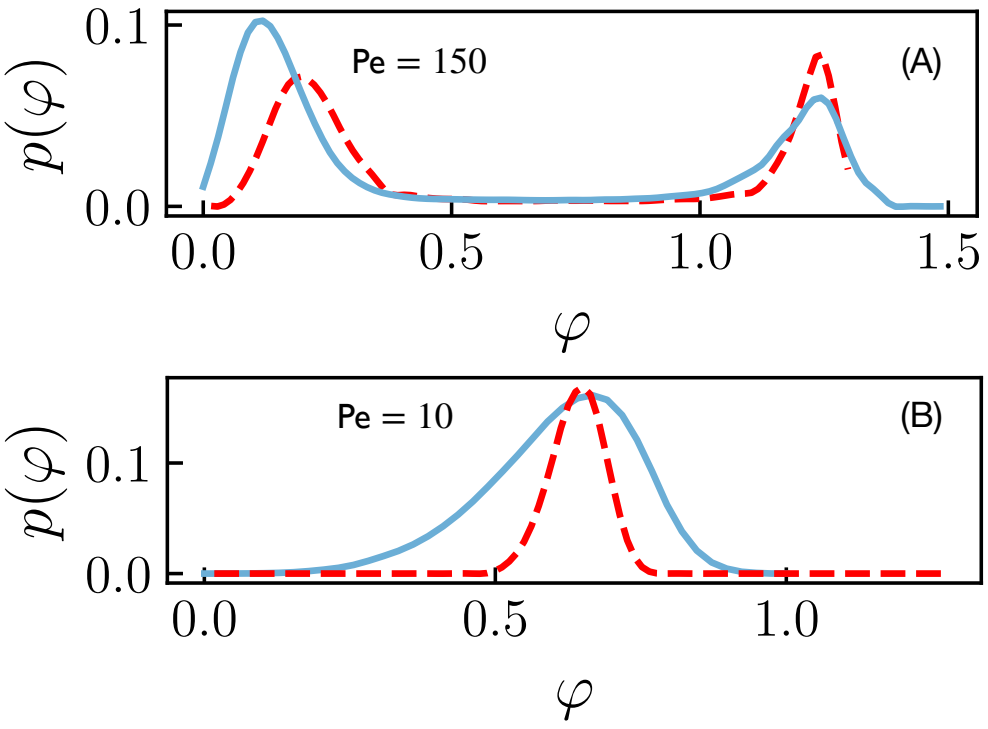


FIGURE C.3: Density distributions for various Peclet Numbers. At lower Pe systems shows peak at average system density $\phi = 0.65$; however, at higher Pe it shows a bimodal distribution indicating dense and dilute phases. The dotted red lines show the results from Redner et al. [108] and solids lines are from the validation study.

The density distribution of the system is calculated by dividing the simulation box into $N \times N$ bins. The local number is calculated in each bin of the system. The frequency distribution of the density below Pe = 10 is uni-modal, whereas, at a higher Pe of 150, the distribution is bimodal, indicating phase separation with the co-existence of dense and dilute regions.

To quantify the dynamical nature of the system, MSD was calculated. Fig. C.4 shows the comparison of MSD for different Pe at $\varphi = 0.65$.

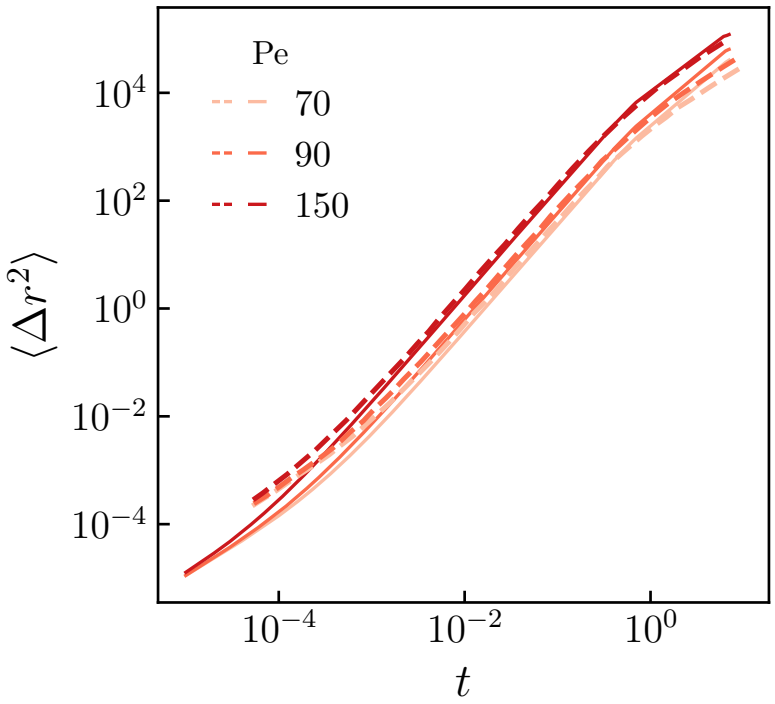


FIGURE C.4: Comparison of Mean square displacement of an active dense system for different Pe at $\varphi = 0.65$. The dotted lines show the results from Redner et al. [108] and solids lines are from the validation study.

The results obtained from the validation study of the developed codes show good qualitative agreement with the well-known results in the literature. However, some deviation in the density distribution and MSD behavior of a dense system of active Brownian particles is due to the different system size, integration schemes for the equation, and sampling rate of the data than the work of Redner et al.[108].

The following sections contain some of the key codes developed during the study. The implementation details of these codes are discussed in Chapter 2.

## Implementation of active force

```
bond_active.cpp       Mon Mar 01 14:33:57 2021       1
/* ----------------------------------------------------
------------------
   LAMMPS - Large-scale Atomic/Molecular Massively Par
allel Simulator
   http://lammps.sandia.gov, Sandia National Laborator
ies
   Steve Plimpton, sjplimp@sandia.gov

   Copyright (2003) Sandia Corporation.  Under the ter
ms of Contract
   DE-AC04-94AL85000 with Sandia Corporation, the U.S.
 Government retains
   certain rights in this software.  This software is
distributed under
   the GNU General Public License.

   See the README file in the top-level LAMMPS directo
ry.
----------------------------------------------------
------------------ */

#include <math.h>
#include <stdlib.h>
#include <string.h>
#include "bond_active.h"
#include "atom.h"
#include "neighbor.h"
#include "domain.h"
#include "comm.h"
#include "force.h"
#include "memory.h"
#include "error.h"

using namespace LAMMPS_NS;

/* ----------------------------------------------------
------------------ */

BondActive::BondActive(LAMMPS *lmp) : Bond(lmp)
{
  reinitflag = 1;
}

/* ----------------------------------------------------
------------------ */

BondActive::~BondActive()
```

## Implementation of active force

**bond_active.cpp Mon Mar 01 14:33:57 2021 2**

```
{
  if (allocated && !copymode) {
    memory->destroy(setflag);
    memory->destroy(k);
    memory->destroy(r0);
    memory->destroy(beta);
  }
}

/* ----------------------------------------------------
------------------ */

void BondActive::compute(int eflag, int vflag)
{
  int i1,i2,n,type;
  double delx,dely,delz,ebond,fbond;
  double rsq,r,dr,rk,theta;

  ebond = 0.0;
  if (eflag || vflag) ev_setup(eflag,vflag);
  else evflag = 0;

  double **x = atom->x;
  double **f = atom->f;
  int **bondlist = neighbor->bondlist;
  int nbondlist = neighbor->nbondlist;
  int nlocal = atom->nlocal;
  int newton_bond = force->newton_bond;

  for (n = 0; n < nbondlist; n++) {
    i1 = bondlist[n][0];
    i2 = bondlist[n][1];
    type = bondlist[n][2];

    delx = x[i1][0] - x[i2][0];
    dely = x[i1][1] - x[i2][1];
    delz = x[i1][2] - x[i2][2];

    rsq = delx*delx + dely*dely + delz*delz;
    r = sqrt(rsq);
   // dr = r - r0[type];

    rk = k[type];
    theta = beta[type];

    // force & energy
```

# Implementation of active force

```
bond_active.cpp       Mon Mar 01 14:33:57 2021       3
     if (r > 0.0) fbond = rk/r;
     else fbond = 0.0;

    // if (eflag) ebond = rk*dr;

     // apply force to each of 2 atoms

    if (newton_bond || i1 < nlocal) {
       if (theta == 1 ){
       f[i1][0] += delx*fbond;
       f[i1][1] += dely*fbond;
       f[i1][2] += delz*fbond;}
       else {
       f[i1][0] += r*cos((theta*3.14)/180)*fbond;
       f[i1][1] += r*sin((theta*3.14)/180)*fbond;
       f[i1][2] += delz*fbond;
       }
     }
/*
     if (newton_bond || i2 < nlocal) {
       f[i2][0] = delx*fbond;
       f[i2][1] = dely*fbond;
       f[i2][2] = delz*fbond;
     }*/

     if (evflag) ev_tally(i1,i2,nlocal,newton_bond,ebon
d,fbond,delx,dely,delz);
   }
}

/* ----------------------------------------------------
------------------- */

void BondActive::allocate()
{
  allocated = 1;
  int n = atom->nbondtypes;

  memory->create(k,n+1,"bond:k");
  memory->create(r0,n+1,"bond:r0");
  memory->create(beta,n+1,"bond:r0");

  memory->create(setflag,n+1,"bond:setflag");
  for (int i = 1; i <= n; i++) setflag[i] = 0;
}

/* ----------------------------------------------------
```

# Implementation of active force

```
bond_active.cpp       Mon Mar 01 14:33:57 2021       4
------------------
   set coeffs for one or more types
------------------------------------------------------
------------------ */

void BondActive::coeff(int narg, char **arg)
{
  if (narg != 4) error->all(FLERR,"Incorrect args for
bond coefficients");
  if (!allocated) allocate();

  int ilo,ihi;
  utils::bounds(FLERR,arg[0],1,atom->nbondtypes,ilo,ih
i,error);

  double k_one = utils::numeric(FLERR,arg[1],false,lmp
);
  double r0_one = utils::numeric(FLERR,arg[2],false,lm
p);
  double beta_one = utils::numeric(FLERR,arg[3],false,
lmp);


  //force->bounds(FLERR,arg[0],atom->nbondtypes,ilo,ih
i);

  //double k_one = force->numeric(FLERR,arg[1]);
  //double r0_one = force->numeric(FLERR,arg[2]);

  int count = 0;
  for (int i = ilo; i <= ihi; i++) {
    k[i] = k_one;
    r0[i] = r0_one;
    beta[i] = beta_one;
    setflag[i] = 1;
    count++;
  }

  if (count == 0) error->all(FLERR,"Incorrect args for
 bond coefficients");
}

/* ------------------------------------------------------
------------------
   return an equilbrium bond length
------------------------------------------------------
------------------ */
```

```
bond_active.cpp       Mon Mar 01 14:33:57 2021       5

double BondActive::equilibrium_distance(int i)
{
  return r0[i];
}

/* ----------------------------------------------------
------------------
   proc 0 writes out coeffs to restart file
----------------------------------------------------
------------------ */

void BondActive::write_restart(FILE *fp)
{
  fwrite(&k[1],sizeof(double),atom->nbondtypes,fp);
  fwrite(&r0[1],sizeof(double),atom->nbondtypes,fp);
  fwrite(&beta[1],sizeof(double),atom->nbondtypes,fp);
}

/* ----------------------------------------------------
------------------
   proc 0 reads coeffs from restart file, bcasts them
----------------------------------------------------
------------------ */

void BondActive::read_restart(FILE *fp)
{
  allocate();

  if (comm->me == 0) {
    fread(&k[1],sizeof(double),atom->nbondtypes,fp);
    fread(&r0[1],sizeof(double),atom->nbondtypes,fp);
    fread(&beta[1],sizeof(double),atom->nbondtypes,fp)
;
  }
  MPI_Bcast(&k[1],atom->nbondtypes,MPI_DOUBLE,0,world)
;
  MPI_Bcast(&r0[1],atom->nbondtypes,MPI_DOUBLE,0,world
);
  MPI_Bcast(&beta[1],atom->nbondtypes,MPI_DOUBLE,0,wor
ld);

  for (int i = 1; i <= atom->nbondtypes; i++) setflag[
i] = 1;
}

/* ----------------------------------------------------
```

# Implementation of active force

```
bond_active.cpp       Mon Mar 01 14:33:57 2021       6
-------------------
   proc 0 writes to data file
-------------------------------------------------------
------------------- */

void BondActive::write_data(FILE *fp)
{
  for (int i = 1; i <= atom->nbondtypes; i++)
    fprintf(fp,"%d %g %g %g\n",i,k[i],r0[i],beta[i]);
}

/* -------------------------------------------------
------------------- */

double BondActive::single(int type, double rsq, int i,
 int j,
                          double &fforce)
{
  double r = sqrt(rsq);
  double dr = r - r0[type];
  double rk = k[type] * dr;
  fforce = 0;
  if (r > 0.0) fforce = -2.0*rk/r;
  return rk*dr;
}

/* -------------------------------------------------
-------------------
    Return ptr to internal members upon request.
-------------------------------------------------------
------------------ */
void *BondActive::extract( char *str, int &dim )
{
  dim = 1;
  if( strcmp(str,"kappa")==0) return (void*) k;
  if( strcmp(str,"r0")==0) return (void*) r0;
  if( strcmp(str,"beta")==0) return (void*) beta;
  return NULL;
}
```

## Implementation of elasto-plastic substrate

```
fix_spring_self.cpp       Sat Feb 27 00:25:22 2021       1
/* ----------------------------------------------------
------------------
   LAMMPS - Large-scale Atomic/Molecular Massively Par
allel Simulator
   http://lammps.sandia.gov, Sandia National Laborator
ies
   Steve Plimpton, sjplimp@sandia.gov

   Copyright (2003) Sandia Corporation.  Under the ter
ms of Contract
   DE-AC04-94AL85000 with Sandia Corporation, the U.S.
 Government retains
   certain rights in this software.  This software is
distributed under
   the GNU General Public License.

   See the README file in the top-level LAMMPS directo
ry.
----------------------------------------------------
------------------ */

/* ----------------------------------------------------
------------------
   Contributing author: Naveen Michaud-Agrawal (Johns
Hopkins University)
----------------------------------------------------
------------------ */

#include <cstdlib>
#include <cstring>
#include "fix_spring_self.h"
#include "atom.h"
#include "update.h"
#include "domain.h"
#include "respa.h"
#include "memory.h"
#include "error.h"
#include "force.h"

using namespace LAMMPS_NS;
using namespace FixConst;

/* ----------------------------------------------------
------------------ */

FixSpringSelf::FixSpringSelf(LAMMPS *lmp, int narg, ch
ar **arg) :
```

```
fix_spring_self.cpp       Sat Feb 27 00:25:22 2021       2
  Fix(lmp, narg, arg),
  xoriginal(NULL)
{
  if ((narg < 5) || (narg > 6))
    error->all(FLERR,"Illegal fix spring/self command"
);

  restart_peratom = 1;
  scalar_flag = 1;
  global_freq = 1;
  extscalar = 1;
  respa_level_support = 1;

  k = force->numeric(FLERR,arg[3]);
  Ecut = force->numeric(FLERR,arg[4]);
  if (k <= 0.0) error->all(FLERR,"Illegal fix spring/s
elf command");

  xflag = yflag = zflag = 1;

  if (narg == 6) {
    if (strcmp(arg[5],"xyz") == 0) {
      xflag = yflag = zflag = 1;
    } else if (strcmp(arg[5],"xy") == 0) {
      zflag = 0;
    } else if (strcmp(arg[5],"xz") == 0) {
      yflag = 0;
    } else if (strcmp(arg[5],"yz") == 0) {
      xflag = 0;
    } else if (strcmp(arg[5],"x") == 0) {
      yflag = zflag = 0;
    } else if (strcmp(arg[5],"y") == 0) {
      xflag = zflag = 0;
    } else if (strcmp(arg[5],"z") == 0) {
      xflag = yflag = 0;
    } else error->all(FLERR,"Illegal fix spring/self c
ommand");
  }

  // perform initial allocation of atom-based array
  // register with Atom class

  xoriginal = NULL;
  grow_arrays(atom->nmax);
  atom->add_callback(0);
  atom->add_callback(1);
```

```
fix_spring_self.cpp       Sat Feb 27 00:25:22 2021       3
  // xoriginal = initial unwrapped positions of atoms

  double **x = atom->x;
  int *mask = atom->mask;
  imageint *image = atom->image;
  int nlocal = atom->nlocal;

  for (int i = 0; i < nlocal; i++) {
    if (mask[i] & groupbit) domain->unmap(x[i],image[i
],xoriginal[i]);
    else xoriginal[i][0] = xoriginal[i][1] = xoriginal
[i][2] = 0.0;
  }

  espring = 0.0;

}

/* ----------------------------------------------------
------------------- */

FixSpringSelf::~FixSpringSelf()
{
  // unregister callbacks to this fix from Atom class

  atom->delete_callback(id,0);
  atom->delete_callback(id,1);

  // delete locally stored array

  memory->destroy(xoriginal);
}

/* ----------------------------------------------------
------------------- */

int FixSpringSelf::setmask()
{
  int mask = 0;
  mask |= POST_FORCE;
  mask |= THERMO_ENERGY;
  mask |= POST_FORCE_RESPA;
  mask |= MIN_POST_FORCE;
  return mask;
}

/* ----------------------------------------------------
```

```
fix_spring_self.cpp       Sat Feb 27 00:25:22 2021       4
------------------- */

void FixSpringSelf::init()
{
  if (strstr(update->integrate_style,"respa")) {
    ilevel_respa = ((Respa *) update->integrate)->nlev
els-1;
    if (respa_level >= 0) ilevel_respa = MIN(respa_lev
el,ilevel_respa);
  }
}

/* ----------------------------------------------------
------------------- */

void FixSpringSelf::setup(int vflag)
{
  if (strstr(update->integrate_style,"verlet"))
    post_force(vflag);
  else {
    ((Respa *) update->integrate)->copy_flevel_f(ileve
l_respa);
    post_force_respa(vflag,ilevel_respa,0);
    ((Respa *) update->integrate)->copy_f_flevel(ileve
l_respa);
  }
}

/* ----------------------------------------------------
------------------- */

void FixSpringSelf::min_setup(int vflag)
{
  post_force(vflag);
}

/* ----------------------------------------------------
------------------- */

void FixSpringSelf::post_force(int vflag)
{
  double **x = atom->x;
  double **f = atom->f;
  int *mask = atom->mask;
  imageint *image = atom->image;
  int nlocal = atom->nlocal;
```

```
fix_spring_self.cpp       Sat Feb 27 00:25:22 2021       5
  double dx,dy,dz;
  double unwrap[3];

  espring = 0.0;
  foriginal[0] = foriginal[1] = foriginal[2] = 0.0;
  force_flag = 0;

  for (int i = 0; i < nlocal; i++)
    if (mask[i] & groupbit) {
      domain->unmap(x[i],image[i],unwrap);
      dx = unwrap[0] - xoriginal[i][0];
      dy = unwrap[1] - xoriginal[i][1];
      dz = unwrap[2] - xoriginal[i][2];
    //  if (!xflag) dx <= 0.707;
      //if (!yflag) dy <= 0.707;
      //if (!zflag) dz <= 0.707;
      double dist = dx*dx + dy*dy + dz*dz ;
      if (dist <=Ecut) {
      f[i][0] -= k*dx;
      f[i][1] -= k*dy;
      f[i][2] -= k*dz;
      espring += k * (dx*dx + dy*dy + dz*dz);}
      else {
      espring+=0.0;
      }
    }

  espring *= 0.5;

}

/* ----------------------------------------------------
------------------- */

void FixSpringSelf::post_force_respa(int vflag, int il
evel, int iloop)
{
  if (ilevel == ilevel_respa) post_force(vflag);
}

/* ----------------------------------------------------
------------------- */

void FixSpringSelf::min_post_force(int vflag)
{
  post_force(vflag);
}
```

## Implementation of elasto-plastic substrate

```
fix_spring_self.cpp       Sat Feb 27 00:25:22 2021       6

/* ----------------------------------------------------
------------------
   energy of stretched springs
----------------------------------------------------
------------------ */
double FixSpringSelf::compute_scalar()
{
  double all;
  MPI_Allreduce(&espring,&all,1,MPI_DOUBLE,MPI_SUM,wor
ld);
  if (force_flag == 0) {
    MPI_Allreduce(foriginal,foriginal_all,3,MPI_DOUBLE
,MPI_SUM,world);
    force_flag = 1;
  }
 return all;
}

/* ----------------------------------------------------
------------------
   memory usage of local atom-based array
----------------------------------------------------
------------------ */

double FixSpringSelf::memory_usage()
{
  double bytes = atom->nmax*3 * sizeof(double);
  return bytes;
}

/* ----------------------------------------------------
------------------
   allocate atom-based array
----------------------------------------------------
------------------ */

void FixSpringSelf::grow_arrays(int nmax)
{
  memory->grow(xoriginal,nmax,3,"fix_spring/self:xorig
inal");
}

/* ----------------------------------------------------
------------------
   copy values within local atom-based array
----------------------------------------------------
```

```
fix_spring_self.cpp       Sat Feb 27 00:25:22 2021       7
------------------- */

void FixSpringSelf::copy_arrays(int i, int j, int delf
lag)
{
  xoriginal[j][0] = xoriginal[i][0];
  xoriginal[j][1] = xoriginal[i][1];
  xoriginal[j][2] = xoriginal[i][2];
}

/* ----------------------------------------------------
------------------
   pack values in local atom-based array for exchange
with another proc
----------------------------------------------------
------------------- */

int FixSpringSelf::pack_exchange(int i, double *buf)
{
  buf[0] = xoriginal[i][0];
  buf[1] = xoriginal[i][1];
  buf[2] = xoriginal[i][2];
  return 3;
}

/* ----------------------------------------------------
------------------
   unpack values in local atom-based array from exchan
ge with another proc
----------------------------------------------------
------------------- */

int FixSpringSelf::unpack_exchange(int nlocal, double
*buf)
{
  xoriginal[nlocal][0] = buf[0];
  xoriginal[nlocal][1] = buf[1];
  xoriginal[nlocal][2] = buf[2];
  return 3;
}

/* ----------------------------------------------------
------------------
   pack values in local atom-based arrays for restart
file
----------------------------------------------------
------------------- */
```

```
fix_spring_self.cpp       Sat Feb 27 00:25:22 2021       8

int FixSpringSelf::pack_restart(int i, double *buf)
{
  buf[0] = 4;
  buf[1] = xoriginal[i][0];
  buf[2] = xoriginal[i][1];
  buf[3] = xoriginal[i][2];
  return 4;
}

/* ----------------------------------------------------
------------------
   unpack values from atom->extra array to restart the
 fix
----------------------------------------------------
------------------ */

void FixSpringSelf::unpack_restart(int nlocal, int nth
)
{
  double **extra = atom->extra;

  // skip to Nth set of extra values

  int m = 0;
  for (int i = 0; i < nth; i++) m += static_cast<int>
(extra[nlocal][m]);
  m++;

  xoriginal[nlocal][0] = extra[nlocal][m++];
  xoriginal[nlocal][1] = extra[nlocal][m++];
  xoriginal[nlocal][2] = extra[nlocal][m++];
}

/* ----------------------------------------------------
------------------
   maxsize of any atom's restart data
----------------------------------------------------
------------------ */

int FixSpringSelf::maxsize_restart()
{
  return 4;
}

/* ----------------------------------------------------
------------------
```

```
fix_spring_self.cpp       Sat Feb 27 00:25:22 2021       9
   size of atom nlocal's restart data
-------------------------------------------------------
------------------- */

int FixSpringSelf::size_restart(int nlocal)
{
  return 4;
}
```

## Implementation of SSLJ potential

```
pair_lj_cut.cpp       Thu Jul 01 23:35:01 2021       1
/* ----------------------------------------------------
-------------------
   LAMMPS - Large-scale Atomic/Molecular Massively Par
allel Simulator
   http://lammps.sandia.gov, Sandia National Laborator
ies
   Steve Plimpton, sjplimp@sandia.gov

   Copyright (2003) Sandia Corporation.  Under the ter
ms of Contract
   DE-AC04-94AL85000 with Sandia Corporation, the U.S.
 Government retains
   certain rights in this software.  This software is
distributed under
   the GNU General Public License.

   See the README file in the top-level LAMMPS directo
ry.
----------------------------------------------------
------------------- */

/* ----------------------------------------------------
-------------------
   Contributing author: Paul Crozier (SNL)
----------------------------------------------------
------------------- */

#include <math.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "pair_lj_cut.h"
#include "atom.h"
#include "comm.h"
#include "force.h"
#include "neighbor.h"
#include "neigh_list.h"
#include "neigh_request.h"
#include "update.h"
#include "integrate.h"
#include "respa.h"
#include "math_const.h"
#include "memory.h"
#include "error.h"

using namespace LAMMPS_NS;
using namespace MathConst;
```

# Implementation of SSLJ potential

```
pair_lj_cut.cpp       Thu Jul 01 23:35:01 2021       2

/* ----------------------------------------------------
------------------- */

PairLJCut::PairLJCut(LAMMPS *lmp) : Pair(lmp)
{
  respa_enable = 1;
  writedata = 1;
}

/* ----------------------------------------------------
------------------- */

PairLJCut::~PairLJCut()
{
  if (allocated) {
    memory->destroy(setflag);
    memory->destroy(cutsq);

    memory->destroy(cut);
    memory->destroy(epsilon);
    memory->destroy(sigma);
    memory->destroy(lj1);
    memory->destroy(lj2);
    memory->destroy(lj3);
    memory->destroy(lj4);
    memory->destroy(offset);
  }
}

/* ----------------------------------------------------
------------------- */

void PairLJCut::compute(int eflag, int vflag)
{
  int i,j,ii,jj,inum,jnum,itype,jtype;
  double xtmp,ytmp,ztmp,delx,dely,delz,evdwl,fpair;
  double alpha,rsq,r2inv,r6inv,forcelj,factor_lj;
  int *ilist,*jlist,*numneigh,**firstneigh;

  evdwl = 0.0;
  if (eflag || vflag) ev_setup(eflag,vflag);
  else evflag = vflag_fdotr = 0;

  double **x = atom->x;
  double **f = atom->f;
  int *type = atom->type;
```

## Implementation of SSLJ potential

```
pair_lj_cut.cpp       Thu Jul 01 23:35:01 2021       3
  int nlocal = atom->nlocal;
  double *special_lj = force->special_lj;
  int newton_pair = force->newton_pair;

  inum = list->inum;
  ilist = list->ilist;
  numneigh = list->numneigh;
  firstneigh = list->firstneigh;

  // loop over neighbors of my atoms

  for (ii = 0; ii < inum; ii++) {
    i = ilist[ii];
    xtmp = x[i][0];
    ytmp = x[i][1];
    ztmp = x[i][2];
    itype = type[i];
    jlist = firstneigh[i];
    jnum = numneigh[i];

    for (jj = 0; jj < jnum; jj++) {
      j = jlist[jj];
      factor_lj = special_lj[sbmask(j)];
      j &= NEIGHMASK;

      delx = xtmp - x[j][0];
      dely = ytmp - x[j][1];
      delz = ztmp - x[j][2];
      rsq = alpha*alpha + delx*delx + dely*dely + delz
*delz;
      jtype = type[j];

      if (rsq < cutsq[itype][jtype]) {
        r2inv = 1.0/rsq;
        r6inv = r2inv*r2inv*r2inv;
        forcelj = r6inv * (lj1[itype][jtype]*r6inv - l
j2[itype][jtype]);
        fpair = factor_lj*forcelj*r2inv;

        f[i][0] += delx*fpair;
        f[i][1] += dely*fpair;
        f[i][2] += delz*fpair;
        if (newton_pair || j < nlocal) {
          f[j][0] -= delx*fpair;
          f[j][1] -= dely*fpair;
          f[j][2] -= delz*fpair;
        }
```

## Implementation of SSLJ potential

```
pair_lj_cut.cpp       Thu Jul 01 23:35:01 2021       4

        if (eflag) {
          evdwl = r6inv*(lj3[itype][jtype]*r6inv-lj4[i
type][jtype]) -
            offset[itype][jtype];
          evdwl *= factor_lj;
        }

        if (evflag) ev_tally(i,j,nlocal,newton_pair,
                             evdwl,0.0,fpair,delx,dely
,delz);
      }
    }
  }

  if (vflag_fdotr) virial_fdotr_compute();
}

/* ----------------------------------------------------
------------------- */

void PairLJCut::compute_inner()
{
  int i,j,ii,jj,inum,jnum,itype,jtype;
  double xtmp,ytmp,ztmp,delx,dely,delz,fpair;
  double alpha,rsq,r2inv,r6inv,forcelj,factor_lj,rsw;
  int *ilist,*jlist,*numneigh,**firstneigh;

  double **x = atom->x;
  double **f = atom->f;
  int *type = atom->type;
  int nlocal = atom->nlocal;
  double *special_lj = force->special_lj;
  int newton_pair = force->newton_pair;

  inum = list->inum_inner;
  ilist = list->ilist_inner;
  numneigh = list->numneigh_inner;
  firstneigh = list->firstneigh_inner;

  double cut_out_on = cut_respa[0];
  double cut_out_off = cut_respa[1];

  double cut_out_diff = cut_out_off - cut_out_on;
  double cut_out_on_sq = cut_out_on*cut_out_on;
  double cut_out_off_sq = cut_out_off*cut_out_off;
```

## Implementation of SSLJ potential

```
pair_lj_cut.cpp       Thu Jul 01 23:35:01 2021       5
  // loop over neighbors of my atoms

  for (ii = 0; ii < inum; ii++) {
    i = ilist[ii];
    xtmp = x[i][0];
    ytmp = x[i][1];
    ztmp = x[i][2];
    itype = type[i];
    jlist = firstneigh[i];
    jnum = numneigh[i];

    for (jj = 0; jj < jnum; jj++) {
      j = jlist[jj];
      factor_lj = special_lj[sbmask(j)];
      j &= NEIGHMASK;

      delx = xtmp - x[j][0];
      dely = ytmp - x[j][1];
      delz = ztmp - x[j][2];
      rsq = alpha*alpha + delx*delx + dely*dely + delz
*delz;

      if (rsq < cut_out_off_sq) {
        r2inv = 1.0/rsq;
        r6inv = r2inv*r2inv*r2inv;
        jtype = type[j];
        forcelj = r6inv * (lj1[itype][jtype]*r6inv - l
j2[itype][jtype]);
        fpair = factor_lj*forcelj*r2inv;
        if (rsq > cut_out_on_sq) {
          rsw = (sqrt(rsq) - cut_out_on)/cut_out_diff;
          fpair *= 1.0 - rsw*rsw*(3.0 - 2.0*rsw);
        }

        f[i][0] += delx*fpair;
        f[i][1] += dely*fpair;
        f[i][2] += delz*fpair;
        if (newton_pair || j < nlocal) {
          f[j][0] -= delx*fpair;
          f[j][1] -= dely*fpair;
          f[j][2] -= delz*fpair;
        }
      }
    }
  }
}
```

## Implementation of SSLJ potential

```
pair_lj_cut.cpp       Thu Jul 01 23:35:01 2021       6
/* ----------------------------------------------------------------------
------------------- */

void PairLJCut::compute_middle()
{
  int i,j,ii,jj,inum,jnum,itype,jtype;
  double xtmp,ytmp,ztmp,delx,dely,delz,fpair;
  double alpha,rsq,r2inv,r6inv,forcelj,factor_lj,rsw;
  int *ilist,*jlist,*numneigh,**firstneigh;

  double **x = atom->x;
  double **f = atom->f;
  int *type = atom->type;
  int nlocal = atom->nlocal;
  double *special_lj = force->special_lj;
  int newton_pair = force->newton_pair;

  inum = list->inum_middle;
  ilist = list->ilist_middle;
  numneigh = list->numneigh_middle;
  firstneigh = list->firstneigh_middle;

  double cut_in_off = cut_respa[0];
  double cut_in_on = cut_respa[1];
  double cut_out_on = cut_respa[2];
  double cut_out_off = cut_respa[3];

  double cut_in_diff = cut_in_on - cut_in_off;
  double cut_out_diff = cut_out_off - cut_out_on;
  double cut_in_off_sq = cut_in_off*cut_in_off;
  double cut_in_on_sq = cut_in_on*cut_in_on;
  double cut_out_on_sq = cut_out_on*cut_out_on;
  double cut_out_off_sq = cut_out_off*cut_out_off;

  // loop over neighbors of my atoms

  for (ii = 0; ii < inum; ii++) {
    i = ilist[ii];
    xtmp = x[i][0];
    ytmp = x[i][1];
    ztmp = x[i][2];
    itype = type[i];
    jlist = firstneigh[i];
    jnum = numneigh[i];

    for (jj = 0; jj < jnum; jj++) {
      j = jlist[jj];
```

# Implementation of SSLJ potential

```
pair_lj_cut.cpp       Thu Jul 01 23:35:01 2021       7
      factor_lj = special_lj[sbmask(j)];
      j &= NEIGHMASK;

      delx = xtmp - x[j][0];
      dely = ytmp - x[j][1];
      delz = ztmp - x[j][2];
      rsq = alpha*alpha +delx*delx + dely*dely + delz*
delz;

      if (rsq < cut_out_off_sq && rsq > cut_in_off_sq)
 {
        r2inv = 1.0/rsq;
        r6inv = r2inv*r2inv*r2inv;
        jtype = type[j];
        forcelj = r6inv * (lj1[itype][jtype]*r6inv - l
j2[itype][jtype]);
        fpair = factor_lj*forcelj*r2inv;
        if (rsq < cut_in_on_sq) {
          rsw = (sqrt(rsq) - cut_in_off)/cut_in_diff;
          fpair *= rsw*rsw*(3.0 - 2.0*rsw);
        }
        if (rsq > cut_out_on_sq) {
          rsw = (sqrt(rsq) - cut_out_on)/cut_out_diff;
          fpair *= 1.0 + rsw*rsw*(2.0*rsw - 3.0);
        }

        f[i][0] += delx*fpair;
        f[i][1] += dely*fpair;
        f[i][2] += delz*fpair;
        if (newton_pair || j < nlocal) {
          f[j][0] -= delx*fpair;
          f[j][1] -= dely*fpair;
          f[j][2] -= delz*fpair;
        }
      }
    }
  }
}

/* ----------------------------------------------------
------------------- */

void PairLJCut::compute_outer(int eflag, int vflag)
{
  int i,j,ii,jj,inum,jnum,itype,jtype;
  double xtmp,ytmp,ztmp,delx,dely,delz,evdwl,fpair;
  double alpha,rsq,r2inv,r6inv,forcelj,factor_lj,rsw;
```

## Implementation of SSLJ potential

```
pair_lj_cut.cpp       Thu Jul 01 23:35:01 2021       8
  int *ilist,*jlist,*numneigh,**firstneigh;

  evdwl = 0.0;
  if (eflag || vflag) ev_setup(eflag,vflag);
  else evflag = 0;

  double **x = atom->x;
  double **f = atom->f;
  int *type = atom->type;
  int nlocal = atom->nlocal;
  double *special_lj = force->special_lj;
  int newton_pair = force->newton_pair;

  inum = list->inum;
  ilist = list->ilist;
  numneigh = list->numneigh;
  firstneigh = list->firstneigh;

  double cut_in_off = cut_respa[2];
  double cut_in_on = cut_respa[3];

  double cut_in_diff = cut_in_on - cut_in_off;
  double cut_in_off_sq = cut_in_off*cut_in_off;
  double cut_in_on_sq = cut_in_on*cut_in_on;

  // loop over neighbors of my atoms

  for (ii = 0; ii < inum; ii++) {
    i = ilist[ii];
    xtmp = x[i][0];
    ytmp = x[i][1];
    ztmp = x[i][2];
    itype = type[i];
    jlist = firstneigh[i];
    jnum = numneigh[i];

    for (jj = 0; jj < jnum; jj++) {
      j = jlist[jj];
      factor_lj = special_lj[sbmask(j)];
      j &= NEIGHMASK;

      delx = xtmp - x[j][0];
      dely = ytmp - x[j][1];
      delz = ztmp - x[j][2];
      rsq =alpha*alpha + delx*delx + dely*dely + delz*
delz;
      jtype = type[j];
```

## Implementation of SSLJ potential

```
pair_lj_cut.cpp       Thu Jul 01 23:35:01 2021       9

      if (rsq < cutsq[itype][jtype]) {
        if (rsq > cut_in_off_sq) {
          r2inv = 1.0/rsq;
          r6inv = r2inv*r2inv*r2inv;
          forcelj = r6inv * (lj1[itype][jtype]*r6inv -
 lj2[itype][jtype]);
          fpair = factor_lj*forcelj*r2inv;
          if (rsq < cut_in_on_sq) {
            rsw = (sqrt(rsq) - cut_in_off)/cut_in_diff
;
            fpair *= rsw*rsw*(3.0 - 2.0*rsw);
          }

          f[i][0] += delx*fpair;
          f[i][1] += dely*fpair;
          f[i][2] += delz*fpair;
          if (newton_pair || j < nlocal) {
            f[j][0] -= delx*fpair;
            f[j][1] -= dely*fpair;
            f[j][2] -= delz*fpair;
          }
        }

        if (eflag) {
          r2inv = 1.0/rsq;
          r6inv = r2inv*r2inv*r2inv;
          evdwl = r6inv*(lj3[itype][jtype]*r6inv-lj4[i
type][jtype]) -
            offset[itype][jtype];
          evdwl *= factor_lj;
        }

        if (vflag) {
          if (rsq <= cut_in_off_sq) {
            r2inv = 1.0/rsq;
            r6inv = r2inv*r2inv*r2inv;
            forcelj = r6inv * (lj1[itype][jtype]*r6inv
 - lj2[itype][jtype]);
            fpair = factor_lj*forcelj*r2inv;
          } else if (rsq < cut_in_on_sq)
            fpair = factor_lj*forcelj*r2inv;
        }

        if (evflag) ev_tally(i,j,nlocal,newton_pair,
                             evdwl,0.0,fpair,delx,dely
,delz);
```

## Implementation of SSLJ potential

```
pair_lj_cut.cpp       Thu Jul 01 23:35:01 2021       10
      }
    }
  }
}

/* ----------------------------------------------------
------------------
   allocate all arrays
----------------------------------------------------
------------------ */

void PairLJCut::allocate()
{
  allocated = 1;
  int n = atom->ntypes;

  memory->create(setflag,n+1,n+1,"pair:setflag");
  for (int i = 1; i <= n; i++)
    for (int j = i; j <= n; j++)
      setflag[i][j] = 0;

  memory->create(cutsq,n+1,n+1,"pair:cutsq");

  memory->create(cut,n+1,n+1,"pair:cut");
  memory->create(epsilon,n+1,n+1,"pair:epsilon");
  memory->create(sigma,n+1,n+1,"pair:sigma");
  memory->create(lj1,n+1,n+1,"pair:lj1");
  memory->create(lj2,n+1,n+1,"pair:lj2");
  memory->create(lj3,n+1,n+1,"pair:lj3");
  memory->create(lj4,n+1,n+1,"pair:lj4");
  memory->create(offset,n+1,n+1,"pair:offset");
}

/* ----------------------------------------------------
------------------
   global settings
----------------------------------------------------
------------------ */

void PairLJCut::settings(int narg, char **arg)
{
  if (narg != 1) error->all(FLERR,"Illegal pair_style
command");

  cut_global = utils::numeric(FLERR,arg[0],false,lmp);

  // reset cutoffs that have been explicitly set
```

## Implementation of SSLJ potential

```
pair_lj_cut.cpp       Thu Jul 01 23:35:01 2021       11

  if (allocated) {
    int i,j;
    for (i = 1; i <= atom->ntypes; i++)
      for (j = i; j <= atom->ntypes; j++)
        if (setflag[i][j]) cut[i][j] = cut_global;
  }
}

/* ----------------------------------------------------
------------------
   set coeffs for one or more type pairs
----------------------------------------------------
------------------ */

void PairLJCut::coeff(int narg, char **arg)
{
  if (narg < 5 || narg > 6)
    error->all(FLERR,"Incorrect args for pair coeffici
ents");
  if (!allocated) allocate();

  int ilo,ihi,jlo,jhi;
  utils::bounds(FLERR,arg[0],1,atom->ntypes,ilo,ihi,er
ror);
  utils::bounds(FLERR,arg[1],1,atom->ntypes,jlo,jhi,er
ror);

  double epsilon_one = utils::numeric(FLERR,arg[2],fal
se,lmp);
  double sigma_one = utils::numeric(FLERR,arg[3],false
,lmp);
  double alpha = utils::numeric(FLERR,arg[4],false,lmp
);
  double cut_one = cut_global;
  if (narg == 6) cut_one =  utils::numeric(FLERR,arg[5
],false,lmp);

  int count = 0;
  for (int i = ilo; i <= ihi; i++) {
    for (int j = MAX(jlo,i); j <= jhi; j++) {
      epsilon[i][j] = epsilon_one;
      sigma[i][j] = sigma_one;
      cut[i][j] = cut_one;
      setflag[i][j] = 1;
      count++;
    }
```

```
pair_lj_cut.cpp       Thu Jul 01 23:35:01 2021       12
  }

  if (count == 0) error->all(FLERR,"Incorrect args for
 pair coefficients");
}

/* ----------------------------------------------------
------------------
   init specific to this pair style
----------------------------------------------------
------------------ */

void PairLJCut::init_style()
{
  // request regular or rRESPA neighbor list

  int irequest;
  int respa = 0;

  if (update->whichflag == 1 && strstr(update->integra
te_style,"respa")) {
    if (((Respa *) update->integrate)->level_inner >=
0) respa = 1;
    if (((Respa *) update->integrate)->level_middle >=
 0) respa = 2;
  }

  irequest = neighbor->request(this,instance_me);

  if (respa >= 1) {
    neighbor->requests[irequest]->respaouter = 1;
    neighbor->requests[irequest]->respainner = 1;
  }
  if (respa == 2) neighbor->requests[irequest]->respam
iddle = 1;

  // set rRESPA cutoffs

  if (strstr(update->integrate_style,"respa") &&
      ((Respa *) update->integrate)->level_inner >= 0)
    cut_respa = ((Respa *) update->integrate)->cutoff;
  else cut_respa = NULL;
}

/* ----------------------------------------------------
------------------
   init for one type pair i,j and corresponding j,i
```

## Implementation of SSLJ potential

```
pair_lj_cut.cpp       Thu Jul 01 23:35:01 2021       13
------------------------------------------------------
------------------- */

double PairLJCut::init_one(int i, int j)
{
  if (setflag[i][j] == 0) {
    epsilon[i][j] = mix_energy(epsilon[i][i],epsilon[j
][j],
                               sigma[i][i],sigma[j][j]
);
    sigma[i][j] = mix_distance(sigma[i][i],sigma[j][j]
);
    cut[i][j] = mix_distance(cut[i][i],cut[j][j]);
  }

  lj1[i][j] = 48.0 * epsilon[i][j] * pow(sigma[i][j],1
2.0);
  lj2[i][j] = 24.0 * epsilon[i][j] * pow(sigma[i][j],6
.0);
  lj3[i][j] = 4.0 * epsilon[i][j] * pow(sigma[i][j],12
.0);
  lj4[i][j] = 4.0 * epsilon[i][j] * pow(sigma[i][j],6.
0);

  if (offset_flag && (cut[i][j] > 0.0)) {
    double ratio = sigma[i][j] / cut[i][j];
    offset[i][j] = 4.0 * epsilon[i][j] * (pow(ratio,12
.0) - pow(ratio,6.0));
  } else offset[i][j] = 0.0;

  lj1[j][i] = lj1[i][j];
  lj2[j][i] = lj2[i][j];
  lj3[j][i] = lj3[i][j];
  lj4[j][i] = lj4[i][j];
  offset[j][i] = offset[i][j];

  // check interior rRESPA cutoff

  if (cut_respa && cut[i][j] < cut_respa[3])
    error->all(FLERR,"Pair cutoff < Respa interior cut
off");

  // compute I,J contribution to long-range tail corre
ction
  // count total # of atoms of type I and J via Allred
uce
```

## Implementation of SSLJ potential

```
pair_lj_cut.cpp       Thu Jul 01 23:35:01 2021       14
  if (tail_flag) {
    int *type = atom->type;
    int nlocal = atom->nlocal;

    double count[2],all[2];
    count[0] = count[1] = 0.0;
    for (int k = 0; k < nlocal; k++) {
      if (type[k] == i) count[0] += 1.0;
      if (type[k] == j) count[1] += 1.0;
    }
    MPI_Allreduce(count,all,2,MPI_DOUBLE,MPI_SUM,world
);

    double sig2 = sigma[i][j]*sigma[i][j];
    double sig6 = sig2*sig2*sig2;
    double rc3 = cut[i][j]*cut[i][j]*cut[i][j];
    double rc6 = rc3*rc3;
    double rc9 = rc3*rc6;
    etail_ij = 8.0*MY_PI*all[0]*all[1]*epsilon[i][j] *
      sig6 * (sig6 - 3.0*rc6) / (9.0*rc9);
    ptail_ij = 16.0*MY_PI*all[0]*all[1]*epsilon[i][j]
*
      sig6 * (2.0*sig6 - 3.0*rc6) / (9.0*rc9);
  }

  return cut[i][j];
}

/* ----------------------------------------------------
------------------
   proc 0 writes to restart file
----------------------------------------------------
------------------ */

void PairLJCut::write_restart(FILE *fp)
{
  write_restart_settings(fp);

  int i,j;
  for (i = 1; i <= atom->ntypes; i++)
    for (j = i; j <= atom->ntypes; j++) {
      fwrite(&setflag[i][j],sizeof(int),1,fp);
      if (setflag[i][j]) {
        fwrite(&epsilon[i][j],sizeof(double),1,fp);
        fwrite(&sigma[i][j],sizeof(double),1,fp);
        fwrite(&cut[i][j],sizeof(double),1,fp);
      }
```

```
pair_lj_cut.cpp       Thu Jul 01 23:35:01 2021       15
    }
}

/* ----------------------------------------------------
------------------
   proc 0 reads from restart file, bcasts
----------------------------------------------------
------------------ */

void PairLJCut::read_restart(FILE *fp)
{
  read_restart_settings(fp);
  allocate();

  int i,j;
  int me = comm->me;
  for (i = 1; i <= atom->ntypes; i++)
    for (j = i; j <= atom->ntypes; j++) {
      if (me == 0) fread(&setflag[i][j],sizeof(int),1,
fp);
      MPI_Bcast(&setflag[i][j],1,MPI_INT,0,world);
      if (setflag[i][j]) {
        if (me == 0) {
          fread(&epsilon[i][j],sizeof(double),1,fp);
          fread(&sigma[i][j],sizeof(double),1,fp);
          fread(&cut[i][j],sizeof(double),1,fp);
        }
        MPI_Bcast(&epsilon[i][j],1,MPI_DOUBLE,0,world)
;
        MPI_Bcast(&sigma[i][j],1,MPI_DOUBLE,0,world);
        MPI_Bcast(&cut[i][j],1,MPI_DOUBLE,0,world);
      }
    }
}

/* ----------------------------------------------------
------------------
   proc 0 writes to restart file
----------------------------------------------------
------------------ */

void PairLJCut::write_restart_settings(FILE *fp)
{
  fwrite(&cut_global,sizeof(double),1,fp);
  fwrite(&offset_flag,sizeof(int),1,fp);
  fwrite(&mix_flag,sizeof(int),1,fp);
  fwrite(&tail_flag,sizeof(int),1,fp);
```

```
pair_lj_cut.cpp       Thu Jul 01 23:35:01 2021       16
}

/* ----------------------------------------------------
------------------
   proc 0 reads from restart file, bcasts
----------------------------------------------------
------------------ */

void PairLJCut::read_restart_settings(FILE *fp)
{
  int me = comm->me;
  if (me == 0) {
    fread(&cut_global,sizeof(double),1,fp);
    fread(&offset_flag,sizeof(int),1,fp);
    fread(&mix_flag,sizeof(int),1,fp);
    fread(&tail_flag,sizeof(int),1,fp);
  }
  MPI_Bcast(&cut_global,1,MPI_DOUBLE,0,world);
  MPI_Bcast(&offset_flag,1,MPI_INT,0,world);
  MPI_Bcast(&mix_flag,1,MPI_INT,0,world);
  MPI_Bcast(&tail_flag,1,MPI_INT,0,world);
}

/* ----------------------------------------------------
------------------
   proc 0 writes to data file
----------------------------------------------------
------------------ */

void PairLJCut::write_data(FILE *fp)
{
  for (int i = 1; i <= atom->ntypes; i++)
    fprintf(fp,"%d %g %g\n",i,epsilon[i][i],sigma[i][i
]);
}

/* ----------------------------------------------------
------------------
   proc 0 writes all pairs to data file
----------------------------------------------------
------------------ */

void PairLJCut::write_data_all(FILE *fp)
{
  for (int i = 1; i <= atom->ntypes; i++)
    for (int j = i; j <= atom->ntypes; j++)
      fprintf(fp,"%d %d %g %g %g\n",i,j,epsilon[i][j],
```

# Implementation of SSLJ potential

```
pair_lj_cut.cpp       Thu Jul 01 23:35:01 2021       17
sigma[i][j],cut[i][j]);
}

/* ----------------------------------------------------
------------------- */

double PairLJCut::single(int i, int j, int itype, int
jtype, double rsq,
                         double factor_coul, double fa
ctor_lj,
                         double &fforce)
{
  double r2inv,r6inv,forcelj,philj;

  r2inv = 1.0/rsq;
  r6inv = r2inv*r2inv*r2inv;
  forcelj = r6inv * (lj1[itype][jtype]*r6inv - lj2[ity
pe][jtype]);
  fforce = factor_lj*forcelj*r2inv;

  philj = r6inv*(lj3[itype][jtype]*r6inv-lj4[itype][jt
ype]) -
    offset[itype][jtype];
  return factor_lj*philj;
}

/* ----------------------------------------------------
------------------- */

void *PairLJCut::extract(const char *str, int &dim)
{
  dim = 2;
  if (strcmp(str,"epsilon") == 0) return (void *) epsi
lon;
  if (strcmp(str,"sigma") == 0) return (void *) sigma;
  return NULL;
}
```