

ENS DE LYON



SAPIENZA
UNIVERSITÀ DI ROMA

AtoSIM Master of Science

École Normale Supérieure de Lyon
VU University Amsterdam
UVA University Amsterdam
Sapienza Università di Roma

Research Project 2014–2015

ENS Lyon
Alon Gal

Emergence of collective motion and pattern formation in flying spin models

Summary: *I will look at two simple continuous-time flying spin models: the continuous-time Vicsek model and the flying XY model. I will show that both these models exhibit a first-order phase transition near the onset of collective motion. I will also show that the Vicsek model has the same phenomenology as its well-studied discrete-time counterpart, while the flying XY model exhibits some novel behavior. In particular, the Vicsek model has a stable “traveling-band” configuration near the transition, which does not exist in the flying XY case. I will also plot a phase diagram for both models, and discuss its correspondence with existing mean-field theoretical predictions.*

Key words: Vicsek Model, flying XY model, active matter, collective motion, flocking, non-equilibrium phase transition.

Supervisor:

Denis Bartolo

ENS Lyon École Normale Supérieure de Lyon, Laboratoire de Physique

ENS de Lyon, UMR CNRS 5672, 46 allée d'Italie, F69007, Lyon,
FRANCE



In Atomic Scale Modelling of Physical, Chemical and Biomolecular Systems

Contents

1	Introduction	3
1.1	Motivation and background	3
1.2	The Vicsek model: first attempt at replicating collective motion	4
1.3	My project: continuous-time models for the emergence of collective motion	6
2	Methods	10
2.1	Simulation	10
2.2	Data analysis	17
2.3	Summary	20
3	Vicsek model	21
3.1	Observed phases	21
3.2	Nature of the transition	22
3.3	Phase diagram	25
3.4	Spatial properties of emergent structures	27
4	Flying XY	29
4.1	Observed phases	29
4.2	Nature of the transition	30
4.3	Phase diagram	32
4.4	Spatial properties of emergent structures	35
5	Discussion	39
5.1	Comparison of the discrete-time and continuous-time Vicsek models	39
5.2	Comparison between the flying XY and the continuous-time Vicsek model	39
5.3	Theoretical models for the transition surface	40
5.4	Order and characteristics of the phase transition	40
6	Perspective	41
6.1	Clustering behavior	41
6.2	Higher-dimensional systems	41
6.3	Correspondence between discrete- and continuous-time Vicsek	41
6.4	Time dynamics and coarsening	41
6.5	Theoretical work regarding the phase transition	42
7	Acknowledgments	42
A	Alignmnent potential	43
B	Vectorial versus angular noise	44
B.1	No noise	44
B.2	Angular Noise	45
B.3	Vectorial Noise	46

1 Introduction

1.1 Motivation and background

My research takes place in the field of physics which is called *Active Matter*, referring to the fact that the individual entities are *self-propelled*, or consume energy in order to move. This field takes inspiration from systems of living organisms on a large range of length scales: flocks of birds, herds of mammals, schools of fish schools, and swarms of insects, as well as bacterial suspensions[1] and molecular motor proteins[2].

One of the interesting facets of active systems is their tendency to form non-trivial spatial and temporal patterns. One may consider “fish balls” or “starling clouds”, but the same is also true on the microscopic level; bacterial colonies may grow into crystalline structures and biofilms can form themselves into honeycomb-shaped networks[3].

We are especially interested in the formation mechanism of such emergent spatial arrangements; these come about even in the absence of a clear biological motivation such as following a leader or seeking a mate, and also in the absence of strong geometrical constraints such as following a narrow trail.



School of organized in a ball¹



Flock of starlings²

Figure 1: Biological systems displaying collective motion.

Beyond the spatial arrangements, we are also interested in the *orientation* of the single moving elements (henceforth referred to as “particles”). A school of fish circling around a center, for example, has a zero net linear momentum, on average. However, the velocities of the fish are strongly correlated across short distances. A charging herd of buffalo, on the other hand, exhibits both correlation of local orientation and a non-zero average velocity.

It is believed that this complex emergent behavior can in fact result from a simple set of local interactions. This is due to the large amount of information which would have to be distributed extremely quickly in order to conceive of this motion as being the result of conscious planning, and the inability of any single member to perceive the entire flock. If such local interactions are indeed discovered that reproduce the relevant attributes of flocking motion in a satisfactory manner, we will be able to apply the well-established tools of statistical physics and the theory of dynamical systems to them, and develop a deeper understanding of their mechanism.

In my thesis I will examine two such models and analyze their emergent behaviors, in terms of spatial patterns and velocity distributions. The questions that will be posed are:

¹Unknown photographer, taken from <http://twistedsifter.com/2010/07/15-fascinating-schools-of-fish/>

²Christine Kvistad, Rome, taken from <https://studiostrange.wordpress.com/2012/12/08/bird-patterns/european-starling-flock/>

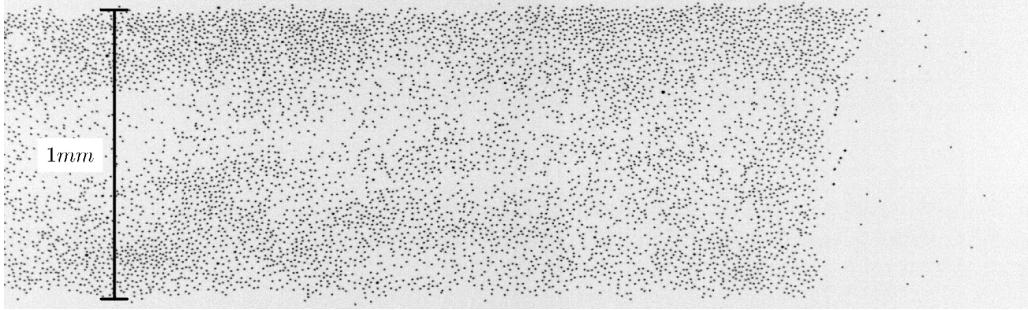


Figure 2: Micron sized rollers swarming in a microfluidic channel.[4]

1. What are the different stable phases of the systems?
2. Does the system undergo a phase transition? If so, of what order?
3. Can we draw the phase diagrams of the models in question?

1.2 The Vicsek model: first attempt at replicating collective motion

When constructing a model for collective motion, we generally include two main ingredients:

1. Short-range pairwise alignment
2. Rotational diffusion (angular Brownian noise)

Perhaps the simplest of these models, and certainly the most studied to date, is known as the Vicsek model[1], introduced by Vicsek et al. in 1995. This is an N particle discrete-time model that obeys the following rules:

1. All particles move at a constant speed, v_0 .
2. At every time step, each particle aligns its velocity with the direction of the mean velocity of its close neighbors.
3. After aligning, each particle makes a small random change (“mistake”) in its new velocity direction.

The magnitude of the mistake in alignment that the particles make, or the system “noise”, can be adjusted as a parameter. This scheme, minimalistic as it is, reproduces many of the desired characteristics of a collective motion model.

1.2.1 Mathematical formulation of the Vicsek model

Let us write out the model explicitly, and introduce some of the language we will use throughout this study. The following are the evolution equations for the original Vicsek model:

$$\mathbf{r}_i(t + \Delta t) = \mathbf{r}_i(t) + \mathbf{v}_i(t)\Delta t, \quad (1.1)$$

$$\theta_i(t + \Delta t) = <\theta_i(t)>_{R^*} + \xi_i(t), \quad (1.2)$$

where $\mathbf{r}_i(t)$ and $\mathbf{v}_i(t)$ correspond to the position and velocity of the i -th particle at time t , respectively, and Δt is a fixed parameter representing the time step size. $\theta_i(t)$ is the direction of motion of particle i at time t , such that: $\mathbf{v}_i(t) \equiv v_0 (\cos \theta_i(t), \sin \theta_i(t))$, where v_0 is fixed and represents the speed of the particles.

The dynamics are controlled by the two right-hand side terms in equation 1.2. The first, $\langle \theta_i(t) \rangle_{R^*}$, is an operator which gives the direction of the mean velocity of all particles within a radius R^* of particle i , including particle i itself. R^* is called the *radius of interaction*, and is generally assumed to be much smaller than the simulation “box” length. This interaction is illustrated in figure 3. If only the first term were present, all particles would be expected to perfectly align in the same direction, provided the density is high enough for them to encounter each other.

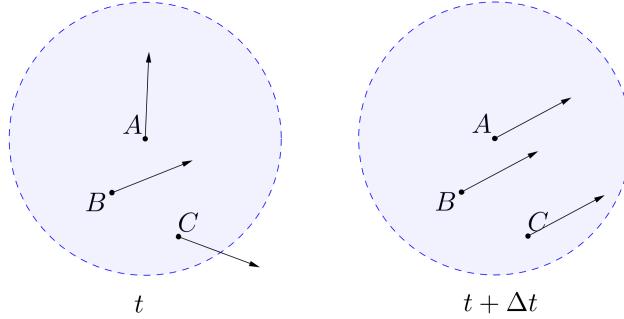


Figure 3: Vicsek alignment dynamics. Particle A ’s radius of interaction is represented by a blue circle. In the next time frame, all particles will align their velocities, assuming there are no additional particles in their vicinity.

The second term in the dynamics is $\xi_i(t)$, which is defined as a uniform random variable in the interval $[-\eta/2, \eta/2]$; more precisely, it is a set of independent random variables, one for each time step and particle. This term is given the name *noise*, since it leads to a decorrelation of the velocities. For a very high value of η we expect the alignment term to be negligible, and we will have a system of N independent (rotational) random walkers.

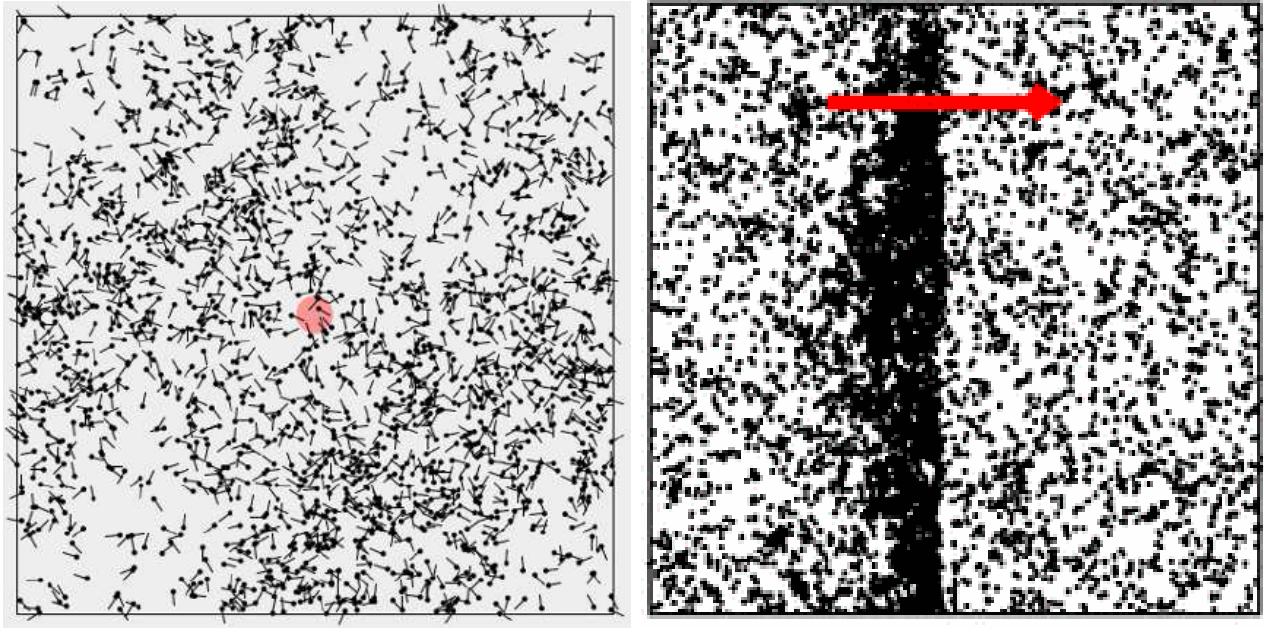
1.2.2 Known observations

The Vicsek model has been studied extensively by Chaté et al.[5], and has been shown to possess a rich phenomenology, of which I will give a brief overview. I will begin by surveying the various stable phases, starting from a high noise parameter to a low one.

For a very high noise setting, we get the expected *gas* configuration, where the system is homogenous and isotropic. At some noise value, local clusters are formed, but are not stable. The system remains homogenous and isotropic, but the local density begins to fluctuate. Reducing the noise parameter some more, we see coherently moving *bands* forming, which is perhaps the most non-trivial result of the Vicsek model. These so-called bands are comprised of a dense region of particles extending across the entire system length, moving in the same direction. In coexistence with the bands we have an isotropic gas phase, filling the space between the bands. These results are shown in figure 4.

Going to yet smaller noises, the bands disappear and an ordered state emerges, known as a *polar liquid*. This state is homogenous in space, but favors a particular direction of motion, and exhibits very large local density fluctuations.

I would also like to briefly mention the existence of other parameters which may play an important part, notably the particle density, speed, and the size and geometry of the system. At this stage I will adhere to a simplified description treating only the noise parameter, and will elaborate on the others as they become relevant to the types of models I simulate in my own work.



(a) Gas[6]

(b) Band[5]

Figure 4: Snapshots of characteristic phases of the discrete Vicsek model. The red circle at the center of 4a represents the radius of interaction R^* .

1.3 My project: continuous-time models for the emergence of collective motion

To motivate the study of continuous-time models, we observe that any realistic system should have a finite response time with regards to alignment, in contrast to the instantaneous reaction of the discrete model. This is significant because this time scale can be compared to the other two time scales in the model, which are the decorrelation time $\eta\Delta t$ and the time it takes to leave the radius of interaction R^*/v_0 . This additional time scale extends the model and may imply a non-trivial modification of the emerging phenomenology.

To study the effect of this alignment time scale I will introduce a continuous-time adaptation of the original Vicsek model, one that is expressed as a set of differential equations (or rather *stochastic* differential equations, as they contain a random noise term). From this point onwards, when referring to the “Vicsek model”, I will mean the continuous-time version, using “discrete-time Vicsek model” to refer to the original.

1.3.1 The continuous-time Vicsek model

The Vicsek model is an N particle dynamical system formulated under the assumption of a constant speed v_0 for all particles. The system evolves in time according to the following equations of motion:

$$\dot{\mathbf{r}}_i = v_0 (\cos(\theta_i), \sin(\theta_i)), \quad (1.3)$$

$$\dot{\theta}_i = A \left[\frac{1}{\mathcal{N}_i} \sum_{|\mathbf{r}_i - \mathbf{r}_j| < R^*} \sin(\theta_j - \theta_i) \right] + \xi_i(t), \quad (1.4)$$

where \mathbf{r}_i corresponds to the position of the i -th particle, θ_i to the direction of its velocity, A is the alignment coupling strength, R^* is the interaction radius, \mathcal{N}_i is the number of neighbors

counted in the sum, and $\xi(t)$ is a random noise term. The noise term is a random variable with zero mean and the following self correlation:

$$\langle \xi_i(t) \xi_j(t') \rangle = 2D\delta_{ij}\delta(t - t'), \quad (1.5)$$

where we introduce D , the *noise parameter*, which is a measure of the variance of $\xi_i(t)$ and in turn the strength of the noise.

Equation 1.3 propagates the i -th particle at a fixed speed v_0 in the θ_i direction, while equation 1.4 controls the dynamics, and contains two terms, which are analogous to those of equation 1.2. The first term is the alignment term, which causes each particles to rotate towards the average direction of motion of its neighbors. The second is an angular diffusion term, which causes the particle velocities to decorrelate after a characteristic time, even in the absence of interactions.

It is worthwhile to note that the discrete-time Vicsek model did not contain the parameter A , but instead featured instantaneous alignment on each time step. In this sense the continuous-time version has a larger parameter space, since we may adjust both the alignment and the noise parameter values. However, fixing A , we can still take the high- and low-noise limits of the continuous-time version, and expect to recover the aforementioned *gas* and *liquid* configurations.

The division by \mathcal{N}_i in the alignment term is important in order to achieve similarity to the discrete-time model in the high A limit. In this limit the time it takes to align with one's neighbors decreases to zero, and we recover the “instantaneous” alignment towards the mean neighbor velocity. In the discrete-time model the alignment is always instantaneous, and in particular it is independent of the number of neighbors; we want to recover this property for the continuous-time equivalent, and the normalization by \mathcal{N}_i ensures that. For an alternative derivation, see appendix B.1.

Another subtlety in the continuous-time model is whether one should count particle i itself in the alignment term, which corresponds to either normalizing by \mathcal{N}_i or $\mathcal{N}_i + 1$. I chose the former, which means that before normalizing I have to make sure that particle i has at least one neighbor, to avoid dividing by zero. I don't believe this choice will significantly alter the phenomenology, although it would shift and perhaps stretch the transition surface. This is because changing the normalization to $\mathcal{N}_i + 1$ makes the alignment force weaker, and it does so in a way that is non-uniform for regions with differing density. However, both normalization terms converge to the discrete-time model in the high A limit.

1.3.2 The flying XY model

The continuous-time Vicsek model which has been introduced suffers from a significant problem: it does not obey the action-reaction principle. When discussing biological flocking models one may imagine such asymmetric interactions, which may be the outcome of other more complex (i.e. cognitive) processes. However, looking at synthetic active matter systems, such as Quincke rollers[4] or Janus colloids[7], the interactions must be purely physical in nature, and so they must obey the action-reaction rule. The dynamics in such systems are therefore controlled by torques which are not modeled by the continuous-time Vicsek model, even qualitatively.

Fortunately, a remedy is readily available, in the following form:

$$\dot{\theta}_i = A \left[\sum_{|\mathbf{r}_i - \mathbf{r}_j| < R^*} \sin(\theta_j - \theta_i) \right] + \xi_i(t), \quad (1.6)$$

which is distinguished from equation 1.4 only by the absence of the $1/\mathcal{N}_i$ term. It is denoted the *Flying XY* model[8] due to its resemblance to the Classical XY Model, the difference being

that the spins are not fixed on a lattice but are free to move. It was previously studied in the context of a mean-field approach by Peruani et al.[9]

While the difference between the Vicsek Model and the Flying XY model may seem trivial at first glance, it carries some important consequences. First, as previously mentioned, the Flying XY model obeys Newton's third law, whereas Vicsek does not. In the Vicsek model, particle i does not affect particle j in the same manner that particle j affects particle i , since $\mathcal{N}_i \neq \mathcal{N}_j$. In fact, the alignment term of the flying XY model may be derived from a potential, as shown in Appendix A, unlike the Vicsek model. In addition, in the Vicsek model there is a fixed maximum alignment torque, regardless of the number of neighbors. This fixes in a sense the relative importance of the alignment versus the noise effect, regardless of the configuration. In the Flying XY model, however, the alignment strength is bounded only by the number of neighbors, so a dense region will be less affected by noise than a dilute region.

Unfortunately, while these observations do give a qualitative picture of the difference between the two models, it remains unclear what observable differences will ensue, if any. In particular, we are interested in the resulting phase diagrams, which may or may not be similar to each other.

1.3.3 Phase space and dimensionless parameters

In order to quantify phase behavior, we have to understand the parameter space we are working in, and in particular those parameters which have dimensions. This analysis is identical for both the Flying XY and the Vicsek models. The parameters are summarized in table 1. It is important to note that in addition to these physical parameters, there are other dimensional scales which are introduced by the simulation, for example Δt and the total simulation time. There are also finite size parameters with dimensions, namely the length or area of the box. While these may play an important role in the simulation outcome, I will not go into detail about them, and will discuss only those parameters which remain in an ideal system which is infinite in spatial extent and calculated with perfect precision.

Parameter	Physical meaning	Units
A	Alignment	T^{-1}
D	Noise	T^{-1}
ρ	Density	L^{-2}
v_0	Speed	L/T
R^*	Interaction radius	L

Table 1: Physical parameters with dimensions. T stands for time and L for length.

We would like to construct dimension-less sizes out of these 5 parameters. Since v_0 is the only parameter which couples time and space dimensions, we can use it to reformulate all the other parameters in terms of length, as is done in table 2.

Parameter	Units
A/v_0	L^{-1}
D/v_0	L^{-1}
ρ	L^{-2}
R^*	L

Table 2: Physical parameters with length dimension

These new quantities have names, in fact: $A/v_0 = 1/l_a$, where l_a is the alignment length, and $D/v_0 = 1/l_p$, where l_p is the persistence length. The alignment length is approximately the distance a particle traverses before reorienting its velocity towards the mean velocity, while the persistence length is the length of the trajectory after which the particle is traveling in a new random direction (disregarding alignment). Note the definition of l_a applies for the Vicsek model, but is more subtle in the flying XY case, as the alignment length may be much shorter, depending on the number of neighbors of a particular particle.

We are down to 4 dimensional parameters, and we may use our freedom of unit choice to select length units in which $R^* = 1$, and accordingly time units in which $v_0 = 1$. We are thus left with a 3-dimensional parameter space, in which every point is identified by a set of values (A, D, ρ) .

On this parameter space, or *phase space*, we will look for *transition surfaces*, which separate phase space into regions with qualitatively different behavior. We will be especially interested in the order-disorder transition surface, which separates a gas state from an ordered state.

2 Methods

The simulation was programmed in the C language, while the result analysis and plotting were done in Python. In this section I will present the main components of the code, as well as some of the various optimizations and numerical techniques I have used.

2.1 Simulation

The program is a Molecular Dynamics simulation, employing the Euler integration scheme. There are many fine points which I have had to deal with while writing it, and this section deals with some of those which I found the most interesting, or most useful to future endeavors in the field.

Some of the issues which I will not go into detail about are output storage format (text vs. binary vs. database), limiting memory usage during the simulation, enabling pausing and resuming of a simulation (which was extremely important from a practical point of view), and deciding how frequently to log the configuration, among others; I am mentioning them here only to indicate that they are significant and should be considered when writing such simulations.

2.1.1 Program outline

The simplest method to numerically evaluate a system of first order differential equations of the form given in equation 1.6 is called the Euler method. I will not attempt to go into detail about the subject of numerical integrators, but I will say that that main idea is to choose a "small" time-step Δt , and then to use the following formula:

$$\mathbf{r}_i(t + \Delta t) = \mathbf{r}_i(t) + v_0 \Delta t (\cos \theta_i(t), \sin \theta_i(t)), \quad (2.1)$$

$$\theta_i(t + \Delta t) = \theta_i(t) + \Delta t \left(\left[A \sum_{|\mathbf{r}_i(t) - \mathbf{r}_j(t)| < R^*} \sin(\theta_j(t) - \theta_i(t)) \right] + \xi_i(t) \right), \quad (2.2)$$

in order to get the system's configuration in the next time frame. Note that this formula is based on equation 1.6, i.e. the flying XY model, but the Vicsek scheme is obtained equivalently.

Before giving an outline of the simulation, we need to discuss the noise term, and in particular how to handle the Dirac delta function which appears in equation 1.5. We must first reformulate the delta function in a discrete-time fashion, going from dt to a finite (but small) Δt . From this perspective, $\delta(t)$ will be defined as a rectangular window of height $1/\Delta t$. Since the series of times $\{t_i\}$ is discrete, the delta function property $\int_{-\infty}^{\infty} \delta(t) dt = 1$ transforms into $\sum_i \delta_{i,0} \frac{1}{\Delta t} \Delta t = 1$, and remains valid in its discrete version.

Choosing a normal distribution for the noise, we get an expression for $\xi_i(t)$:

$$\xi_i(t) = \sqrt{\frac{2D}{\Delta t}} \varphi_i(t), \quad (2.3)$$

where $\varphi_i(t)$ is a normal random variable with mean 0 and variance 1. Finally we can rewrite equation 2.2 as

$$\theta_i(t + \Delta t) = \theta_i(t) + A \left[\sum_{|\mathbf{r}_i - \mathbf{r}_j| < R^*} \sin(\theta_j - \theta_i) \right] \Delta t + \sqrt{2D\Delta t} \varphi_i(t), \quad (2.4)$$

which can be implemented directly as part of an Euler integrator.

An outline of the main simulation loop is given in algorithm 1, which contains the main ingredients of the program.

Algorithm 1: Main program loop

Data: Randomly generated $\{\mathbf{r}_i(t), \theta_i(t)\}$, A, D (alignment and noise parameters)
Result: Set of configurations for $t > 0$

```

for number of iterations do
    Calculate torques;
    Update velocities;
    Update positions;
    Write positions and velocities to file;
end

```

2.1.2 Precision assessment

When assessing the accuracy of a numerical integrator, we generally define the error in the following way: we calculate the next configuration, and then recalculate it again using several smaller time steps (or a higher order integration method). The difference between these two outcomes is defined as the error of the integrator. To control accuracy we adjust Δt to keep this error below a particular threshold. For a full treatment, see [10], chapter 17.

This approach for error estimation is unfortunately not applicable in our case, at least not directly, due to the random noise term. Since we have a stochastic variable in the equation, even a theoretical perfect-precision algorithm would not reproduce the same result when run several times. A comparison of two such configurations is thus a poor way of error assessment.

We therefore resort to dimensional analysis to exert control over simulation accuracy. Starting with the noise term, we know that we have an associated characteristic time, $1/D$, which is the velocity decorrelation time of a non-interacting particle. If we use a time step such that $\Delta t \gg 1/D$, we are sure to grossly misinterpret the dynamics associated with angular diffusion, since our sampling is too crude. We have thus come up with a reasonable criterion to maintain accuracy: by setting $D\Delta t \ll 1$. In the same fashion, to capture the alignment dynamics, we make sure that $A\Delta t \ll 1$. This is as far as we need to go for the Vicsek model, but the flying XY model requires some more attention, and will be treated in the following section.

2.1.3 Adaptive time step

When simulating the Vicsek model, the mutual alignment term in equation 1.4 is bounded by 1, since:

$$\left| \frac{1}{\mathcal{N}_i} \sum_{|\mathbf{r}_i - \mathbf{r}_j| < R^*} \sin(\theta_j - \theta_i) \right| \leq \frac{1}{\mathcal{N}_i} \sum_{|\mathbf{r}_i - \mathbf{r}_j| < R^*} 1 = 1, \quad (2.5)$$

where we set $A = 1$ for simplicity. This means whatever the configuration, we can set Δt to achieve a desired accuracy and will get comparable precision. However the situation changes in the flying XY case, where we do not have the normalization term $1/\mathcal{N}_i$, and we get an upper bound of \mathcal{N}_i , rather than the convenient 1. \mathcal{N}_i may be a very large number, since it scales with the density, on average; in fact, even with a low mean-density we may still observe very dense regions, resulting in low and inconsistent precision.

One possible solution is to choose a “small enough” Δt . For example, we can determine a characteristic neighbor number $\bar{\mathcal{N}}$, and choose a new time step size by: $\Delta t^{\text{FlyingXY}} \equiv \Delta t^{\text{Vicsek}} / \bar{\mathcal{N}}$. This approach is problematic for several reasons:

1. We are forced to select a value for $\bar{\mathcal{N}}$ before the simulation, possibly without possessing the required data to make an informed choice.

2. We have to generously ‘‘round up’’ our estimate of $\bar{\mathcal{N}}$, or else the precision will suffer.
3. In configurations where the neighbor count is relatively low, such as the transient at the start of the simulation, we will progress unnecessarily slowly.

To overcome all of the above issues we introduce an adaptive time step, which gets updated as the simulation progresses. A naive way of setting the time step size is to find the particle with the largest number of neighbors, $\mathcal{N}^{\max} \equiv \max_i \mathcal{N}_i$, and set $\Delta t(t) = \Delta t^{\text{Vicsek}} / \mathcal{N}^{\max}(t)$ on each iteration. This introduces several new difficulties:

1. The elapsed simulation ‘‘time’’ must be saved along with the configuration data, whereas before it could be simply deduced by $t = n\Delta t$ for iteration n .
2. The sampling is no longer uniform in time, meaning that when taking averages we need to consider weighting according to the time differences between configurations.
3. The simulation ‘‘length’’ is no longer decided by the number of iterations, but by some other criterion such as simulation time or ‘‘real’’ computation time.

None of these present any serious problems, but they need nevertheless be considered when implementing such a scheme.

Flying XY simulations tend to be very slow, especially at low noises (see also section 2.1.5), mainly due to a small Δt for configurations with dense regions. It may in fact be unnecessarily small, given the following argument: while it is true that the upper bound on the torque is \mathcal{N}^{\max} , this worst case scenario occurs when all sine terms evaluate to 1. Physically, it means our particle is traveling in a direction which is perpendicular to every single one of its neighbors. This is unlikely for any particle, and far less likely for our \mathcal{N}^{\max} representative which is in a dense region, where particles are expected to be fairly well aligned.

Thus, speed can be improved by choosing the normalization factor to be

$$M(t) \equiv \max_i \left| \sum_{|\mathbf{r}_i - \mathbf{r}_j| < R^*} \sin(\theta_j - \theta_i) \right|, \quad (2.6)$$

where the sum is over j ’s. Note that $M(t) \leq \mathcal{N}^{\max}$. This is the strongest alignment torque experienced by any single particle, and is generally much smaller than the maximum number of neighbors.

A possible pitfall, for a perfectly aligned system for example, is to have a very small M and thus a very large Δt . This hurts the precision of the other terms, namely the Brownian noise, which has an associated time scale as well. The final choice for $\Delta t(t)$ is therefore

$$\Delta t(t) = \min \left\{ \Delta t^{\text{Vicsek}}, \frac{\Delta t^{\text{Vicsek}}}{M(t)} \right\}. \quad (2.7)$$

2.1.4 Sweep and Prune optimization

The computational bottleneck of the simulation is the pair-wise torque term calculation, i.e. the sum in equation 2.2. The naive calculation scales quadratically with N , as presented in algorithm 2. As a side note, I would like to mention that Algorithm 2 represents a typical Flying XY calculation, but does not apply to the Vicsek model. The problem is the normalization according to the number of neighbors, which is not known until all neighbors have been looped over, meaning the algorithm needs to be slightly modified. This fact reflects

Algorithm 2: Naive torque calculation

```

Data:  $\{\mathbf{r}_i(t), \theta_i(t)\}$ 
Result: Calculate alignment torque
Init: set  $\theta_i(t + \Delta t) = \theta_i(t)$  for all i;
for all particles particle pairs  $i, j$  do
    if  $|\mathbf{r}_i - \mathbf{r}_j| < R^*$  then
         $\delta \leftarrow A \times \sin(\theta_j - \theta_i);$ 
         $\theta_i(t + \Delta t) \leftarrow \theta_i(t + \Delta t) + \delta;$ 
         $\theta_j(t + \Delta t) \leftarrow \theta_j(t + \Delta t) - \delta;$ 
    end
end

```

a deeper physical difference between the two models; Newton's third law, which is visibly manifest in the symmetric addition of the δ terms, does not apply in the Vicsek model.

For N particles, this algorithm runs at $O(N^2)$, since the number of pairs is $\binom{N}{2} = \frac{1}{2}(N^2 - N)$. However, most of these pairs will not contribute to the torque, since the particles they represent are too far apart. There are several standard Molecular Dynamics techniques to optimize the calculation of forces with a cut-off radius, most notably Verlet lists[11] and cell lists[12]. These algorithms undoubtedly deserve consideration, but they are complicated to implement, prone to bugs, and I felt they were out of scope for this particular Master's project.

I decided to implement a simpler optimization algorithm, although arguably just as effective, which is inspired by a related problem from the world of computer game design, called "collision detection". This is the more intricate problem of determining whether two complicated three dimensional solids intersect, or "collide", so that their movement needs to be adjusted accordingly. However, since determining whether two polygons intersect is computationally costly, computer game developers use the idea of a "bounding box". Switching back to 2D, this means each object is bounded by a rectangular box aligned with the coordinate axes, as seen in figure 5a. We can safely assume that if two objects' bounding boxes do not cross, then there is no collision, and thus avoid the expensive polygon overlap calculation. This preliminary check is referred to as "broad phase" collision detection in computer science jargon.

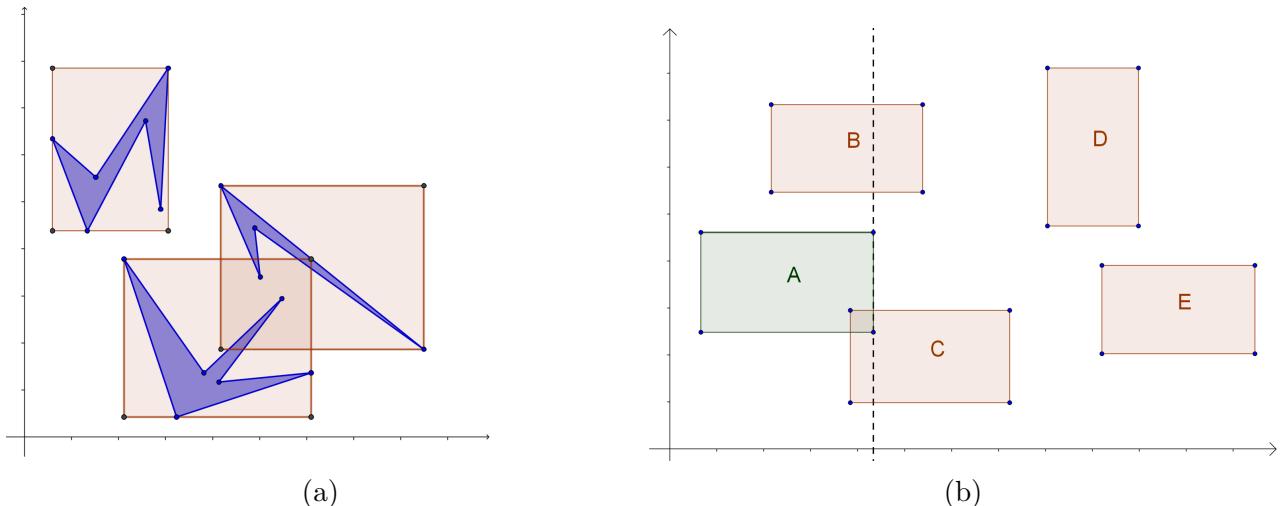


Figure 5: In 5a we can see the bounding box optimization for collision detection; we only need to check for overlap those bodies with overlapping bounding boxes. In 5b we see an example of sweep and prune. The bodies are represented by their bounding boxes. A is only tested against B and C, whose x axis projections overlap with its own.

Nothing of what we have seen so far reduces the $O(N^2)$ complexity of the algorithm. To move forward, we extend naturally the bounding box idea to an algorithm called Sweep and Prune (SAP), introduced by David Baraff in his PhD thesis[13]. The idea is the following: the objects are sorted on each iteration in ascending order according to their x position (their left-most point). To check for intersections, we start at the left most object, and compare it only to those objects which are “close enough” on the x axis. An illustration of this idea is presented in figure 5b.

It is unfortunately not possible to “sweep” on both the x and the y axes, but one of the two must be chosen. The problem is not the computational cost of sorting the elements twice. Sorting is done using an elementary algorithm named “insertion sort”, which runs at $O(n^2)$ in the worst case scenario; it is however very efficient for “almost sorted” input. Since our time increment is small, the particle order does not change much between consecutive iterations, making the sorting time effectively $O(N)$. The problem is, rather, the fact that no algorithm exists to verify whether two particles overlap on both axes, which doesn’t fall back into quadratic time (to the best of my knowledge). This is worthwhile if the bodies are complicated solids, but for our case of simple circles we may go ahead and check the distance after detecting an overlap on the x axis. The basic logic is presented given in algorithm 3.

This a stripped-down version of the original idea, but an extremely effective one, orders of magnitudes faster than the naive approach. Implementing it has allowed me to go from 400 particle simulations to 16000. A $t = 100$, 4196 particle Vicsek simulation was timed at 100 seconds with SAP, but took 600 seconds without it, a factor of 6. Of course the actual improvement could be much better, depending on the particular configuration, density, spatial inhomogeneities, and most importantly particle number.

Algorithm 3: Basic SAP scheme

```

Data:  $\{\mathbf{r}_i(t)\}, \{\theta_i(t)\}$ 
Result: Calculate alignment torque
for all particles  $i$  (in ascending  $x$  order) do
    for all particles  $j$  to the right of  $i$  (in ascending  $x$  order) do
        if  $|x_i - x_j| < R^*$  then
            | break (leave inner loop);
        end
        if  $|\mathbf{r}_i - \mathbf{r}_j| < R^*$  then
            |  $\delta \leftarrow A \times \sin(\theta_j - \theta_i);$ 
            |  $\theta_i(t + \Delta t) \leftarrow \theta_i(t + \Delta t) + \delta;$ 
            |  $\theta_j(t + \Delta t) \leftarrow \theta_j(t + \Delta t) - \delta;$ 
        end
    end
end
```

It should be noted that algorithm 3 is only an initial framework. In a real implementation there are several things to look out for, such as handling periodic boundary conditions, and saving the sorted particle list rather than starting with a completely unsorted list on each iteration.

As a final boost to performance, and inspired by the band-like structures which tended to emerge in parallel with one of the axes in Vicsek simulations, I added a possibility to spontaneously switch between sorting on the x or the y axis. This is especially relevant in a band configuration since sweeping on the axis which is normal to the mean polarization implies there will be far fewer potential neighbors detected, as shown in figure 6.

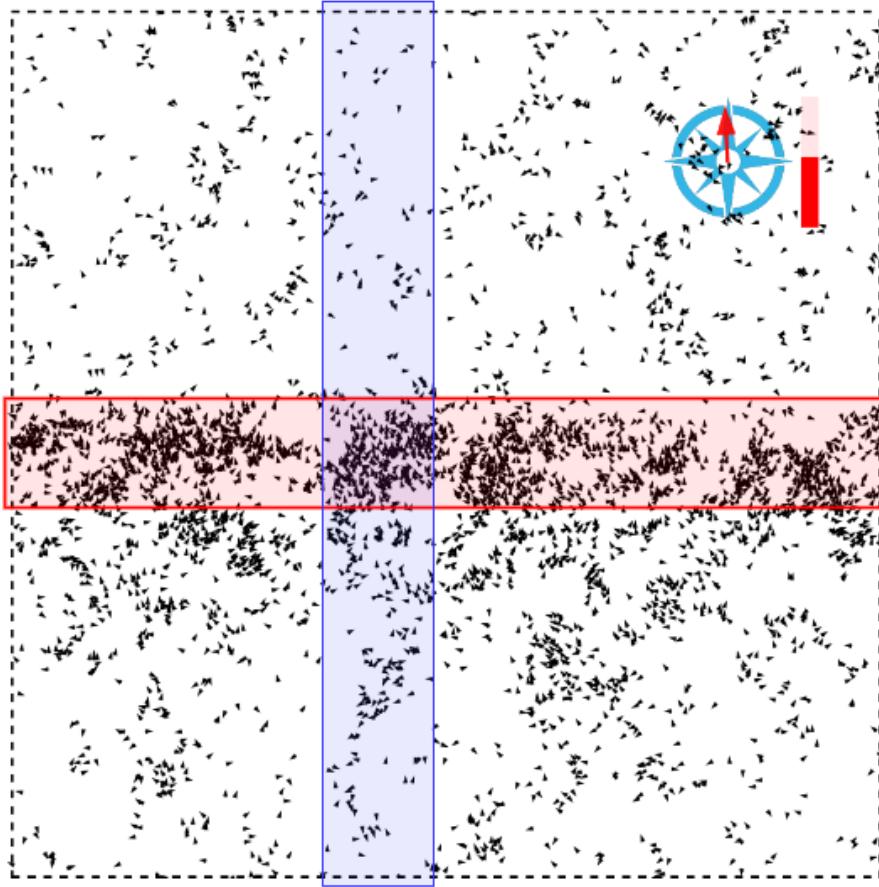


Figure 6: The importance of axis choice in SAP. The blue bar corresponds to sweeping on the x axis, while the red bar corresponds to the y axis. All particles inside the bar need to be checked to see if they are neighbors or not. Sweeping on the x axis is clearly more efficient. The compass in the corner points in the direction of the average velocity and the red bar shows the value of the order parameter Π , as defined in equation 3.1. Snapshot is taken from a Vicsek simulation with $A = 1$, $D = 0.2$, $\rho = 0.5$, $N = 4096$.

In that spirit, I measured the efficiency of each sweep, and once in a while (say every 1000 iterations) spontaneously switched axis, to see if the results improve. If they do, we continue SAP using the new axis, and if they do not, we switch back to the original. To determine the efficiency of a scan I computed the ratio of particles pairs examined to particle pairs which were actually neighbors, or the *hit rate*.

The algorithm naturally extends to higher dimensions. For a homogenous system in any dimension n with volume L^d , the number of potential neighbors that are looked at for each particle is $\frac{R^*}{L}N$, rather than N with the naive approach. This is potentially very powerful, especially in the presence of an anisotropic effect such as gravity or in the case of thin traveling sheets, such as those observed for the discrete-time Vicsek model in three dimensions[5].

2.1.5 Code parallelization

The simulation was done on an 8 processor computer, but the code as described up to this point only takes advantage of one. The simplest way to employ all eight is to run several simulations in parallel, which requires no changes to the code. This is done by a small script that launches the program 8 times in parallel, with different parameters.

There are two main limitations to this approach. The first is that if I wish to run one long simulation I cannot do so efficiently. The more acute issue, however, arose when I switched

from simulating the Vicsek model to the flying XY model, the relevant difference being the adaptive time step used in the latter. This means that for the flying XY model, low-noise (or high-density) simulations take far longer to complete than high-noise (or low-density) ones. The root of this difference is the combined effect of more neighboring pairs (since the low-noise systems cluster up) and the consequently small time step. To illustrate the situation, after running a week-long flying XY simulation for several noise values, the lowest noise value simulation reached $t = 700$ while the highest reached $t = 21,000$. This effect actually occurs in a fixed time step simulation as well, when using SAP, since a configuration with dense regions implies there are more neighbors to sum over; however this effect is negligible compared to the one resulting from the change in Δt .

The conclusion was that if I wished to attain reasonably long simulations for low-noise Flying XY simulations, I had to further speed up the code. The pair-wise torque calculation was still by far the biggest time consumer, despite the SAP optimization, and hence the most attractive target for optimization.

Fortunately, the torque calculation falls under the category of an *embarrassingly parallel* problem, meaning the main loop in both algorithms 2 and 3 can be split up into independent parts which may be computed in parallel. I implemented this using threads in the simulation code itself and got about a 400% increase in performance, which was enough for my purposes.

This is a far as I got, but there is much more that can be done on the parallelization front. For example, I divided the particle list into equally-sized sub-lists, but better would be to partition by trying to capture similar amounts of neighbors in each sub-list, meaning the division will be finer around denser regions. The next step would be to sum directly on a shared memory resource; in my current scheme each thread maintains a partial sum, all of which are added together when all threads have finished. This is a non-parallel process which takes $O(N \times n_{\text{threads}})$ time. Finally, the ultimate optimization would be mass parallelization on a computer cluster, which has the potential to open up a whole new range of system sizes.

2.2 Data analysis

Data analysis was performed in Python using the Numpy and Scipy modules for numerical work, and Matplotlib for drawing plots and making video clips. While most of the analysis is fairly straight-forward, I will elaborate on some of the more interesting facets of it as well as some techniques I used which others might find useful.

2.2.1 Integrity checks

Before applying the code to large systems and running long simulations, several tests were done to verify that the program was behaving as expected. The first of these was a single particle diffusion check. The single particle system can be solved analytically, and so we can check that the numerical results correspond to the theoretical values. In particular, we may examine the mean square displacement and the velocity auto-correlation function, whose analytical solutions are given by[14]

$$\langle [\mathbf{r}(t' + t) - \mathbf{r}(t')]^2 \rangle = 2 \frac{v_0^2}{D^2} (Dt - 1 + e^{-Dt}), \quad (2.8)$$

$$\langle \hat{\mathbf{p}}(t' + t) \cdot \hat{\mathbf{p}}(t') \rangle = e^{-Dt}. \quad (2.9)$$

A comparison between these expressions and numerical results can be seen in figure 7. We

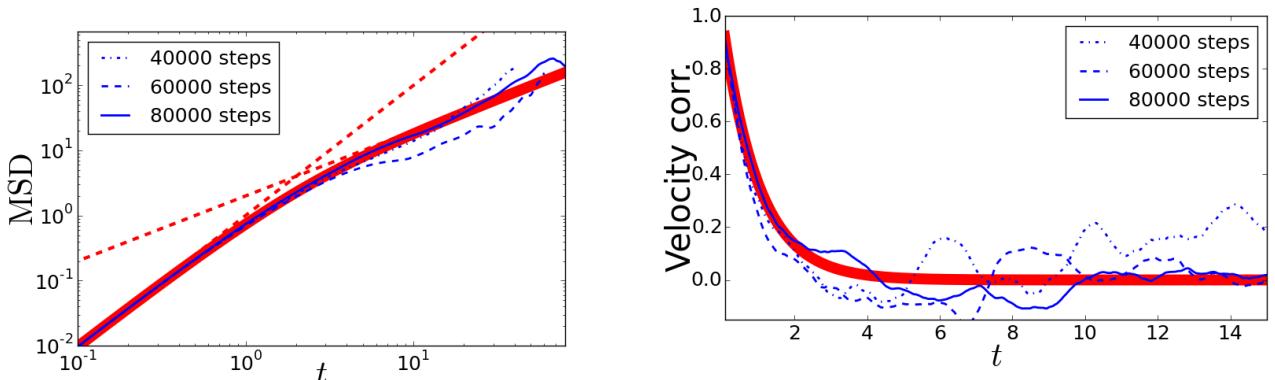


Figure 7: Statistical behavior of a single particle undergoing angular diffusion, inside a very large box, with $D = 1$, $v_0 = 1$, and $dt = 10^{-3}$. On the left we have the MSD as a function of time for 3 simulation runs of differing length (blue lines), the theoretical prediction (solid red line), and the asymptotic limits $MSD = t^2$ and $MSD = 2t$ (dashed red lines). These limits correspond to the ballistic and diffusive regime, respectively. On the right we see velocity autocorrelation vs time plotted for the same data (blue lines), along with the theoretical prediction (red line).

can also make use of these analytical solutions to verify the appropriate scaling behavior of the simulation. As an example in figure 8 we can see the MSD and velocity autocorrelation curves for different v_0 and D values, so we can finally be assured that the “noise” part of the simulation is working as expected.

Once we introduce the inter-particle alignment term, there are no more analytical predictions to compare with; however, at the very least we may draw some trajectories and convince ourselves that our “alignment” term is truly that, as is shown in figure 9.

2.2.2 Local density

To sample the local density mentioned in section 3.2, I partitioned the simulation box into a grid and “smeared” each particle over a finite region using a Gaussian. This method is limited

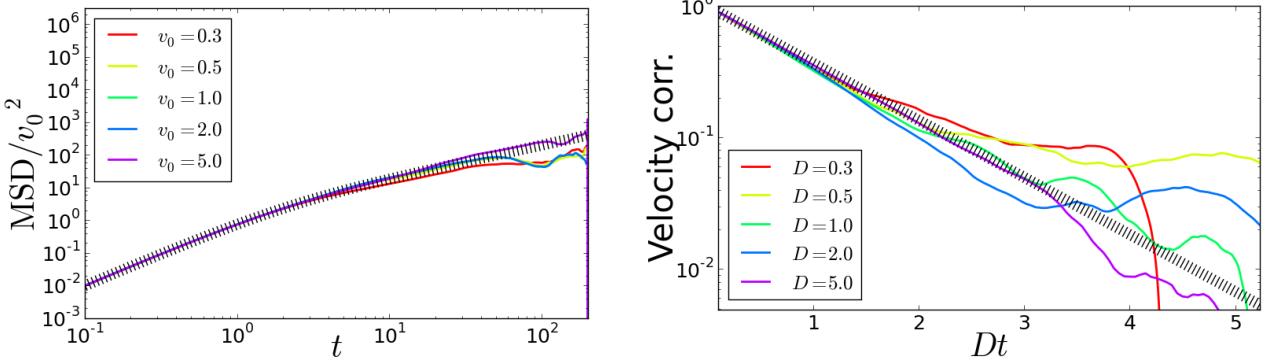


Figure 8: MSD and Velocity autocorrelation for different v_0 and D values, plotted with the theoretical curves (in black). Results are normalized to fall on the same master curve, as indicated in the axis titles.

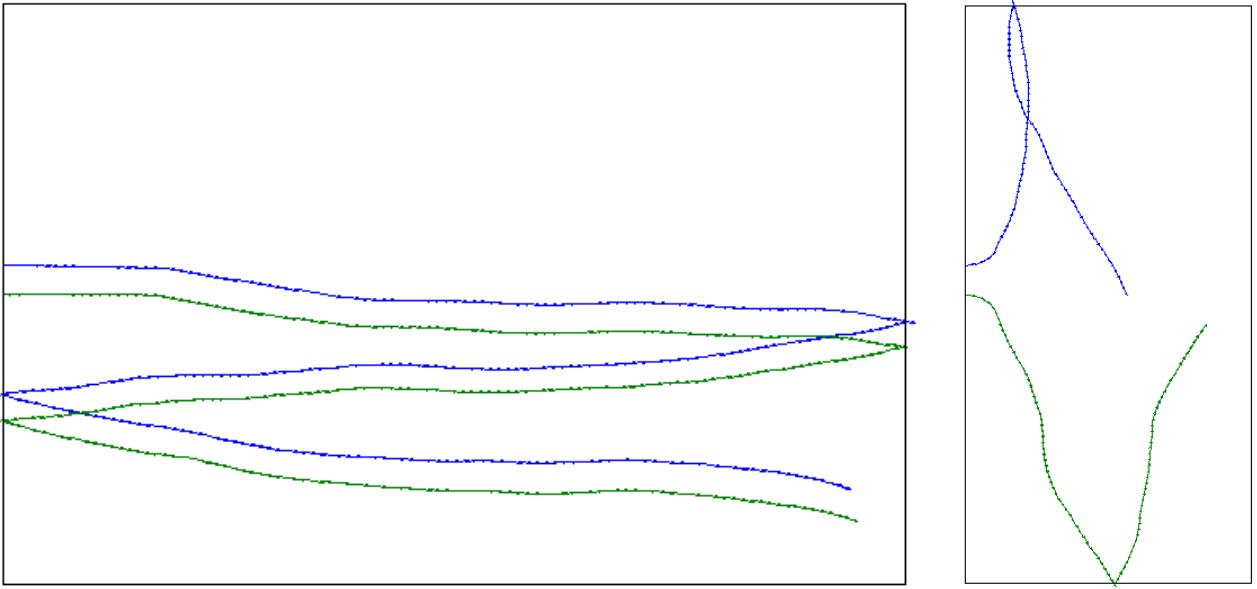


Figure 9: Example trajectories for a low noise system with reflecting boundary conditions, and alignment interaction turned on. The left picture show two trajectories with a positive A , while the right has a negative A (anti-alignment).

in various ways. First, it is slow: if space is partitioned into an $n \times n$ grid, and there are N_{part} particles, then to get the density as a function of position I need to sample $n^2 \times N$ Gaussians. This introduces the next problem, which is that this method introduces two somewhat arbitrary parameters:

1. The number of grid points n
2. The particle spatial extent (i.e. the standard deviation of the Gaussian)

These two parameters values are related to each other; for example, if I set a very fine grid (high n) and use very localized particles, most grid boxes will contain either one or zero particles, and no "smoothing" will be achieved. We would like some overlap between the particles, within a radius of order R^* , but not so large as to connect clusters of particles to each other.

Finally, there is a problem with maintaining consistency across system scales. A 50×50 grid may be very fine when treating a small simulation box, but very coarse when treating a large one. I did not spend time on this problem as I did not go into a scaling analysis of

the local density, but it does not mean that it shouldn't be considered for studies which put a bigger emphasis on this observable.

2.2.3 Cluster analysis

Given a particular configuration, we are interested in identifying *clusters* of particles, as will be elaborated in section 4.4. Mathematically, we wish to define an equivalence relation \sim on the set of all particles, such that $a \sim b$ implies that a and b belong to the same cluster. There are several possible ways to accomplish this, but perhaps the most natural is the following:

Definition 1 Let a, b be particles in the system. Then $a \sim b$ if $\text{dist}(a, b) \leq R^*$, and this relation is transitively extended, so that if $a \sim b$ and $b \sim c$ then $a \sim c$.

where $\text{dist}(a, b)$ is the inter-particle distance. In other words, two particles are in the same cluster if they are less than a distance R^* from each other, or if we can create a path from one to the other which passes through neighboring pairs only. This is illustrated in figure 10.

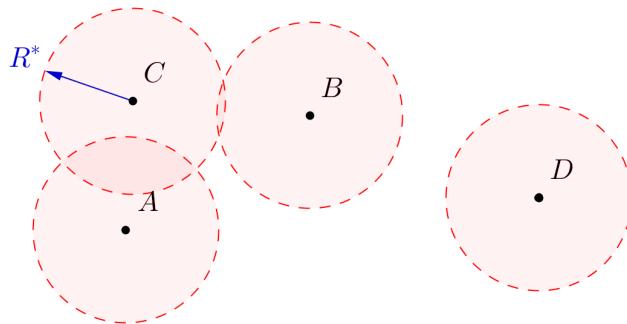


Figure 10: Illustration of particle clusters, as defined in definition 1. Particles A and C are in the same cluster because they are neighbors, and the same is true for B and C . Particles A and B are not neighbors, but they are in the same cluster, since we can go from A to B by passing through C . D is alone in its cluster, since it has no neighbors.

We now need to concoct an algorithm which takes as input particle positions and returns a partition of the particle list into clusters, which are the equivalence classes of relation 1. This is not a trivial problem to handle efficiently.

It can, however, be reduced to a known problem in mathematical graph theory: the problem of finding disconnected components in an undirected graph. Graphs are mathematical objects that represent networks, which are ubiquitous in computer science. This problem can be solved in linear time using known algorithms[15], and in particular using a Python library that handles graphs[16]. To translate the problem into a graph problem, we will define our graph in the following way:

Definition 2 We define the undirected graph G such that it has one vertex for each particle, and two vertices are connected if their corresponding particles are neighbors.

This definition is illustrated in figure 11. After constructing the graph, we can efficiently generate a list of its disconnected components, which will correspond exactly to the clusters we are interested in.

Note that this technique is not in fact a linear time solution for the entire problem, as constructing the graph requires knowledge of all neighboring particles, which is the problem treated in 2.1.4. I worked on the data after the simulation, so that I had to reconstruct the neighbor lists on each frame that I analyzed, making the treatment effectively $O(n^2)$. This was still feasible since the number of data frames I saved was much smaller than the entire simulation

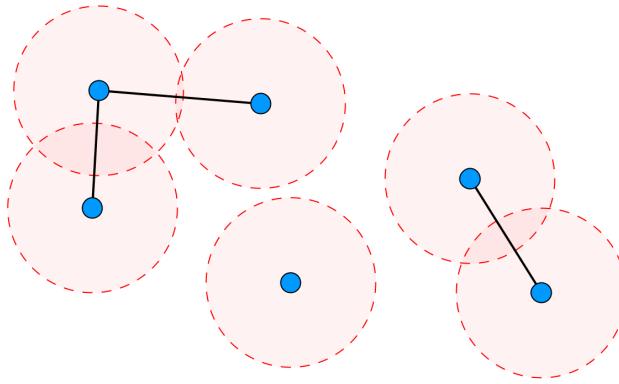


Figure 11: A graph formed by connecting neighboring particle, according to definition 2. In this example, there are 3 clusters, which correspond to the graph's 3 disconnected components.

(about every 20th frame in constant Δt simulations and much less when using adaptive time steps).

However, if clustering behavior is a focus of study for large simulations, one may save neighbor lists as part of the simulation output, which would not add much computational overhead since these are obtained anyway in the alignment calculations. This will indeed be quicker, but still not ideal, for three reasons: the first is that these lists may be quite long and require a lot of storage space, the second is that they complicate the data output format, and the third is that in the worst case, say if the system collapses to a very small spatial region, the pair list is simply $O(N^2)$ in size, and no trick can reduce that calculation time (at least not using a graph algorithm).

2.3 Summary

I consider my contributions to the methodology of the simulation of Vicsek and similar models to be the following:

1. A protocol for the simulation of continuous-time Vicsek and flying XY models
2. Guidelines for the parallelization of said protocol
3. An adaptation of the Sweep and Prune algorithm to N body simulations with short range interactions
4. A scheme for using adaptive time steps in such simulations
5. An outline for performing cluster analysis using known graph algorithms

It is my hope that this Methods section will be of use as reference for future projects in the field.

3 Vicsek model

In this section I will present the numerical results for the Vicsek model, and introduce the various order parameters and physical observables that will guide us in our following analysis and discussion.

3.1 Observed phases

When referring to a particular *phase* in this text, I mean a certain large-scale behavior of the system. This definition is somewhat open to interpretation, and part of my work has indeed been selecting appropriate *order parameters* that may quantify our notion of a phase.

To get an idea of the system's behavior near the transition point, we will explore the $A = 1, \rho = 0.5$ line through phase space, and examine the emergent structures near the transition point. The values of D were chosen so as to sample on both sides of the transition. To estimate this range I started with smaller simulations (around 100 particles) and a relatively high noise value, and verified I was in a gas region; I then systematically decreased the noise until structures began to form. After setting these rough boundaries, I went on to larger simulations and refined the appropriate range of D values.

Several final configuration snapshots are presented in figure 12. These configurations are taken from the very end of the simulation, where the system has presumably stabilized in its state. It has been observed previously that the discrete-time Vicsek model produces extended traveling bands at parameter values near the transition point; this is a non-trivial result which is clearly reproduced in the continuous-time model.

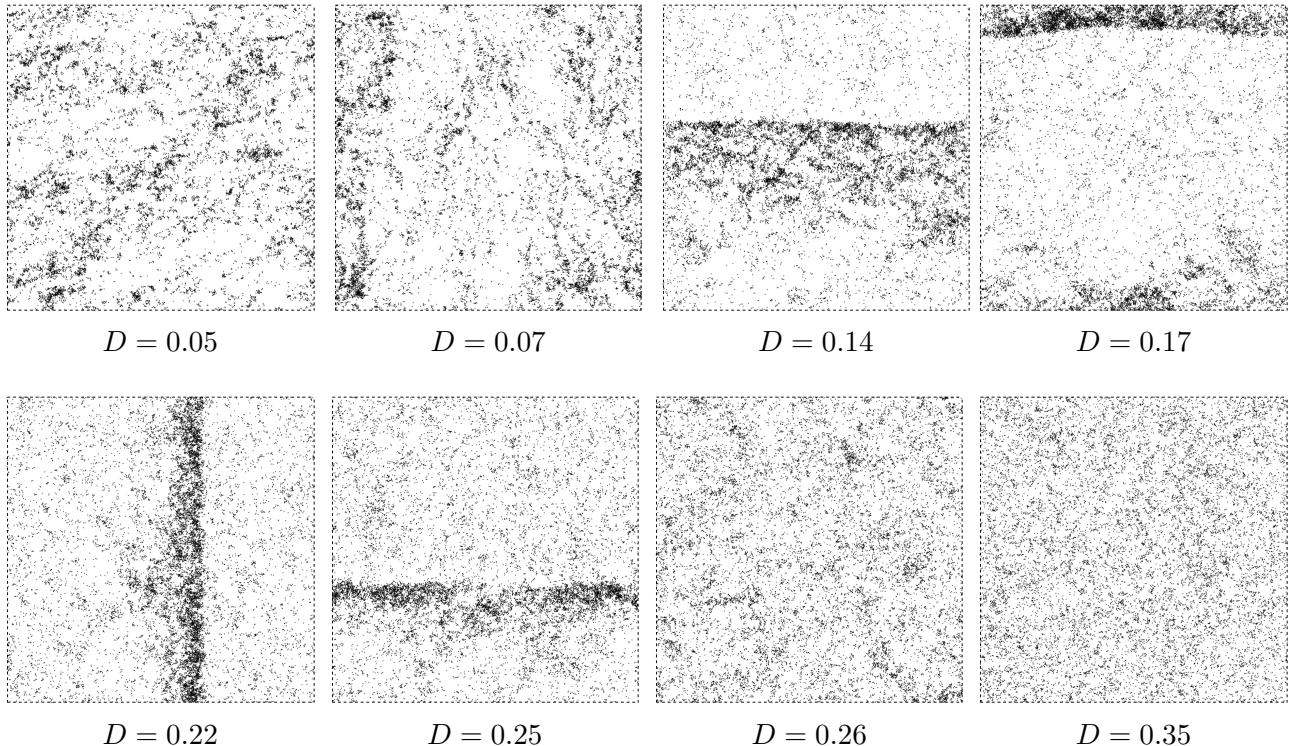


Figure 12: Snapshots of final simulation configurations on the $A = 1, \rho = 0.5$ line, Vicsek model. The number of particles in these simulation is 16384.

The polarized liquid state has not been studied in detail. Qualitatively, we do see a polarized inhomogenous state for very low noises, and the bands do disappear below a certain noise value. However, my work will focus on the vicinity of the transition surface, and I will not discuss this low-noise region further.

A band state can be unambiguously distinguished from a gas by looking at our first order parameter, the overall polarization Π :

$$\Pi(t) = \frac{1}{N} \left| \sum_j e^{i\theta_j(t)} \right|, \quad (3.1)$$

where N is the number of particles in the system. Note that $\Pi(t) \in [0, 1]$. Despite its simplicity, $\Pi(t)$ will prove to be quite versatile, and will serve as the primary order parameter in my work. It is a good measure of a significant aspect of the Vicsek model phase transition, namely the *breaking of rotational symmetry*. From an isotropic initial state, the system spontaneously polarizes in a particular direction. In fact bands tend to align their motion along either the x or y axis, or less frequently along one of the two main diagonals, resulting in 8 possible outcomes for the polarization orientation of a stable state. The reason this happens has to do with the toroidal topology of our system (see [17], [6] for further discussion).

3.2 Nature of the transition

To begin our study of the phase transition, we plot the mean overall polarizations as a function of noise, as is shown in figure 13. The mean is taken after discarding the initial transient part of the simulation, since we are interested in steady state statistics. We observe a clear transition point at around $D = 0.25$, in agreement with previous observations of the discontinuity of the Vicsek transition. To further confirm the first-order nature of the transition, we plot the same

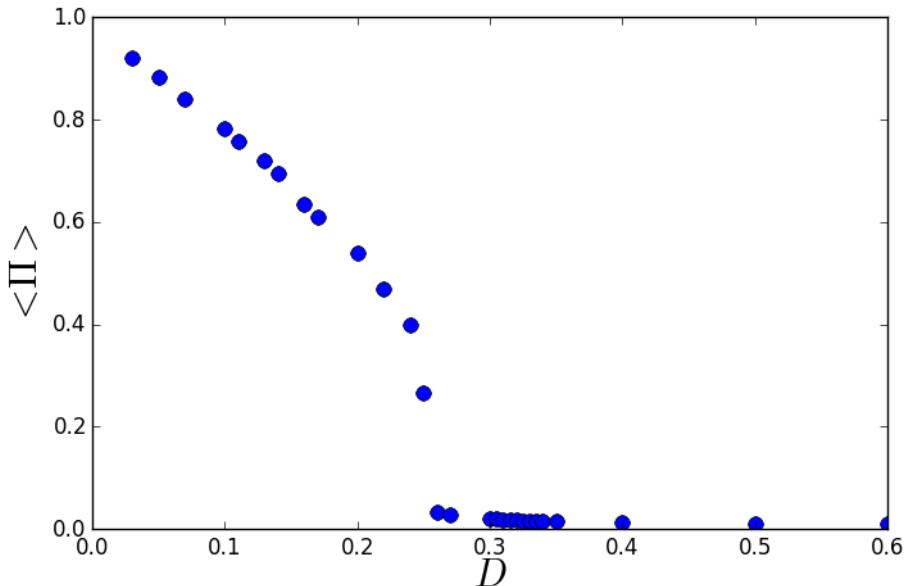


Figure 13: Mean polarization for different D values along the $A = 1, \rho = 0.5$ line. There are 16384 particles in the simulation.

data for several system sizes, shown in figure 14. It has been observed by Chaté et al.[5] that the discrete-time model transition becomes sharper for larger system sizes; we would like to see if this scaling behavior is reproduced in our case. Results are shown in 14, which show the same conclusion: the slope steepens with increasing system size, indicating a discontinuous transition.

Here we are also introducing the Binder cumulant[18], which is defined as:

$$G \equiv 1 - \frac{\langle \Pi^4 \rangle_t}{3 \langle \Pi^2 \rangle_t^2}. \quad (3.2)$$

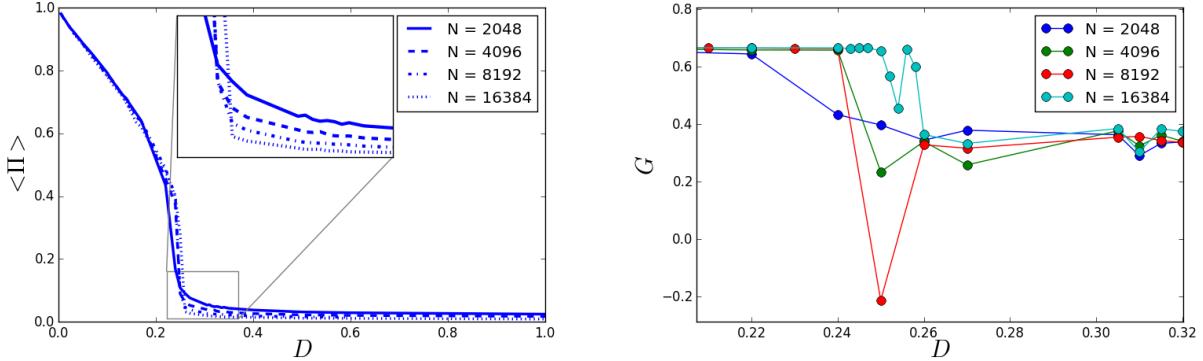


Figure 14: Scaling behavior of the Vicsek transition. On the left we plot the mean polarization $\langle \Pi \rangle$ along the $A = 1, \rho = 0.5$ line, and on the right we plot the Binder cumulant G .

The Binder cumulant is useful in deciding whether a phase transition is continuous, by comparing systems of different size. In a continuous transition the Binder plots should all overlap in the transition point, while in a discontinuous transition we expect G to drop to negative values close to the transition. We thus see a clear indication of a discontinuous transition up until the $N = 16384$ scale; I suspect that these simulations were not long enough to properly capture the fluctuations, as the dynamics of large system seem to be slower.

In addition to the rotational symmetry breaking, we observe a *breaking of translational symmetry*; from an initially homogenous state, the system evolves into a band structure, which is highly localized on the axis parallel to its velocity. We would like to introduce another order parameter which will probe the inhomogeneity of the system, rather than its anisotropy, and watch its behavior near the transition point.

We start by defining a time dependent density field $\rho(\mathbf{r}, t)$:

$$\rho(\mathbf{r}, t) = \sum_i \delta(\mathbf{r} - \mathbf{r}_i), \quad (3.3)$$

using a sum of Dirac delta functions over the particles. In order to get a smoother field, which can be studied numerically, we coarse grain $\rho(\mathbf{r}, t)$ and get an effective field $\rho_l(\mathbf{r}, t)$ in the following manner:

$$\rho_l(\mathbf{r}, t) = \rho(\mathbf{r}, t) * f(\mathbf{r}), \quad (3.4)$$

where $f(\mathbf{r})$ is some localized function (I use Gaussians, but square or triangle windows could work as well), and $*$ is a spatial convolution. A convolution with a delta function is equivalent to an origin translation, meaning we end up summing over Gaussians centered around the particle positions. This is easily shown:

$$\begin{aligned} f(\mathbf{r}) * \delta(\mathbf{r} - \mathbf{r}_i) &= \int d\mathbf{y} f(\mathbf{y}) \delta(\mathbf{y} - (\mathbf{r} - \mathbf{r}_i)) \\ &= f(\mathbf{r} - \mathbf{r}_i). \end{aligned} \quad (3.5)$$

For completeness, the final expression I used for density calculations is:

$$\rho_l(\mathbf{r}, t) = \sum_i f(\mathbf{r} - \mathbf{r}_i(t)), \quad (3.6)$$

where my chosen f function is a simple Gaussian:

$$f(\mathbf{r}, t) = \alpha e^{\frac{-|\mathbf{r}^2|}{2\sigma^2}}, \quad (3.7)$$

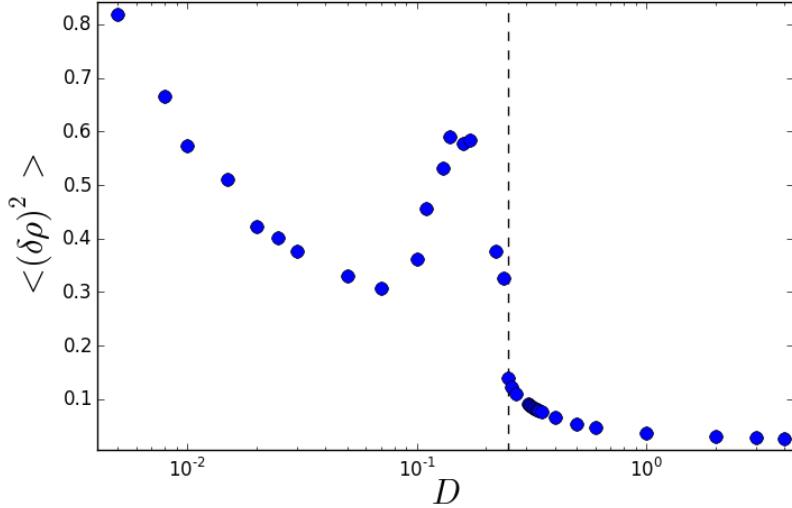


Figure 15: Local density variance for different D values along the $A = 1, \rho = 0.5$ line, $N = 4096$. The dotted vertical line is the approximate transition point, as determined from figure 14.

with α being a normalization factor intended to make sure the integral over all of space is 1, so as to obey the condition $\int d\mathbf{r} \rho(\mathbf{r}) = N$. Notice that there is a parameter in this Gaussian: σ , which controls the spatial extent of each particle. I found that a value around $\sigma = 5R^*$ works well for my purposes, but of course this is somewhat arbitrary.

The expression in equation 3.6 can be calculated directly, and lead us to our desired inhomogeneity observable. In particular, we evaluate $\rho_l(\mathbf{r}, t)$ on a grid, and take the variance, which we will refer to as $\delta\rho_l^2$. The variance will be low for a homogenous state and will rise as bands start to form. By sampling on a grid we introduce another external parameter, namely the number of points in the grid, which affects both the accuracy and the cost of the computation. I used a 50×50 grid. Note that this variance scales with the overall density, so when plotting it we will normalize $\delta\rho_l^2$, dividing by ρ . This does not matter so much at this stage since we are holding $\rho = 0.5$ fixed, but it will when we explore other regions of phase space.

Results are shown in figure 15. It is interesting to note the peak in the variations near the transition lines, where bands form. The low noise data should not be taken at face value, since I suspect longer simulations are required for the system to properly stabilize; we do in any case see an increase in spatial inhomogeneities for low D values. Since $\langle \Pi \rangle \approx 1$ for low noise values, it seems all particles are aligned but inhomogeneously distributed around the system, which is certainly a stable configuration in the $D \rightarrow 0$ limit.

As final evidence towards the first-order nature of the transition, I performed a hysteresis simulation by varying the noise continuously, starting from a high-noise gas state, in a linear fashion. In figure 16 we can see the polarization as a function of the noise, and observe a clear hysteresis loop, once again indicating a first-order transition.

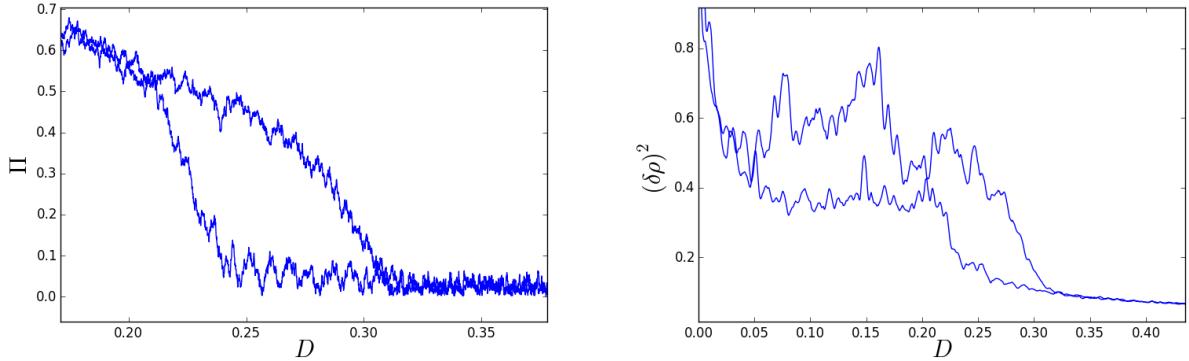


Figure 16: Hysteresis in continuous-time Vicsek model. Total simulation time is $t = 5 \times 10^4$. The right plot is smoothed for readability. The parameters are $A = 1, \rho = 0.5, 8192$ particles. The initial noise is $D = 1.5$, which is then linearly reduced down to $D = 10^{-4}$, and taken back up to 1.5 in a similar fashion, or in other words $D(t)$ is a triangle function.

3.3 Phase diagram

One of my main goals is to plot the phase space diagram, and in particular the surface separating the polarized state from the gas state. In order to identify a point on this surface, we will perform several simulations along a single line segment in parameter space, measuring Π at each point. We will then study how the order parameter varies along the trajectory. If we choose the location and direction of our segment carefully, we might penetrate through a phase transition surface, and we can then use the order parameter to pinpoint the precise point in our path where we crossed it. This is illustrated in figure 17; it is in fact what we did in the previous section, from which we can deduce our first transition surface point: $A = 1, D = 0.25, \rho = 0.5$.

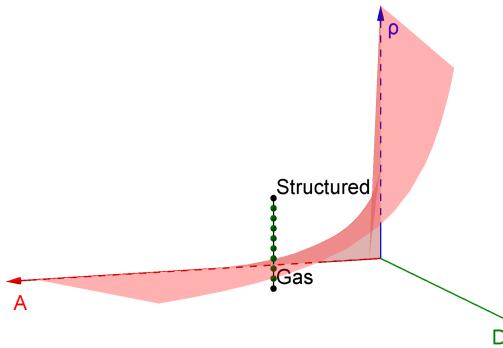


Figure 17: Finding the transition point along a fixed A, D trajectory. Each point is a simulation with a given ρ value. The surface is drawn for illustration purposes, and does not represent the actual Vicsek transition surface.

Although we can go in any arbitrary direction, it is convenient to travel parallel to one of the planes, keeping two of the parameters fixed. The preferred plane depends on the approximate form of the transition surface near the point we are looking at; the transition point will be clearest if we penetrate the surface along its normal direction, and vague if we graze it along a tangent. I searched for transition points exclusively on the $A = 1.0$ and $D = 0.2$ planes, so my final result will be two transition curves along the surface. A map of the simulations is given in figure 18.

Note the red curves in figure 18, which represent the transition curves. These lines were calculated by going along a vertical or horizontal line on either of the planes shown in figure 18, plotting out the order parameter along this path, and locating the transition. We chose to

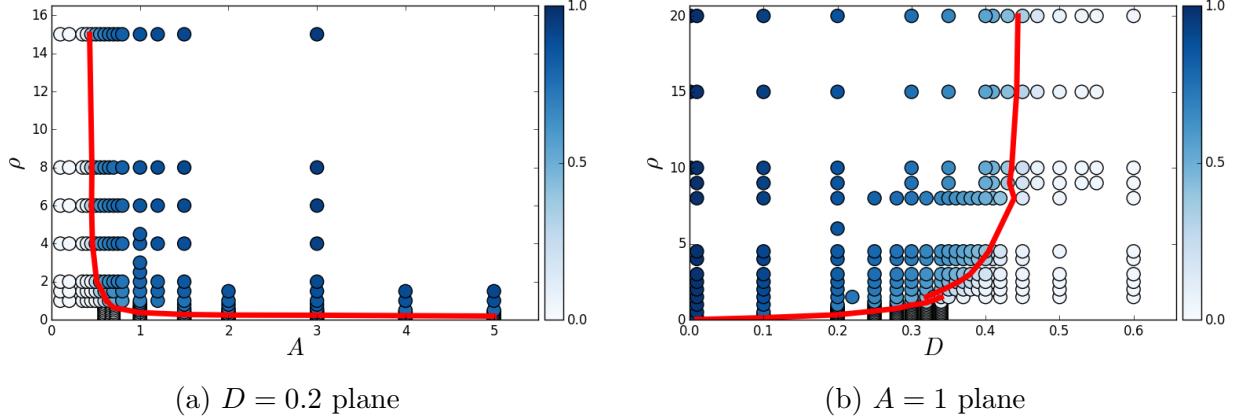


Figure 18: Vicsek model phase diagrams, based on 4096 particle simulations. Each circle represents a simulation, colored according to its $\langle \Pi \rangle$ value. Simulations are between 1500 and 5000 time units in length. The red curves show the transition lines, and are comprised of two parts, one calculated along vertical lines (fixed A or D) and the other across horizontal lines (fixed ρ).

define this transition point as the inflection point of this graph, calculated by fitting the curve locally to a cubic polynomial. “Locally” means I selected by hand the range of points that will be included, to correspond to an interval around the transition. If the transition is indeed discontinuous, as we suspect, than such a definition will converge with the sharper transition point observed in the previous section, if we were to increase the system size. An example of such a plot is given in figure 19.

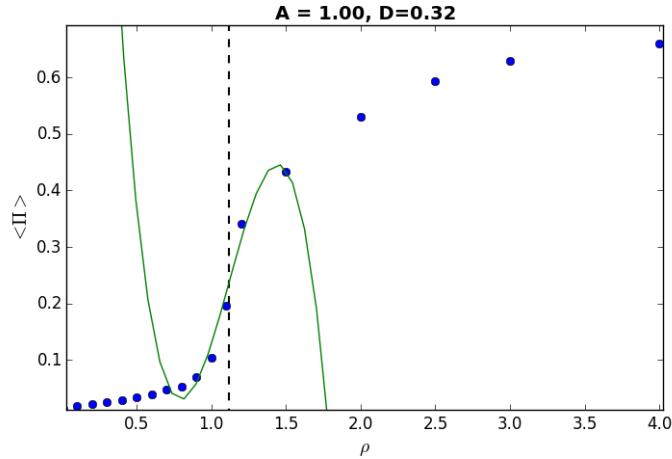


Figure 19: Finding the transition point using the inflection point of $\langle \Pi \rangle$ along a path in phase space. The green curve is the local cubic fit and the vertical dashed line is its inflection point.

Going back to figure 18, we notice both planes in phase space show a vertical asymptote. I believe its roots lie in the normalization of the alignment term in the Vicsek dynamics, which sets an upper bound on the alignment strength, independent of how many neighbors a particle has. Let us take a look at the fixed A plane as an example. If we increase the noise beyond a certain threshold value, the alignment torque simply becomes negligible, even in a very dense system. In such a case we cannot expect collective motion to occur. The fixed D case is analogous.

My final effort to characterize the phase transition will consist of a hypothesis, which is that

D/A is an interesting dimensionless parameter of the system. This is in part inspired by the hydrodynamic theory associated with Vicsek-like models, which will be discussed further in the flying XY analysis; it also has an intuitive significance, as it is a measure of the relative importance of alignment and noise. Finally, we observe that both vertical asymptotes in figure 19 have approximately the same value of D/A , which certainly merits further investigation.

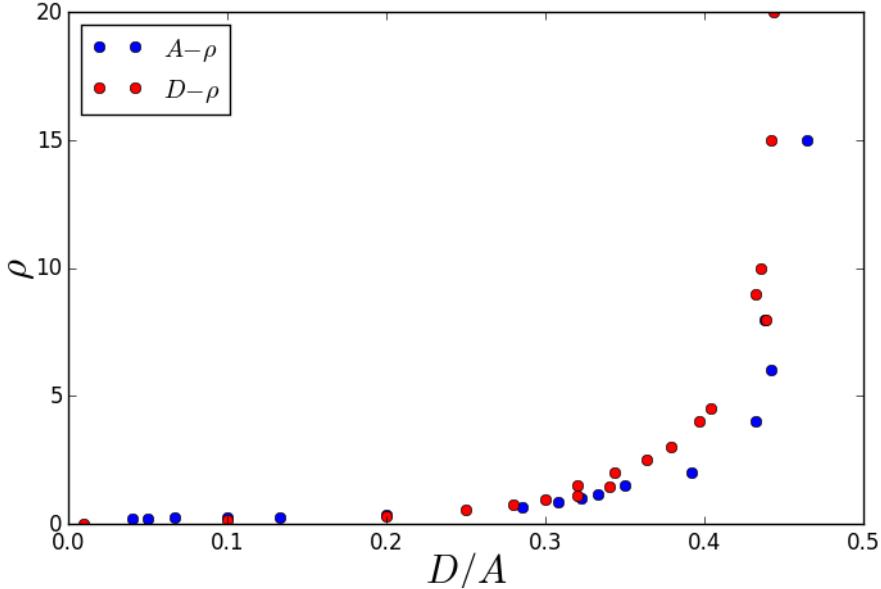


Figure 20: Transition density as a function of D_c/A_c .

We plot the transition points shown in figure 18 on the D/A vs ρ plane, seen in figure 20. The most noticeable feature of this plot is the vertical asymptote at $D/A \approx 0.45$. It implies that regardless of density, collective motion cannot occur below the $D = 0.45A$ plane.

3.4 Spatial properties of emergent structures

We would like to analyze the emerging band patterns, and compare them to those observed in [5]. We are especially interested in the form of a band *density profile*. To define this profile we look at the local density ρ_l and average it on the direction perpendicular to the mean orientation, x_\perp , so we have $\rho_\perp(x_\parallel, t) \equiv <\rho_L(\mathbf{r}, t)>_\perp$. This density profile is then shifted so that its peak is in the center and finally time averaged, so that we are left with the mean density profile, seen in figure 21.

We see bands that are sharp on one side and have tails which seem to decay exponentially, which is confirmed by the fitting to $\rho_\perp(x_\parallel, t) \approx a_0 + a_1 e^{-x_\parallel/w}$, shown in 21a. This is in accord with the observation made in [5]. Note that we take noise values which are close to the transition but on the “ordered” side of it, since there are no stable bands to the “right” of the transition.

We can estimate the width of the bands using the fitting parameter w , and their height by looking at the density maximum. Both are plotted against noise in figure 22. The heights, which correspond to the density at the band front, seem to stay constant close to the transition. For low enough noises we see a decline, which may be a sign of another phase transition, i.e. the band-liquid transition. The widths, corresponding to the spatial extent of the bands, increase slowly as the noise is lowered away from the transition point, and the bands get more “smeared” along their direction of motion. This effect seems to be linear close to the transition but becomes stronger for low enough noise values.

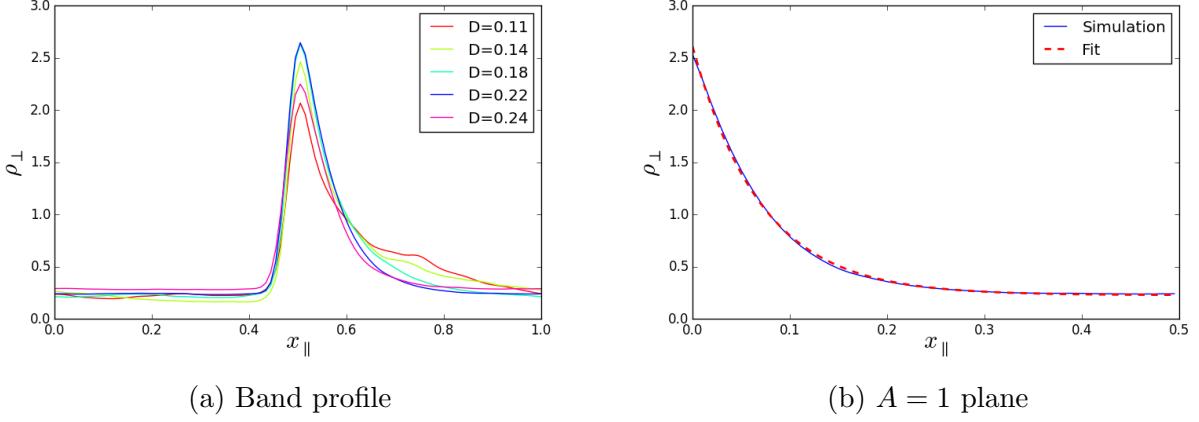


Figure 21: In 21a: Time-averaged band profile for $A = 1.0$, $\rho = 0.5$, using 8192 particles. The sampling grid resolution is 100×30 on the parallel and perpendicular axes respectively. x_{\parallel} is measured in units of L , the box side length. In 21a we see the decaying tail for $D = 0.22$ with an exponential fit.

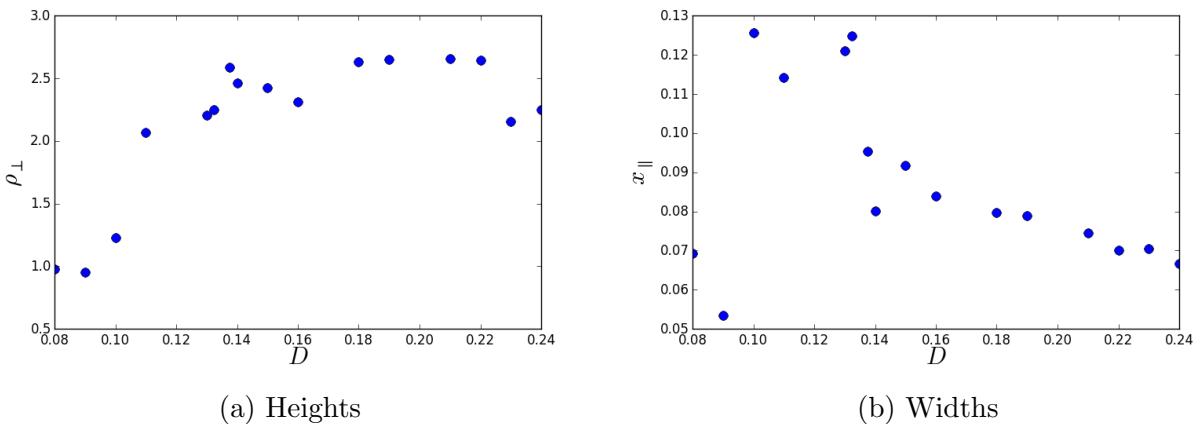


Figure 22: Band heights and widths for $A = 1.0$, $\rho = 0.5$. The order-disorder transition is at $D = 0.25$. The two lowest noise values do not display stable bands.

4 Flying XY

Having analyzed the Vicsek model, we would like to look at the flying XY model, which is somehow more “physical”. This is primarily to its being derivable from a potential, and its adherence to the principle of action-reaction; also, unlike the Vicsek model, it shows stronger alignment interaction in dense regions, which is physically reasonable. We will perform an identical analysis to that done in section 3, and point out the similarities and differences between the two models as we progress.

4.1 Observed phases

Following the plan of performing the same analysis for both models, I would have liked to show the simulation outcomes along the $A = 1, \rho = 0.5$ line through phase space. However, it turns out that this path is ineffective for the flying XY model, since it does not enter the transition surface at a right angle, and the transition is therefore not as distinct. Instead I will show a much clearer transition along the line $A = 1, D = 2$, varying ρ . Final snapshots are presented in figure 23.

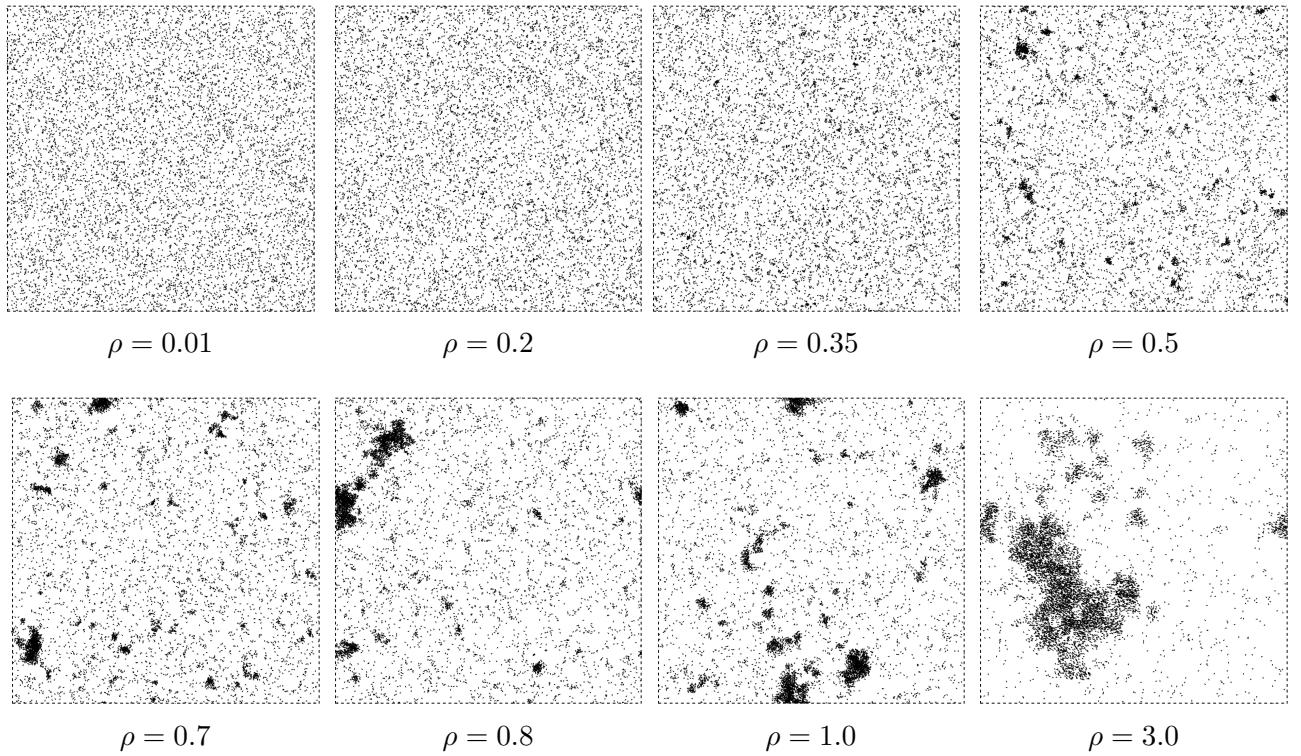


Figure 23: Snapshots of final simulation configurations on the $A = 1, D = 2$ line, flying XY. The number of particles in these simulation is 8000. We locate the transition at around $\rho_c = 0.5$ based on the data in figure 24.

At a glance we see a significant difference: there are no bands near the transition, unlike both the discrete- and continuous-time Vicsek models. I should note that just because there are no bands on this trajectory is not reason enough to conclude that they don’t appear on some other place near the transition surface; I can say, however, that despite a thorough search, I have been unable to observe bands for any parameter range in flying XY simulations³. It appears that the flying XY model favors a “clustering” configuration, in which localized particle clusters are formed, rather than extended structures.

³A band has been observed by [8] for the flying XY model, in the point $A = 0.16, D = 2, \rho = 30, v_0 = 2.1$. I

Another notable difference is the more continuous nature of the snapshots when crossing the transition. In the Vicsek model the bands disappeared abruptly upon crossing the threshold D value, while in the flying XY case clusters begin to appear even at low densities, and grow gradually. This may have to do with the fact that bands have an associated length scale, namely the box side length, while clusters can be shrunk continuously.

4.2 Nature of the transition

The next step is to calculate the mean overall polarization along this line, as shown in figure 24. Although the transition is not as sharp as that seen in figure 13, it is clear that at some point around $\rho = 0.4$ we begin to observe non-zero polarization.

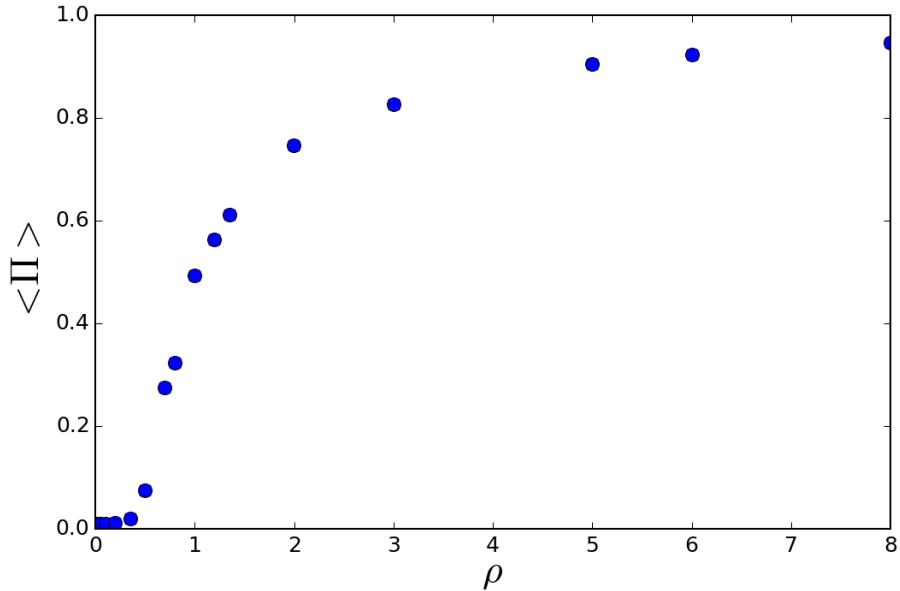


Figure 24: Mean polarization for different ρ values along the $A = 1, D = 2$ line. There are 8000 particles in the simulation.

This transition is in fact surprisingly clear, and perhaps suspiciously so; I had not expected Π to be a very effective order parameter for the detection of local clusters, which polarized only locally. Assuming these clusters are numerous enough, and uncorrelated between themselves, we expect to have roughly zero overall polarization, on average, which is not what we observe. I will come back to this point later on.

To continue our analysis, we will present scaling behavior for these same parameter values, given in figure 25. Interestingly, we observe once more a discontinuous transition, as indicated by the slope steepening when we go to larger system sizes. The Binder cumulant does show a decrease in value around the transition, but I believe that larger simulations are needed to produce a conclusive drop to negative values, if indeed one occurs.

Since the $\langle \Pi \rangle$ plot is already quite indicative of a discontinuous transition, we expect an even clearer picture when plotting $\langle \delta\rho_l^2 \rangle$. This is because while it is not clear that rotational symmetry is broken for a configuration of traveling clusters, translational symmetry clearly is; therefore we expect the density variations to become pronounced at the transition. Results are shown in figure 26.

have reproduced this simulation and confirm the result. However, this system has a very small box size, $L = 10$, and I believe this band is due to a finite-size effect, and would disappear in a larger system (keeping the density fixed). In addition the band is smeared across the box, much unlike the highly localized Vicsek bands, and

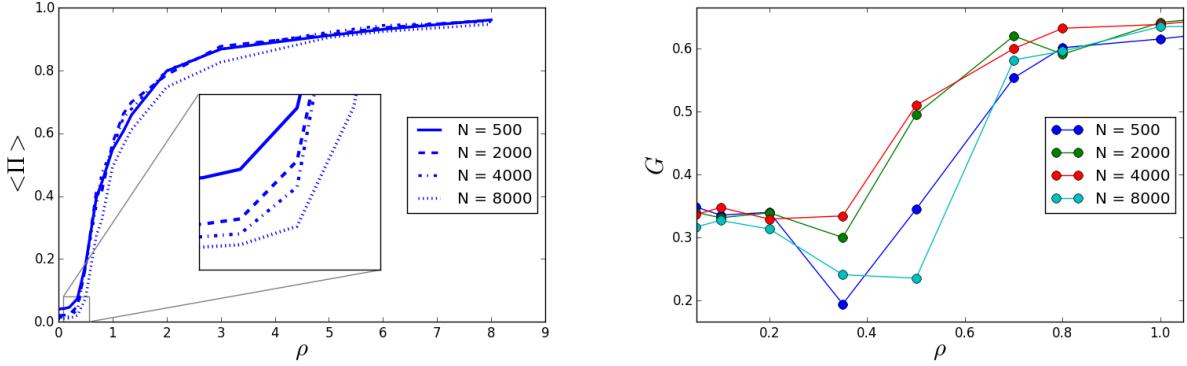


Figure 25: Scaling behavior of the Flying XY transition, $A = 1, D = 2$.

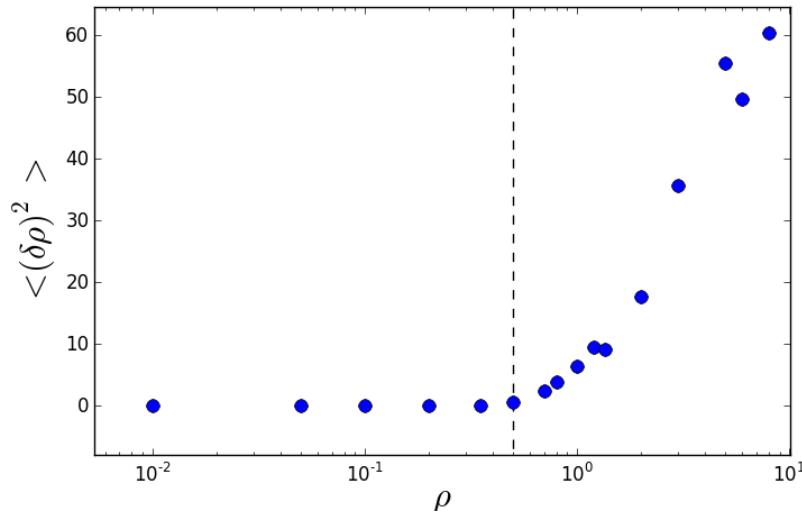


Figure 26: Local density variance for different ρ values along the $A = 1, D = 2$ line, with $N = 8000$.

Interestingly, the transition is much smoother when seen through the local density variance. This indicates that the spatial configuration does not necessarily change abruptly at the onset of collective motion, although the overall polarization does. Both these observations are non-trivial, and will be considered in the following discussion.

Finally we look at a hysteresis plot. The results presented in figure 27 are taken from the $A = 1, \rho = 0.5$ line, for technical reasons⁴. Even though this path across the transition is not ideal, we can still clearly discern hysteresis loops, indicating a first-order phase transition.

To summarize our findings on the flying XY transition, we observe a discontinuous transition in Π , a continuous transition in $\delta\rho_l^2$, and a clear hysteresis in both. The Binder cumulant is to be regarded as inconclusive, due to the limited length of the $N = 8000$ simulations and the lack of sufficient resolution near the transition.

these should not in my opinion be considered similar phenomena.

⁴Varying ρ during a simulation is a subtle task, since it means either changing the number of particles or the box dimensions, neither of which can be done trivially without altering the statistics. It would be possible to vary A instead, to see a sharper transition line, say along the $D = 2, \rho = 5$ line; unfortunately, time did not permit.

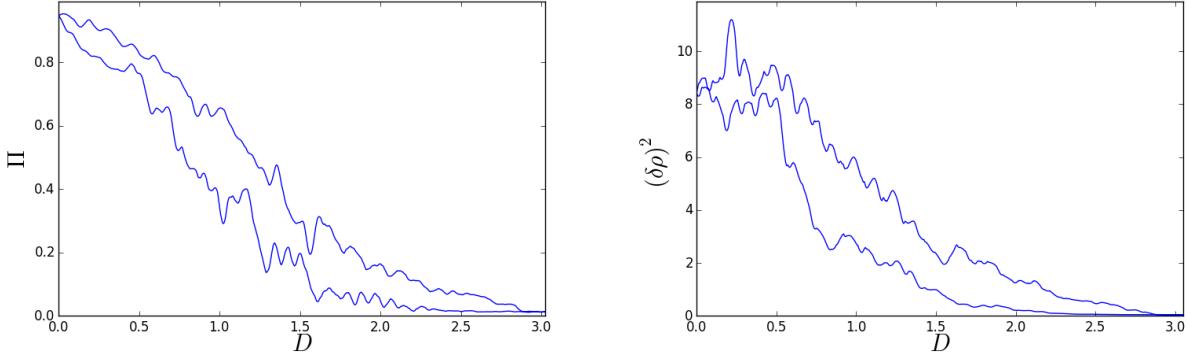


Figure 27: Hysteresis in the flying XY model, on the $A = 1, \rho = 0.5$ line with 8192 particles, from $D = 3.5$, down to $D = 10^{-4}$ and back, with total simulation time $t = 10000$. The results have been smoothed for readability. The transition point is around $D = 1.2$.

4.3 Phase diagram

The phase diagram will be sketched on the $A = 1$ and $D = 2$ planes, using 1000 particles simulations, with lengths between 1500 and 5000 time units. Most transition points were found using “vertical” lines, keeping A and D fixed, due to form of the transition surface in the region we looked at; it is computationally expensive to simulate large density systems, since this forces a large number of particles in turn (otherwise the box gets very small, and we wish to avoid a simulation where $L \approx R^*$). The region where ρ values are small on the transition surface is nearly parallel to the $A - D$ plane, and therefore “vertical” paths are a convenient choice. The phase diagram is shown in figure 28.

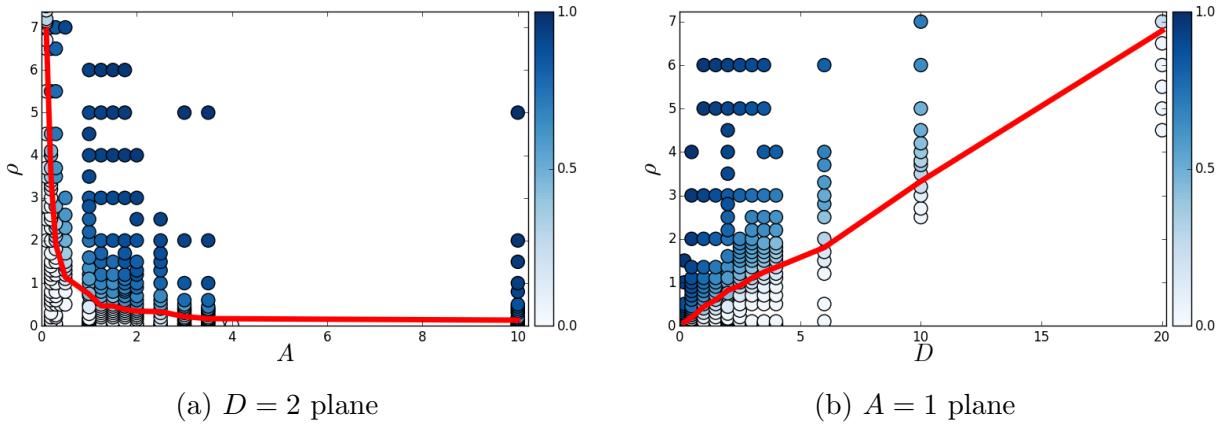


Figure 28: Flying XY transition curves, based on 1000 particle simulations. Each circle represents a simulation, and the color coding corresponds to the $\langle \Pi \rangle$ value of that particular simulation. The red lines represents the transition curves.

This phase diagram is different from the Vicsek one in that neither plot shows a vertical asymptote for finite parameter values. This is to be expected, as the flying XY dynamics in equation 1.6 contains no neighbor normalization, and so the alignment strength can grow arbitrarily with increasing density. The alignment and noise can therefore always be set to be on the same order of magnitude, by choosing an appropriate density. Another interesting observation is the roughly linear form of the transition curve on the $A = 1$ plane.

There is a theoretical result corresponding to a mean field hydrodynamic coarse-grained description of the flying XY model (see [8], [19]), which predicts the transition curve $\rho_c = \alpha \frac{D_c}{A_c}$,

where (A_c, D_c, ρ_c) is a point on the transition surface and α is some proportionality constant. This motivates us to plot all the transition points we found on a ρ vs D/A plot, as is done in figure 29. We observe a remarkable correspondence with our results; the linear fit quality is $R^2 > 0.99$. The numerically obtained proportionality constant is $\alpha = 0.339 \pm 0.003$.

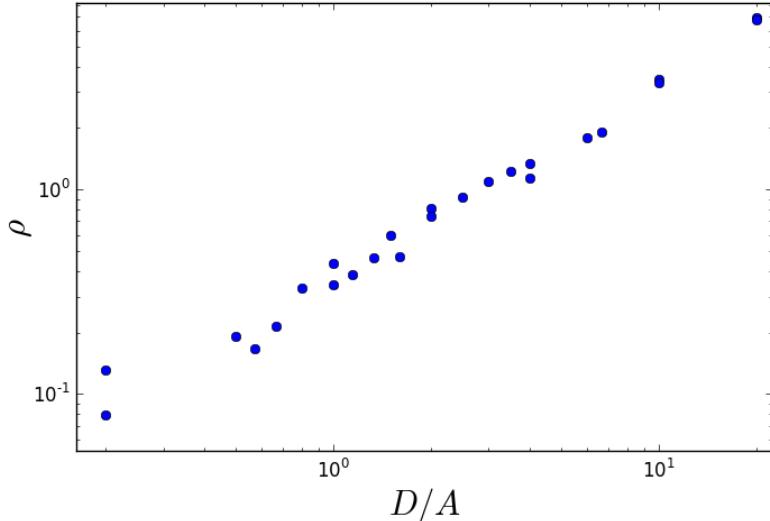


Figure 29: D/A conjecture for Flying XY transition. Transition points are taken from the $A = 1$ and $D = 2$ planes.

The success of this mean-field description is unexpected, since the theory does not take into account the spatial inhomogeneities of the transition nor its discontinuous nature. To understand where this theory comes from, we will take a look at the mean-field continuum equivalent of the flying XY. The configuration will be described using the *local momentum* field $\mathbf{W}(\mathbf{r}, t)$, called the *momentum field*. It is a coarse-grained description of the quantity $\rho(\mathbf{r}, t)\mathbf{v}(\mathbf{r}, t)$, where $\rho(\mathbf{r}, t)$ is the local density as defined in equation 3.3, and $\mathbf{v}(\mathbf{r}, t)$ is the local velocity field, defined as:

$$\mathbf{v}(\mathbf{r}, t) = \sum_i \mathbf{v}_i(t) \delta(\mathbf{r} - \mathbf{r}_i), \quad (4.1)$$

where $\mathbf{v}_i(t)$ is the i -th particle's velocity at time t . The coarse-graining is done in a manner analogous to that shown in section 3.1.

The dynamic evolution is taken from [8], and reads as follows (after dropping all space derivatives):

$$\frac{\partial}{\partial t} \mathbf{W} = \left(\frac{\pi}{2} A \rho - D \right) \mathbf{W} + \frac{\pi^2 A^2}{8D} W^2 \mathbf{W}, \quad (4.2)$$

where \mathbf{W} is the momentum field, and the parameters have been translated from the original paper to correspond to my units.

We want to analyze the linear stability of the disordered state, for which $W = 0$. By introducing a small perturbation and neglecting the higher order second term, we can solve and get

$$\mathbf{W}(t) = \mathbf{W}_0 \exp \left(\frac{\pi}{2} A \rho - D \right) t, \quad (4.3)$$

whose stability depends on the sign of the exponent. We can therefore deduce the transition surface to be

$$\rho_c = \frac{2}{\pi} \frac{D_c}{A_c} \approx 0.64 \frac{D_c}{A_c}, \quad (4.4)$$

where the proportionality constant $\frac{2}{\pi} \approx 0.64$ is approximately a factor of two from the measured value $\alpha \approx 0.339$.

Since the fit is so tight, we can attempt to plot the 3D transition surface, as is done in figure 30. We see the same two transition lines from figure 28, lying on their respective planes. Below this surface we have a gas state, and above it a globally polarized state.

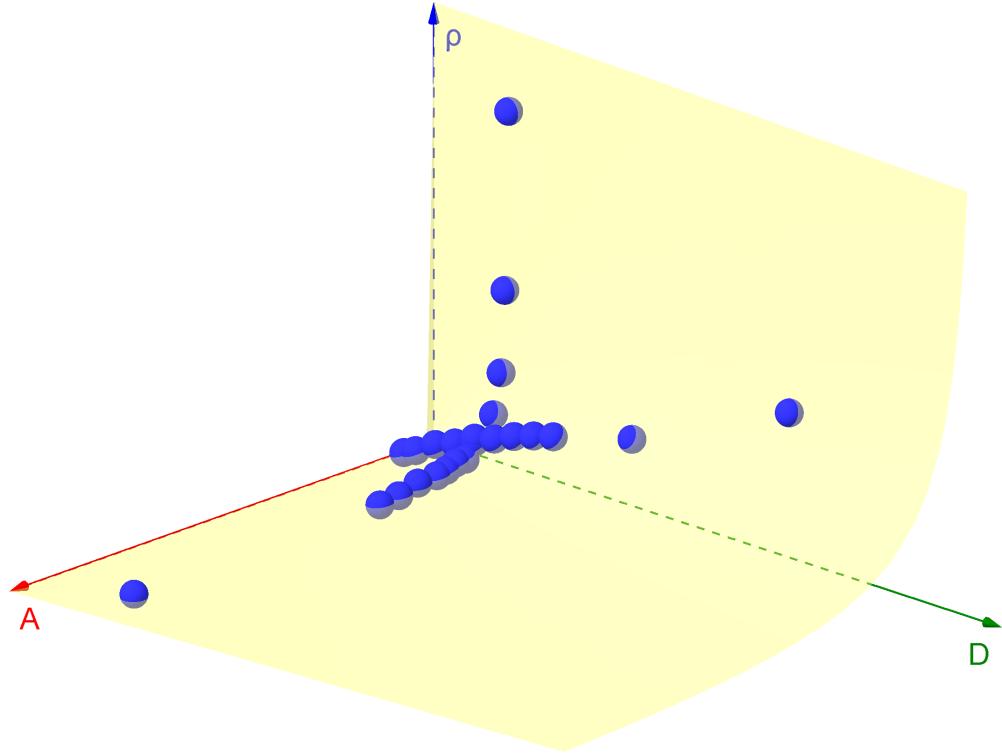


Figure 30: Flying XY Transition surface, fitted to the function $\rho = \alpha \frac{D}{A}$. We can see the transition points as red spheres along the $A = 1.0$ and $D = 2.0$ planes.

Plotting this surface is somewhat ambitious, considering that it is only based on two curves; it remains the subject of future studies to see whether additional points in other parameter ranges will indeed agree with it.

4.4 Spatial properties of emergent structures

At this point we must divert from our policy of analyzing the Vicsek and Flying XY model in an identical manner, since their spatial patterns are different. It is evident that in our parameter range, simulation scale, and geometry, the flying XY model produces local clusters; I would like to analyze this clustering behavior, and address the following questions:

1. How many particles are there in a typical cluster? Is there a characteristic cluster size?
2. What is the spatial extent of a cluster relative to R^* , and to the box size?
3. Do clusters sizes exhibit any critical behavior near the transition point, such as large fluctuations?

We will look at several observables that reflect the clustering behavior of the system. The first is the *cluster fraction*, which is defined as the fraction of particles in the simulation which “belong to a cluster”. A particle belongs to a cluster if it has more neighbors than it would if the particles were randomly placed in the box. More formally, the expected number of neighbors \tilde{N} in a randomly distributed system is proportional to the area of interaction of a single particle:

$$\tilde{N} \equiv \rho\pi(R^*)^2. \quad (4.5)$$

Using N_i , the number of neighboring particles within distance R^* from particle i , we say that a particle is inside a cluster if $N_i > \tilde{N}$. The fraction of particles fulfilling this condition is a measure of how phase-separated the system is. If a very low fraction of particles belong to clusters, we have a gassy system, and we expect this fraction to grow with the density. Results are shown in figure 31.

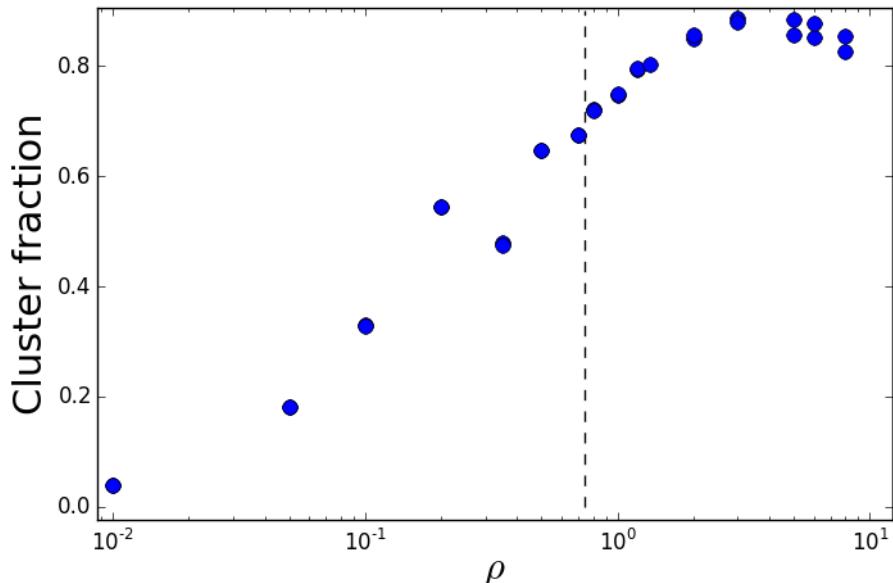


Figure 31: Mean cluster fraction along the $A = 1, D = 2$ line, $N = 4000$. The dashed vertical line shows the transition point.

Interestingly, while we see the expected increase in cluster membership, not much happens near the transition itself. There is no inflection point or sudden increase in the clustering

percentage, for example. This is in line with our qualitative observation that clusters grow continuously.

From here we will employ our notion of clusters as defined in section 2.2.3. This definition is equivalent to that used by Huepe and Aldana[20], and states that two particles are in the same cluster if there is a path between them passing only through neighboring particles, i.e. particles less than R^* apart. The “size” of such clusters may be defined either in terms of *number of particles* in a cluster or by its *gyration radius* R_g^A , which is a measure of its spatial extent, defined as:

$$(R_g^A)^2 \equiv \frac{1}{N_A} \sum_{i \in A} |\mathbf{r}_i - \mathbf{r}_{\text{c.m.}}^A|^2, \quad (4.6)$$

where N_A is the number of particles in cluster A , and $\mathbf{r}_{\text{c.m.}}^A$ is its center of mass position. We plot the mean values of both sizes in figure 32.

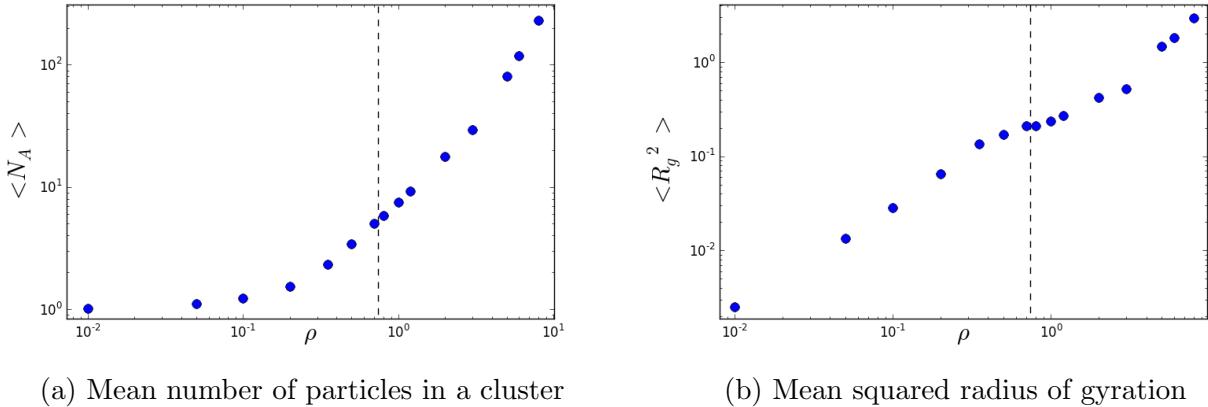


Figure 32: Clustering statistics, $A = 1.0$, $D = 2.0$, $N = 4000$.

We see the expected increase in cluster size with density, which appears gradual, and the most noteworthy feature is the possible inflection point in figure 32b near the transition. I would also point out that even in the higher density configurations we see very low values for both sizes, with respect to R^* and the number of particles in the simulation. This suggests that the mean is dominated by small clusters, and returns us to the question of how cluster sizes are distributed, which may be more informative than analyzing the mean values.

This is a good time to re-examine the question that was first raised after looking at figure 24: how can a system made of isotropically-oriented, independent clusters have a finite value of $\langle \Pi \rangle$? I propose two possible explanations:

1. Clusters are not independent but rather have correlated orientations.
2. The statistics are controlled entirely by a small number of very large clusters.

Starting with the second hypothesis, we will look at the distribution of cluster sizes. If this second explanation is correct, we expect a distribution with a “fat tail”. Results are plotted in figure 33, taking the number of particles as the measure of size (averaged over time, after discarding the initial transient).

Our first observation is that the distributions converge above a certain ρ value, which is lower than the transition. The second observation is this ordered state distribution is a power law. Fitting to $P(n) \approx a_0 n^{-p}$ gives an exponent $p = 1.73 \pm 0.04$, where the calculation is done on the slopes between $\rho = 0.7$ and $\rho = 5.0$. The smallest clusters were discarded (up

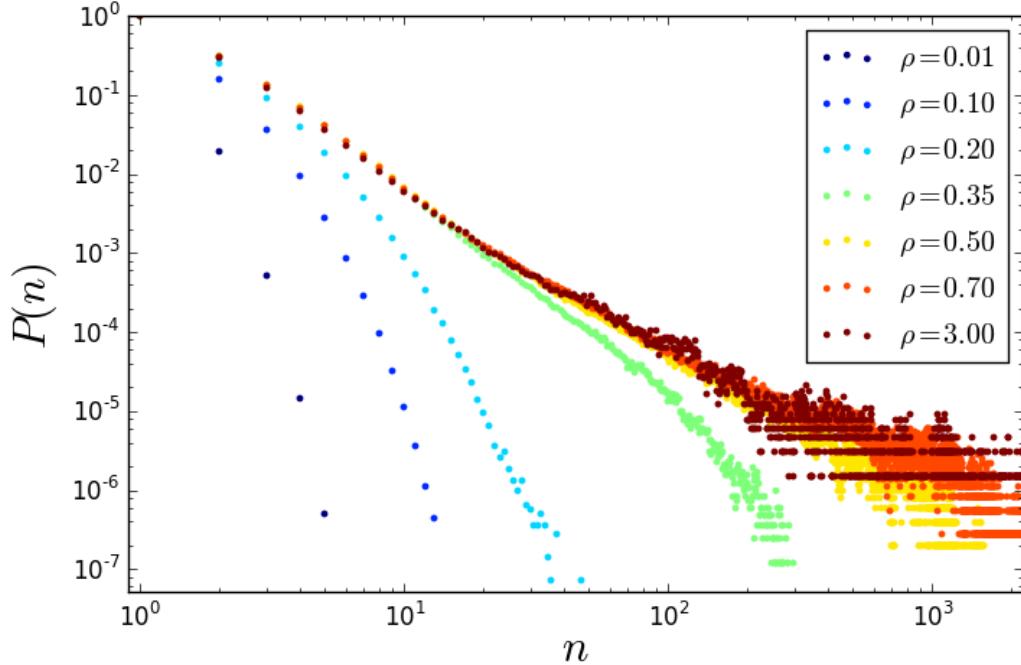


Figure 33: Cluster size distribution for several ρ values on the $A = 1, D = 1$ line, $N = 4000$. The transition is at $\rho \approx 0.74$. The data sets are normalized to coincide for the size-one clusters.

to $N_A = 20$), as they seems to follow a different slope. One slope was calculated for each ρ value, and the error estimate is the standard deviation of these slopes. The fit had on average $R^2 \approx 0.98$.

This result is consistent with [20], in which a power law distribution was similarly observed, $p \in (1.5, 1.9)$. Interestingly, there were not using the flying XY model but rather the discrete-time Vicsek model, for small noise values (too small for bands to stabilize). In such small noises some clustering can be observed, near the polar-liquid transition. I have not looked deeply enough into this region in the continuous-time Vicsek model, but a comparative study between low-noise Vicsek and near-transition flying XY clustering behavior can certainly be an interesting direction of inquiry.

Going back to the first possible explanation, we will look at the distribution of cluster orientations. If clusters tend to align towards the mean direction of polarization, we would have at least a partial for the finite $\langle \Pi \rangle$ value. To that end we define the mean orientation and cluster orientation:

$$\Theta(t) = \arg \left[\sum_{j=1}^N e^{i\theta_j(t)} \right], \quad \Theta_A(t) = \arg \left[\sum_{j \in A} e^{i\theta_j(t)} \right], \quad (4.7)$$

where the first sum is over all particles is the second over the particles in cluster A , and $\arg(re^{i\theta}) = \theta$ gives the argument of a complex number. We finally define a cluster's *relative* orientation as:

$$\Theta_A^*(t) = \Theta_A(t) - \Theta(t). \quad (4.8)$$

We can plot the distribution of $\Theta_A^*(t)$ to learn how the clusters tend to align with respect to the overall direction of polarization. Results are shown in figure 34, and show an isotropic orientation of the clusters with respect to $\Theta(t)$.

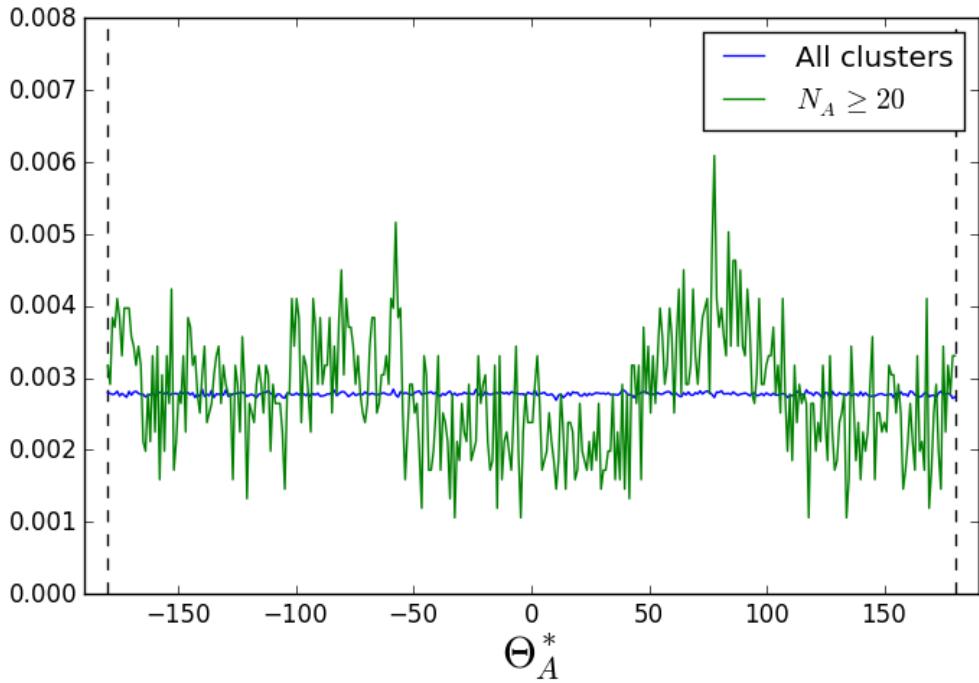


Figure 34: Distribution of relative cluster orientation, $A = 1, D = 2, D = 0.7, N = 4000$. Angles are measured in degrees. The blue curve contains data from all clusters while the green one only considers clusters with 20 particles or more.

To confront the question of whether very small randomly-aligned clusters are the primary reason for this uniformity, I made a separate set of measurements counting only clusters with more than 20 particles; however, both sets show a uniform distribution, suggesting that the overall polarization is not due to the clusters moving on average towards the mean orientation, or even to “big” clusters doing so.

In summary, it seems that the second explanation is the likelier one, as indicated by the power-law size distribution and lack of coherence between the cluster orientations with $\Theta(t)$, as assessed in figure 34.

5 Discussion

In this section I will present my main observations about the two models. I will address their qualitative behavior, and how they compare against each other and against the discrete-time Vicsek model. I will also discuss the nature of the phase transitions and the emerging spatial patterns.

5.1 Comparison of the discrete-time and continuous-time Vicsek models

I will begin this discussion with the basic observation that moving to a continuous-time Vicsek model does not alter the phenomenology. We still observe the three main phases of the Vicsek model, which are the disordered gas, the traveling bands, and the polarized liquid[5]. I have not looked deeply into the liquid state, but we do see the bands disappearing far enough below the transition surface.

While it is true that we have introduced an additional parameter in the continuous model, namely the alignment strength A , it seems that by fixing it to a constant we recover the discrete model's phase diagram, as seen in figure 18a (compare to Figure 8 in [5]). In particular, the transition line reaches a vertical asymptote for a certain value of D . This is trivial in the discrete case, since setting $\xi(t)$ in equation 1.1 to have a large enough variation implies that velocities fully decorrelate on each time step; we are thus left with a system of independent random rollers, and should not expect to observe a collective behavior. This asymptote is reproduced in the continuous-time model, due to the fact that the alignment term in equation 1.4 is bounded (on an $A = \text{const}$ plane), while the noise term grows with D ; therefore for large enough noise values the Brownian term dominates the dynamics, regardless of the density. Interestingly, a vertical asymptote can be observed in the fixed D case as well, for a similar reason, as seen in figure 18; this asymptote does not have a discrete-time counterpart.

It has been observed in [5] that a different noise term, termed *vectorial noise*, results in a much clearer discontinuity of the phase transition. This vectorial noise term is applied to each measured neighbor angle in the alignment term sum of equation 1.2, resulting in the evolution equation

$$\theta_i(t + \Delta t) = \frac{1}{N_i} \sum_{|\mathbf{r}_i(t) - \mathbf{r}(t)_j| < R^*} \sin [\theta_j(t) - \theta_i(t) + \xi_{ij}(t)], \quad (5.1)$$

which represents an error in the measurement of the individual neighbors' velocities, as opposed to an error in aligning one's direction of motion. This logic can be applied to the continuous model as well, with far less insightful results, as described in appendix B. The final expressions seen in equations B.15 and B.21 are only trivially different, and there is no reason to expect them to display different phase transition behavior. I did in fact perform some simulations using the vectorial noise term, but the results were too similar to the angular noise case to motivate further inquiry.

The last observation is that the bands in the two models have similar structure, with a steep front and an exponentially-decaying tail.

5.2 Comparison between the flying XY and the continuous-time Vicsek model

The most obvious difference between the two models are the spatial formations that appear close to the order-disorder transition surface. The Vicsek model displays extended bands while

the flying XY model displays clusters which grow continuously when approaching the transition. This difference is not only aesthetic, but also qualitative: the bands do not grow gradually, as they cannot exist as localized structures. That means they can only appear when they are stable enough to span the entire system size. Clusters, on the other hand, can also exist on small scales.

Looking at the phase diagrams in figures 18 and 28, we see several other notable discrepancies. It is reasonable that for given values of A and ρ the transition noise D_c would be higher for the flying XY model than for the Vicsek model. This is because the alignment torque is always stronger in the flying XY case, for equivalent particle configurations. We do not however witness a mere shifting or stretching of the transition surface, but some qualitative changes as well. The most evident of these is the absence of asymptotes for finite parameter values in the flying XY phase diagrams. This is due to the fact that unlike in the Vicsek model, the alignment torque strength grows arbitrarily with density, and so even a very noisy system can display collective motion.

5.3 Theoretical models for the transition surface

For the Vicsek model transition, we do not have a complete theory, we do have a mean-field prediction for the high-noise limit, which results in an asymptotic bound. To take the corresponding limit of the mean-field equation 4.2, we divide the alignment term by $\pi\rho$, to get (to first order)

$$\frac{\partial}{\partial t} \mathbf{W} = \left(\frac{1}{2}A - D \right) \mathbf{W}. \quad (5.2)$$

This change is justified by the following argument: we want to divide the alignment torque by a characteristic number of neighbors, which is $\rho\pi(R^*)^2 = \rho\pi$, since $R^* = 1$. This will represent, on average, the normalization by \mathcal{N}_i in equation 1.4. Note that there is no unit inconsistency since we are using unitless parameters. We therefore predict that in this regime the linear stability will depend on the sign of $\frac{1}{2}A - D$, predicting the asymptote $D/A = 0.5$, which is quite close that seen in figure 20, located at approximately $D/A = 0.45$.

For the flying XY model we have a more complete theory, also based on a mean-field approach, which results in the transition surface $\rho_c = \alpha \frac{D_c}{A_c}$. Here we find a difference of a factor 2 between the numerical results and the theoretical prediction, which give $\alpha \approx 0.339$ and $\alpha \approx 0.64$ respectively. This discrepancy is quite reasonable, since it sets the mean-field transition density to be higher than the observed order-disorder transition density. Since the latter results from localized clusters, it can be expected to precede the homogenous transition, and perhaps coincide with it for higher densities.

5.4 Order and characteristics of the phase transition

All three models (discrete-time Vicsek, continuous-time Vicsek, and flying XY) show a clear discontinuous transition at the onset of collective motion, as gauged by the mean polarization $\Pi(t)$. This observation is based on the scaling behavior analysis, the observed hysteresis, and to a lesser extent the Binder cumulant as plotted near the transition.

The continuous-time Vicsek model exhibits a sudden appearance of extended bands at sub-critical noise values, while the flying XY appears to scale up its clusters continuously. However, we do see a transition in the cluster size distribution, which converges to a power-law for densities near the transition, as seen in figure 33. This convergence may not coincide exactly with the transition exhibited in the the $\langle \Pi \rangle$ discontinuity, and further study is required to determine the relation between the two.

6 Perspective

The flying XY and continuous-time Vicsek are perhaps the two most basic continuous-time models of active matter. In light of the abundance of recent publications which use variations and complications of these two models, it is perhaps worthwhile to study in more detail the phenomenology of these two basic models. Such understanding might serve as a more sturdy foundation for future theoretical work. To that end, I would like to suggest several possible directions of study, in no particular order.

6.1 Clustering behavior

Clustering behavior has been observed by me in the flying XY model and by others in the discrete-time Vicsek model, in the low-noise regime. It would be interesting to compare the clustering behavior of the flying XY model to that of the Vicsek model. For example, I suspect that Vicsek clusters might show a stronger overall tendency to align towards the overall polarization direction, and could form using a different mechanism than the flying XY ones. This is because there is a discontinuity in the spatial formation in the Vicsek model which I have not observed in the flying XY model, i.e. the band phase.

Similarly, the cluster size distribution around the transition could be further studied. In particular, we could examine higher moments (i.e. the variance and kurtosis), in order to discern possible critical behavior. We could also look at size distributions for sub-critical values of ρ , which do not correspond to a power-law: are they Gaussian? Do they have an associated length scale? How does the distribution tend to a power-law with increasing ρ ?

The cluster orientation distribution could also be compared to the distribution of the local polarization field $\mathbf{W}(\mathbf{r}, t)$, to learn about the relative importance of dense and dilute regions setting the overall polarization. We might also plot spatial correlation functions for the cluster orientation, perhaps factored somehow by the cluster size. We could hypothesize that there is an associated correlation length, which may show some critical behavior near the transition.

Finally, we could study the evolution of Θ in time, using finer time sampling, to learn whether it is a smooth function on time scales shorter than the alignment time. The purpose of such a study is to learn about the effect of large clusters joining and splitting on the overall polarization, and perhaps to motivate a more general study of cluster dynamics.

6.2 Higher-dimensional systems

The discrete-time Vicsek model has been studied in three dimensions and has shown similar phenomenology, with extended “sheets” instead of bands near the transition. While I would assume that the continuous-time version would be similar, it is not at all obvious to me how the flying XY model phenomenology would change in a 3D system.

6.3 Correspondence between discrete- and continuous-time Vicsek

I have by no means surveyed the entire phenomenological study which was done on the Vicsek model. Many other peculiarities have been observed, such as multiple bands (for different geometries and boundary conditions), giant density fluctuations, and transverse superdiffusion[5], among others. All of these could potentially be verified on the continuous-time model as well.

6.4 Time dynamics and coarsening

In my work I have observed very large fluctuations of $\Pi(t)$ in time for systems close to the transition; in fact, the point of maximum fluctuations could be used as an alternative definition

for the transition point (rather than the $\langle \Pi \rangle$ inflection point). However, a more detailed study could be done, addressing the following questions:

1. Are these fluctuations random, or do we see jumping back and forth between the ordered and disordered state?
2. Performing a spectral analysis of these fluctuations, how the system size affects the frequency of fluctuations?

In addition, we might be interested in the dynamics of *coarsening*, in the flying XY case, and how they compared to known models and theories. This question can be asked about the Vicsek model as well: how long does it take for the system to settle in its stable ordered phase? Can we develop a theory for the time evolution of the order parameter?

6.5 Theoretical work regarding the phase transition

While some predictions exist for both models, a complete theory describing the phase transition surface does not yet exist. This effort could be complemented by a comparative numerical study of the continuum model with agent-based simulations like my own. Even without a complete theory, perhaps we could explain various aspects of my results, such as the tight linear fit of figure 29.

7 Acknowledgments

I would like to thank my adviser Denis Bartolo for guiding me through my thesis and providing helpful advice and support. I would also like to thank all the people who worked alongside me in the ENS Lyon Physics laboratory. In particular I thank Vijayakumar Chikkadi, who helped me understand my results and answered a lot of my questions, Joost Weijns, who advised me on numerical techniques, and Jean-Baptiste Caussin, who followed my progress and helped me with several theoretical aspects. I also wish to thank the AtoSIM Erasmus Mundus program administrators, who took care of me and my fellow students for the last two years, that were full of both academic and technical challenges.

A Alignment potential

The alignment term in equation 1.6 can be derived from a potential energy of the form:

$$E_{\text{align}} = -A \sum_{|\mathbf{r}_i - \mathbf{r}_j| < R^*} \mathbf{v}_i \cdot \mathbf{v}_j. \quad (\text{A.1})$$

To see that we indeed get the same equation of motion, we use the general formula $\dot{\mathcal{B}} = -\frac{\partial E}{\partial B}$, and so get:

$$\dot{\theta}_i = -\frac{\partial}{\partial \theta_i} E_{\text{align}} = \sum_{|\mathbf{r}_i - \mathbf{r}_j| < R^*} \frac{\partial}{\partial \theta_i} (\mathbf{v}_i \cdot \mathbf{v}_j) \quad (\text{A.2})$$

where the summation is done only on j , and we have discarded all the terms which don't contain \mathbf{v}_i as they will vanish when taking the derivative. We also set $A = 1$ for notational brevity. We continue with only the term in the sum:

$$\begin{aligned} \frac{\partial}{\partial \theta_i} (\mathbf{v}_i \cdot \mathbf{v}_j) &= \mathbf{v}_j \cdot \frac{\partial}{\partial \theta_i} \mathbf{v}_i = \mathbf{v}_j \cdot \frac{\partial}{\partial \theta_i} (\cos \theta_i, \sin \theta_i) \\ &= (\cos \theta_j, \sin \theta_j) \cdot (-\sin \theta_i, \cos \theta_i) \\ &= \sin \theta_j \cos \theta_i - \cos \theta_j \sin \theta_i \\ &= \sin(\theta_j - \theta_i) \end{aligned} \quad (\text{A.3})$$

the identity $\sin(\alpha - \beta) = \sin \alpha \cos \beta - \cos \alpha \sin \beta$ was used in the last step. We ended up with equation 1.6.

The Vicsek model is not derivable from a similar potential, because of the normalization \mathcal{N}_i , which destroys the pairwise symmetry. To see this, we write the form of a generic potential which is a sum of pairwise interactions, and which is also invariant under rotations (so it contains only angle differences):

$$E_{\text{Vicsek}} = \sum_{i < j} U(\theta_i - \theta_j), \quad (\text{A.4})$$

where $U(\theta)$ is some pairwise interaction potential. This potential would adhere to the action-reaction law:

$$F_{ji} = \frac{\partial}{\partial \theta_i} U(\theta_i - \theta_j) = -\frac{\partial}{\partial \theta_j} U(\theta_i - \theta_j) = -F_{ij}, \quad (\text{A.5})$$

where F_{ij} is the torque felt by particle i due to particle j . Therefore, no such potential exists for the Vicsek model.

B Vectorial versus angular noise

Let us focus on the Brownian noise term. The way we have written the equations of motion 1.4 and 1.6, we have added a noise term after calculating the alignment term noiselessly. This could correspond to a particle measuring exactly the direction of motion of its neighbors, but then making a mistake when trying to align its own velocity towards the mean direction. This is referred to as the *Angular* noise model and is reminiscent of the one employed in the original Vicsek model. However, it is only one way to add a Brownian noise, and we will look at several alternatives.

B.1 No noise

As a first step we will rewrite the noiseless equations of motion in vector form:

$$\dot{\mathbf{v}}_i = A \left[\mathbf{n}_i \left(\sum_{|\mathbf{r}_i - \mathbf{r}_j| < R^*} \mathbf{v}_j \right) \right] \mathbf{n}_i, \quad (\text{B.1})$$

where \mathbf{v}_i is the i -th particle momentum, and \mathbf{n}_i is a unit vector normal to \mathbf{v}_i . We observe that the force is always perpendicular to the momentum vector (thereby not altering the speed), and any neighbor that is moving in the same direction as particle i does not cause a torque, as is to be expected. It is useful to rewrite B.1 using projection operators:

$$\dot{\mathbf{v}}_i = A \left[O_i \sum_{|\mathbf{r}_i - \mathbf{r}_j| < R^*} \mathbf{v}_j \right], \quad (\text{B.2})$$

where $O_i \mathbf{a} = [\mathbf{n}_i \cdot \mathbf{a}] \mathbf{n}_i$ is a projection operator on the direction normal to \mathbf{v}_i . We observe the similarity to the discrete-time Vicsek dynamics in B.2: dividing the sum by N_i , we have precisely a torque towards the mean direction of the neighbors. In the $A \rightarrow \infty$ limit, we have instantaneous alignment; this is a more formal way of describing the link between the discrete- and continuous-time Vicsek models.

The projection operator is sometimes written in the form: $O_i = I - \hat{\mathbf{p}}_i \times \hat{\mathbf{p}}_i$, where I is the identity matrix, $\hat{\mathbf{p}}_i$ is the normalized i -th particle momentum, and \times denotes an outer product. To show the equivalence of this form, we take

$$\mathbf{n}_i \equiv \hat{\mathbf{z}} \wedge \hat{\mathbf{p}}_i = \begin{pmatrix} -\sin \theta_i \\ \cos \theta_i \end{pmatrix}, \quad (\text{B.3})$$

so that

$$O_i \mathbf{a} = [\mathbf{n}_i \cdot \mathbf{a}] \mathbf{n}_i = \left[(-\sin \theta_i \quad \cos \theta_i) \begin{pmatrix} a_x \\ a_y \end{pmatrix} \right] \begin{pmatrix} -\sin \theta_i \\ \cos \theta_i \end{pmatrix}, \quad (\text{B.4})$$

and expanding:

$$= [-a_x \sin \theta_i + a_y \cos \theta_i] \begin{pmatrix} -\sin \theta_i \\ \cos \theta_i \end{pmatrix} = \begin{pmatrix} a_x \sin^2 \theta_i - a_y \sin \theta_i \cos \theta_i \\ -a_x \sin \theta_i \cos \theta_i + a_y \cos^2 \theta_i \end{pmatrix}. \quad (\text{B.5})$$

Using the identity $\sin^2 \alpha + \cos^2 \alpha = 1$, we have:

$$\begin{aligned} O_i \mathbf{a} &= \begin{pmatrix} a_x - a_x \cos^2 \theta_i - a_y \sin \theta_i \cos \theta_i \\ a_y - a_x \sin \theta_i \cos \theta_i - a_y \sin^2 \theta_i \end{pmatrix} \\ &= \begin{pmatrix} a_x - a_x p_{ix} p_{ix} - a_y p_{ix} p_{iy} \\ a_y - a_x p_{ix} p_{iy} - a_y p_{iy} p_{iy} \end{pmatrix} \\ &= \begin{pmatrix} 1 - p_{ix} p_{ix} & -p_{ix} p_{iy} \\ -p_{ix} p_{iy} & 1 - p_{iy} p_{iy} \end{pmatrix} \begin{pmatrix} a_x \\ a_y \end{pmatrix}. \end{aligned} \quad (\text{B.6})$$

Going back to the first form of O_i , we decompose all vectors and operators into their components. A and v_0 will be set to 1 for notational brevity.

$$\begin{aligned} O_i &= \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} - \begin{pmatrix} p_{ix} \\ p_{iy} \end{pmatrix} \begin{pmatrix} p_{ix} & p_{iy} \end{pmatrix} \\ &= \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} - \begin{pmatrix} p_{ix}p_{ix} & p_{ix}p_{iy} \\ p_{ix}p_{iy} & p_{ix}p_{iy} \end{pmatrix} \\ &= \begin{pmatrix} 1 - p_{ix}p_{ix} & -p_{ix}p_{iy} \\ -p_{ix}p_{iy} & 1 - p_{iy}p_{iy} \end{pmatrix}, \end{aligned} \quad (\text{B.7})$$

which is equivalent to equation B.6, thus showing that $O_i \mathbf{a} = [\mathbf{n}_i \cdot \mathbf{a}] \mathbf{n}_i = (I - \hat{\mathbf{p}}_i \times \hat{\mathbf{p}}_i) \mathbf{a}$.

Continuing with the derivation, we let O_i operate on the vector \mathbf{v}_j :

$$O_i \mathbf{v}_j = \begin{pmatrix} 1 - p_{ix}p_{ix} & -p_{ix}p_{iy} \\ -p_{ix}p_{iy} & 1 - p_{iy}p_{iy} \end{pmatrix} \begin{pmatrix} p_{jx} \\ p_{jy} \end{pmatrix}. \quad (\text{B.8})$$

Perhaps now is a good time to move to angle notation, based on equation 1.3:

$$\begin{aligned} O_i \mathbf{v}_j &= \begin{pmatrix} 1 - \cos^2 \theta_i & -\sin \theta_i \cos \theta_i \\ -\sin \theta_i \cos \theta_i & 1 - \sin^2 \theta_i \end{pmatrix} \begin{pmatrix} \cos \theta_j \\ \sin \theta_j \end{pmatrix} \\ &= \begin{pmatrix} \sin^2 \theta_i & -\sin \theta_i \cos \theta_i \\ -\sin \theta_i \cos \theta_i & \cos^2 \theta_i \end{pmatrix} \begin{pmatrix} \cos \theta_j \\ \sin \theta_j \end{pmatrix} \\ &= \begin{pmatrix} \sin^2 \theta_i \cos \theta_j - \sin \theta_i \cos \theta_i \sin \theta_j \\ -\sin \theta_i \cos \theta_i \cos \theta_j + \cos^2 \theta_i \sin \theta_j \end{pmatrix}, \end{aligned} \quad (\text{B.9})$$

and doing the same for left hand side of equation B.1:

$$\dot{\mathbf{v}}_i = \begin{pmatrix} \dot{p}_{ix} \\ \dot{p}_{iy} \end{pmatrix} = \frac{d}{dt} \begin{pmatrix} \cos \theta_i \\ \sin \theta_i \end{pmatrix} = \dot{\theta}_i \begin{pmatrix} -\sin \theta_i \\ \cos \theta_i \end{pmatrix}. \quad (\text{B.10})$$

Putting everything together we have

$$\dot{\theta}_i \begin{pmatrix} 1 \\ 1 \end{pmatrix} = \sum_{|\mathbf{r}_i - \mathbf{r}_j| < R^*} \begin{pmatrix} \sin \theta_j \cos \theta_i - \cos \theta_j \sin \theta_i \\ \sin \theta_j \cos \theta_i - \cos \theta_j \sin \theta_i \end{pmatrix}, \quad (\text{B.11})$$

and employing the identity $\sin(\alpha - \beta) = \sin \alpha \cos \beta - \cos \alpha \sin \beta$:

$$\dot{\theta}_i = \sum_{|\mathbf{r}_i - \mathbf{r}_j| < R^*} \sin(\theta_j - \theta_i), \quad (\text{B.12})$$

and we recovered the original Flying XY alignment term. The procedure for Vicsek is identical, except for a factor of \mathcal{N}_i .

B.2 Angular Noise

If we want to introduce angular noise and write the dynamics in vector form, we can try the following:

$$\dot{\mathbf{v}}_i = \left[\mathbf{n}_i \left(\sum_{|\mathbf{r}_i - \mathbf{r}_j| < R^*} \mathbf{v}_j \right) + \xi_i^p \right] \mathbf{n}_i, \quad (\text{B.13})$$

where ξ_i^p is a random variable fulfilling the condition given in equation 1.5. Conceptually, this corresponds to the idea that after computing the neighbor interaction without error, we make a random mistake in aligning towards the average direction. Linearity allows us to treat the two terms separately: where ξ_i^p is a random variable fulfilling the condition given in equation 1.5. Conceptually, this corresponds to the idea that after computing the neighbor interaction without error, we make a random mistake in aligning towards the average direction. Linearity allows us to treat the two terms separately:

$$\mathbf{v}_i = O_i \left(\sum_{|\mathbf{r}_i - \mathbf{r}_j| < R^*} \mathbf{v}_j \right) + \xi_i^p \mathbf{n}_i. \quad (\text{B.14})$$

The components of the second term can be obtained by taking $\mathbf{n}_i = \begin{pmatrix} -\sin \theta_i \\ \cos \theta_i \end{pmatrix}$, and we have, in the spirit of equation B.12:

$$\dot{\theta}_i = \sum_{|\mathbf{r}_i - \mathbf{r}_j| < R^*} \sin(\theta_j - \theta_i), + \xi_i^p \quad (\text{B.15})$$

and we recovered the noise term of equations 1.4 and 1.6.

B.3 Vectorial Noise

The next noise type we will look at is not introduced separately from the alignment interaction, as a scalar, but rather as a vectorial “error in measurement”:

$$\mathbf{v}_i = \left[\mathbf{n}_i \sum_{|\mathbf{r}_i - \mathbf{r}_j| < R^*} (\mathbf{v}_j + \boldsymbol{\xi}_i) \right] \mathbf{n}_i. \quad (\text{B.16})$$

Now we have to discuss separately the Vicsek and the Flying XY situation, since they would differ by a factor of \mathcal{N}_i . This has non-trivial implications for the flying XY model, since it means that not all particles will experience the same Brownian noise (i.e. those with more neighbors will be more perturbed). For the purpose of this discussion we will consider the Flying XY model.

Once again, due to linearity, the terms may be treated separately:

$$\mathbf{v}_i = O_i \left(\sum_{|\mathbf{r}_i - \mathbf{r}_j| < R^*} \mathbf{v}_j \right) + \mathcal{N}_i O_i \boldsymbol{\xi}_i, \quad (\text{B.17})$$

and so we can clearly see the difference between the two noise terms in equations B.15 and B.17:

Angular Noise	Vectorial Noise
$\xi_i \mathbf{n}_i$	$[\mathbf{n}_i \cdot \boldsymbol{\xi}_i] \mathbf{n}_i$ (Vicsek)
	$\mathcal{N}_i [\mathbf{n}_i \cdot \boldsymbol{\xi}] \mathbf{n}_i$ (Flying XY)

In the angular noise case ξ_i is a scalar random variable, while the vectorial $\boldsymbol{\xi}_i$ is a vector of dimension 2. We can construct an analog of equation 2.3 for the vectorial case:

$$\boldsymbol{\xi}_i(t) = \sqrt{\frac{2D}{\Delta t}} \boldsymbol{\varphi}_i(t), \quad (\text{B.18})$$

where we can take ξ_i to be, for example, a random vector on the unit circle (or rather a circle with radius $\eta = \sqrt{2D}$). Notice that we no longer have the freedom to choose any random distribution, since there is no particular mean value to center around. Let us label ξ 's two components using an angle: $\xi = \xi(\cos \alpha, \sin \alpha)$, and calculate the vectorial noise term:

$$\begin{aligned} O_i \xi_i &= \left[(-\sin \theta_i \quad \cos \theta_i) \begin{pmatrix} \cos \alpha \\ \sin \alpha \end{pmatrix} \right] \begin{pmatrix} -\sin \theta_i \\ \cos \theta_i \end{pmatrix} \\ &= (-\sin \theta_i \cos \alpha + \cos \theta_i \sin \alpha) \begin{pmatrix} -\sin \theta_i \\ \cos \theta_i \end{pmatrix} \\ &= \sin(\alpha - \theta_i) \begin{pmatrix} -\sin \theta_i \\ \cos \theta_i \end{pmatrix}, \end{aligned} \tag{B.19}$$

So we get the angular evolution equation:

$$\dot{\theta}_i = \sum_{|\mathbf{r}_i - \mathbf{r}_j| < R^*} \sin(\theta_j - \theta_i) + \eta \sin(\alpha - \theta_i). \tag{B.20}$$

Finally, since α is a random variable in $[0, 2\pi]$, we can equivalently use the expression:

$$\dot{\theta}_i = \sum_{|\mathbf{r}_i - \mathbf{r}_j| < R^*} \sin(\theta_j - \theta_i) + \eta \sin \alpha, \tag{B.21}$$

which describes the angular dynamics in the case of vectorial noise.

References

- [1] Tamás Vicsek, András Czirók, Eshel Ben-Jacob, Inon Cohen, and Ofer Shochet. Novel type of phase transition in a system of self-driven particles. *Phys. Rev. Lett.*, 75:1226–1229, Aug 1995.
- [2] Yoshie Harada, Akira Noguchi, Akiyoshi Kishino, and Toshio Yanagida. Collective motion of self-propelled particles interacting without cohesion. *Nature (London)*, 326:805, Apr 1987.
- [3] N. G. Cogan and Charles W. Wolgemuth. Pattern formation by bacteria-driven flow. *Biophys J*, 88(4):2525–2529, Apr 2005. 53348[PII].
- [4] Antoine Bricard, Jean-Baptiste Caussin, Desreumaux Nicolas, Olivier Dauchot, and Bartolo Denis. Emergence of macroscopic directed motion in populations of motile colloids. *Nature*, 503(7474):95, Nov 2013.
- [5] Hugues Chaté, Francesco Ginelli, Guillaume Grégoire, and Franck Raynaud. Collective motion of self-propelled particles interacting without cohesion. *Phys. Rev. E*, 77:046113, Apr 2008.
- [6] M. Aldana, H. Larralde, and B. Vázquez. On the emergence of collective order in swarming systems: A recent debate. *International Journal of Modern Physics B*, 23(18):3661–3685, 2009.
- [7] Félix Ginot, Isaac Theurkauff, Demian Levis, Christophe Ybert, Lydéric Bocquet, Ludovic Berthier, and Cécile Cottin-Bizonne. Nonequilibrium equation of state in suspensions of active colloids. *Phys. Rev. X*, 5:011004, Jan 2015.
- [8] F. D. C. Farrell, M. C. Marchetti, D. Marenduzzo, and J. Tailleur. Pattern formation in self-propelled particles with density-dependent motility. *Phys. Rev. Lett.*, 108:248101, Jun 2012.
- [9] F. Peruani, A. Deutsch, and M. Br. A mean-field theory for self-propelled particles interacting by velocity alignment mechanisms. *The European Physical Journal Special Topics*, 157(1):111–122, 2008.
- [10] William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery. *Numerical Recipes 3rd Edition: The Art of Scientific Computing*. Cambridge University Press, New York, NY, USA, 3 edition, 2007.
- [11] Loup Verlet. Computer “experiments” on classical fluids. i. thermodynamical properties of lennard-jones molecules. *Phys. Rev.*, 159:98–103, Jul 1967.
- [12] W. Mattson and B. M. Rice. Near-neighbor calculations using a modified cell-linked list method. *Computer Physics Communications*, 119:135–148, June 1999.
- [13] David Baraff. *Dynamic Simulation of Non-Penetrating Rigid Bodies*. PhD thesis, Department of Computer Science, Cornell University, March 1992.
- [14] Fernando Peruani and Luis G. Morelli. Self-propelled particles with fluctuating speed and direction of motion in two dimensions. *Phys. Rev. Lett.*, 99:010602, Jul 2007.
- [15] John Hopcroft and Robert Tarjan. Algorithm 447: Efficient algorithms for graph manipulation. *Commun. ACM*, 16(6):372–378, June 1973.

- [16] NetworkX: Python software for complex networks. <http://networkx.github.io/>.
- [17] Máté Nagy, István Daruka, and Tamás Vicsek. New aspects of the continuous phase transition in the scalar noise model (snm) of collective motion. *Physica A: Statistical Mechanics and its Applications*, 373:445–454, 2007.
- [18] K. Binder. Finite size scaling analysis of ising model block distribution functions. *Zeitschrift für Physik B Condensed Matter*, 43(2):119–140, 1981.
- [19] Eric Bertin, Michel Droz, and Guillaume Grégoire. Hydrodynamic equations for self-propelled particles: microscopic derivation and stability analysis. *Journal of Physics A: Mathematical and Theoretical*, 42(44):445001, 2009.
- [20] Cristián Huepe and Maximino Aldana. Intermittency and clustering in a system of self-driven particles. *Phys. Rev. Lett.*, 92:168701, Apr 2004.