

**A Thesis Submitted for the Degree of PhD at the University of Warwick**

**Permanent WRAP URL:**

<http://wrap.warwick.ac.uk/177148>

**Copyright and reuse:**

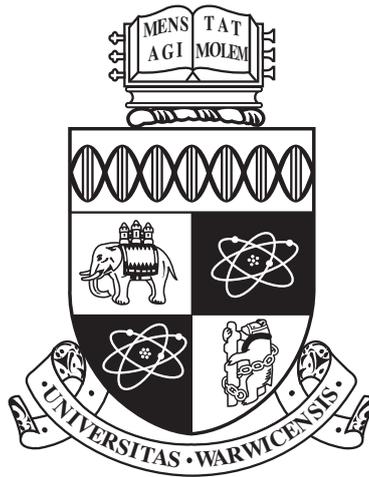
This thesis is made available online and is protected by original copyright.

Please scroll down to view the document itself.

Please refer to the repository record for this item for information to help you to cite it.

Our policy information is available from the repository home page.

For more information, please contact the WRAP Team at: [wrap@warwick.ac.uk](mailto:wrap@warwick.ac.uk)



**Top-down and Bottom-up Models of Collective  
Motion**

by

**Harvey L Devereux**

**Thesis**

Submitted to the University of Warwick

for the degree of

**Doctor of Philosophy**

**Warwick Mathematics Institute**

October 2022

THE UNIVERSITY OF  
**WARWICK**

# Contents

<b>List of Tables</b>	<b>iv</b>
<b>List of Figures</b>	<b>v</b>
<b>Acknowledgments</b>	<b>xvii</b>
<b>Declarations</b>	<b>xviii</b>
<b>Abstract</b>	<b>xix</b>
<b>Abbreviations</b>	<b>xx</b>
<b>Chapter 1 Introduction</b>	<b>1</b>
1.1 Active Matter . . . . .	1
1.1.1 Experimental and Real World Examples . . . . .	1
1.1.2 Theoretical Approaches . . . . .	3
1.2 Top Down to Bottom Up . . . . .	10
1.2.1 Intrinsically Motivated Collective Motion . . . . .	11
1.2.2 Intelligent Decisions and Causal Entropy . . . . .	15
1.3 Outline . . . . .	15
<b>Chapter 2 A Data Driven Study of Whirligig Beetles</b>	<b>19</b>
2.1 Introduction . . . . .	19
2.1.1 Experimental Data . . . . .	20
2.2 Towards a Measure of Local Density . . . . .	23
2.3 Self-Propulsion Statistics . . . . .	29
2.3.1 Inertial Delay . . . . .	29
2.3.2 Speed and Density . . . . .	30
2.3.3 Features in the Speed Density Relationship . . . . .	30
2.4 Corralled Active Brownian Particles . . . . .	37

2.5	Fitting the Model . . . . .	38
2.6	Phase Behaviour . . . . .	41
2.7	Conclusions . . . . .	42
<b>Chapter 3 Collective Motion from Future Path Entropy Maximisation</b>		<b>44</b>
3.1	Visual Input and the Cognitive Process . . . . .	45
3.1.1	Discretised Sensor Entropy . . . . .	51
3.1.2	Compression Entropy . . . . .	52
3.2	Both Path Entropy Methods are Phenomenologically Equivalent . .	59
3.2.1	State Count Path Entropy . . . . .	64
3.2.2	Compression Entropy . . . . .	65
3.3	Phenotypes . . . . .	67
3.4	Conclusions . . . . .	69
<b>Chapter 4 Visual Information and the Cognitive Process</b>		<b>74</b>
4.1	Perturbing Visual Information Deterministically: Occlusion . . . . .	75
4.1.1	Explicit Occlusion Model . . . . .	76
4.1.2	Effect of the Occlusion Threshold . . . . .	79
4.2	The Role of Noise in the Cognitive Process . . . . .	89
4.2.1	Vicsek-Like Noise . . . . .	90
4.2.2	Cognitive Noise . . . . .	92
4.3	Conclusions . . . . .	94
<b>Chapter 5 Monte-Carlo Sampling of Future Paths</b>		<b>96</b>
5.1	Monte-Carlo Future State Maximisation on a Finite Action Space . .	96
5.1.1	Sampling Actions . . . . .	97
5.1.2	Sampling Paths Without Replacement . . . . .	100
5.2	Partially Sampling Deeper Tree Leads to Higher Order . . . . .	103
5.3	Sampling Actions: Applying a Monte-Carlo Tree Search . . . . .	105
5.3.1	FSM and Monte-Carlo Tree Searching . . . . .	107
5.3.2	Does Uniform Action Sampling Lead to Ordering? . . . . .	111
5.4	Continuous Action Spaces . . . . .	112
5.4.1	Phenotypes . . . . .	117
5.5	Conclusions . . . . .	117
<b>Chapter 6 Conclusions and Future Work</b>		<b>120</b>
<b>Appendix A Online Entropy Calculation</b>		<b>134</b>



# List of Tables

2.1	Best fit parameters corresponding to the data in figure 2.8. Note the exponent $\alpha$ is a dimensionless number, and we have re-scaled lengths to the beetle length/ particle diameter. . . . .	40
-----	---	----

# List of Figures

1.1	Four macroscopic examples of Active Matter (a) a group of Barracuda fish, Rob Hughes, cc-by-sa-2.0 (b) a man-made swarm of “Kilobots” capable of group communication and individual programming [2], cc by-sa 4.0 (c) a Honey Bee swarm cc by 3.0 (d) a swarm of Starlings and predator, cc by-sa 4.0. Each image was sourced from Wiki-media Commons. . . . .	2
1.2	The four phenotypes found from the Couzin model, adapted from [36]. (A) agents swarm by remaining cohesive but not ordered. (B) milling agents characterised by circular motion. (C) a roughly aligned group and (D) a group showing almost total alignment. . . . .	6
1.3	Visual summary of intrinsically motivated collective motion, reproduced from [71] (A) visual depiction of the action space. (B) a diagram showing an example visual state calculated by first finding the visual projection (thick blue lines) and then determining the coverage of each sensor (segments defined by thin black lines) for the central (red) agent. (C) The future states tree generated by actions (A) showing one complete branch up to time $\tau = 5$ into the future, as well as one ballistic agent and future collisions with it (dashed red circles). . . . .	12

1.4	Detailed formulation of the visual projection angles as defined in [73] and used in [71] (red, dashed triangles) and the tangent line method (blue, solid triangles). a, b, and c show the construction for increasing inter-agent spacing $x$ along the $x$ -axis for clarity. The methods differ at small distances, we will use both in this thesis, partly for exact comparisons sake and secondly to show the minor impact of choosing one over the other. Motivation for the arc-tangent approach is that particles are defined as “phantoms” with no constraints on overlaps, even a small overlap will lead to the arc-sine formulation becoming undefined. . . . .	13
2.1	Snapshot of experimental footage of $N = 200$ Whirligig beetles within the circular arena of diameter 95.25 cm. A typical beetle measures $12 \pm 1$ mm along its body length. The body itself is roughly elliptical with a 2 : 1 aspect ratio between its major and minor axes. The beetles are able to propel themselves freely on the waters surface and are not observed diving in the footage used for our analysis. The experiments used for this chapter do not involve startling of the beetles or any other stimulus except for the constant illumination of the arena (730 lx). The camera is placed for a top-down view of the tank, captured at spatial resolution $1920 \times 1080$ and time resolution of 30 frames per second. . . . .	20
2.2	An overlay showing the Delaunay Tessellation local density computation applied directly to the data shown in figure 2.1 once positions were extracted. The inset details the geometry relating to the starred beetle indicated by particle index $i$ (associated with a position coordinate $\mathbf{x}_i$ ), we show the internal angle $\theta_i^{(j)}$ made by the $j$ -th Delaunay triangle to beetle $i$ . This triangle has area denoted by $A_i^{(j)}$ . The index $j$ is understood to enumerate the set of Delaunay triangles which share the point $\mathbf{x}_i$ as a common vertex. We calculate the weighted average triangle area across the indices $j$ , weighted by the angles they make at $i$ . Our density measure is then the inverse of this mean suitably normalised by beetle length $r$ . Reproduced from [94]. . . . .	27

2.3	Density computed using the DTFE (a.1, b.1) and the proposed method (a.2, b.2) for a lattice of circular discs (radius $r = 1$ ), panels a.x show a lattice of $N = 3^2$ and panels b.x show a lattice of $N = 6^2$ . Panels a.3 and b.3 show a naked Delaunay tessellation of the input data for reference (computed using Qhull [95]), note due to the co-linearity of points the tessellation is not unique. The colour scale is normalised to the output of the DTFE method in both cases (giving an arbitrary scale $[0, 1]$ and colour scale). Note the variance in the DTFE method (b.1) and the uniform nature of the local density in the proposed method (b.2). . . . .	28
2.4	Orientation velocity correlation function $C(\delta t)$ plotted for large (a) and small (b) time lags $\delta t$ . Dashed line in (b) shows $\delta t = 0$ . Note we find a small but positive value of $\delta t$ at which the correlation function attains a maximum. Data plotted is for the $N = 200$ case only, we fit a Gaussian function to the data to obtain a delay of 13 ms. The small delay would be a natural consequence of inertial effects. . . . .	30
2.5	Relationship between beetle self-propulsion speed $v(\rho)$ and local density $\rho$ . Local density is computed using the method proposed in section 2.2 and outlined in figures 2.2 and 2.3. Data is plotted for three groups ( $N = 50, 100, 200$ ) on a log-log scale, a guide to the eye showing a proportionality of $v(\rho) \sim \rho^{-0.4}$ is also plotted. We see a decay in self-propulsion speed with increasing density up to around $\rho \sim 1$ where a notable uptrend is seen, this occurs over two orders of magnitude. Approximately from $\rho \sim 10^{-2.5}$ to $\rho \sim 1$ . The upturn may be due to collisions or a change in beetle behaviour near collision. Data is plotted after first binning the density axis (on a log-scale) and averaging speeds in the bins, error bars show the standard deviation within bins. . . . .	31
2.6	Speed density relationship for non-inertial (a) and inertial (b) ABPs. In (b) the inertial parameters are $M = 1, J = 1, \eta_r = \eta_t = 1$ In both cases a linear $v(\rho) = a\rho + b$ and power-law $v(\rho) = a\rho^b + c$ have been fitted (least squares) as solid and dashed lines respectively. For both (a) and (b) common parameters are $v_0 = 10, D_r = 5 \times 10^{-3}, r = 1, \mu = 3000$ and time step $\Delta t = 10^{-5}$ . Particles were initialised in a box with packing fraction 0.4 such that n particles were overlapping at time $t = 0$ . Notable is the non-linear best fit for inertial particles, and the lack of an increasing speed at higher densities. . . . .	35

2.7	Results for simulating equations 2.6 and 2.7 using masses $M$ and moments of inertia $J$ , $D_r = 5 \times 10^{-3}$ , and $v_0 = 10$ and $r = 1$ . Colours show the difference in root-mean-square error between the linear and power law models scaled to $[0, 1]$ where 1 indicates the power law has a lower fitting error. . . . .	36
2.8	(a) Schematic diagram of the model showing the geometric centre (GC) the soft-body force including overlap, and the velocity, orientation, and GC director (not normalised here) data used to compute the re-orientation $\mathbf{V}$ , $\theta$ , $\mathbf{R}$ and the rounded arrows respectively. (b) resultant density obtained from fitting the model to the $N = 200$ data set <i>only</i> . The error bars on the data curve (blue) indicates a standard deviation of kernel density estimates taken at $n = 100$ temporal windows in the data, each window had equal width of $T/n$ where $T$ is the total number of time-frames in the data. The error bars on the model indicate one standard deviation between three separate simulations with different initial conditions. Reproduced from [94] . . . . .	39
2.9	The CABP model in $\alpha - \tau$ space. For $N = 200$ (a), $N = 100$ (b) and $N = 50$ (c). We see a region of uni-modal densities (left of boundary) and another region of bi-modal densities (right of boundary). Data with circles indicate values for which we believe a steady state has not been reached, it is likely this is due to the effect of too weak a reorientation causing dispersal. Filled squares indicate unexplored parameters. The unfilled square in (a) indicates the best fit for the $N = 200$ case. . . . .	41
3.1	(a) A sketch of a visual state, the outline of the central agent indicates sensors which are considered “activated” or “filled” or not by an opaque black outline of translucent black outline respectively. Faint lines emanating from the central agent indicate pairs of angles defining the $n_s = 40$ sensors. (b) A sketch of an agent’s future states tree (FST) (red). The blue and green agents represent two models of other agents. The bold black lines highlight the paths following on from one particular initial action. Anticipated collisions can occur on the FST, indicated by the Green agents fourth state (transparent). We choose to either continue enumerating paths and states past these “collisions” or prematurely stop enumerating. Both cases will be discussed, by default we perform the latter. . . . .	45

3.2	Circles represent visual states over $[0, 360^\circ)$ , black regions are 1 and white/blank regions are 0. Here $1, 90^\circ, 0, 180^\circ, 1, 90^\circ$ means a run of ones anti-clockwise from angle 0 to $90^\circ$ , a run of zeros for $180^\circ$ and finally another run of $90^\circ$ back to the 0 $360^\circ$ boundary. The length of the run-length encoding is proportional to the number of boundaries on the visual state, if the 0, $360^\circ$ boundary is always considered a boundary, and decimal run-lengths are approximated to a constant number of digits. Note the run-lengths in each encoding contain a list of bits and a corresponding angular length (decimal value) where this length must also be converted to a sequence of bits, implying a discretisation. I.e on a computer these decimal values are approximated as 32 or 64 bit floats, even analytically this process necessarily implies a discretisation (like the sensor model) at some level even if it is much finer than e.g 40 sensors, otherwise decimal and transcendental numbers will yield extremely large even infinite codes. E.g even picking a code word, say 00, to represent $\pi$ requires storing the value of $\pi$ once for later decoding which we can only approximate in reality.	55
3.3	Example clustering of a group (in open space) using the method outlined below. (a) the group is cohesive (b) a smaller group splits from the main group and is identified as a second cluster (c) multiple groups have split from the main cluster, some are single-agent clusters. We use DBScan utilising “density” in space and orientation-space to detect order and cohesive groups as clusters. Lone agents are identified as clusters for accurate fragmentation statistics. . . . .	61
3.4	Order ( $\phi$ ), opacity ( $\Theta$ ), and time averaged convex hull number density ( $\rho$ ), equations 3.16, 3.18, 2.1 respectively, using the path entropy method (SCE) across a broad range of model parameters. Here note $\rho$ is plotted on a $\log_{10}$ scale. We find across this parameter set average order $\sim 0.95$ (SD $\sim 0.03$ ) with a minimum and maximum values of 0.86, 0.99 respectively. Lower orders are driven by fragmentations of the main group. Each data point corresponds to 3 simulations which were run for $T = 1000$ time steps, statistics were calculated over time steps $T = 500$ to $T = 1000$ . . . . .	64

- 3.5 Long run-time simulations ( $T = 30000$ ) of (left)  $N = 50$  and (right)  $N = 250$  groups, in each case  $n_s = 40, \Delta\theta = 15^\circ, v_0 = 10, \Delta v = 2$ , for increasing  $\tau$  values from  $\tau = 2$  to  $\tau = 6$ . We find a rapid increase in global order past  $\tau > 3$  caused by a decreased rate of fragmentation across time. Note that fragmentation is increased a large time-scales for large groups, this represents individual agents leaving the main group at a slow rate across time. Error bars indicate one standard deviation on three replicates. . . . . 65
- 3.6 Three example simulations with  $n_s = 40, \Delta\theta = 15^\circ, v_0 = 10, \Delta v = 2, \tau = 6$  with (a)  $N = 50$  (b)  $N = 100$  (c)  $N = 250$ . In each case the Eigenvectors used to calculate the morphology are indicated in (dashed) red scaled by a common constant for visibility. (b) and (c) include some fragmentations ( $\approx 1$  and  $2$  respectively) not pictured. In these three cases the (global) order, morphology and density  $(\phi, \mathcal{M}, \rho)$  are (a)  $0.986, 1.89, 0.092$  (b)  $0.975, 1168, 0.068$ , and (c)  $0.954, 57.23, 0.051$ . The reduction in order for (b) and (c) and band-like morphology in (b) is due to fragmentation of  $\approx 1$  and  $7$  agent/s from the main group respectively, accounting for these fragmentations by computing statistics on the largest cluster, gives (main cluster  $\phi, \mathcal{M}, \rho$ ) of (b)  $0.981, 8.96, 0.068$  and (c)  $0.98, 81.9, 0.05$ . . . . . 66
- 3.7 The order  $\phi$  (blue circles) and largest cluster convex hull density  $\rho_C$  (red triangles) depend on (a) the number of sensors  $n_s$  and (b) the resolution parameter for the unique state counts and compression entropy models respectively. Indicated as a dashed line is the value of  $n_s = 40$  and  $s = \pi/40$  which is indicative of the size of a projection (or union of projections) needed to activate a sensor in both the USC and SCE models (with discretised sensors) and a sensor count of  $n_s = 40$ . Here  $\tau = 6, N = 50$  and error bars correspond to one standard deviation of the order across 10 replicates. Note also density begins to increase before order in (b). This measurement indicates decorrelated motion but with a sense of group cohesion, like a swarm of insects for example. . . . . 66

- 3.8 Parameter sweep for the CE model, confer with figure 3.4. Order ( $\phi$ ), opacity ( $\Theta$ ), and convex hull number density ( $\rho_C$ ) using the path entropy method (SCE) across a broad range of model parameters. Here note  $\rho_C$  is plotted on a  $\log_{10}$  scale. We find across this parameter set average order  $\sim 0.95$  (SD  $\sim 0.03$ ) with a minimum and maximum values of 0.86, 0.99 respectively. Lower orders are driven by fragmentations of the main group. Each data point corresponds to 3 simulations which were run for  $T = 1000$  time steps, statistics were calculated over time steps  $T = 500$  to  $T = 1000$ . . . . . 68
- 3.9 Order (first column), largest cluster size (second column), and largest cluster order (third column) for  $\Delta\theta = 15^\circ, 25^\circ, 30^\circ$  (across rows) for groups with  $\tau = 5$ ,  $N = 50$ ,  $n_s = 40$ . Each group was simulated for 5000 time steps with colour as the mean across three replicates, normalised according to the colour bar. Each statistic,  $\phi$ ,  $\mathcal{C}$ , and  $\phi_C$  are defined in the range  $[0, 1]$ . We find two regions of high order and high cohesion and low order but still high cohesion. These are indicative of highly ordered cohesive groups with persistent directed motion (the former) and a swarming phenotype (the latter). The swarming phenotype is more stable across parameter space ( $v_0, \Delta v$ ) at increased  $\Delta\theta \approx 30^\circ$ . . . . . 72
- 3.10 Various phenotypes observed by varying model parameters, common parameters are  $\tau = 5$  and  $n_s = 40$ , all include trials plotted for 100 timesteps. (a) the ordered and cohesive phenotype with parameters  $N = 50$ ,  $\Delta\theta = 15^\circ$ ,  $v_0 = 10$ ,  $\Delta v = 2$ . (b) Persistent rotation here with velocity reversals i.e.  $N = 50$ ,  $\Delta\theta = 15^\circ$ ,  $v_0 = 1$ ,  $\Delta v = 2$  implying speeds  $-1, 1, 3$ , rotation is long-lived but not always with the same sign across different initial conditions, and possibly run-time. (c) Not cohesive but not completely fragmented group,  $N = 50$ ,  $v_0 = 1$ ,  $\Delta v = 1$ ,  $\Delta\theta = 15^\circ$  (i.e. including a stopping action  $v^t = 0$ ). (d,e)  $N = 250, 50$  respectively with common parameters  $\tau = 5$ ,  $\Delta\theta = 30^\circ$ ,  $\Delta v = 0.75$ , and  $v_0 = 1$ . We see coexistence of multiple clusters breaking and forming whilst swarming around a largely stationary central point, in (e) the two clusters are about to reform and then break again, reducing  $\Delta\theta$  to  $15^\circ$  and lower results in large scale fragmentations within  $\sim 100$  time steps. (f)  $N = 50$   $\Delta\theta = 30^\circ$  but with a different actions space, two speeds  $v^t = 1$  or  $v^t = 2$  and two angle increments  $\pm\Delta\theta$ . . . . . 73

4.1	Schematic representation of the occlusion problem as overlapping visual projections, from the perspective of the blue agent (b). The diagram on the left represent agent position and the diagram on the right the resultant visual state. Note the overlaps between the visual projection of the orange (o) and green (g) agents. . . . .	76
4.2	Sketch of the occlusion algorithm as applied to the situation in figure 4.1. (a) projection intervals are sorted by distance from the agent calculating the visual state. (b) overlaps are identified, i.e. by the dashed line covering the overlap between the orange and green intervals (c) for each overlap the furthest interval is contracted to remove the overlap, indicated by the faded region on the green interval. . . .	77
4.3	Parameter sweep for SCE using four occlusion thresholds, (a,b,c,d) $\mathcal{O} = \mathcal{O}^*$ (no occlusion), 0.0, 0.5, 0.95 respectively. This data suggests (1) occlusion promotes order in larger groups ( $N = 200$ ) in this case (2) the standard set and systems with higher $\tau = 5, 6, 7$ appear largely unchanged (3) the model appears less robust to changes in $\Delta\theta, \Delta v$ and the presence of orientational noise post decision, but this decreases with $\mathcal{O}$ increasing (4) density appears to have decreased slightly on average, but with significant variation. (5) opacity appears reduced on average but with significant variation. . . . .	81
4.4	Varying occlusion threshold has decreasing effect for future time horizons $\tau \gtrsim 4$ , except in larger groups $N \gtrsim 500$ . Figures (a-c) show the order parameter, equation 3.16 resulting from simulations with $\tau = 2, 4, 6$ futures respectively. Figures (a-c) <i>insets</i> show the time averaged size of the largest cluster as a fraction of $N$ , a low value (approaching $1/N$ ) indicates mass fragmentation which is usually associated with low order, a value of 1 indicates total cohesion. Error bars represent the standard deviation of 3 simulations, where agents are positioned uniformly at random in a box of side length $L$ with initial number density is $N/L^2 = 0.005$ , and orientations are drawn from a normal distribution with mean 0 and standard deviation of $\Delta\theta^\circ$ . . . . .	82

4.5	(a) Fine structure in the effect of occlusion for $\tau = 6$ . Error bars represent the standard deviation of 3 simulations, where agents are positioned uniformly at random in a box of side length $L$ with initial number density is $N/L^2 = 0.005$ , and orientations are drawn from a normal distribution with mean 0 and standard deviation of $\Delta\theta^\circ$ . (b) Long time (10,000 times steps) simulations for $N = 500$ , $\tau = 6$ , individuals for varying occlusion thresholds, error bands indicate one standard error on the mean of 4 simulations. The increased order $\phi$ is maintained at least up to $t = 10,000$ , although all series continue to lose some order across time. Compare also with the inset of figure 3.5 (b). . . . .	84
4.6	Figures (a-c) show the morphology parameter (equation 3.20) $\log_{10}$ scale, the ratio of eigenvalues in equation 3.19, for the $\tau$ values 2, 4, 6 respectively. Left to right on the $x$ axis represents an increasingly strong occlusion threshold (from no occlusion). Figures (a-c) <i>insets</i> represent the opacity $\Theta$ defined as the ratio of active sensors to inactive sensors. Error bars represent the averaging over 3 simulations, where agents are positioned uniformly at random in a box of side length $L$ such that the initial number density is $N/L^2 = 0.005$ , and orientations are drawn from a normal distribution with mean 0 and standard deviation of $\Delta\theta^\circ$ . . . . .	85
4.7	Density of the entire group $\rho$ and of the largest cluster $\rho_{c^*}$ for $\tau = 4$ (a,c) and $\tau = 6$ (b,d). Error bars are the standard error on the mean for three repeats. . . . .	87
4.8	Snapshots for various parameters and group sizes showing the varied morphology we observe. (a) An $N = 50$ group with $\tau = 4$ futures and no occlusion. (b) A group of $N = 1000$ agents with $\tau = 6$ and an occlusion threshold of 0.95 (c) $N = 100$ agents with an occlusion threshold of 0.5 and including states past future collisions (d) $N = 1000$ agents with $\tau = 6$ but occlusion off (e) $N = 1000$ agents with $\tau = 6$ , occlusion 0.95 and enumeration past collisions in the future states tree (collisions off) (f) $N = 50$ and a occlusion threshold of 0.95. Note also the small fragmentation in (c) at approximate coordinates $(x = 9000, y = 1800)$ . The arrow in (c) denotes the average heading direction of the group (by eye), observe the stretching in (d) is along the direction of travel (although this sometimes fluctuates in the simulation). . . . .	88

4.9	Order and largest cluster size for different groups sizes with $\tau = 6$ for (a) collision branch cuts on the FST (b) no collision branch cuts on the FST. Data for the largest cluster sizes. Error bars indicate one standard error on the mean computed over 5 replicates for time steps 500 to 1000. . . . .	90
4.10	Effect of post-decision orientational noise on global order $\phi$ , largest cluster size $\mathcal{C}$ (inset) for $N = 50, 100, 250$ group sizes (a,b,c.i) and $\tau$ values 2, 4, 6. In each case a rotational additive noise term is applied directly to the output of the FSM decision process sampled from a normal distribution with variance $\eta^2$ , statistics are calculated after a burn-in of 500 time steps up to time step 1000. $\tau = 2$ is generally not enough for ordered motion to arise, with $\tau = 4, 6$ we see a transition from high order $\phi \sim 0.98$ to disorder $\phi \sim 1/\sqrt{N}$ (dashed line) with a slightly more robust ordered phase with $\tau = 6$ compared to $\tau = 4$ . We see from $\mathcal{C}$ that this is driven by large scale fragmentation. Error bars indicate 1 standard error computed from 3 replicates. We also plot the global order and order of the largest cluster ( $\phi_{\mathcal{C}}$ ) for low noise values in the $N = 250$ case (c.ii,c.iii). . . . .	91
4.11	Impact of increasing “cognitive” noise on global order $\phi$ and largest cluster size $\mathcal{C}$ for groups of sizes $N = 50, 100, 250$ (a,b,c.i), for each a burn-in of 500 was used with statistics calculated up to time step 1000. Figures c.ii and c.iii detail the effects with low rotational noise $\eta$ and 0 speed noise $\zeta$ . Cognitive noise is noise applied to each agent’s model of other agents one or both of orientational noise (Gaussian and white with variance $\eta^2$ ) or speed noise (Gaussian and white with variance $\zeta^2$ ) applied to the base ballistic model. i.e. $\eta = \zeta = 0$ implies a ballistic model and non-zeros values indicate diffusive trajectories/ increasing uncertainty. . . . .	93
5.1	A visualisation of four walkers descending a binary tree of depth two. In this case each leaf has received at least one walker, the probability of the pictured event is $\approx 0.09375$ , in contrast if the tree was only depth one (so the third row is cutoff and the second row becomes the leaves) then the probability is 0.875 of all leaves being visited. . . . .	99

5.2	(a, Main plot) Global order increases with number of samples drawn with replacement. The order does not reach the same as in explicit enumeration due to sample degeneracy. Here $N = 50$ agents, and $\tau = 4$ , were simulated with differing numbers of walkers, but identical starting conditions. Data points and error bars correspond to the mean of 5 simulations, and one standard deviation of 5 simulations respectively. (a, Inset) Probability of all paths being sampled in a 5-ary tree of depth 4 with $S$ walkers, this is the form of the future states tree for the default FSM parameter set. For reference 99% probability is achieved at approximately 6894 walkers. Theory indicates evaluating equation 5.7, and the simulations are by simulating the process according to equation 5.5. . . . . .	101
5.3	Global order, $\phi$ , as a function of the number of paths (in the FST) sampled, $p$ , for time horizons $\tau = 4, 5, 6$ . Note the extra x-axis (top) marking $S = 5^4, 5^5, 5^6$ show necessary but not always sufficient (depending on $\tau$ ) conditions for complete enumeration when $\tau = 4, 5, 6$ respectively. In particular, we see that for a common value of $p$ and increased horizon $\tau$ promotes order. In particular observe the increases in order across 100 to 625 paths for increasing $\tau$ . For each scatter point $N = 50$ agents were simulated for 1000 time steps, with standard parameters and paths were sampled uniformly at random without replacement. The error bars indicate one standard error on the mean over $n = 5$ replicates. Statistics were accumulated for time steps 500 to 1000. . . . . .	102
5.4	Order is increased due to a decrease in fragmentation in large groups ( $N = 250$ and $N = 1000$ ) when extending a constant number of samples, $S = 5^4$ here, to deeper tree depths. Paths were sampled without replacement, error bars show one standard error on the mean over six replicates. The inset shows the largest cluster sizes for the same simulations . . . . .	104
5.5	Order for $N = 50$ agents using the Boltzmann sampler with future entropy to judge actions to sample. (a) and (b) use $\tau = 6$ and $\tau = 10$ respectively. (c) and (d) use the average count of boundaries on future states with $\tau = 6$ and $\tau = 10$ respectively. Error bars indicate one standard error on the mean for 6 repeated simulations. Increasing $\beta$ has an unclear impact at $\tau = 6$ . With $\tau = 10$ increased $\beta$ appears to have a slight benefit implying uniform sampling is advantageous. . .	112

5.6	Two examples of a continuous action space in our context. (a) a model where agents may increment their orientation by any angle $\theta \in [-\Delta\theta, \Delta\theta]$ and choose speeds in the range $v \in [v_0 - \Delta v, v_0 + \Delta v]$ . (b) the same as in (a) but now since $\Delta v = v_0$ the speed can be any value in $[0, 2v_0]$ . (c) An example sampled path to depth $\tau = 4$ (red, thick arrows) starting with a set of $n_a = 8$ initial actions to choose from (black, thin arrows with first red thick arrow). . . . .	113
5.7	Example output from the continuous model with $N = 50, \tau = 5, n_a = 5, \Delta\theta = 15^\circ, \Delta v = 2.0, v_0 = 10.0$ (a), we find $\phi = 0.974$ . The distribution of realised actions is shown in (b) for the speed and (c) for the orientation, in each case the truncated normal distributions $\alpha_v \sim \mathcal{N}_{v_0-\Delta v}^{v_0+\Delta v}(v_0, \Delta v^2)$ and $\alpha_\theta \sim \mathcal{N}_{-\Delta\theta}^{\Delta\theta}(0, \Delta\theta^2)$ are plotted as orange dotted lines, in (b) the black line indicates a uniform distribution over the range. The red dashed lines indicate truncated normal's with the same means but different variances: $\alpha_v \sim \mathcal{N}_{v_0-\Delta v}^{v_0+\Delta v}(v_0, 4)$ and $\alpha_\theta \sim \mathcal{N}_{-\Delta\theta}^{\Delta\theta}(0, 361)$ which are better (b) and arguably better (c) fits. For flocks fragmenting both $\alpha_\theta$ and $\alpha_v$ tended to be uniform distributions. Statistics for (a) and (b) were collected for all time-steps (1-1000) in (a). . . . .	115
5.8	Order of an $N = 50$ group with $\tau = 5$ dependent on the number of initial action $n_a$ an agent selects from for 625 and 3125 sampled paths. Here $v_0 = 10, \Delta v = 2$ and $\Delta\theta = 15^\circ$ to compare directly with the SCE model with discrete action space. $n_a = 5$ is marked for comparison. There is a trade-off between how many initial actions an agent selects from and versus how many paths need to be sampled to effectively judge between them. . . . .	116
5.9	Phenotypes in the continuous model, we see swarming (a.iv) and highly ordered cohesive groups (a.i,a.ii) for low and high values of $v_0$ respectively, with fragmenting groups (a.iii) at intermediate values of $v_0$ . $\Delta v/v_0$ modifies this only slightly at $v_0 \approx 1$ (c.i) and (c.ii). In (b) we see the result of biasing the orientation selection to a positive sign. In all cases $N = 50$ and $v_0 = 10, \Delta v = 2, \Delta\theta = 45^\circ, S = 625, n_a = 5$ , and $\tau = 10$ unless stated otherwise. . . . .	118
5.10	Bias applied to orientation leads to a cohesive flock following a circular path. $N = 50$ and $v_0 = 10, \Delta v = 2, \Delta\theta = 45^\circ, S = 625, n_a = 5$ , and $\tau = 10$ unless stated otherwise. . . . .	119

# Acknowledgments

I would like to particularly thank my supervisors Matthew Turner and Shashi Thutupalli for their continued support and stimulating discussions throughout my research for this thesis, without whom this would not have been possible.

I'd also like to thank Shashi and his lab for their generous hospitality and fantastic atmosphere of collaboration, particularly Shashwat Kumar, Pawan Nandakishore and Noor Alkazwini. And I would also like to extend my gratitude to NCBS and its students in general for my time spent there, with Saptarshi Dasgupta, Sujan Kumar and Alex Rautu in particular.

To my partner and parents I owe the greatest thanks for their unconditional support of me throughout my PhD, before it, and beyond.

# Declarations

I declare that this thesis is my own work and has not been submitted for any degree or qualification at any other University. Work based partly on collaborative research or previously published work is clearly indicated here and within this thesis.

Content from chapter 2 has been previously published in *Devereux HL, Twomey CR, Turner MS, Thutupalli S. 2021 Whirligig beetles as corralled active Brownian particles. J. R. Soc. Interface 18: 20210114..* In this section, raw data collection (experiments) was not performed by myself and is outside the scope of this thesis. Processing of the raw data into trajectories was partially collaborative. I did not contribute to the detection methodology or human tracking of the data. I did collaborate, but did not lead, research on the algorithmic tracking. My contributions appear after section 2.2.

Content in chapters 3 and 4 has been submitted for peer review at Physical Review X, at the time of submission.

# Abstract

Active matter is an expanding field of physics covering a diverse range of complex and beautiful phenomena. From examples we see in our everyday lives, such as the flight of birds and organisation of insects, to more esoteric bacteria and other micro-scale biological systems. What we can learn about the physical rules that pin these diverse systems together is important not just for our understanding of physics but our ability to utilise the natural world around us. The core of our understanding of Active matter spans between out-of-equilibrium analogues of well-known thermodynamics to the realm of complex intelligent decision-making. From a top-down view point, we observe phenomena such as aggregation, ordered motion, dynamic pattern formation, leader-follower relationships, long range interactions, collisions avoidance, and coordinated motion to name a few, and model these directly within a mathematical formalism. From a bottom-up perspective we attempt to explain the generation of these phenomena from intrinsic process driving individual agents. In this thesis we consider a data-driven analysis of collective motion in an insect system, a top-down approach, as well as developing a model of individual decision making based upon future path entropy, a bottom-up approach. The latter results in the spontaneous emergence of some basic features of collective motion seen in real world examples, lending explanatory power.

# Abbreviations

AM Active Matter

AP Active Particle

ABP Active Brownian Particle

PBP Passive Brownian Particle

SPP Self-Propelled Particle

MIPS Motility Induced Phase Separation

IMCM Intrinsically Motivated Collective Motion

FSM Future State Maximisation

FST Future States Tree

HTM Hierarchical Triangular Mesh

RL Reinforcement Learning

MCTS Monte-Carlo Tree Search

FAI Fractal Artificial Intelligence

CEF Causal Entropic Force

DT Delaunay Triangulation

DTFE Delaunay Tessellation Field Estimator

CVC Contiguous Voronoi Cell

G-JF Grønbech-Jensen Farago

SDE Stochastic Differential Equation

GC Geometric Centroid

CABP Corralled Active Brownian Particle

CE Compression Entropy

RLE Run-length Encoding

SCE State Count Entropy

# Chapter 1

## Introduction

### 1.1 Active Matter

The term *Active Matter* (AM) refers to physical systems that (1) consume energy and dissipate this energy in order to move and (2) do this at the level of the individual constituents of the system [1]. Examples include groups of animals, insects, or bacteria which we encounter regularly in our everyday lives. As a Physicist AM is a fundamentally out-of-equilibrium theory of systems, and particularly their constituent objects, that are driven not solely from an external field but are intrinsically driven by energy in- and out-flow of their constituents.

The constituents themselves are variously termed as *Self-Propelled Particles* (SPPs) or *Active Particles* (APs) or by reference to some specific model of AM such as *Active Brownian Particles* (ABPs), or by the details and context of their movement as in *Micro-Swimmers*. Examples in these categories span many length scales from microscopic ( $\mu\text{m}$  to  $\text{nm}$ ) to macroscopic ( $\text{mm}$  to  $\text{m}$ ), and are found both in nature as well as in man-made form.

#### 1.1.1 Experimental and Real World Examples

The requirement for individual constituents to consume energy and move is a very general statement about AM, covering a diverse range of naturally occurring and man-made examples. Each case usually involves a particular method of self-propulsion, often inextricably linked to particular environmental conditions, which introduce case-by-case idiosyncratic phenomenology to model and explain. Nevertheless the commonalities across these diverse systems often lend to general descriptions or at least common approaches to theoretical modelling.

Arguably interest began in natural systems at the macro-scale; one does not

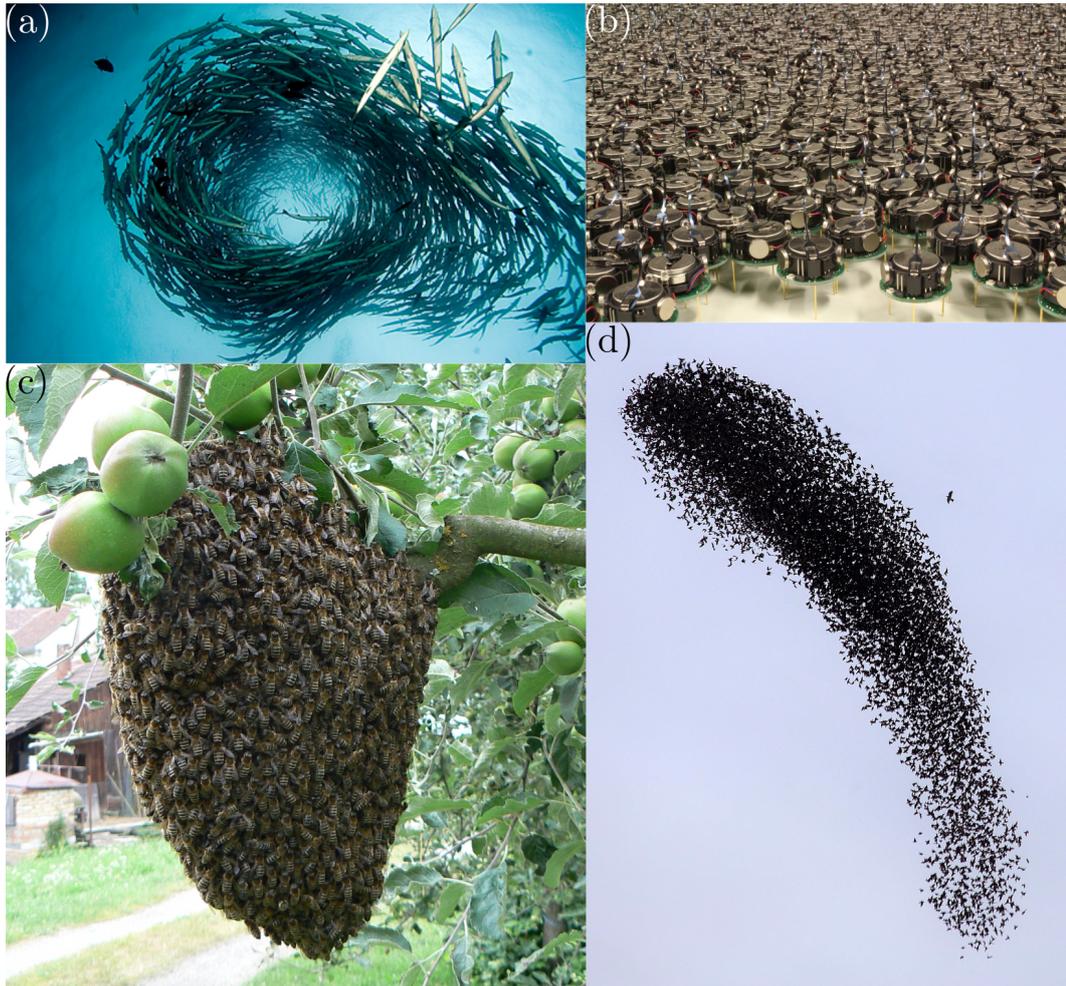


Figure 1.1: Four macroscopic examples of Active Matter (a) a group of Barracuda fish, Rob Hughes, cc-by-sa-2.0 (b) a man-made swarm of “Kilobots” capable of group communication and individual programming [2], cc by-sa 4.0 (c) a Honey Bee swarm cc by 3.0 (d) a swarm of Starlings and predator, cc by-sa 4.0. Each image was sourced from Wiki-media Commons.

need to be a scientist to marvel at the complex collective dynamics exhibited by birds, fish, insects, and of course humans. The beginnings of the scientific literature are often prescribed to Reynolds [3] who modelled swarms of birds or fish by coining his eponymous rules: (1) Avoid collisions (2) Match velocity (3) Remain cohesive. Soon followed by interest in the Statistical Mechanics literature by Vicsek *et al* [4]. Experimentally speaking, for birds, perhaps the archetypal example are Starling flocks, which have given rise to understanding of scale free correlations [5], information transfer among members of flocks [6], and complex pattern formation [7]. Other birds also exhibit similar behaviours, for example hierarchical motion in

carrier pigeon flocks [8], and for grounded birds the structure of and appearance of travelling waves in Emperor penguin huddles [9]. Human behaviour and movement also occupy a place in AM research in particular the study of pedestrian dynamics [10, 11], leading to applications in pedestrian tracking and the understanding of human crowd movement. Fish and insects comprise particularly useful examples in experimental terms, compared with birds, due to the quasi-two dimensional nature a top-down camera perspective can bring, for example Whirligig Beetles [12, 13], and fish schools [14, 15] in shallow water tanks. At the microscale studied natural examples include *Escherichia coli* [16] and spermatozoa [17, 18], among many others. Understanding and manipulation of the properties of both systems have important potential for health applications e.g. fertility for Spermatozoa.

Motivated by such real-world examples there are now several human fabricated examples at the microscale and macro-scales such as “Janus particles”, which exploit their two-faced nature to cause self propulsion by e.g. light stimulation of one face [19] or [20] by chemical reaction on one face, heating of one face via laser stimulation [21], and external electric field acting on one face [22]. Other mechanisms include external forcing via magnetic field [23], local charge symmetry breaking [24], and flow (Marangoni) across droplet-fluid interfaces due to droplets containing Bromine [25], to name a few [26]. Experiments at the macro-scale are commonly conducted using a “shaker” which rapidly vibrates a plate containing specially fabricated particles, which have lead to realisation of active Brownian particles with and without significant inertial effects [27, 28, 29, 30, 31, 32]. Alternatively particles are driven locally e.g. by vibration motors [33] which have even been designed for programmable control of individuals and communication between individuals [2].

### 1.1.2 Theoretical Approaches

The expansion of real world data on microorganisms, synthetic micro-swimmers, bird motion, and human motion *etcetera*, have supplied ample stimulation for the construction of phenomenological models of these particular examples and AM in general. The lack of open access peer-reviewed data repositories in the field is noteworthy, with a culture of data being made available on request to authors of individual studies. Notable exceptions do exist however, such as the work of Davidson *et al.* [34].

## Reynolds, Vicsek, and Couzin

Initially, as mentioned above, Reynolds [3] and Vicsek *et al.* [4] developed theoretical approaches based upon individual agent equations of motion via agent-based approaches. Reynolds' work proposed a framework for realistic simulation of large groups of autonomous agents, aimed at computer graphics, whilst Vicsek *et al.* developed these ideas into a simple dynamical system, given by equations 1.1 and 1.2.

$$\frac{d\mathbf{x}_i(t)}{dt} = v_0 \begin{bmatrix} \cos \theta_i(t) \\ \sin \theta_i(t) \end{bmatrix}, \quad (1.1)$$

$$\theta_i(t) = \langle \theta_j(t) \rangle_{j \in \mathcal{N}_i(t)} + \eta_i(t). \quad (1.2)$$

There are  $i \in 1, 2, \dots, N$  circular agents of radius  $r$  in two dimensions with positions  $\mathbf{x}_i(t)$  and orientations  $\theta_i(t)$  at time  $t$ . The model implements Reynolds' velocity matching rule by performing orientational averaging over the neighbour set  $\mathcal{N}_i(t) = \{j : \|\mathbf{x}_i(t) - \mathbf{x}_j(t)\|_2 < r_c\}$ . Where  $r_c$  defines a distance based cutoff point at which velocity averaging stops.  $\eta_i(t)$  is a uniform noise term with zero time or inter-agent correlations, taking values in  $[-\eta/2, \eta/2]$  for a noise strength  $\eta$ . Thus, the model describes individuals with a finite interaction radius,  $r_c$ , that move in such a way to take the average orientation of their neighbours within this interaction radius, while a controllable noise disrupts this process. The averaging process itself should be understood as the value of  $\arctan(\langle \sin \theta_i(t) \rangle / \langle \cos \theta_i(t) \rangle)$  taken over the neighbour set  $\mathcal{N}_i(t)$  which includes  $i$  itself.

The interest in this model stems from the transition from order to disorder with increasing noise,  $\eta$ . Order here was defined by the order parameter

$$\phi(t) = \frac{1}{Nv_0} \left\| \sum_{i=0}^N v_i(t) \right\|_2, \quad (1.3)$$

where  $v_0(t)$  is the average agent speed, and  $v_i(t)$  is agent  $i$ 's velocity at time  $t$ .  $\phi(t)$  shows a transition, in a non-equilibrium system, remarkably like those seen in equilibrium systems close to phase boundaries, when  $N$  is large.

The key contribution was a dynamical analogue of familiar physics of ferromagnetic alignment; an out of equilibrium system analogous to an equilibrium process. Instead of static ferromagnetic alignment of magnetic spins, as found in the Ising or XY models [35], alignment of velocity occurs between self moving individual particles. Further this phase transition, in  $\langle \phi \rangle$ , was found also to be dependent on density  $N/L^2$  (for system length  $L$ ) in direct analogy to continuous phase transitions

in classic equilibrium models.

This Vicsek model uncovered interesting physics directly from a system solely employing Reynolds' 2nd rule, velocity alignment, by adding a Stochastic variable. Couzin *et al* probed further the collision avoidance and cohesion rules, proposed by Reynolds, by defining a model using three spherical interaction zone: (with a blind angle):  $z_r$  a zone of repulsion,  $z_o$  a zone of alignment,  $z_a$  a zone of attraction. Each zone was a sphere/circle centred at an agents' position  $\mathbf{x}_i(t)$  with radii  $r_r$ ,  $r_o$ , and  $r_a$  respectively, with  $z_o$  and  $z_a$  also given the possibility of a "blind angle" behind the agent. The dynamics are that each individual selects a self-propulsion direction based on the populations of agents within these regions,

$$\mathbf{d}_r(t) = - \sum_{\substack{j=1 \\ j \neq i}}^{n_r} \hat{\mathbf{x}}_{ij}, \quad (1.4)$$

$$\mathbf{d}_o(t) = \sum_{j=1}^{n_o} \hat{\mathbf{v}}_j, \quad (1.5)$$

$$\mathbf{d}_a(t) = \sum_{\substack{j=1 \\ j \neq i}}^{n_a} \hat{\mathbf{x}}_{ij}. \quad (1.6)$$

The values of  $\mathbf{x}_{ij} = \mathbf{x}_j - \mathbf{x}_i$  indicate relative position vectors from agent  $i$  to  $j$ , with hats denoting unit vectors. The summations should be understood as over the sets of agents:  $\{j \neq i \in 1, 2, \dots, N : |\mathbf{x}_{ij}| < r_r\}$  which has size  $n_r$ ,  $\{j \in 1, 2, \dots, N : r_r \leq |\mathbf{x}_{ij}| < r_o\}$  with size  $n_o$ , and finally  $\{j \in 1, 2, \dots, N : r_o \leq |\mathbf{x}_{ij}| \leq r_a\}$  with size  $n_a$  with  $|\mathbf{x}|$  the Euclidean norm of the vector  $\mathbf{x}$ . The model prioritised collision avoidance by stipulating an agent's velocity in the next time step,  $\mathbf{v}_i(t + \Delta t) = \mathbf{d}_r(t)$  whenever  $n_r > 0$ . Otherwise, given  $n_a > 0$  and/or  $n_o > 0$  then  $\mathbf{v}_i(t + \Delta t) = \frac{1}{2}(\mathbf{d}_o(t) + \mathbf{d}_a(t))$ . For the case of no agents within the interaction zones, an the focal agent continues ballistically. By analysing the dynamics which emerges from simulations under these rules, Couzin *et al.* [36] found four main phenotypes by varying the model parameters: (1) a swarm characterised by a low group polarisation  $\frac{1}{N} |\sum_{i=1}^N \mathbf{v}_i(t)|$  and low angular momentum  $\frac{1}{N} |\sum_{i=1}^N \mathbf{r}_{ic}(t) \times \mathbf{v}_i(t)|$  (where  $\mathbf{r}_{ic}$  is the vector from particle  $i$  to the groups centre) resulting in a group that is disordered but cohesive (2) A milling group rotating around a central point on toroidal trajectories (low order and high angular momentum) (3) Dynamic parallel groups with high order and low angular momentum characterised by higher mobility than the toroidal group. Finally, a highly aligned group which moves in straight line trajectories. Further they demonstrated an ability to control the group level dynamics (transitions between the four phenotypes) by altering individual level behaviour

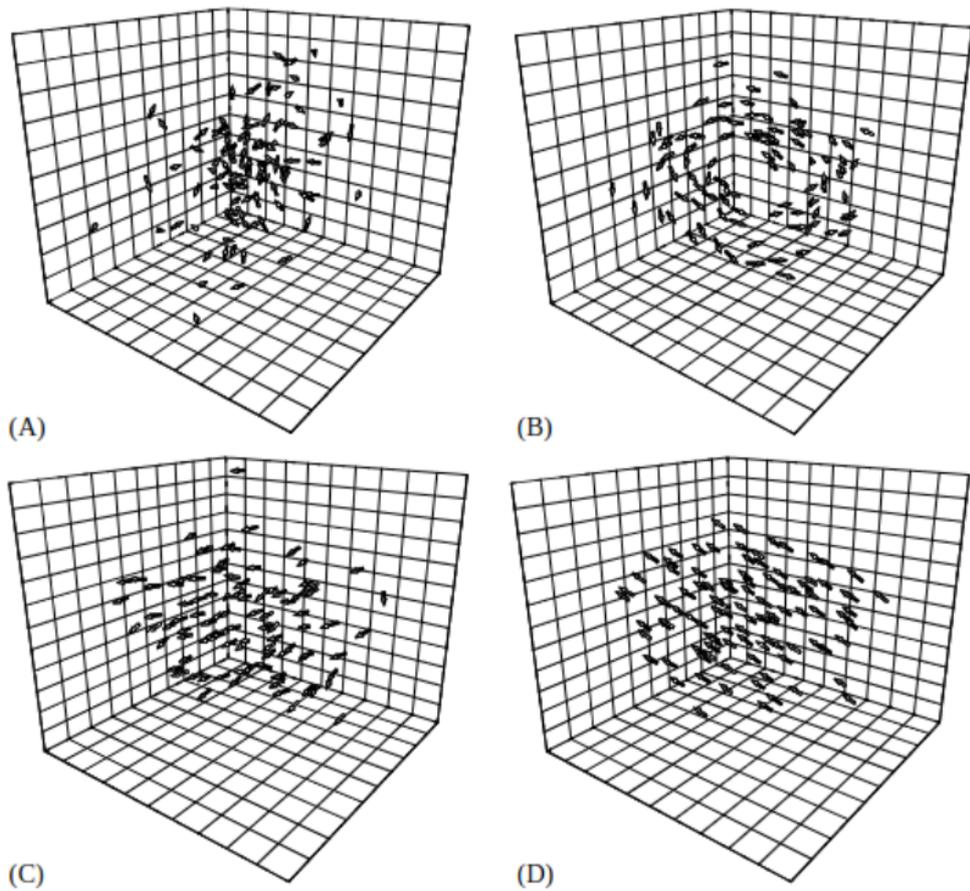


Figure 1.2: The four phenotypes found from the Couzin model, adapted from [36]. (A) agents swarm by remaining cohesive but not ordered. (B) milling agents characterised by circular motion. (C) a roughly aligned group and (D) a group showing almost total alignment.

by controlling  $r_o$ , including a hysteresis effect by  $r_o$ . Thus, the model successfully demonstrated transitions between dynamical phenotypes reminiscent of nature, e.g. bird flocks for (1) and milling fish for (3), controlled by the three interaction zones in question.

Following the Couzin and Vicsek models and Reynold’s rules, alternative models have been proposed which often share some similar structure. For example by suppressing or amplifying the effect of a Couzin-like cohesive “social force” by a measurement of an individual’s centrality to a flock [37]. With the idea being that an agent on the periphery of a group may more strongly attract to the group than one within the bulk. Additionally, Hildenbrandt’s model includes a more detailed treatment of environmental forces. For example by explicitly including gravitational and drag forces (from the air). And like the Reynolds’ model, a treatment of bird movement e.g. by modelling banked turns. Other variations have focused on the geometry of agent visual sensing. The Couzin model included a set angle  $\alpha$  defining a spherical sector. This sector defined lines of sight along which agents would not interact with others in their perception zones. More detailed models of visual sensing include two zones, one for each “eye”, which allow for non-contiguous region of visual sensing [38].

Finally, both the Vicsek and Couzin models include metric interactions. That is agents interact with others based upon specific distances, beyond which agents do not interact. Agent interactions based upon explicit metric criteria have been questioned, with experimental data indicating Starlings interact on average with a fixed number of individuals (6-7), not with all flock-mates within a specific distance [39]. This has resulted in topological variations of the same interaction rules as e.g. the Vicsek model. The topological variation of the Vicsek model replaces the alignment interaction with all neighbours  $\mathcal{N}_i(t)$  within the cut-off distance  $r_c$ . This is done by identifying all agents  $j$  that are Voronoi neighbours to an agent  $i$  and using this definition in place of  $\mathcal{N}_i(t)$  as defined in equation 1.2 above [40].

A particular ubiquitous model that we shall draw upon in latter chapters is the *Active Brownian Particle* (ABP). The model is based mathematically on the work of Paul Langevin, in particular his eponymous underdamped/overdamped Langevin equation introduced in 1908 [41]. The modern day simulation of the ABP model is outlined by Volpe *et al.* [42], and Bechinger *et al.* offer an extensive review on theoretical research into ABP models as well as their experimental relevance [26]. The model itself consists in two spatial dimensions, without inertial effects or

hydrodynamics, as the coupled stochastic partial differential equation in equations

$$\frac{\partial \mathbf{x}(t)}{\partial t} = v_0 \mathbf{n}(t) + \sqrt{2D_T} \boldsymbol{\eta}_T, \quad (1.7)$$

$$\frac{\partial \theta(t)}{\partial t} = \sqrt{2D_R} \eta_R, \quad (1.8)$$

also known as the *overdamped-Langevin* equations.

Here we are considering a single particle with position vector  $\mathbf{x}(t) \in \mathbb{R}^2$ , for a two-dimensional system although three and higher is possible, dependent on time, a self-propulsion speed of constant value  $v_0$ , with heading direction vector (sometimes called polarity vector)  $\mathbf{n}(t) = [\cos \theta(t) \sin \theta(t)]^T$  for angle  $\theta(t)$ , with translational and rotational diffusion coefficients  $D_T$  and  $D_R$ .  $\boldsymbol{\eta}(t)$  is a two-dimensional Wiener process and  $\eta_R$  a one dimensional Wiener process with no self or cross correlations in time.

At its core the ABP models a particle with some agency over its velocity, given by  $v_0 \mathbf{n}(t)$ , this is the active part. The model also includes a translational diffusion term giving the model its Brownian character. In particular this mimics a standard thermal noise process <sup>1</sup> e.g. as seen pollen grains [43]. It is the interplay between the ABP's ability to drive itself and the thermal fluctuations that give rise to its fundamentally out-of-equilibrium nature [44, 45]. Indeed retaining non-zero noise terms  $\boldsymbol{\eta}_T$  and  $\eta_R$  in equations 1.7 and 1.8, but setting zero self-propulsion speed  $v_0 = 0$  gives us the *passive* Brownian particle (PBP) within standard equilibrium statistical mechanics.

ABPs have been applied widely to simulate the dynamics seen in many forms of micro (e.g. [19]) and sometimes macro (e.g. [31]) systems, with examples of aggregation [29, 46, 47]. In particular what is now known as motility induced phase separation (MIPS) occurs in active systems like ABPs [48, 49]. MIPS is a process that can lead to a co-existence of aggregated (clustered/high density) regions with sparse regions, even in systems with purely repulsive interactions. This particular phenomenology resembles a theoretical treatment of run and tumble bacteria [50] that shows phase separation, and more recently computational work [48], on a-polar self-propelled (2D circular) particles, with no translational diffusion and small rotational diffusion rate. The model undergoes a phase transition (controlled by the systems packing fraction) from non-aggregated Brownian motion to the co-existence of aggregated and non-aggregated regions. Particularly striking was the fact this system included only repulsive steric contact forces and no alignment or

---

<sup>1</sup>i.e collisions between the comparatively macro-particle and the microscopic particles of the medium (gas or liquid) in which it resides.

other interactions. This behaviour was also seen experimentally in liquid colloids [51], similarly experiments on nematic liquid crystals [47] further showed dynamical properties, such as clustering, could present in systems without communication mechanisms and only collision based interactions.

The mechanism by which the dense phase forms in these systems can be understood by considering the competition between reorientation time facilitating escape from a cluster and further accretion of particles. When a particle collides with another cluster, the persistence of each particle’s motion delays the escape of the particle. If particles collide before others can escape a cluster can form. The formation and stability of small clusters, pairs, or triplets will be approximately related to the ratio of particle reorientation time to the rate of collisions. The former will arise from diffusion statistics and the latter from both local/system density and particle propulsion speed. As the mean collision time falls sufficiently lower than the particle re-orientation time scale large clusters will begin to form and grow. This effect can be considered as an “effective cohesive” force which was carefully examined with reference to experiment and computational work across intermediate densities by Ginot *et al.* [52]. This effect is also closely linked to the fact that active particles in general will aggregate in regions where they move more slowly [53]. When combined with a self propulsion speed inversely related to local density [49] there is a natural feedback loop between density, increased by particle slowing, and slowing, induced by particle density.

Further *inertial* ABPs [54] have seen a growing interest in recent years both experimentally: in synthetic vibrated systems [31], beetles in flight on the air-water interface [55] and whirling fruits [56], as well as theoretically [57]. Particular theoretical interest lies in relation to (kinetic) temperature differences which appear to be motility induced and the presence of inertia, causing MIPS to break down with large self-propulsion (giving a re-entrant phase diagram) [58, 59]. In this case we use the underdamped-Langevin equations,

$$\frac{M}{\nu} \frac{\partial^2 \mathbf{x}(t)}{\partial t^2} = v_0 \mathbf{n}(t) + \sqrt{2D_T} \boldsymbol{\eta}_T - \frac{\partial \mathbf{x}(t)}{\partial t}, \quad (1.9)$$

$$\frac{J}{\nu_r} \frac{\partial^2 \theta(t)}{\partial t^2} = \sqrt{2D_R} \eta_R - \frac{\partial \theta(t)}{\partial t}. \quad (1.10)$$

Where we have introduced acceleration terms, a mass  $M$ , moment of inertia  $J$ , and translational and rotational drag coefficients  $\nu$  and  $\nu_r$ .

## 1.2 Top Down to Bottom Up

Reynolds’ rules, the Vicsek and Couzin models, and ABPs are all, in a sense, a “top-down” approach to active matter. For example in the Couzin model it is assumed that zones of attraction, repulsion, and alignment exist. Reynolds begins with collision avoidance, velocity matching, and cohesion as a given. Similarly, the Vicsek model examines a group which by definition follows velocity alignment. These assumed behaviours are quite reasonable, and clearly do give predictive power, but from where do they arise? Reynolds writes “[flocking is evolutionary selected for due to] protection from predators, statistically improving survival of the (shared) gene pool from attacks from predators, profiting from a larger effective search pattern in the quest for food, and advantages for social and mating activities”. These attributions do have some evidence to support them. Examples of predator avoidance are known in biological experiments [60] and in attraction-repulsion (Couzin-like) simulations [61, 62, 63] there is evidence for group behaviours as a flock benefiting the individual, in particular the ability of the group to rapidly alter its shape is attributed to confusing predators. Similarly, it is known Emperor penguins huddle for warmth (a clear benefit of collective behaviour) [64, 9], and Phalaropes circulate in groups to draw up food [65]. But still much numerical work relies upon an imposition of attraction-repulsion based (or other) rules as a given in the model, making them dependent on these rules arising in the first place.

Multiple studies have been made on unpicking individual equations of motion and responses to group mates as a bottom-up, data-driven, approach. For example by attempting to fit Vicsek-Couzin-Reynolds like equations of motion to real data [66]. Or by examining closely the average force response of individuals correlated with nearest neighbour position [67, 68, 69, 15, 70]. The motivation being that an animal may be choosing how to self-propel based on its relative position to its neighbours. In a sense determining these behavioural “force response maps” from data is a generalisation of Couzin-like models, i.e. the nature of the force response is not limited to circular regions or to a linear or non-linear function.

A bottom up approach, as we understand it in this thesis, to collective motion involves the definition of a generative or motivational model intrinsic to the agents that results in emergent collective dynamics. That is naturally generating collision avoidance, velocity alignment, and cohesion (Reynolds rules), or even predator/object avoidance. In particular a physical model that leads to collision avoidance or velocity alignment without these characteristics being “baked in” to the equations of motion (or decision-making process) would be an example of the

bottom up approach. Data-driven examples as we introduced above are quite varied in the literature but often still involve using data-driven force responses to fit pre-determined equations of motion.

Understood in these terms, bottom-up and top-down are approaches to modelling which are relative to the phenomena being investigated. The top being starting from an assumption that the particular phenomena exists, and typically fitting some explicit model of that phenomena to data. On the other hand the bottom begins without the assumption of the phenomena existing and instead some more basic property. For example, we may impose a repulsion term in a model to explicitly include collision avoidance, perhaps with some strength parameter which can be fitted to data (top-down). Alternatively as we will see in the next section, we could define a general principle, “keeping option open”, by which collisions may reduce the number of options leaving a rational agent to avoid them (bottom up). Both approaches investigate a phenomenon (collision avoidance). The top-down approach is more suited to investigating the effects of the phenomenon, whilst the bottom-up approach is more suited to investigating why it may exist at all.

### 1.2.1 Intrinsically Motivated Collective Motion

One major example of a bottom up approach, directly applied to active matter is Intrinsically Motivated Collective Motion (IMCM) [71, 72], that will be redeveloped in the latter chapters of this thesis and explored in further detail. The approach outlines a model (Future State Maximisation, FSM) constructed by each agent’s intrinsic motivation to (loosely speaking) “maximise their future options” with reference to their estimations of the future. As an example to gain intuition consider playing a game of Chess. When losing the game a player will most likely experience a progressive waning of their pieces directly limiting their future options, until eventually there is no option to take; the game is lost. This example also given by Charlesworth and Turner labours the point that “winning” and “losing” can often be cast into a paradigm of increasing or decreasing viable futures (viable options). It is precisely this general mechanism IMCM proposes as a generative process for collective motion. The result is that collision avoidance, velocity alignment and cohesion emerge without being imposed directly; a bottom-up model.

IMCM considers  $N$  agents moving in two-dimensional space and able to select from a small set of actions (1) ballistic movement  $\mathbf{v}_i(t) = v_0 \mathbf{n}_i(t)$  (2) slow speed  $\mathbf{v}_i(t) = (v_0 - \Delta v) \mathbf{n}_i(t)$  (3) fast speed  $\mathbf{v}_i(t) = (v_0 + \Delta v) \mathbf{n}_i(t)$ , and a positive (4) or negative (5) increment to the angle between the agent’s direction  $\mathbf{n}_i(t)$  and the  $x$ -axis. These actions are visualised in figure 1.3 (A). The core idea was that

each agent independently decides based upon current and *future* visual information which of the 5 actions to take. Each agent does this by reference to visual states defined by tracing lines to other (circular) agents representing their visual size [73]. A sketch of this process is shown in figure 1.3 (B). The method used in [73] and

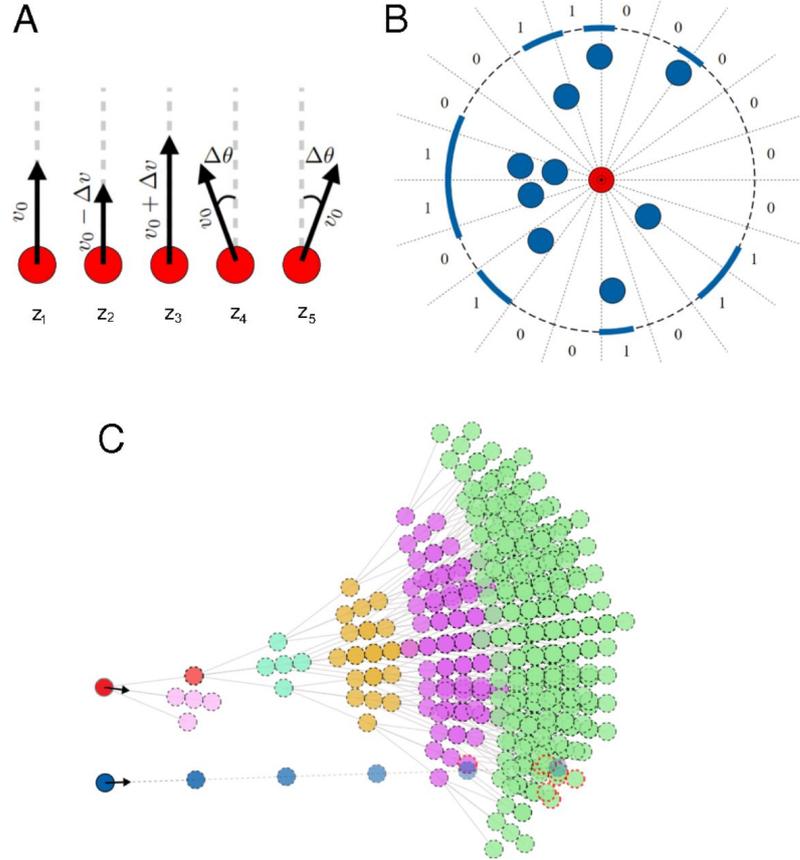


Figure 1.3: Visual summary of intrinsically motivated collective motion, reproduced from [71] (A) visual depiction of the action space. (B) a diagram showing an example visual state calculated by first finding the visual projection (thick blue lines) and then determining the coverage of each sensor (segments defined by thin black lines) for the central (red) agent. (C) The future states tree generated by actions (A) showing one complete branch up to time  $\tau = 5$  into the future, as well as one ballistic agent and future collisions with it (dashed red circles).

[71] invokes the formulation shown by the red dashed triangles in figure 1.4, that is the size of the projection is calculated using the value of  $\arctan(r/x)$  (the angle made a point  $x$  in the figure by the red dashed triangle). An alternative is to use the tangent lines resulting in the value  $\arcsin r/x$  (blue solid triangles), with the difference reducing as a function of distance. One benefit (as far as numerics are

concerned) of the arctangent formulation is that it exists up to  $x = (0, 0)$  when agents are perfectly coincident, whereas the arcsin formulation fails to exist if one agent's centre is within the radius of another. The mathematical difference can be noted by converting the arcsin to an arctangent via  $\arcsin(r/x) = \arctan(1/\sqrt{x^2 - r^2})$ . The visual state produced is visualised in figure 1.3 (b) as the blue regions, which

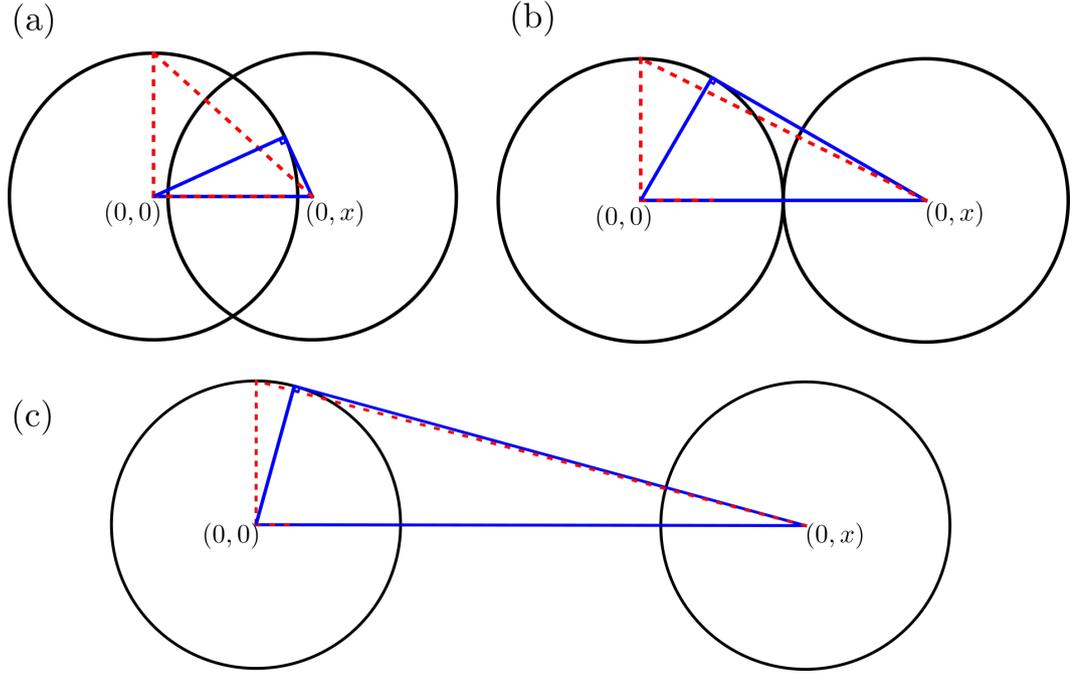


Figure 1.4: Detailed formulation of the visual projection angles as defined in [73] and used in [71] (red, dashed triangles) and the tangent line method (blue, solid triangles). a, b, and c show the construction for increasing inter-agent spacing  $x$  along the  $x$ -axis for clarity. The methods differ at small distances, we will use both in this thesis, partly for exact comparisons sake and secondly to show the minor impact of choosing one over the other. Motivation for the arc-tangent approach is that particles are defined as “phantoms” with no constraints on overlaps, even a small overlap will lead to the arc-sine formulation becoming undefined.

are then processed into a finite binary vector (0's and 1's) representing  $n_s$  sensors (segments defined by black lines). Each agent also projects the current positions of all other  $N - 1$  agents *ballistically*  $\tau$  time steps into the future forming a future states tree (FST) by all possible sequences of  $\tau$  actions, visualised in figure 1.3 (c). Thereby at a node at depth  $t' \in \{0, 1, 2, \dots, \tau\}$  in the future states tree (an imagined future time  $t+t'$ ) an agent computes a future visual state by the projection method, using its position having taken that  $t'$  length sequence of actions and the ballistic positions of the rest of the flock. Collecting these states the decision on

which action to take at time  $t$  is to select the action with the largest count of unique states<sup>2</sup> on the FST branch corresponding to taking that action initially (or randomly among degenerate maxima). This simple criterion was shown to result in highly ordered ( $\frac{1}{N}|\sum_{i=1}^N \mathbf{v}_i(t)| \sim 0.98$ ) and cohesive groups which move across two-dimensional space, at groups sizes  $N = 50$  to  $N = 500$  studied in the model. Whether cohesive and ordered groups emerge at a particular value of  $N$  depends crucially on the value of  $\tau$ , the time horizon.  $\tau < 4$  results groups fragmenting into smaller ordered subgroups. Sufficiently large  $\tau \geq 5$  maintains an ordered and cohesive group. Groups larger than  $N \gtrsim 100$  exhibit a slow rate of individual fragmentations from the main group depending on  $\tau$ , which can largely be eliminated with  $\tau \gtrsim 6$ .

Although functioning in two-dimensional space, in principle, the model of vision can be extended to a three-dimensional setting. Models such as that by Reynolds, and Hildenbrandt *et al.* [3, 37] both incorporate detailed models of bird flight in three-dimensional space. The Couzin model also function in three-dimensional space without such a detailed flight model. With IMCM the visual state is the most difficult part to extend to three-dimensional space. The visual state defined in IMCM, based upon an earlier work [73], can be extended by considering not pairs of tangent lines to other agents, but tangent cones. However, distributing  $n_s$  sensors within a sphere as apposed to a circle is not quite as simple. Nor is calculating overlap between unions of tangent cones and sensors on the sphere. In the two-dimensional case discretising the range  $[0, 2\pi)$  gives regularly sized sensors. In 3D we would need to find an analogous method to distributed regularly sized sensors on a sphere. One approach to spherical tessellation is the Hierarchical Triangular Mesh (HTM) [74] which recursively subdivides a triangular mesh, beginning with an Octahedron. A mesh such as the HTM could be used to place sensors. However, these sensors would not be uniformly spaced or sized. Additionally, the HTM allows meshes with  $8 \cdot 4^{d-1}$  “Trixels” for recursion depth  $d \geq 1$ . For IMCM in two dimensions  $n_s = 8$  is too few sensors for order to emerge,  $n_s = 32$  is, and  $n_s = 128$  is too many sensors. It is not clear *a priori* how many sensors would be optimal in three dimensions.

The significance of the result was to show a process by which velocity alignment and cohesion could be obtained through a bottom-up physical approach. In this case the bottom up criterion was the notion of maximising available future states.

---

<sup>2</sup>Although states at which a collision is detected between an agents future position and a flockmates future ballistic position are discounted, as well as subsequent states.

### 1.2.2 Intelligent Decisions and Causal Entropy

Agents or systems of many agents making autonomous intelligent decisions has applications in a variety of domains, especially with the rise in reinforcement learning (RL) [75] (including the multiple agent case), deep learning [76], and their combination with other methodologies like Monte-Carlo tree searches (MCTS) [77] to solve increasingly complex and abstract problems such as Go [78]. Indeed, attempts have been made to develop general frameworks for defining what an “intelligent decision maker” is. For example Fractal Artificial Intelligence (FAI) [79], which draws heavily from work on postulated Causal Entropic Forces (CEFs) by *Wissner-Gross et al* [80]. CEFs were also one motivation for IMCM [71].

CEFs envisage a general thermodynamic system by defining macro-states  $\mathbf{X}$  by relating any paths (through phase space)  $\mathbf{x}(t)$  to any other path  $\mathbf{x}'(t)$  if and only if the two paths share the same initial state  $\mathbf{x}(0) = \mathbf{x}'(0)$ . This generates a unique set of macro-states defined by unique initial conditions  $\mathbf{x}(0)$ . The Causal Path Entropy of  $\mathbf{X}$  is defined as the integral equation 1.11 which integrates the entropy of paths, given the initial state, over all possible paths up to some future time horizon. A gradient is then taken, in equation 1.12, which defines the causal entropic force. Using their theory *Wissner-Gross* and *Freer* demonstrate a particle moving to a central position in a confining box under the influence of  $\mathbf{F}$ , maximising the diversity of causal paths that the system could access (in this case via Brownian motion). Similarly, they demonstrate a cart-pole system spontaneously up-righting itself; maximising path diversity. And finally show emergent behaviours claimed to demonstrate tool-use and social cooperation, both thought of as hallmarks of “intelligence”.

$$S(\mathbf{X}, \tau) = -k_B \int_{\mathbf{x}(t)} \mathcal{P}(\mathbf{x}(t)|\mathbf{x}(0)) \log \mathcal{P}(\mathbf{x}(t)|\mathbf{x}(0)) d\mathbf{x}(t) \quad (1.11)$$

$$F(\mathbf{X}_0, \tau) = T_c \nabla_{\mathbf{X}} S(\mathbf{X}, \tau) \Big|_{\mathbf{X}_0} \quad (1.12)$$

## 1.3 Outline

In chapter 2 we will begin by studying experimental data from a real world active matter system, namely groups of *Dineutus discolor* (*Coleoptera: Gyinidae*) more commonly known as the Whirligig beetle, swimming freely on the surface of water. The experiments consisted of observations of motion of beetle groups ranging in size within a circular arena of shallow water, and we will consider undisturbed groups. In terms of author contribution, the data acquisition phase was completed prior to

this thesis and is outside its scope. The main hypothesis of the chapter is that large groups of Whirligig beetles are exhibiting a type of motility induced/ dynamic phase separation into high and low density (slow and fast moving) regions. We will explore evidence for this hypothesis and confounding factors by a top-down analysis of the dynamics. Using motion statistics from video data we compare to extant literature on active phase separation and develop a self-propelled particle model inspired by the phenomenology. Concretely we introduced a Delaunay tessellation based approach to computing a local density value for each beetle. We use this method to show how a local density dependent speed is present within each group, that is a decreasing speed with increasing local density; a crucial ingredient for motility induced phase separation. We also find the presence of weak inertial effects. Some evidence is found for the same velocity-density curve we measure in inertial systems of active Brownian particles, but cannot rule out a possible role of more complex behavioural effects. Finally, we introduced a modified ABP model including a local density dependent re-orientation term inspired by the paths of individuals take on excursion from the main cluster. We show that bi-modal local density distributions (averaged over time) naturally arise within this model for large  $N \gtrsim 100$ , and we are able to fit the model to experimental data.

Next in chapter 3 we change focus to a bottom-up approach to active matter. We develop the ideas of FSM and IMCM by defining an agent based model in which individuals take actions which maximise future path entropy over the future. Our initial hypothesis in this chapter is that we can define such a model by considering all paths on the future states tree generated by each possible sequence of actions into the future and forming an empirical count distribution of visual states along these paths. We show that this does lead to cohesive and highly ordered collective motion of comparable character. As a second hypothesis we test whether we can define a notion of entropy by a measure of compressibility of the (continuous) visual state function itself, i.e. we ask whether an analogous model can be formed avoiding discretisation of the visual states into binary vectors. We find that this is indeed possible, generating cohesive and highly ordered collective motion, and relates to the number of boundaries on the visual state. However, we find that in order to achieve this we must directly impose a “resolution” parameter on the visual state where agents appearing smaller than a given angular size threshold are ignored, effectively creating a level of discretisation. We find the resolution required creates an effective discretisation comparable to the discretisation implied by the sensor model, and in fact the two models are largely equivalent in motion statistics. We interpret this as the sensors, and resolution parameter, being one step up in an agents cognitive

process, meaning the exact number of sensors should be compared to an abstract cognitive process rather than for example, the number of cells in the retina.

Armed with this new model, in chapter 4 we examine in detail two hypotheses based upon perturbing the input into and output of the cognitive processes of agents. First we hypothesise that incorporating an explicit model of visual occlusion will, to leading order, have little effect on the stability and order and other motion characteristics of the model. But we expect higher order changes could be possible in opacity and morphology of the agents, and if occlusion can be more or less severe we expect order to drop with more severe occlusion. To test this we define an explicit occlusion model based upon the visual state and the circular nature of our agents that is tune-able in “strength”. Whereby other agents are not considered in an agent’s cognitive processing if they are obscured by more than a certain percentage. We find that our hypothesis holds in small groups  $N \lesssim 100$  but in larger groups we find significant increases in order and stability coincident with drastic changes to morphology ( $N = 500, 1000$ ) even with severe occlusion, although of course to a limit. We hypothesise this is due to a building of errors in the ballistic modelling assumption in these large groups, that is occlusion naturally regulates the amount of information an agent processes and this leads to a more accurate picture of the future.

Our second hypothesis in chapter 4 is that by applying noise processes to the input and output of each agent’s cognitive process we will observe a transition from cohesive and ordered motion to disordered and fragmenting groups. We examine two cases: first a Vicsek-like noise process applied by perturbing agent orientations after actions are selected, and secondly we replace the ballistic modelling assumption with a stochastic process where modelled agents choose random speeds and rotations that on average forms a ballistic trajectory. In both cases we find order-disorder transitions, but crucially we find small noise values at which order is paradoxically promoted. In the ballistic case we interpret this increase in order as a self-consistent noise value, i.e. the noisy predictions are of higher accuracy than the ballistic assumption.

Finally, in chapter 5 we hypothesise that the path entropy model can be recast as a sampling algorithm. In particular, we hypothesise that paths from the FST can be sampled directly at random to result in ordered cohesive motion. That we can apply a Monte-Carlo Tree Search (MCTS) to build paths from individual action selections for more efficient sampling, and that we can apply these ideas to sample from continuous action spaces. We find that sampling paths with replacement is not a workable solution, it results in rarely ordered groups that are highly unstable, this

however is completely rectified by sampling paths without replacement. In doing so we find significant advantages to sampling paths deeper rather than fully enumerating at lower tree depths. That is a depth first tree search results in cohesive highly ordered motion with less evaluation of states (in some cases up to 40% less) than a breadth first search of the FST, i.e. enumerating all nodes at a given time horizon before enumerating nodes at greater time horizons. We interpret this as our model predicting, for agents in such groups, it is more efficient to prioritise computing partial information about their state further into the future than full information about the immediate future.

Encouraged by this we apply an MCTS to the problem of action selection by building sample paths actions-per-action. We do this by sampling the next action to evaluate on the FST by a probability weighted by the current estimate of future entropy subsequent to it. We hypothesise this will be a more efficient approach to sampling paths without replacement and that we need to regulate the distribution by the number of subsequent unsampled paths. In fact, we find marginal benefit for any MCTS compared to uniform sampling without replacement. Especially considering the increased computational load for tracking entropy estimates and selecting actions to sample.

Given this knowledge we close chapter 5 by applying sampling to continuous actions spaces. We hypothesise that actions composed of continuous orientation increments and speeds selected from a continuous range can be used along with uniform sampling of path actions to result in ordered cohesive motion. We found however that fully uniform sampling of path actions only resulted in ordered and cohesive motion for an extreme number of futures,  $\tau \approx 512$ . We found this was due to a trade of between the number of initial actions an agent judged and the number of paths sampled. Controlling for this we found that when limiting the number of initial actions to judge, e.g. to 5 randomly chosen initial actions, we do realise ordered motion with future time horizons and a number of samples comparable to the discrete sampling case.

## Chapter 2

# A Data Driven Study of Whirligig Beetles

### 2.1 Introduction

In this chapter we will examine a top-down study of group motion of *Dineutus discolor* (*Coleoptera: Gyrimidae*) or “Whirligig Beetles”. The data involved is composed of experimental footage, filmed from above, of different groups of Whirligig beetles ranging over three population sizes. The beetles were allowed to move freely on the (shallow) waters surface of a circular arena, and their motion observed. We will outline the tracking of individual trajectories, the analysis of group motion statistics with a careful focus on a measure of local density, and the construction of a top-down model based upon the Active Brownian Particle formalism. We will then discuss the process of fitting the model’s free parameters to data and the conclusions we can draw from the model and analysis on the mechanism of the dynamical clustering behaviour seen in the laboratory footage.

The theoretical basis for the direction of research is the growing body of work on Motility Induced Phase Separation (MIPS) in active particles, which attempts to answer whether MIPS or a MIPS-like mechanism could explain the dynamical aggregation we see in the data. As outlined in the introduction to this thesis, MIPS is a fundamentally out of equilibrium phenomenon seen in AM. In particular MIPS arises due to the natural tendency of active particles to aggregate due to a self-propulsion speed inversely related to local density [53, 49]. It is also known that inertial effects can greatly change the phase behaviour in ABP systems compared to the non-inertial case, particularly regarding the onset of MIPS [58, 59]. We use this to determine a measure of local density and therefore seek to quantify the local

density dependent self-propulsion speed of beetles, including inertial effects.

### 2.1.1 Experimental Data



Figure 2.1: Snapshot of experimental footage of  $N = 200$  Whirligig beetles within the circular arena of diameter 95.25 cm. A typical beetle measures  $12 \pm 1$  mm along its body length. The body itself is roughly elliptical with a 2 : 1 aspect ratio between its major and minor axes. The beetles are able to propel themselves freely on the waters surface and are not observed diving in the footage used for our analysis. The experiments used for this chapter do not involve startling of the beetles or any other stimulus except for the constant illumination of the arena (730 lx). The camera is placed for a top-down view of the tank, captured at spatial resolution  $1920 \times 1080$  and time resolution of 30 frames per second.

An example snapshot of the movie data is shown in figure 2.1. The setup

consisted of a circular arena (diameter 95.25 cm) filled with water to allow the beetles to move freely on the surface. The beetles were filmed from a top-down perspective at resolution  $1920 \times 1080$  and 30 frames per second, after acclimatisation periods of 20 minutes. No stimulus was applied in the data we use except for the constant 730 lx illumination of the tank for filming purposes.

In terms of experimental design a shallow tank was chosen in part to mitigate diving behaviour. And in the data (footage) tracked from the experiment we do not see this behaviour. The circular nature of the tank acts to reduce boundary effects. For example a square shaped tank would likely cause a tendency for beetles to aggregate at the boundaries. This is a generic feature of many self-propelled agents, but it is possible beetle behaviour may result in aggregation in corners as well. The tank’s camera facing base was chosen to contrast well with the dark black colouration of the beetles, to aid in later post-processing. The recorded data was post-processed into individual frames, which were then thresholded into binary images, both using the ImageMagick software [81].

### Processing Video Data to Extract Trajectories

The processed data we use can be split into two types: (1) fully tracked by hand and (2) tracked by algorithm. The human tracked data is complete with both detections (beetle positions) and tracks (temporal linking of detections). The algorithmic tracking and detection shall be outlined here, and is part of a work in preparation to be published [82]. The detection workflow consisted of training a neural network to detect beetles, which was trained upon the human detections. This was done by coding input images as being a beetle or not (classification encoding) in one input channel, and using the distance (horizontal and vertical separately) between a pixel and the head or tail as four more input channels. These human created input data were fed into a neural network utilising the DenseBox method architecture [83]. The output of this methodology are sets of position-orientation coordinates,  $X^t = \{\mathbf{x}_1^t, \mathbf{x}_2^t, \dots, \mathbf{x}_{N_t}^t\}$  for  $\mathbf{x}_i^t \in \mathbb{R}^2 \times [0, 2\pi)$ , for each frame (discrete time point)  $t$ . The notation  $N_t$  denotes the number of detections at a given frame  $t$ , which can and does vary even though we have a constant number of beetles  $N$ . This provides one way to identify detection errors e.g.  $\frac{(N-N_t)}{N}$ . The neural network itself was trained on the error in position and orientation of a given detected beetle with comparison to the actual detection of a beetle by hand at the pixel level. The exact form of the error is a sum of the classification and regression error, i.e. the pixel level error between the classification channels combined with total error across the four regression channels. As will be discussed shortly with our tracking method-

ology, accurately assigning detections to tracks in a multi-object setting requires accurate spatio-temporal data. The camera sets a spatial (pixel resolution) and temporal (framerate) timescale. Objects approaching the pixel resolution in size, or approaching inter-frame displacements of one or more body lengths per frame, become increasingly impossible to accurately detect and track. Additionally, given a reasonable resolution the detector will also introduce error, i.e. by assigning individual position coordinates offset from the ground-truth. A position error much less than the size of individual objects is desired. Deviations from the ground-truth larger than the typical object size will lead to a greater number of identity swap errors on tracking. Identity swaps occur when a trajectory following one individual “swaps” to tracking another. Typically, the detector error reached within 5% of one beetle body length.

The processed (detected) data consisted of sets of detected beetles at separate time points. In order to obtain trajectory data across time for individuals the next step was to perform a multi-object tracking procedure. That is to link a detection of beetle  $i$ ,  $\mathbf{x}_i^t$ , to the same beetle at time  $\mathbf{x}_i^{t'}$  for all  $t' \neq t$ , with reference to the human tracked results. Multi-object tracking remains a difficult problem that is not fully resolved in many disciplines, with numerous approaches detailed in the literature. Particular examples are idTracker [84] which uses machine learning to generate “fingerprints” for each tracked object, Kalman Filter based approaches [85] utilising the Kalman Filter for a linear motion model, global data association techniques [86], and with high spatio-temporal resolution the Hungarian/Kuhn-Munkres algorithm [87] can be used for nearest-neighbour matching.

Our data was tracked using a modification of the global data association methodology of Zhang *et al.* [86]. The modification was to enforce a global constraint on the number of objects within each frame,  $N$ , and to handle missing objects in any particular frame by adding a linear motion assumption to allow for resolving the problem when any particular track has a missing segment. The method was based solely on distance data, i.e. the distance between an objects position at time  $t$  and  $t + 1$ , since we seek to track almost indistinguishable objects at the resolution of our camera. When resolution is high enough, or objects are in some other way distinguishable, then it is possible to use methods such as idTracker which use this information to build unique identities of individuals by machine learning methods [84]. Alternatively, it is not impossible for experimenters to introduce that distinguish-ability by simple marking individuals, although of course care must be taken this marking does not confound whatever the purpose of the experiment is e.g. by somehow modifying inter-agent behaviours.

The goal of our multi-object tracking based on distance data here is to identify  $N$  true positive detections (i.e. the actual known number of beetles) and join them into time sequences in such a way to match the human tracked data as closely as possible. The difficulty of this task is in a sense directly related to the spatio-temporal resolution, and in particular on average inter-frame displacements of agents compared to their body size. That is if each agent has a size scale of  $r$  and the average inter-frame displacement  $d$  is strictly less than this length scale  $d < r$  then multi-object tracking is (largely) trivially solved by nearest neighbour assignment between pairs of frames, with further improvements achieved using the Hungarian/Kuhn-Munkres algorithm [87] with inter-frame distance costs. This is because (distance based) multi-object tracking errors (with a perfect detector) typically occur when one object (A) in one track moves closer to another object’s (B’s) previous position in the next frame than (B) is, which can result in an identity swap where one track swaps to tracking another object in error.

Our data is unfortunately quite far outside the regime where nearest-neighbour and Hungarian based methods work, this is due to our temporal resolution. Here we find (1) individuals move in a highly non-linear way (2) average inter frame displacement exceeds 2 – 3 times beetle body length. Issue (1) also precludes linear motion model methods such as the Kalman filter [85], and although non-linear motion model trackers exist we find that a method based on (global) data-association [86] is easier to work with since we do not need to find a motion model to track *a priori*. Compared to the human tracked data we do find a  $\approx 99.8\%$  accuracy rate when computing the multi-object tracking accuracy statistic [88] an improvement over the 88.9% accuracy that can be obtained from employing a Hungarian matching approach with inter-frame distance costs and the same linear motion assumption for missing detections. The difference being the global nature of our modified method over the “greedy” local nature of inter-frame Hungarian matching.

## 2.2 Towards a Measure of Local Density

We begin by collecting density and motion statistics. In this section we will discuss the novel methodology we use for local density and the results obtained from it. The calculation of density will be of great interest due to the fact MIPS is intricately linked to a self-propulsion speed that is dependent on local density [49]. In the following discussions we will discuss the typical methodologies for global and group density in active matter, providing specific examples. Following this we will describe our method for local density, and how this relates to other methods. Unless

otherwise stated, we assume a two-dimensional geometry due to the nature of our final application - tracking beetles in 2D. It is possible to use the methods outlined below in three and sometimes higher dimensions, although we shall not discuss this.

### Global Density

To understand our local density calculation we will begin with global density. Global density of a group of confined individuals (in 2D) is usually computed in the MIPS literature via a packing fraction. This measure takes the coverage of the system size  $A$  by particles of size  $r$ . For  $N$  circular particles of uniform radius  $r$  this calculation gives  $\frac{N\pi r^2}{A}$  for a system area  $A$ . This is a particularly useful measure in simulations of Active Brownian particles, including MIPS simulations, confined to “hard” or periodic boundaries, and is a primary parameter of interest for the onset of MIPS [48]. This measure becomes difficult to use in unbounded or effectively unbounded systems where either  $A \rightarrow \infty$  or  $A \gg Nr^2$  (approximating particles as circles). In both cases the value of the packing fraction is driven to or close to 0 even if the system has spontaneously clustered into a cohesive group (e.g. bird flocks or fish schools), or even a MIPS like assembly. In computational work a truly unbounded space can be achieved, i.e. by simply not including boundary conditions in the simulation code, whereas in nature there is often a lack of a well-defined system size. E.g. consider a flock of starlings, the analogous packing fraction in 3D would scale as  $\sim \frac{Nr^3}{V}$  but in this case what should one take for  $V$ ? One could take the volume of the earth’s atmosphere, or less extremely the volume of atmosphere around a city. In either case the density approaches zero for realistic  $N$ . Yet a group of starlings can remain at high density; clearly a different approach should be used.

In these situations ( $A^{1/d} \gg r$ ) another method of group density computation is usually used with the methodology of determining the “shape of the group” as a whole and using the area (or volume) of this shape as the area denominator. Two particular approaches to this we will report as the Convex Hull group density,  $\rho_{CH}$ , and the  $\alpha$ -shape group density  $\rho_\alpha$ . These are closely related. The Convex Hull is a geometric object that can be defined for an arbitrary set of points  $\mathbf{X}$  in  $\mathbb{R}^d$  as (i) the unique minimal convex set containing  $\mathbf{X}$  or (ii) the union of all simplices with vertices in  $\mathbf{X}$  [89]. The analogy usually used to describe this is to imagine a rubber band being stretched to enclose all points in  $\mathbf{X}$  then allowing it to become taut around the points, this is the convex hull. As such  $\rho_{CH}$  will use the area of the convex hull in the density calculation. This can be computed in 3D for the Starlings considered above giving a natural interpretation of density, provided the group does not fragment.

The Delaunay Triangulation (DT) and  $\alpha$ -shape are intricately related, given that the latter is constructed from the former [90]. Briefly the DT is a set of triangles (in 2D) that triangulate a set of points  $\mathbf{X}$ , such that no point in  $\mathbf{X}$  is contained within a circumcircle of any triangle in the DT. An  $\alpha$ -shape can be constructed via a “carving”-process on the DT using a “spoon” or “scalpel” defined by a circle/sphere with radius  $\alpha$ . The carving proceeds to remove segments of the triangulation’s shape via the circular scalpel whenever one (or many) may occupy that space without enclosing a point of the Triangulation (a point of  $\mathbf{X}$ ) [91]. This means the  $\alpha$ -shape can form holes within the body of the original convex hull. For example a “donut” shaped point cloud (2D) or a torus can be recovered using an  $\alpha$ -shape construction, whereas a convex hull will not generate the hole. Similarly, we now use the area of the  $\alpha$ -shape (or more accurately the  $\alpha$ -hull) as the denominator for our density. The advantage of the  $\alpha$ -shape is its more form fitting nature, as well as its ability to “carve out holes” in the shape of data, however this comes at the cost of the  $\alpha$  hyperparameter which must be selected [90].

The  $\alpha$ -shape approach was used by Sosna *et al.* [92] to compute a group density measure applied to fish schools. Of note here is their similar experimental setup involving a top-down perspective on freely moving animals where the tank area is much larger than the size of an individual, and indeed the “area of the group”.

The  $\alpha$ -shape methodology provides a more form fitting “shape” than the convex hull, and therefore a more accurate calculation of (number) density. This comes at the cost of a hyperparameter  $\alpha$  which is not clear *a priori*. One method to fit the  $\alpha$  parameter is pick  $\alpha$  by minimising the area subject to all data points remaining within the resulting  $\alpha$  shape. This methodology leaves the only subjectivity as when to end minimisation. One approach being to select a suitable minimal change in the area between optimisation iterations. But optimisation comes at the cost of computing power. Applications involving large data sets, or otherwise high-throughput algorithms, may be unable to handle this additional hyperparameter optimisation. Such applications may judge the convex hull more efficient at the cost of its less form fitting nature.

## Local Density

We have seen how group density can be defined for unbounded systems, using many related concepts such as the convex hull, DT, and the  $\alpha$ -shape construction. We will now discuss one particular methodology for local density and finally construct our method of local density. The most natural extant method is the Delaunay Tes-

sation Field Estimator (DTFE) which constructs a density estimate for a point  $\mathbf{x}_i$  by dividing the (normalised) mass or weight of  $i$  by the area of the Contiguous Voronoi Cell (CVC). This is defined as the union of Delaunay triangles which share the data point  $\mathbf{x}_i$  as a common vertex (pictured in figure 2.2 inset). The density field for the points can then be constructed via linear interpolation procedure taking advantage of the triangulation to perform a type of gridded interpolation [93]. This methodology has seen success in estimating density fields in astrophysics applications. Here we are less concerned about constructing a field and more focused on the local density at data points  $i$ , and are in particular concerned about boundary effects in the algorithm. The boundary effect we are most concerned about is that an agent internal to the triangulation (not on the convex hull) will naturally have all  $360^\circ$  around it within some Delaunay triangle, whereas points on the convex hull often have a much smaller angular coverage. I.e. compare the starred agent (for which the CVC is shown) and the bottom right agent (with the length scale marker) in figure 2.2. What's more the Delaunay triangles associated with boundary points are more often acute triangles (triangles of tiny area), which can cause large density values to be computed when applying the DTFE. For these reasons we take the step to improve upon the ideas of the DTFE method rather than use it directly.

### Weighted Delaunay Tessellation Local Density

The method used here starts by considering both the Delaunay triangles  $T_i^{(j)}$  associated with the  $i$ th data point  $\mathbf{x}_i$  which form a set indexed by  $j$ , and the set of angles  $\theta_i^{(j)}$  made in the triangle at the vertex  $\mathbf{x}_i$ . We assign the value  $A_i^{(j)}$  to be the area of triangle  $T_i^{(j)}$ , and use these to form the density

$$\rho(\mathbf{x}_i) = \frac{1}{2} \frac{\sum_j \theta_i^{(j)}}{\sum_j \theta_i^{(j)} A_i^{(j)}}. \quad (2.1)$$

This notation is visualised on actual data in figure 2.2. Using this notation we define our density measure by considering beetle  $i$  as contributing  $\theta_i^{(j)}$  of its mass to the triangular area  $A_i^{(j)}$ , this gives a normalised area per particle of  $\frac{1}{\pi} A_i^{(j)}$  contribution for  $j$ . We then calculate the average (over  $j$ ) of these areas weighted by the angles made and invert it to form the local density for this  $i$ . We further normalise this density by  $2\pi$ , since an internal point  $i$  will have  $\sum_j \theta_i^{(j)} = 2\pi$ , but a boundary point will have  $\sum_j \theta_i^{(j)} \leq \pi$ . The aim of this construction is to compute an accurate density, whilst accounting for any boundary effects. Note that since we are implicitly taking the mass as 1 at each point we generate a number density. Similarly,, the

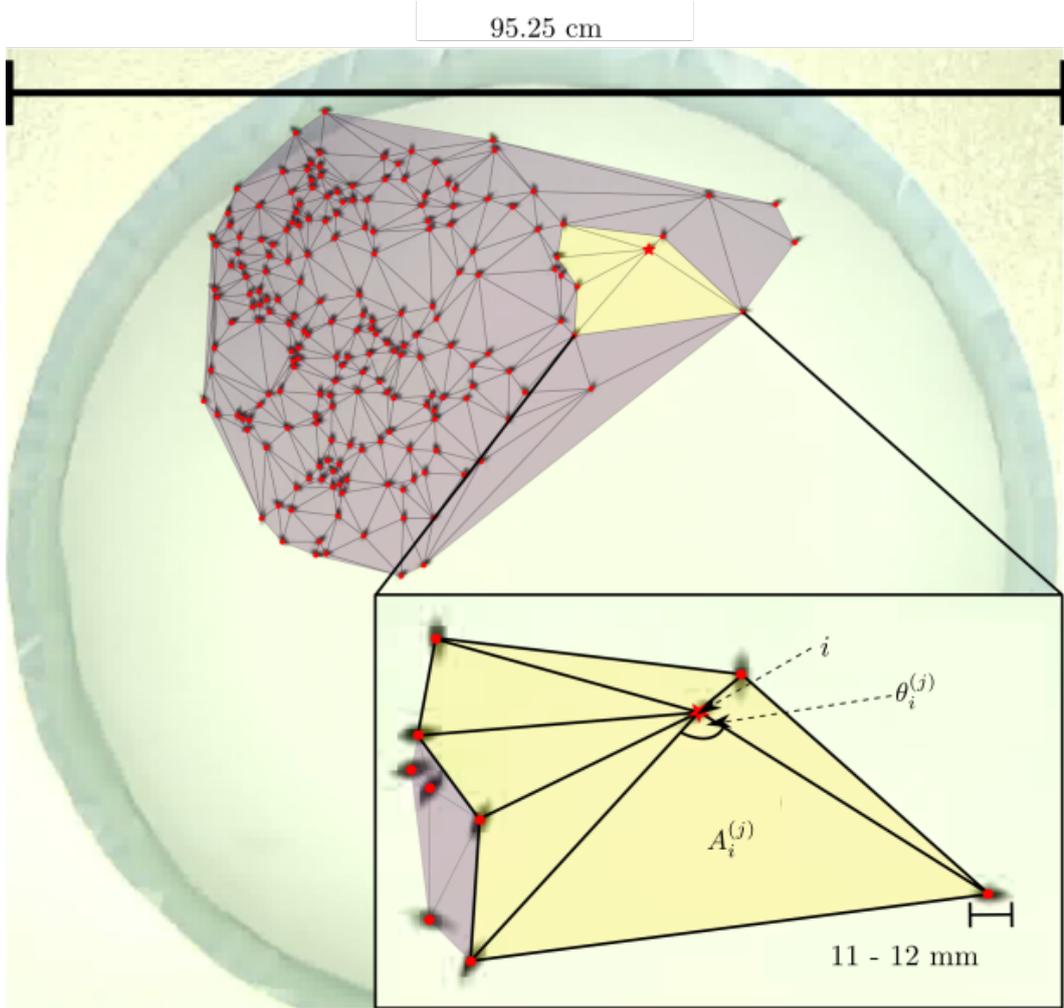


Figure 2.2: An overlay showing the Delaunay Tessellation local density computation applied directly to the data shown in figure 2.1 once positions were extracted. The inset details the geometry relating to the starred beetle indicated by particle index  $i$  (associated with a position coordinate  $\mathbf{x}_i$ ), we show the internal angle  $\theta_i^{(j)}$  made by the  $j$ -th Delaunay triangle to beetle  $i$ . This triangle has area denoted by  $A_i^{(j)}$ . The index  $j$  is understood to enumerate the set of Delaunay triangles which share the point  $\mathbf{x}_i$  as a common vertex. We calculate the weighted average triangle area across the indices  $j$ , weighted by the angles they make at  $i$ . Our density measure is then the inverse of this mean suitably normalised by beetle length  $r$ . Reproduced from [94].

units of area will also be inherited through the areas  $A_i^{(j)}$ . We will be pre-scaling our spatial units to the body length of beetles, or the diameter of circular particles in simulations, as will be detailed whenever necessary.

The result of this methodology can be examined for a particular case as given

in figure 2.3 (b) and (c). Here we calculate the density for a compact square lattice of circular particles each with radius  $r = 1$ , using  $3^2$  and  $6^2$  particles in panels a and b. This example is slightly pathological in terms of the DT since it leaves the triangulation non-unique (due to point co-linearity), however it does illustrate the motivation behind our use of the triangle areas  $A_i^{(j)}$  and angles  $\theta_i^{(j)}$  which are motivated by similar methodologies in condensed matter physics when analysing the stoichiometry of crystal lattices.

In this example one may calculate a density for the green (dashed) particle by noting its four associated triangles all have area  $A_i^{(j)} = 2$  and the angles made at  $\mathbf{x}_i$  are all equal  $\theta_i^{(j)} = \pi/2$ . Therefore, it is simple to calculate  $\rho(\mathbf{x}_i) = \frac{1}{2} \frac{2\pi}{4 \cdot (\frac{\pi}{2} \cdot 2)} = \frac{1}{4}$ , which should be interpreted as a number density  $\frac{1}{4}$  of particles per particle area.

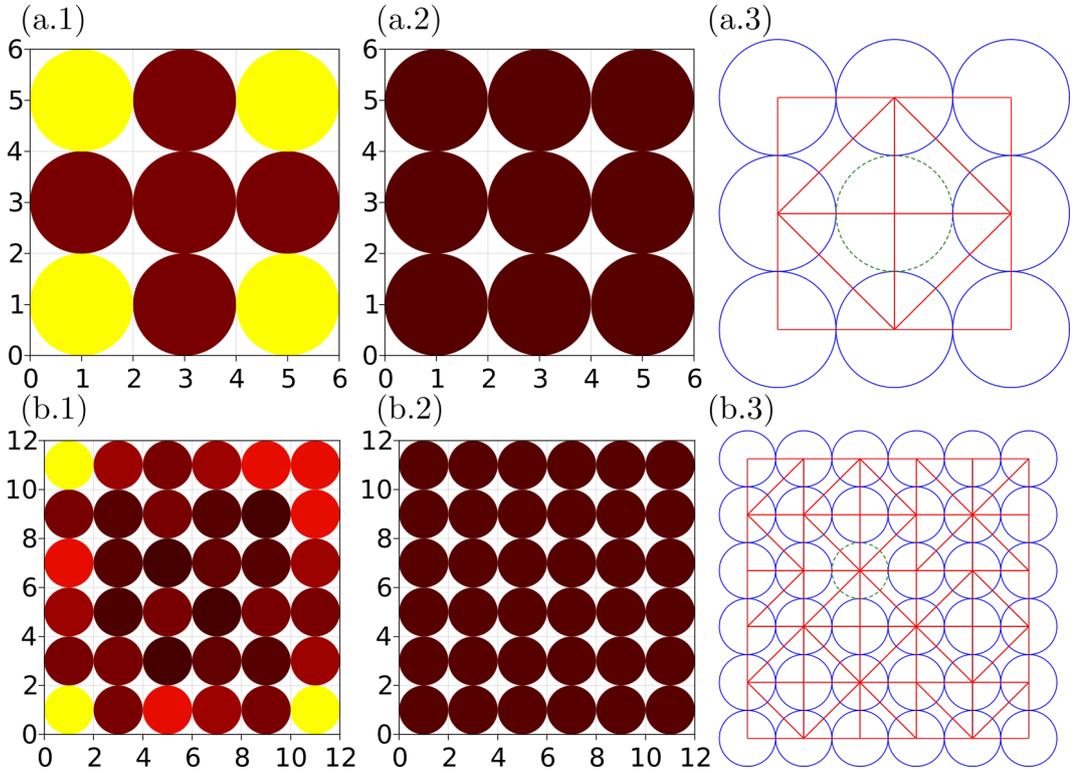


Figure 2.3: Density computed using the DTFE (a.1, b.1) and the proposed method (a.2, b.2) for a lattice of circular discs (radius  $r = 1$ ), panels a.x show a lattice of  $N = 3^2$  and panels b.x show a lattice of  $N = 6^2$ . Panels a.3 and b.3 show a naked Delaunay tessellation of the input data for reference (computed using Qhull [95]), note due to the co-linearity of points the tessellation is not unique. The colour scale is normalised to the output of the DTFE method in both cases (giving an arbitrary scale  $[0, 1]$  and colour scale). Note the variance in the DTFE method (b.1) and the uniform nature of the local density in the proposed method (b.2).

## 2.3 Self-Propulsion Statistics

We define self-propulsion speed by using a two point stencil [96] definition for the first derivative of beetle position as a function of time i.e

$$\mathbf{v}_i(t) = \frac{\mathbf{x}_i(t + \Delta t) - \mathbf{x}_i(t - \Delta t)}{2\Delta t}, \quad (2.2)$$

and taking its norm. Here we identify  $\Delta t$  as the inter-frame time which is  $\frac{1}{30}$  seconds. We use the 2-point stencil as apposed to a forward or backward Euler method for lower error,  $\mathcal{O}(\Delta t^2)$ . We could use a higher-order stencil however increasing orders requires the use of data points at larger temporal gaps which means more data cannot be used. The 2nd order stencil was chosen to give increased accuracy at shorter timescales (our frame rate is 1/30) without reducing the quantity of data significantly around the boundaries (start and end of the video data). We analyse the statistics of speed  $|\mathbf{v}_i(t)|$  to investigate the role of inertia in our data.

### 2.3.1 Inertial Delay

Interest in inertial particles in active matter, and particularly inertial ABPs, has seen a growing interest in the context of motility induced phase separation. One characteristic of inertial SPPs is the delayed correlation between actual particle velocity and self-propulsion direction, which can be quantified with the correlation function [31]

$$C(\delta t) = \langle \hat{\mathbf{V}}_i(t + \delta t) \cdot \hat{\mathbf{n}}_i(t) \rangle_{i,t}. \quad (2.3)$$

That is for a given time lag  $\delta t$  we measure the dot product between the major axis orientation of a beetle  $\hat{\mathbf{n}}(t)$  at time  $t$ , and its normalised velocity  $\hat{\mathbf{v}}_i(t + \delta t)$  at time  $t + \delta t$ . Examining this function across positive and negative time lags  $\delta t$ , averaged over beetle index  $i$  and time  $t$ , should elicit a peak correlation at lag  $\delta t = 0$ , for non-inertial particles, and some finite time lag, for inertial particles. By computing the correlation function  $C(\delta t)$  on the  $N = 200$  beetle data (shown in figure 2.4) we obtain a small but positive delay time of 13 ms between beetle orientation and velocity, this means that the velocity is on average lagging behind beetle orientation by a corresponding timescale.

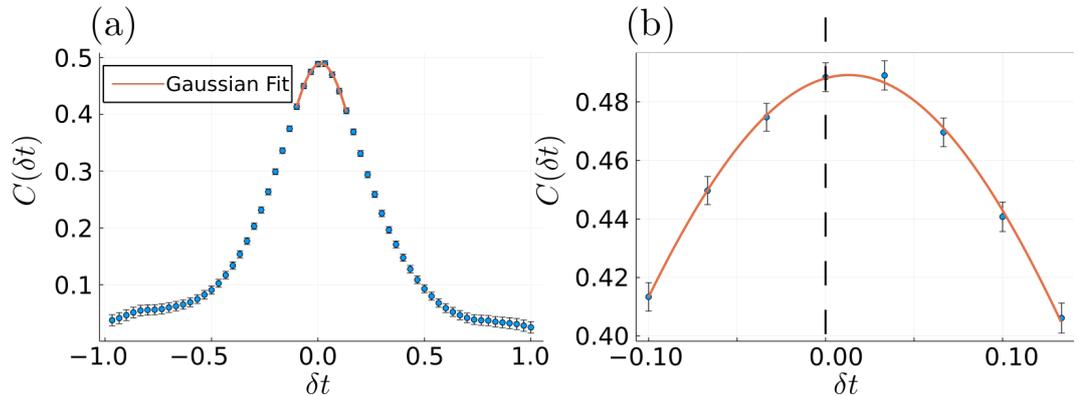


Figure 2.4: Orientation velocity correlation function  $C(\delta t)$  plotted for large (a) and small (b) time lags  $\delta t$ . Dashed line in (b) shows  $\delta t = 0$ . Note we find a small but positive value of  $\delta t$  at which the correlation function attains a maximum. Data plotted is for the  $N = 200$  case only, we fit a Gaussian function to the data to obtain a delay of 13 ms. The small delay would be a natural consequence of inertial effects.

### 2.3.2 Speed and Density

As previously mentioned the relationship between speed and local density is characteristic of MIPS. Armed with a measure of local density we will now use this to examine the relationship between speed and density that we see in the trajectory data.

Using our definition of self-propulsion speed,  $v_i(t) = \|\mathbf{v}_i(t)\|_2$  from equation 2.2, for beetle  $i$  at time  $t$ , we can associate this data with local densities  $\rho_i(t)$ . This data is then averaged over both time  $t$  and beetle index  $i$ , by first binning the density on a log-spaced scale and taking the average value of the bins, to form the density dependent speed  $v(\rho)$  which we plot against density on a log-log scale in figure 2.5.

### 2.3.3 Features in the Speed Density Relationship

Three features are most prominent in the data (1) we see a self-propulsion speed decay (roughly  $\rho^{-0.4}$ ) with increasing local density, this lasts between approximately  $\rho \sim 10^{-2.5}$  to  $\rho \sim 1$ , two orders of magnitude (2) we also see a significant and abrupt upward trend in self-propulsion speed after around  $\rho \sim 1$ . Finally (3) the  $N = 50$  is notably offset from the  $N = 100, 200$  groups, and the  $N = 100$  groups is somewhat offset from the  $N = 200$  group, both in terms of an increased speed for a common density.

For the three most prominent features we can interpret that there is a significant dependence of beetle self-propulsion speed with local density in all three group

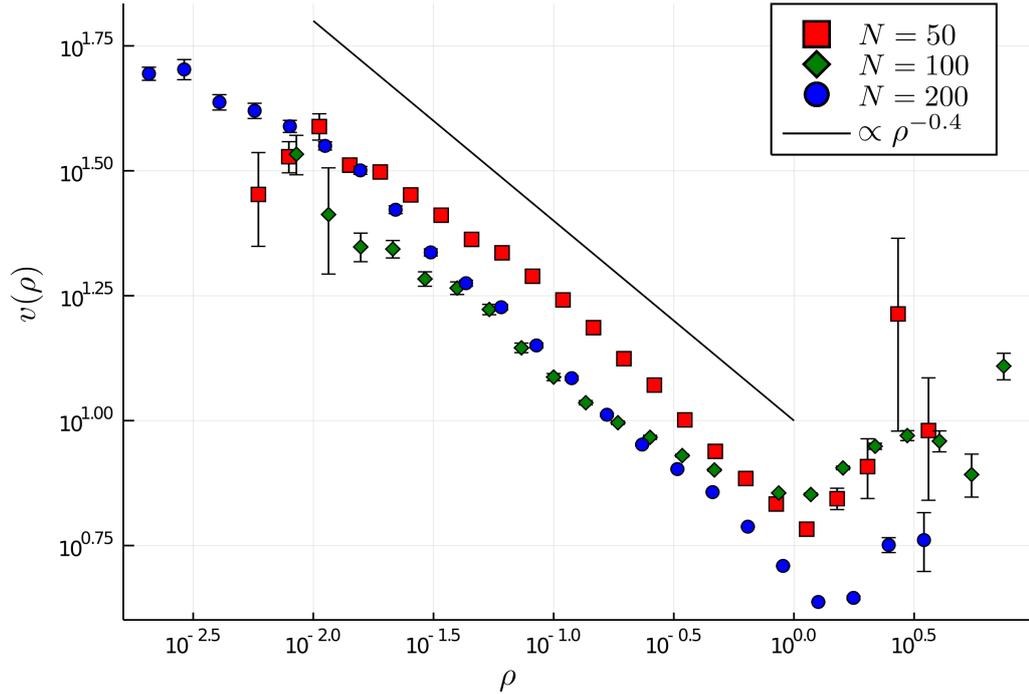


Figure 2.5: Relationship between beetle self-propulsion speed  $v(\rho)$  and local density  $\rho$ . Local density is computed using the method proposed in section 2.2 and outlined in figures 2.2 and 2.3. Data is plotted for three groups ( $N = 50, 100, 200$ ) on a log-log scale, a guide to the eye showing a proportionality of  $v(\rho) \sim \rho^{-0.4}$  is also plotted. We see a decay in self-propulsion speed with increasing density up to around  $\rho \sim 1$  where a notable uptrend is seen, this occurs over two orders of magnitude. Approximately from  $\rho \sim 10^{-2.5}$  to  $\rho \sim 1$ . The upturn may be due to collisions or a change in beetle behaviour near collision. Data is plotted after first binning the density axis (on a log-scale) and averaging speeds in the bins, error bars show the standard deviation within bins.

sizes. This phenomenology is commensurate with MIPS or a MIPS like mechanism. Our SPPs (beetles) display a decreasing speed with increasing local density.

Two further questions immediately spring from figure 2.5 (1) why does the power law arise (2) why does the power law break and speed increase at density  $\sim 1$ . For (1) we could conjecture that either this is a generic property of ABPs with collisions or it is somehow related to beetle behaviour, or perhaps over motion phenomena such as inertial effects. For (2) we could also conjecture the same, perhaps collisions cause this increase in speed. Although conservation of momentum suggests otherwise, or perhaps this is due to some sort of behavioural response of the beetles e.g. recognising immediate collisions and attempting to avoid them by increasing speed away from the possible collisions.

### Velocity density relations in ABP Simulations

Analysis of whether behaviour is causal for (1) and (2) is not answerable directly from our data, we would require a dedicated experiment. However, we can give a partial answer by determining if this  $v(\rho)$  phenomenology is seen in ABP assemblies. If so this would at least allow us to say with confidence that these features *can be* the result of collision forces and active diffusion processes alone, rather than just “behaviour”.

To do this we will simulate an assembly of  $N$  ABPs. We assume circular particles of radius  $r$ , with equations of motion

$$\frac{\partial \mathbf{x}_i(t)}{\partial t} = v_0 \begin{bmatrix} \cos \theta_i(t) \\ \sin \theta_i(t) \end{bmatrix} + \mu \sum_{j \neq i} \mathbf{F}_{ij}(t), \quad (2.4)$$

$$\frac{\partial \theta_i(t)}{\partial t} = \sqrt{2D_r \Delta t} \eta_i(t). \quad (2.5)$$

Each particle,  $i$ , has a position  $\mathbf{x}_i(t)$  and orientation  $\theta_i(t)$  at time step  $t$ . The force term is  $\mathbf{F}_{ij}(t) = 0$  when  $i = j$  or when  $\|\mathbf{r}_{ij}\|_2 > 2r$  with  $\mathbf{r}_{ij}(t) = \mathbf{x}_j(t) - \mathbf{x}_i(t)$  otherwise  $\mathbf{F}_{ij} = (\|\mathbf{r}_{ij}\|_2 - 2r)\hat{\mathbf{r}}_{ij}$ , where hats denote unit vectors throughout. Finally,  $\eta_i(t)$  is Gaussian white noise with no time (or inter-agent) correlations. For the sake of this experiment we will hold constant the free parameters  $N = 200$ , the number of particles,  $\mu = 3000$ , the velocity close to first contact (chosen to be as large as possible for computational stability),  $\Delta t = 1 \times 10^{-5}$  the time step which is chosen while balancing run time with numerical instability, speed  $v_0 = 10$ , rotational diffusion  $D_r = 0.005$ , and finally  $r = 1$  the particle radius. The value of  $\mu$  was chosen (with  $\Delta t$  for stability/runtime) to ensure minimal particle overlaps, since this is a soft-potential. The typical particle overlap  $2 - \delta$  can be discerned from

relating active and collision forces,  $v_0/\mu \sim 2r - \delta$ . Which for our selected speed gives  $\ll 1\%$  of a radius  $r$  for  $v_0 = 10$ . The spring potential itself is chosen since as part of the Fily and Marchetti model [48] it represents a purely repulsive minimal model of MIPS in ABPs.

Implicitly we have neglected inertia, hydro-dynamics, and translational noise which could all play a role. Numerically we solve the first order Stochastic differential equation (SDE) using the Euler-Maruyama method [97], which with a time step of  $10^{-5}$  is sufficient to avoid numerical instability. Our units normalised from the particle diameter  $2r$  (length) and by using the self-propulsion speed  $2r/v_0$  (time).

On simulating the trajectory data for  $N = 1000$  and  $R = 5$  separate replicates (using a random seed) we then compute speed and density statistics using the same methodology as for the beetle data. Except we account for the periodic boundaries so that a particle at the simulation boundary do not generate extremely large anomalous speeds or under-estimated densities. To this end we take a speed and density measurement on a trajectory if at all three times (for the 2-point stencil velocity)  $t - \Delta t, t, t + \Delta t$  a boundary is not crossed. For the speed we could use the nearest image convention to account for boundary crossing. However, for the density methodology this is in principle possible but more complex to implement. We would need to enclose the system by all nine image cells, realising the nearest images. Then compute the Delaunay triangulation on this modified position data. After which we would need to discount the data corresponding to the nine image cells in our statistics. Which is effectively the same as not collecting statistics at the boundary.

The speed and density data is plotted in figure 2.6 (a). We see a downward trend in speed with increasing local density, but this dependence is better explained by a linear fit than by a power law (see inset) unlike the beetle data. Theoretically the linear scaling might make sense due to the linear nature of the force response, although this data is not the force response itself but an aggregate of speeds at given densities.

### **Including Inertia**

It is clear that non-inertial ABPs do not show a clearly non-linear relationship between speed and density. In fact, it appears that a linear relationship best explains the data. Since we find a small inertial effect in our data, the natural question is can inertial effects account for the non-linearity in  $v(\rho)$ ?

To answer we can also conduct the same computer experiment but with inertial ABPs. Inheriting the same terminology from equations 2.4 and 2.5. We

define a system of  $N$  inertial ABPs with translational and rotational equations of motion

$$M \frac{\partial^2 \mathbf{x}_i(t)}{\partial t^2} = \xi_t \left( v_0 \begin{bmatrix} \cos \theta_i(t) \\ \sin \theta_i(t) \end{bmatrix} - \mathbf{V}_i(t) \right) + \mu \sum_{j \neq i} \mathbf{F}_{ij}(t), \quad (2.6)$$

$$J \frac{\partial^2 \theta_i(t)}{\partial t^2} = \xi_r \left( \sqrt{2D_r \Delta t} \eta_i(t) - \omega_i(t) \right). \quad (2.7)$$

Where  $\mathbf{V}_i(t) = \frac{\partial \mathbf{x}_i(t)}{\partial t}$  and  $\omega_i(t) = \frac{\partial \theta_i(t)}{\partial t}$  are a particle's velocity and angular velocity respectively. Note compared to non-inertial case we now have some extra parameters: the particle mass  $M$ , moment of inertia  $J$ , and the translational and rotational friction coefficients  $\eta$  and  $\eta_r$ . For clarity on the units, for the translational equation we can measure space and time as  $2r\bar{\mathbf{x}}_i(\bar{t}) = \mathbf{x}_i(t)$ ,  $2r/v_0\bar{t} = t$ . For the rotational equations note  $J$  has units of mass  $\times$  length<sup>2</sup> which it inherits from the units taken in the translational equation. We take  $\theta$  in radians meaning  $D_r$  will be transformed with the chosen time units.

We set  $M = J = \xi_t = \xi_r = 1$  initially as a simple example, with large inertial effects. We also choose, as in the non-inertial case,  $D_r = 5 \times 10^{-3}$ ,  $v_0 = 10$  and  $r = 1$ . Due to the second order nature of the equations of motion numerical integration needs to be completed with more care to avoid instabilities<sup>1</sup>. For stochastic second order equations, particularly those in molecular dynamics arising from the Langevin formalism, the Euler-Maruyama is accurate up to first order in the time step. Unfortunately some higher order methods such as the Milstein method [97] reduce to EM since our diffusion term is constant in space and orientation e.g. our rotational SDE for one particle can be written in the form  $d\omega = a(\omega, t)dt + \text{Constant} \times dW(t)$  for  $d\theta(t)/dt = \omega(t)$ , and Wiener increment  $dW(t)$ . However simple methods accurate up to second order in the time step can be found such as the Grønbech-Jensen Farago (G-JF) method [98] which is analogous to Verlet integration schemes, but in the stochastic setting. To apply the G-JF scheme we must be careful of the periodic boundary conditions due to the integrator using positions at time  $t$  and  $t - \Delta t$ , one way to solve this is to set the previous position equal to the current at the time when the position wrapping happens, a second would be swapping to an Euler-Maruyama step for that time point. In either case we are careful not to include particles near

---

<sup>1</sup>We are also careful with the initial positions of particles, each particle is randomly placed in a box of constant size  $L = \sqrt{N\pi r^2/\phi}$  for density  $\phi = 0.1$ , subject to the condition it does not overlap with any previously placed particles. Overlaps in the initial configuration can cause dramatic numerical instability in the inertial simulation, much more so than the non-inertial case, and so much so it bears writing explicitly for reproducibility's sake. We also do the same in the non-inertial case to be safe.

the boundary in our statistics anyway. We opt for the velocity free form, equation 24 in [98], and take a time step of  $\Delta t = 10^{-5}$ . Note the relevant time scales are given by  $M/\xi_t$ ,  $J/\xi_r$ ,  $\sqrt{M/\mu}$  and  $v_0/D_r$  and we must have  $\Delta t$  small enough to resolve these. For the inertial system we find a non-linear relationship between  $v(\rho)$  and  $\rho$ ,

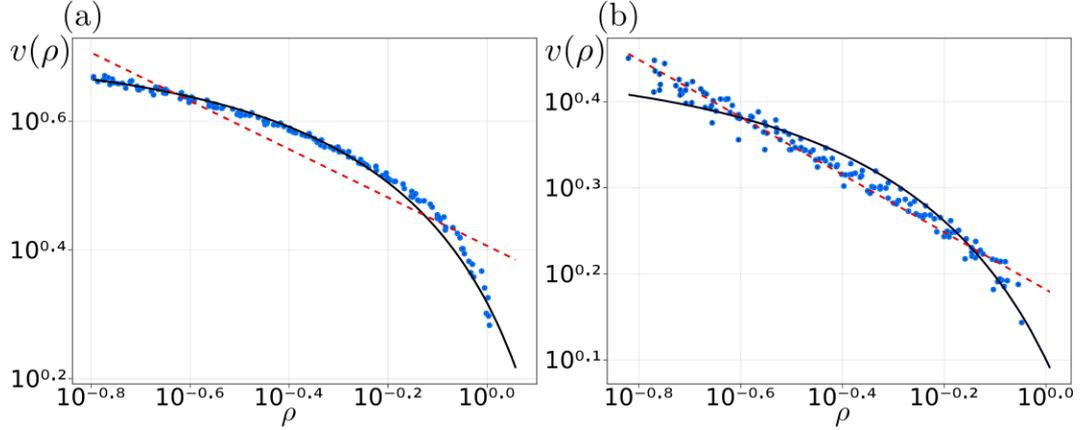


Figure 2.6: Speed density relationship for non-inertial (a) and inertial (b) ABPs. In (b) the inertial parameters are  $M = 1$ ,  $J = 1$ ,  $\eta_r = \eta_t = 1$ . In both cases a linear  $v(\rho) = a\rho + b$  and power-law  $v(\rho) = a\rho^b + c$  have been fitted (least squares) as solid and dashed lines respectively. For both (a) and (b) common parameters are  $v_0 = 10$ ,  $D_r = 5 \times 10^{-3}$ ,  $r = 1$ ,  $\mu = 3000$  and time step  $\Delta t = 10^{-5}$ . Particles were initialised in a box with packing fraction 0.4 such that  $n$  particles were overlapping at time  $t = 0$ . Notable is the non-linear best fit for inertial particles, and the lack of an increasing speed at higher densities.

shown in figure 2.6 (b). From this we can conclude that a non-linear relationship is a better fit to the data than a linear one, for these parameters.

### The Effect of Mass and Moment of Inertia

The experiments above take the simplifying step of setting a constant  $M = \xi_t = 1$  and constant  $J = \xi_r = 1$  to examine one particular case with large inertial effects. To explore more systematically we will examine the effect of varying mass  $M$  from small  $10^{-3}$  to large masses 1, and analogously for  $J$ , but keeping the friction coefficients  $\xi_t = \xi_r = 1$ . We might expect that a mass and moment of inertia approaching zero would result in similar statistics as simulating equations 2.4 and 2.5, namely the non-inertial equations, with  $M = 0$  equation 2.6 mathematically reduces to equation 2.4. Numerically we need to be careful when reducing  $M \rightarrow 0$  (or  $J$ ) since numerically solving equation 2.6 implies a division by  $M$ ; if  $M$  is too small the numerical solver becomes unstable unless  $\Delta t$  is greatly reduced to compensate. For this reason we

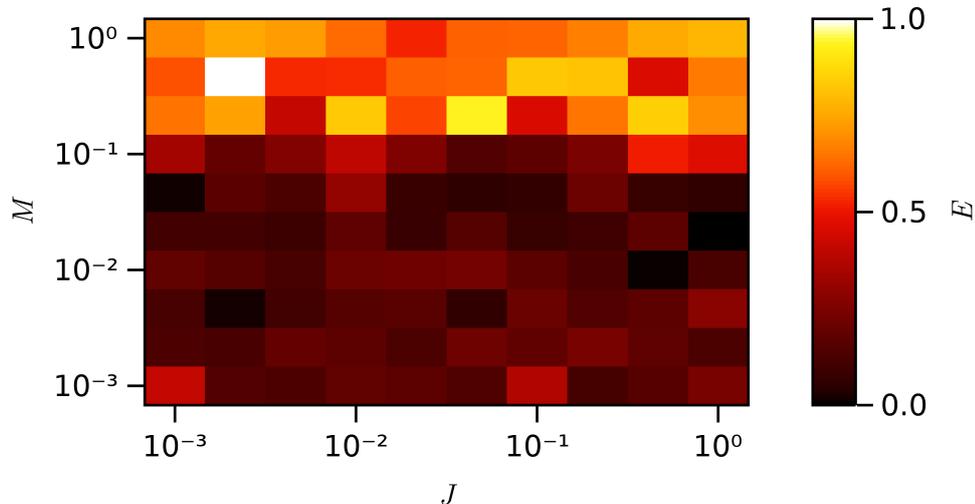


Figure 2.7: Results for simulating equations 2.6 and 2.7 using masses  $M$  and moments of inertia  $J$ ,  $D_r = 5 \times 10^{-3}$ , and  $v_0 = 10$  and  $r = 1$ . Colours show the difference in root-mean-square error between the linear and power law models scaled to  $[0, 1]$  where 1 indicates the power law has a lower fitting error.

choose masses  $M \in [0.001, 1.0]$  and similarly for the moment of inertia  $J$ .

Figure 2.7 is used to summarise the results. The plot details the linear and power law fits in each grid point with a background colour indicating the (normalised) improvement of the power law over the linear fit. We find that at  $M \sim J \sim 1$  a power law emerges as a better model for  $v(\rho)$  whereas for increasingly small inertial effects a linear fit is superior.

In the simulations data presented above the particles are either moving at speed  $v_0$  (non-inertial), accelerating to speed  $v_0$  (inertial), or are slowed due to collision forces. There is no active decision-making in this model. From this it makes sense to examine the effect of a collision on speed, in one spatial dimension we can examine the collision response quite easily.

If we assume a particle that is a unit circle, which is initially at  $x(0) = 0$  with speed  $v(0) = v_0$ , and we hold its orientation constant  $\theta(t) = 0$  along the positive  $x$ -axis. If an immovable object, a potential, sits at  $x = 2$  applying the force  $-k(2 - r)$ , keeping  $t \sim 0$ , we obtain a simple ordinary differential equation

$$\frac{dx}{dt} = v_0 - kx, \quad (2.8)$$

for the particle's velocity. Since this equation is the speed of the particle it is clear

we expect a linear speed dependence on  $x$ . However, considering an inertial particle we instead obtain a second order equation which we can easily solve for the function  $x(t)$  and therefore the speed giving solutions of the form

$$\frac{dx}{dt} = \alpha c_1 e^{\alpha t} + \beta c_2 e^{\beta t}. \quad (2.9)$$

The constants are  $\alpha, \beta = 1/2(\pm\sqrt{4k+1} - 1)$  and  $c_1, c_2$  given by the initial values  $x(0) = 0, v(0) = v_0$ . To connect these equations to  $v(\rho)$  we would need to find the speed as a function of density,  $dx/dt(\rho(t))$ . For this toy model we cannot compute a DT (we cannot even form a triangle), and so cannot use our density methodology directly. But by construction of the problem, if we imagine the potential as a particle, initially increasing  $t$  from 0 (i.e. increasing  $x$  from 0) represents an increasing density, implicitly. It is possible then that the collision response gives the linear and non-linear character in the non-inertial and inertial models respectively.

It appears that a power law  $v(\rho)$  could be explained by inertial effects in the data however we already have determined a small inertial effect in our data, on the order of a 10-millisecond delay time.

## 2.4 Corralled Active Brownian Particles

To model our data we wish to use as simple a model as possible that we know displays MIPS, but also one that functions in fully unbounded space ( $\mathbb{R}^2$ ), with open boundary conditions. This necessitates some form of net attractive force in the dynamics, to avoid the density of the system tending to zero. To meet these goals we define our model to be as close to the basic model of Fily and Marchetti [48] as possible to make as clear contact with MIPS as we can numerically. But we modify the model to function in unbounded space with open boundary conditions. The model assumes  $N$  circular particles of radius  $r$  following equations of motion

$$\frac{\partial \mathbf{x}_i(t)}{\partial t} = v_0 \begin{bmatrix} \cos \theta_i(t) \\ \sin \theta_i(t) \end{bmatrix} + \mu \sum_{j \neq i} \mathbf{F}_{ij}(t), \quad (2.10)$$

$$\frac{\partial \theta_i(t)}{\partial t} = \sqrt{2D_r \Delta t} \eta_i(t) + \kappa_i(t). \quad (2.11)$$

We use a density dependent re-orientation term in the rotational dynamics. i.e. a phenomenological torque of the form  $\kappa_i(t) = \tau \rho_i(t)^{-\alpha} (\mathbf{v}_i(t) \times \hat{\mathbf{R}}_i(t))$  where  $\alpha \geq 0$  and  $\tau \geq 0$  are free parameters,  $\rho_i(t)$  is the local density of particle  $i$  at time step  $t$ ,  $\mathbf{v}_i(t)$  is the velocity of particle  $i$  at time  $t$ , and  $\hat{\mathbf{R}}_i(t)$  is the *unit* vector pointing from particles

$i$ 's position toward the geometric centroid (GC)  $\mathbf{R}_i(t) = \langle \mathbf{x}_j(t) \rangle_j - \mathbf{x}_i(t)$ . We choose a density dependent strength, and are careful to use a unit GC vector to create a “metric-free” torque, or at least a torque not explicitly increasing with distance from the GC (which leads to large torques, and spinning particles). The parameter  $\alpha$  is a dimensionless exponent which controls the re-orientations dependence on local density, and the coefficient  $\tau$  is a rate ( $s^{-1}$ ) parameter controlling the strength of re-orientations.

The motivation for the geometric centroid dependence *biologically* can be seen in experiments on Whirligig Beetle dispersal events [12] and in particular Romey *et al* [13] examine which of: “(1) the first individual to startle, (2) the geometric centre or (3) the point of highest density” correlate most strongly with the direction of travel startled beetles take in a flash-expansion event, they report beetles moving more strongly away from the GC. We argue this experimental evidence at least motivates knowledge of the relative *direction* of the GC (the unit vector) is biologically reasonable. Of course in reality beetles will likely suffer some error on their perception of the GC based on their visual input and cognitive process.

Before continuing we remark on two potential issues that exist, (1) the case where  $\rho_i(t) = 0$  and (2) the case where  $\mathbf{x}_i(t) = \langle \mathbf{x}_j(t) \rangle_j$ . Note (1) will not usually be the case unless the group has fully fragmented which can happen when  $\alpha \sim 0$  but even in this case the group would need to disperse far enough for Delaunay triangle areas to exceed machine precision ( $\sim 10^{-16}$ ), and (2) will imply the cross product,  $\mathbf{v}_i(t) \times \hat{\mathbf{R}}_i(t)$ , is zero and so avoids an infinite torque (at least for the purposes of the numerics). Our model is then defined with the equations 2.10 and 2.11 using the same definitions as in section 2.3.3 except we have *no* boundary conditions. We treat  $\alpha$ ,  $\tau$ , and also  $\mu$ , as free parameters.  $\mu$  is free here due to the error in approximating (roughly) 2 : 1 elliptical beetles with circles that leads to overlaps, to account for this geometric error in fitting.  $v_0$  and  $D_r$  we “fit” using human analysis for motion data as apposed to using a fitting procedure which we use for  $\alpha$ ,  $\tau$  and  $\mu$ . We take  $r = 1$  and scale our parameters extracted from beetle data appropriately by body lengths.

## 2.5 Fitting the Model

We have a model, data, and free parameters to fit the model. Such problems have wide-ranging solutions in the literature. One broad distinction of fitting procedures is “Gradient” vs “Gradient-free”. The former involves *a priori* knowledge of the analytic gradients of the optimisation target  $f(\mathbf{p})$  with respect to the parameters  $\mathbf{p}$ .

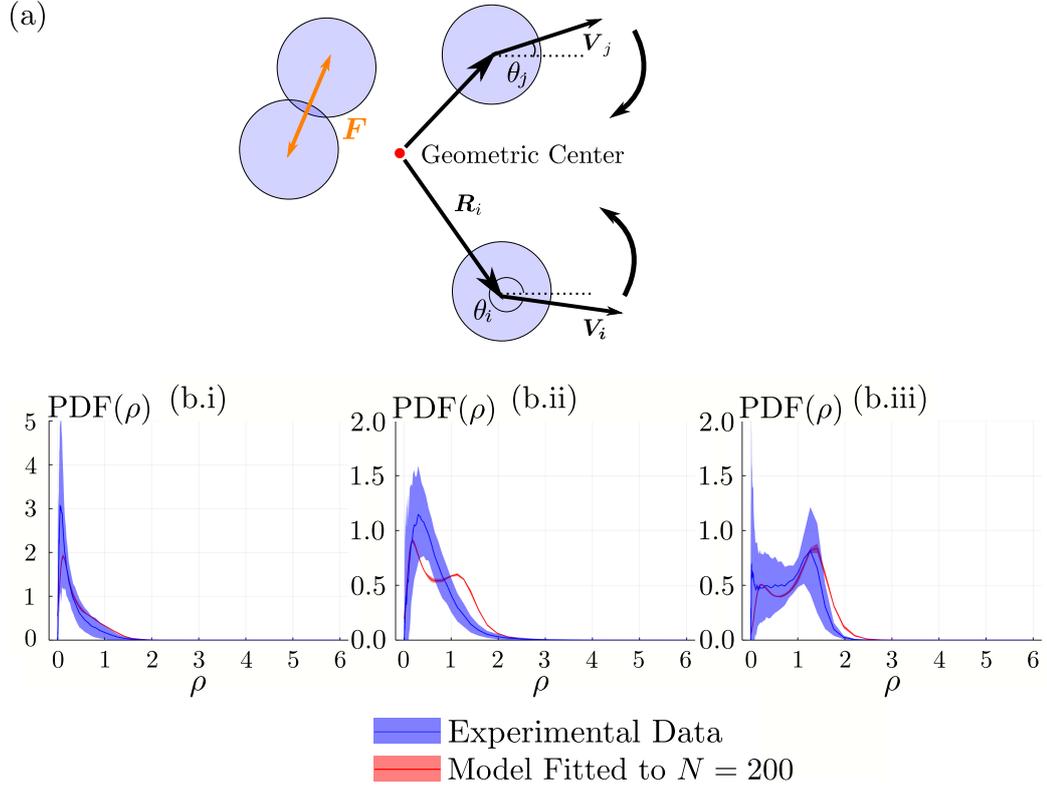


Figure 2.8: (a) Schematic diagram of the model showing the geometric centre (GC) the soft-body force including overlap, and the velocity, orientation, and GC director (not normalised here) data used to compute the re-orientation  $\mathbf{V}$ ,  $\theta$ ,  $\mathbf{R}$  and the rounded arrows respectively. (b) resultant density obtained from fitting the model to the  $N = 200$  data set *only*. The error bars on the data curve (blue) indicates a standard deviation of kernel density estimates taken at  $n = 100$  temporal windows in the data, each window had equal width of  $T/n$  where  $T$  is the total number of time-frames in the data. The error bars on the model indicate one standard deviation between three separate simulations with different initial conditions. Reproduced from [94]

Parameter	$\alpha$	$\tau (s^{-1})$	$\mu k (s^{-1})$	$v_0 (s^{-1})$	$D_r (\text{rad}^2 s^{-1})$
best-fit value	1.1	19.6	316	13.19	2.34

Table 2.1: Best fit parameters corresponding to the data in figure 2.8. Note the exponent  $\alpha$  is a dimensionless number, and we have re-scaled lengths to the beetle length/ particle diameter.

For example back-propagation in neural network fitting is a gradient based approach. The latter does not take gradients as input, but may however compute gradients (or estimates of them in Stochastic systems) *a posteriori*, during optimisation. In our case our objective function involves computing a (Stochastic) simulation with parameters  $\mathbf{p}$  to give trajectories  $\mathcal{T}$ , computing analysis of the simulation, i.e. some observable  $f(\mathcal{T}|\mathbf{p})$ , and comparing to experimental realisations of  $\mathbf{f}$ , simply put we cannot analytically compute gradients for an optimiser, at least not without great difficulty.

Knowing this we opt to use gradient-free methods by using Bayesian optimisation, we use a popular package to implement this procedure [99]. For us the only non-trivial aspects of the optimisation process are the plethora of local minima, and the large computational cost of a simulation (on the order of one minute) and its Stochasticity. The latter meaning we must simulate each parameter set multiple times to reduce fluctuations. Typically, we take 3 – 5 runs per parameter set. Even with GPU acceleration optimisation typically lasted 24 to 48 hours. To solve these we use a large initial random search of roughly 50% of the parameter queries which gives the optimisation procedure a large amount of seed data to find the best minimum, similarly due to the Stochasticity we set the optimiser to take an average of three runs of each parameter set (three to balance the large computational cost). Finally, for us the objective function  $f$  is the mean square error between the Gaussian kernel density estimate of the probability distribution function of density for the candidate simulation and the experimental data. Thus, we aim to find parameters  $\alpha, \tau$  and  $\mu$  which produce a simulation with as close to the density distribution we see in the experimental data as possible.

The fitting results can be seen in figure 2.8, where the model was fitted only to the  $N = 200$  group. We do find that the model computed on non-training data ( $n = 50, N = 100$ ) is broadly accurate, i.e we recover a largely un-clustered phase for  $N = 50$  but see slight bi-modality in  $N = 100$ . We expect that the fitting parameters place a phase boundary (separating unclustered and phase-separated/ bi-modal) at roughly  $N \sim 100$ . In the next section we will explore this and the variance of the model in  $\alpha - \tau$  space.

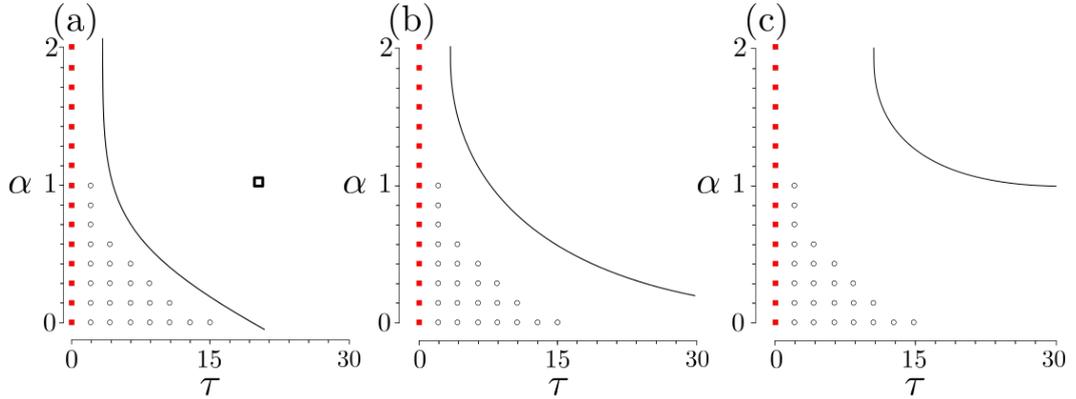


Figure 2.9: The CABP model in  $\alpha - \tau$  space. For  $N = 200$  (a),  $N = 100$  (b) and  $N = 50$  (c). We see a region of uni-modal densities (left of boundary) and another region of bi-modal densities (right of boundary). Data with circles indicate values for which we believe a steady state has not been reached, it is likely this is due to the effect of too weak a reorientation causing dispersal. Filled squares indicate unexplored parameters. The unfilled square in (a) indicates the best fit for the  $N = 200$  case.

## 2.6 Phase Behaviour

To understand the model we can look at how the density distribution varies across different choices of  $\alpha$  and  $\tau$ . The other parameters we will keep fixed. We show this data in figure 2.9

Varying  $\alpha$  and  $\tau$  gives us information on the quality of the fitting process and a measure of uncertainty on the best fit parameters. The same data gives us a phase diagram of the model, i.e. unimodal to bi-modal density distributions across parameter values  $\alpha$  and  $\tau$ , which define the strength of reorientation. To see this fact *a-priori* consider  $\tau = 0$ , clearly as  $\tau$  is the pre-factor of the re-orientation term, in this case there will be *no* re-orientation. With no reorientation the model reduces to active Brownian motion in unconfined space meaning the density will decrease with time to zero. As in this regime the model is unconfined active Brownian motion, the density should tend in distribution to a strongly peaked uni-modal distribution around  $\rho = 0$ . The phase boundary 2.9 (a) (drawn by eye) indicates a rough cutoff between uni-modal and bimodal density distributions controlled by the reorientation term. Notably there are examples of much more extreme bimodal distributions that do not fit the data, and that bimodal density is quite a generic property of the model. The best fit distribution is bi-modal ( $N = 250$ ), so we expect then that there is some intermediate values where uni-modal densities give way to bi-modal densities. The best fit simulation result is well within the phase

boundary, but we can see a significant variance in parameters with similar results. Similarly, for  $N = 50$  and  $N = 100$  we also compute the phase diagrams in figures 2.9 (c) and 2.9 (b) respectively. The  $N = 50$  case does not show clearly bimodal distributions. For  $N = 100$  the bi-modality is clearer but still not as obvious as for  $N = 200$ . Both  $N = 100$  and  $N = 50$  include much larger areas of clear unimodal distributions and in particular larger regions where the density is tending to a concentrated distribution around 0 density meaning the group has not remained cohesive.

## 2.7 Conclusions

In this chapter we have followed a top-down approach to understand the phenomenology seen in a particular example of active matter. We began with an outline of the experimental data collection and processing from raw data into trajectory data. Then with the context of MIPS we discussed various extant methodologies for the calculation of density in active matter, from global to local definitions. Following this we introduced our own method for local density calculation, based upon the DT, which we then use alongside beetle speed statistics to examine the relationship between beetle local density  $\rho$  and beetle self-propulsion speed  $v(\rho)$ . This elicited numerous features, at leading order a speed decaying with density that is one core constituent of MIPS, and at higher order what appears to be a power-law decay that breaks to an increasing speed at high densities. To attempt to explain these features we studied simulation data for non-inertial and inertial ABP's. Following the same methodology, we found a decaying speed with increasing local density in all cases, with a linear dependence in the non-inertial case and a non-linear, perhaps power law, dependence in the inertial case. In both we did not find the upward trend at high densities  $\rho \sim 1$  as seen in the beetle data. Although we cannot rule out non-trivial beetle behaviour generating the features we see in the experimental data for  $v(\rho)$  (figure 2.5), we can at least say the power-law phenomenology *can be* explained by active diffusion processes with collisions at sufficient density combined with inertial effects.

Our behavioural data for Whirligig beetles fulfill one criterion associated with MIPS; a self-propulsion speed decaying with increasing local density. We show this could be explained by inertial ABP's. Interestingly though we did find that inertial effects, which alter the MIPS phase diagram, are weakly represented in the experimental data, as we found in examining the correlation function between beetle orientation and velocity in figure 2.4. Whatever the case the density decay is present

in the experimental data consistent with our hypothesis of a MIPS like mechanism.

After examining the experimental data, we moved on the fitting of a model inspired by the phenomenology observed in the data. We identify the fact that beetles re-orientate back into the cluster when they have large excursions, and present the CABP model to include such a re-orientation term dependent on local density. We found this is sufficient to explain the emergence of the bi-modal density distribution, and examining the model in the space of the re-orientation term's free parameters we find a clear transition from uni- to bi-modality in the density distribution.

One unanswered question in our analysis is what exactly does generate the behaviour we see in the local density dependent speed. Future research could investigate to what extent, if at all, this is natural behaviour by Whirligig beetles perhaps by a detailed assessment of Whirligig biomechanics in large groups and especially just after collisions. However, this and what we have seen in this chapter are top-down approaches to the problem. Another avenue of future research from a bottom-up perspective could be to determine the equations of motion in a completely non-biased way. For example in the literature we find examples of determining interaction rules by fitting equations of motion by comparing individual trajectories [66] and/or responses to neighbour position [67, 68, 69, 15, 70]. It would be interesting to do the same in Whirligig beetles, especially over different groups sizes. But more importantly fitting equations of motion by comparing individual trajectories or individual responses start with an already biased model, that is one informed by the context of the field up to now, giving a pre-defined equation of motion.

One novel direction would be to attempt the fitting of an "intelligence" based method, that is instead of fitting response forces to those seen correlated with nearest neighbour positions, and using this in a pre-determined equation of motion, one could attempt to generate these response maps from sensory input, say visual input using a neural network or other machine learning approach in an unsupervised fashion. The difficulty will likely be in the quantity and quality of data required to do this, and in how decipherable the fitted model is. A second would be to use recent methods for data-driven discovery of differential equations and their parameters [100, 101], and extensions to work with Stochastic differential equations [102]. The torque term we introduce as part of our CABP model is an ideal candidate for these discovery methodologies. The data we have used in this thesis is however the limiting factor in applying these methodologies, more data with higher spatio-temporal resolution would be required to pursue this direction.

## Chapter 3

# Collective Motion from Future Path Entropy Maximisation

In this chapter we will introduce a future path entropy maximisation algorithm to be applied to active matter. The model will be built with a bottom-up approach, and yields behaviour reminiscent of bird flocks, insect swarms, and fish. The model is built from the stipulation that all members of the group, individually, aim to maximise their estimations of visual state entropy over future paths that they perceive. Physically this is motivated by the idea of “keeping options open” as a heuristic for intelligent decision-making. Here we cast this idea directly as an entropy over future paths, the paths being formed by a particular set of actions an agent may take (e.g. choosing a speed, changing velocity direction), calculated by the empirical count distribution of future visual states agents anticipate over these paths. The maximisation of path entropy is then understood as an agent selecting the action leading to the highest path entropy (or in some way selecting among degenerate maxima). The paths and states along them are also conveniently thought of as part of a Future States Tree (FST). The algorithm itself, individuals maximising future path entropy, is completely deterministic. With the possible exception of action selection with degenerate maxima. Nevertheless, we will see this deterministic algorithm leads to cohesive, highly-ordered, group motion. As well as other motion phenotypes. Biologically we propose this model as a reasonable cognitive process real-world active matter may follow, such as birds or insects. The visual state represents input to an individual’s retina, or as will be discussed an abstract representation of this “higher” up an agents cognitive processing. The future paths represent a reasonable judgement individuals can make about the potential future given their information about the present. Throughout and in the conclusion to this

chapter, as well as the next two chapters (4 and 5), we will continuously return to discussion of how reasonable our model is in the real-world and the predictions it makes.

We will outline two methods in which to achieve the model outlined above, at a surface level the first takes as input discrete visual states (into sensors) along paths, and forms an empirical entropy distribution from the counts of each state along paths. The second will construct a compression-based entropy by devising a suitable encoding for non-discretised visual states. We will see that both result in spontaneous collective motion. But the compression approach necessitates a notion of “eye-resolution”; an effective discretisation of the visual state. The effective discretisation is commensurate with the number of sensors in the discrete state approach.

### 3.1 Visual Input and the Cognitive Process

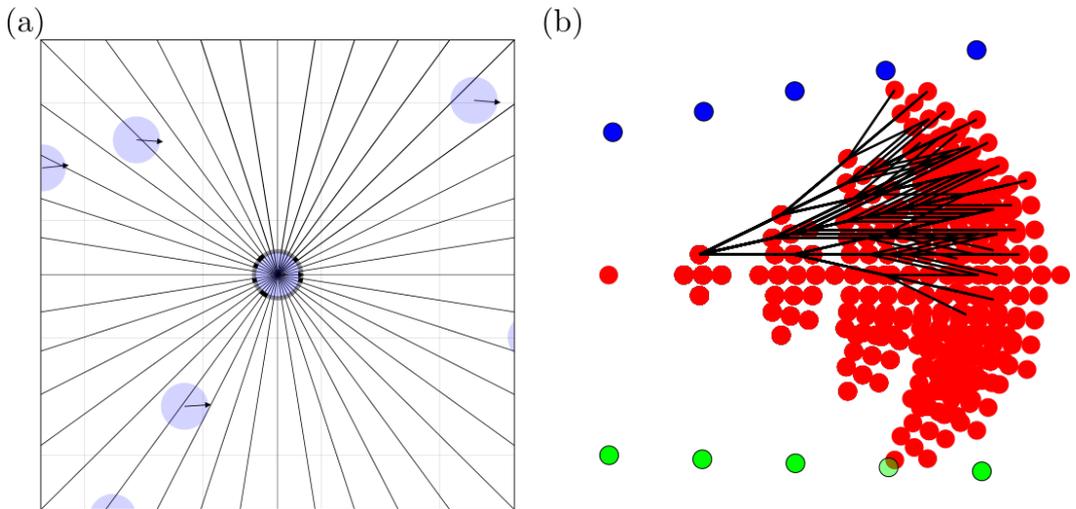


Figure 3.1: (a) A sketch of a visual state, the outline of the central agent indicates sensors which are considered “activated” or “filled” or not by an opaque black outline of translucent black outline respectively. Faint lines emanating from the central agent indicate pairs of angles defining the  $n_s = 40$  sensors. (b) A sketch of an agent’s future states tree (FST) (red). The blue and green agents represent two models of other agents. The bold black lines highlight the paths following on from one particular initial action. Anticipated collisions can occur on the FST, indicated by the Green agents fourth state (transparent). We choose to either continue enumerating paths and states past these “collisions” or prematurely stop enumerating. Both cases will be discussed, by default we perform the latter.

We imagine the  $i$ -th agent, as part of a larger group of  $N$  agents, at a given position  $\mathbf{x}_i^t \in \mathbb{R}^2$ , in unbounded space, and orientation  $\theta_i^t \in [0, 2\pi)$  for discrete time  $t$ . Each is modelled as a disc of radius 1 which defines our units of length. For time we will use a time step  $\Delta t$ , typically 1 unless otherwise stated, defining our time units. Each agent is able to compute a visual state defined by the visual projection method briefly outlined in the introduction, which we sketch in figure 3.1 (a). Mathematically this visual state can be expressed as an integral

$$\psi_i^n = \Theta \left[ \int_{\sigma_n} \Theta \left[ \sum_j I_{ij}(\chi) I'_{ij}(\chi) \right] d\chi - \frac{\pi}{n_s} \right]. \quad (3.1)$$

The terms in equation 3.1 have the following conceptual purposes (1) the integrand indicates 1 if for angle  $\chi \in \sigma_n = [2\pi(n-1)/n_s, 2\pi n/n_s]$ , at least one other agent  $j \neq i$  is intersected by a line of sight from  $i$  at angle  $\chi$ , where  $\Theta[x] = 1$  iff  $x > 0$  and 0 otherwise is the Heaviside step function (2) the integral counts the proportion of angles  $\chi \in \sigma_n$  which intersect some agent  $j \neq i$  (3) the outer Heaviside function registers a 1 for sensor  $n$  if at least half of all lines of sight from  $i$  at angles  $\chi \in \sigma_n$  intersect some agent  $j \neq i$ . That is if for a given sensor, defining a set of angles, if at least half of the lines of sight along those angles intersect at least one agent  $j \neq i$ , the sensor  $\psi_i^n$  is considered activated.

In detail, the notation  $\psi_i^n$  denotes a visual state's  $n$ -th component for an agent  $i$ , with  $n_s$  sensors. Each sensor defines a range of angles  $\chi \in \sigma_n$ , along which lines of sight  $\hat{\mathbf{n}}_i = R(\chi)\hat{\mathbf{v}}_i$  are projected. Here  $\hat{\cdot}$  denotes a unit vector and  $R(\chi)$  is the rotation matrix

$$R(\theta) = \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix}. \quad (3.2)$$

To determine if a given line of sight intersects at least one agent  $j \neq i$  two indicator functions are used

$$I_{ij}(\chi) = \Theta[1 - |\mathbf{x}_{ij} \times \hat{\mathbf{n}}_i(\chi)|], \quad (3.3)$$

$$I'_{ij}(\chi) = \Theta[\mathbf{x}_{ij} \cdot \hat{\mathbf{n}}_i(\chi)] \quad (3.4)$$

The first determines if the shortest distance between the ray  $\mathbf{n}_i(\chi)$  from  $i$  is less than the agent radius (here set to 1). The second restricts the first to apply only along the direction from  $i$  towards  $j$ . This is required since  $I_{ij}$  will be satisfied along either direction.

For each indicator function the relative position vectors  $\mathbf{x}_{ij} = \mathbf{x}_j - \mathbf{x}_i$  are used. Where time indices have been dropped for clarity. Finally the subtraction of  $\frac{\pi}{n_s}$  can be generalised to  $\frac{2\pi}{n_s}\theta_c$ , here  $\theta_c$  defines the proportion a sensor must be filled to register a 1 in the visual state. Unless otherwise stated  $\theta_c = 1/2$  throughout this thesis.

Core to our model is the computation of future states by each agent. To write this down we will adopt a convention that *realised* positions, orientations, and velocities of agent  $i$  at times  $t$  will be written as  $\mathbf{x}_i^t$ ,  $\theta_i^t$ , and  $\mathbf{v}_i^t$  respectively. For futures, we will adopt the notation  $\mathbf{q}_i^t$ ,  $\phi_i^t$ , and  $\mathbf{p}_i^t$  for the positions, orientations, and velocities of agents  $i$  at *future times*  $t$ . For speeds, we will use the shorthand  $\|\mathbf{v}_i^t\|_2 = v_i^t$  and  $\|\mathbf{p}_i^t\|_2 = p_i^t$ , and if we need to distinguish between actual time  $t$  and future time  $t'$  we will use primes. For realised actions we will use  $\alpha_i^t \in \mathcal{A}$  to denote agent  $i$ 's action at time  $t$  and for future actions, as yet un-realised, we will use  $\beta_i^t \in \mathcal{A}$ .

The agent has available a selection of actions with which it can modify its velocity. The set of actions  $\mathcal{A}$  indexed by  $\alpha \in \{1, 2, \dots, n_\alpha\}$ , an agent to update its velocity vector by the operator

$$A_{\alpha_i^t}[\mathbf{v}_i^t] = v_{\alpha_i^t} R(\theta_{\alpha_i^t}) \hat{\mathbf{v}}_i^t. \quad (3.5)$$

$$(3.6)$$

These agents select speeds  $v_{\alpha_i^t}$  and rotation angles  $\theta_{\alpha_i^t}$  (possibly zero). Our default action set is  $v_1 = v_4 = v_5 = v_0$ ,  $v_2 = v_0 + \Delta v$ ,  $v_3 = v_0 - \Delta v$  and rotations  $\theta_1 = \theta_2 = \theta_3 = 0$ ,  $\theta_4 = -\Delta\theta$  and  $\theta_5 = \Delta\theta$ . This is written as the set  $\mathcal{A}$ . This gives agents access to three speeds and a positive or negative reorientation, which are measured in our dimensionless length and time units defined by particle radius  $r = 1$  and time-step  $\Delta t = 1$ . Each agent then follows the equations of motion

$$\mathbf{x}_i^{t+1} = \mathbf{x}_i^t + \mathbf{v}_i^{t+1} \Delta t, \quad (3.7)$$

$$\mathbf{v}_i^{t+1} = A_{\alpha_i^t}[\mathbf{v}_i^t] \quad (3.8)$$

The core of our model then, is how the actions  $A_{\alpha_i^t}$  are selected. We first build a Future States Tree (FST) from the actions  $\mathcal{A}$ . *A priori* agents may take all possible  $\tau$  length combinations of  $\beta \in \mathcal{A}$ ; paths in the FST. Figure 3.1 (b) shows a sketch of an FST with  $\tau = 4$  futures. One path of length  $\tau$  can be written explicitly as the sum of products

$$\mathbf{q}_i^{t+\tau'} = \mathbf{q}_i^t + \sum_{t'=t}^{t-1+\tau'} p_{\beta_i^{t'}} \prod_{t''=t}^{t'} R(\theta_{\beta_i^{t''}}) \mathbf{p}_i^t. \quad (3.9)$$

This represents summing the  $\tau$  selected velocity vectors resulting from action choices  $\{\beta_i^{t'=t}, \beta_i^{t'=t+1}, \dots, \beta_i^{t'=t+\tau}\}$  starting from the present time  $t' = t$ . We can use equation 3.9 to determine future positions at leaf nodes in the FST (at time horizon  $\tau$ ) when  $\tau' = \tau$ , and all internal nodes  $\mathbf{q}_i^{t+\tau'}$  where  $1 \leq \tau' \leq \tau$ .

In each agent's cognitive process we apply equation 3.9 to the position of other agents  $j \neq i$  and define some model as to how agents select actions. That is each agent  $i$  imagines the future positions  $\mathbf{q}_{ij}^t$  of the other agents, where we write  $ij$  subscripts to indicate  $i$ 's prediction of  $j$ 's future state. In principle, we can define any model for how actions  $\beta_{ij}^t$  are chosen for agent  $j$  in agent  $i$ 's cognitive process, by default however we will use a ballistic assumption. In our notation this is  $\beta_{ij}^t = 1$ . The FST then consists of  $|\mathcal{A}|$  branches defined by the initial actions  $\beta_i^{t'=t}$  with paths given by equation 3.9.

In order to make a decision about which action to take in the present time  $t$ , an agent can group the future paths by their initial action  $\beta_i^{t'=t} \in \mathcal{A}$ . In this way an agent can generate, via its cognitive process, a set of  $(\tau - 1)|\mathcal{A}|^{\tau-1}$  visual states (which may not be unique), denoted as the sets  $\mathbf{B}_{\beta_i^t}^t = \{\psi_1, \psi_2, \dots, \psi_{(\tau-1)|\mathcal{A}|^{\tau-1}}\}$ , for each initial action  $\beta_i^t$  defining a branch of the FST. This set contains all the visual states generated from the  $|\mathcal{A}|^{\tau-1}$  paths after having taken action  $\beta_i$  at time  $t' = t$ . We refer to the visual states as  $\psi_k$  rather than explicitly as the vectors  $\psi_i$  defined by equation 3.1 calculated with different sets of relative position vectors  $\{\mathbf{q}_i^{t'} - \mathbf{q}_j^{t'}\}$ .

We wish to develop a notion of path entropy based upon the visual states an agent predicts during its cognitive process,  $\mathbf{B}_{\beta_i^t}^t$ , for each action  $\beta_i^t$ , and use this measure to judge which action to take i.e to choose  $\alpha_i^t$ . There are two ways to do this in the current setting which will turn out, in a sense, to be equivalent. It is noteworthy that both lead to ordered and cohesive motion. First we can use the discretised visual states  $\psi$ , binary vectors of length  $2^{n_s}$ , where  $n_s \in \mathbb{N}$  defines the visual sensors. Then an entropy for a branch on the FST is formed by the empirical distribution of these predicted possible discrete states an agent calculates. Secondly in an attempt to avoid discretisation by integral equation 3.1, we define a compression-based entropy for an arbitrary visual projection function  $\psi$ , without discretisation into sensors, and use this to find an entropy along each branch. This will be done in sections 3.1.1 and 3.1.2 respectively. The motivation for attempting this is to understand the importance of the number of sensors,  $n_s$ ,

and in particular why 40 sensors appears optimal. Biologically speaking  $n_s$  is far smaller than the number of cells in the retina of higher animals or insects, e.g. in compound insects eyes like *Drosophila*'s there are 700 units making up the eye [103]. We have previously eluded that the sensors represent a step up in the cognitive process, and not simply a count of retina cells. Our hypothesis is that by forming a compression entropy,  $n_s$  can be avoided, leaving only discretisation by the computational accuracy of decimal values (e.g. floating point arithmetic).

Under either of these entropies the algorithm is the same. Each agent individually builds a prediction of all possible futures states to a finite time horizon  $\tau$ , calculates an entropy for each possible action and chooses the action in the present with the highest entropy. Algorithm 1 details the agent action selection by the path entropy method. The algorithm makes use of a function to collect visual states along a particular branch of the FST. This is detailed as a recursive algorithm in algorithm 2. The tree search enumerates all states along a branch of the FST. At any given node in the FST algorithm 2 collects the visual state associated with the appropriate positions given by equation 3.9. In the case of internal nodes, where multiple paths exist deeper into the tree, this visual state is added multiple times. This reflects the fact the state is present in multiple future paths. Algorithm 3 details calculation of a visual state. The algorithm produces states commensurate with equation 3.1. The algorithm avoids the costly integral by instead determining visual projection intervals for each  $j \neq i$ . Overlapping intervals are removed as they are redundant. Finally, the coverage of each sensor  $n \in [1, n_s]$  by the non-overlapping projections intervals defines the sensor array  $\psi_i$ .

Both (1) and (2) can lead to flocking behaviour of similar character which

we will discuss in detail over the following sections.

---

**Algorithm 1: FSM Decision**

---

**Result:** Action selection

```

1 Function decision( $i, X_t, \mathcal{A}$ ) is
2    $\psi_0 \leftarrow \text{visualState}(i, X_t)$ ;
   /* Initialise future ballistic positions of  $j \neq i$ , and action positions
   for  $i$  */
3   FST  $\leftarrow \text{futureStatesTree}(i, X_t)$ ;
4   for  $\alpha \in \mathcal{A}$  do
5     distributions[ $\alpha$ ]  $\leftarrow [\psi_0]$ ;
6      $u \leftarrow \text{nextNode}(\text{root}, \text{FST})$ ;
7     distributions[ $\alpha$ ].append(statesOnBranch( $u, \text{FST}$ ));
   /* Unique states found on the FST */
8      $\Psi[\alpha] \leftarrow \text{uniqueValues}(\text{distributions}[\alpha])$ ;
9     norm  $\leftarrow 0$ ;
   /* Compute the count distribution for these states */
10    for  $\psi \in \Psi[\alpha]$  do
11       $c \leftarrow \text{counts}(\psi, \Psi[\alpha])$ ;
12       $P[\alpha][\psi] \leftarrow c$ ;
13      norm  $\leftarrow \text{norm} + c$ ;
14    end
15    normalise( $P[\alpha], \text{norm}$ );
16     $S[\alpha] \leftarrow \text{entropy}(P[\alpha])$ ;
17  end
18  return  $\text{argmax}_{\alpha} S$ ;
19 end

```

---

**Algorithm 2: FSM State enumeration**

---

**Result:** Visual states along one branch of the FST

```

1 Function statesOnBranch( $\text{Node}, \text{FST}$ ) is
   /* Empty state array */
2   states  $\leftarrow []$ ;
3    $t' \leftarrow \text{Node.depth}$ ;
4    $\tau \leftarrow \text{FST}.\tau$ ;
5   if  $\neg \text{isCollided}(\text{Node}, q[t'])$  then
   /* at a non-collided state */
6      $p \leftarrow \text{futurePaths}(\text{Node}, \text{FST})$ ;
7     states.append(visualState( $i, q[t'], p$ ));
8     if  $\neg \text{isleaf}(\text{Node})$ ;
9     then
10      for  $\alpha \in \mathcal{A}$  do
11         $u \leftarrow \text{nextNode}(\text{Node}, \alpha, \text{FST})$ ;
12        states.append(statesOnBranch( $u, \text{FST}$ ));
13      end
14   return states;
15 end

```

---

---

**Algorithm 3: FSM Visual State Vector**


---

**Result:** Visual State for agent  $i$  from agent positions  $X$

```

1 Function  $visualState(i, X)$  is
  | /* Obtain visual projections for each  $j \neq i$  */
2 for  $j \in [1, N] \setminus i$  do
  | | /* 0,  $2\pi$  boundary must be handled with care */
  | |  $\theta[j][0], \theta[j][1] \leftarrow project(i, j, X);$ 
3 end
  | /* Merge overlapping intervals, to avoid multiple counting sensor
  | | overlaps */
4  $mergeOverlaps(\theta);$ 
  | /* Compute sensor coverage */
5 for  $s \in [0, n_s - 1]$  do
  | |  $\psi[s] \leftarrow 0;$ 
  | |  $start \leftarrow s \cdot \frac{2\pi}{n_s};$ 
  | |  $end \leftarrow (s + 1) \cdot \frac{2\pi}{n_s};$ 
  | | /* Compute coverage of sensor  $s$  by merged intervals */
  | |  $c \leftarrow overlap(start, end, \theta);$ 
  | | if  $c > \theta_c$  then
  | | |  $\psi[s] \leftarrow 1;$ 
  | | end
6 end
7 return  $\psi;$ 
8 end

```

---

### 3.1.1 Discretised Sensor Entropy

For an action  $\beta_i^t$  we consider the set  $\mathbf{B}_{\beta_i^t}^t \in \Psi(n_s)^{|\mathbf{B}_{\beta_i^t}^t|}$  by equation 3.1, as a distribution of states generated by action  $\beta_i^t$ . Where the set  $\Psi(n_s)$  contains all possible discretised visual states for  $n_s$  sensors. That is  $\Psi(n_s)$  is precisely the list of all binary numbers up to  $2^{n_s} - 1$  with their digits interpreted as the  $n_s$  sensors states<sup>1</sup>. This provides for an efficient way to implement the visual state computationally, by manipulating a single  $n_s$ -bit integer value for each visual state.

We then form a count distribution by the function  $n_{\beta_i^t}^t(\boldsymbol{\psi})$  which counts how many times an arbitrary visual state  $\boldsymbol{\psi}$  appears in  $\mathbf{B}_{\beta_i^t}^t$ . Our entropy is then simply calculated as a sample entropy of the empirical distribution given by

$$H_{\beta_i^t} = - \sum_{\boldsymbol{\psi} \in \mathbf{B}_{\beta_i^t}^t} P_{\beta_i^t}^t(\boldsymbol{\psi}) \log_b P_{\beta_i^t}^t(\boldsymbol{\psi}), \quad (3.10)$$

---

<sup>1</sup>i.e  $\Psi(1) = \{1, 0\}$ ,  $\Psi(2) = \{00, 11, 10, 01\}$ , and  $\Psi(3) = \{000, 111, 100, 010, 001, 110, 011, 101\}, \dots$

where the probability is

$$P_{\beta_i^t}^t(\boldsymbol{\psi}) = \frac{n_{\beta_i^t}^t(\boldsymbol{\psi})}{\sum_{\boldsymbol{\psi} \in \mathbf{B}_{\beta_i^t}^t} n_{\beta_i^t}^t(\boldsymbol{\psi})}. \quad (3.11)$$

We find that using this entropy measure the agents generally attain cohesive and highly order collective motion. We do however see some fragmentation in large groups at finite  $\tau$ .

Using this definition of path entropy we may also calculate the maximum entropy possible for a given action set size  $|\mathcal{A}|$  and time horizon  $\tau$  which is independent of  $N$ . In appendix B we derive this maximum entropy by assuming all visual states on the FST are unique. The value we obtain can be used To normalise entropies for finite tau. Ultimately though we find in the limit  $\tau \rightarrow \infty$  the maximum entropy is undefined. Which can also be seen from equation 3.11. Since if the time horizon is infinite, so is both the number of unique visual states, and count of each. That is we obtain a probability  $P_{\beta_i^t}^t(\boldsymbol{\psi}) \rightarrow \infty/\infty$  when  $\tau \rightarrow \infty$ . Biologically  $\tau \rightarrow \infty$  would require an organism capable of projecting future movements of a group infinitely far into the future. Therefore, practically speaking the case of an infinite time horizon should be infeasible in any case. Similarly, for  $|\mathcal{A}| \rightarrow \infty$  a similar problem will emerge. Although in this case the maximum entropy tends to  $\infty$ .

### 3.1.2 Compression Entropy

To develop the compression based approach we first outline a method of compression of the visual state, using a run-length encoding in section 3.1.2. We will then use the length of this compressed form to estimate the entropy of that visual state via Shannon's source coding theorem [104]. Finally, we will assume selecting the branch  $\beta_i^t$ , which maximises the sum of these entropies over the visual states in  $\mathbf{B}_{\beta_i^t}^t$  (all paths subsequent to it), will be equivalent to maximising path entropy. To apply the compression we will re-define a non-discretised visual state as a function  $\psi_i : [0, 2\pi] \mapsto \{0, 1\}$ , given a set of relative position vectors  $\{\mathbf{q}_i^t - \mathbf{q}_j^t\}$ .

Firstly we will recall Shannon's source coding theorem. Shannon proves his source coding theorem that implies a message  $X = \{x_1, x_2, \dots, x_L\}$  where each symbol  $x_i$  is independent and identically distributed can be compressed into no less than  $L \cdot H(X)$  bits without risk of information loss, where the Shannon entropy

$H(X)$  is given by

$$H(X) = - \sum_i P(x_i) \log_2 P(x_i), \quad (3.12)$$

for some probability distribution  $P(x_i)$  of the symbols.

Stated another way, if we take a message and compress it losslessly into  $b$  bits we should find that  $b$  is proportional to the entropy of the message, and at minimum  $L \cdot H(X)$ . In particular the entropy, empirical or analytic, itself gives rise to Huffman codes [105] and Shannon (Shannon-Fano) codes. These use the entropy given by the empirical/analytic probability distribution of the alphabet making up a message to define a minimal lossless encoding. For example taking a corpus of English text, we compute the probability of any given letter (like  $P(\text{“a”})$ ,  $P(\text{“b”}) \dots$ ) and calculate the information content  $-\log_2 P(x)$  for each symbol  $x \in \{\text{“a”}, \text{“b”}, \text{“c”}, \dots, \text{“z”}\}$ . Symbols with higher information content are prioritised as code words in the encoding. E.g “e” is the most common letter, in the English language, so will have the shortest encoding.

### Run-length Encoding

To estimate entropy of a visual state we will define a lossless compression scheme of a non-discretised visual state (to be defined later). To do this we will first outline one such scheme, the run-length encoding.

Entropy based codes require knowledge of the underlying probability distribution  $P(x_i)$  to find an optimal encoding. However, some lossless encoding strategies do exist that do not require knowledge of  $P(x_i)$  empirical or otherwise. One example is the run-length encoding (RLE) which has historically been widely used to compress data, especially image data [106]. The intuitive idea is that if a stream of data includes long runs of repetition, e.g. 00000011111 or 111111001111, the data can be compressed by calculating run-lengths of repeated data. The “runs“ may then be replaced by a run-length at a corresponding bit. The utility of such a code for our purposes is to use it to compress a non-discretised visual state and use the length of the compressed form to approximate the entropy. We will outline this process by first outlining the RLE in this section.

To illustrate the RLE we compress one example sequence of binary values. Consider the sequence 00111110000001111111000000. We can compress this sequence (using integers as code-words) by finding the run lengths of each bit (0, 1) to produce the sequence 0, 2, 1, 5, 0, 6, 1, 7, 0, 6, that is two zeros, five ones, six zeros, etc. Further we can recognise that we only need to know the parity of the sequence,

i.e. if we know the first run is of 0's we can include that information by writing the encoding as 0, 2, 5, 6, 7, 6 and noting after each run-length integer we flip the bit outputted by the decoder, starting with that bit as the first term in the sequence. To return to binary we can realise each integer has a binary form and write our compressed data as 0, 010, 101, 110, 111, 110 where we implement our decoder to take the first bit (parity of the data) and then parse the code in chunks of three bits, the run lengths. Generalising this, our data will have some maximum run length, and we represent each run length as a bit string of the lengths binary representation with zeros prepended if necessary. So if our largest run length is 30 (11110 in binary), and one run length is 3 we write three as 00011 instead of 11.

In the example above we started with a sequence of 26 bits and compressed it to a sequence of 16 bits, a compression ratio of  $16/26 \approx 0.6$ . For bit strings of e.g. 26 1's or 0's we would find compressed sequences 111010 and 011010 respectively, which both have compression ratios of  $6/26 \approx 0.357$ , whereas a string of alternating 1's and 0's would have the compressed form 110111011101...11, i.e a compression ratio of  $> 1$  indicating it is better to not compress under this scheme. In these three cases the empirical entropies are  $-\log_2(12/26) - \log_2(14/26) \approx 0.996$ , (all ones)  $-\log_2(1) - 0 \log_2(0) = 0$ , and (alternating)  $-\log_2(1/2) = 1$  respectively.

In general for any binary sequence of data the encoding computed in this way will yield a sequence of length  $1 + b \cdot k$  where 1 is for the parity,  $b \in \{0, 1, 2, \dots\}$  is the total number of boundaries in the sequence (where the bit flips) and  $k$  is the total length of the binary representation of the largest run-length. This method is somewhat naïve. By carefully choosing our encoding of the integer run-lengths we could shorten the code length further by carefully choosing code words. One example is Elias- $\gamma$  coding [107] used when the largest run-length is unknown, in this case the run-length 1 is encoded as 1 and 2 as 010, however 7 would be 00111, a little longer. In general choosing code words carefully often leads to some encodings being more optimal and some less and one must be careful to keep codes uniquely decodable e.g  $(1 \mapsto 10, 2 \mapsto 00, 3 \mapsto 11, 4 \mapsto 110, \text{ for symbols } 1, 2, 3, 4)$ , and even better instantaneous  $(1 \mapsto 0, 2 \mapsto 10, 3 \mapsto 110, 4 \mapsto 111, \text{ for symbols } 1, 2, 3, 4)$  where when a decoder reads 0 it knows the decoding must be 1 since 0 is not a prefix to any other code [108]. For simplicity, we will take the 0-prepended binary representation for code words since in any case the compressed size will be proportional to the number of run-lengths, and therefore the entropy as required.

### Run length Encoding of a Visual State

Having recalled the idea of lossless compression and the run-length encoding as particular examples, we will now apply this to the visual state of an agent in the next section. First we will make solid the idea of a non-discretised visual state  $\psi$ . To do this we simply take the integrand in equation 3.1 interpreted as a function  $\psi_i(\chi) : [0, 2\pi] \mapsto \{0, 1\}$ . That is

$$\psi_i(\chi) = \Theta \left[ \sum_j I_{ij}(\chi) I'_{ij}(\chi) \right]. \quad (3.13)$$

This function, given some relative position vectors  $\{\mathbf{q}_j^t - \mathbf{q}_i^t\}$ , will be 1 where at least one agent  $j$  is intersected by some line of sight directed at angle  $\chi$  from  $i$ . And will be zero otherwise.

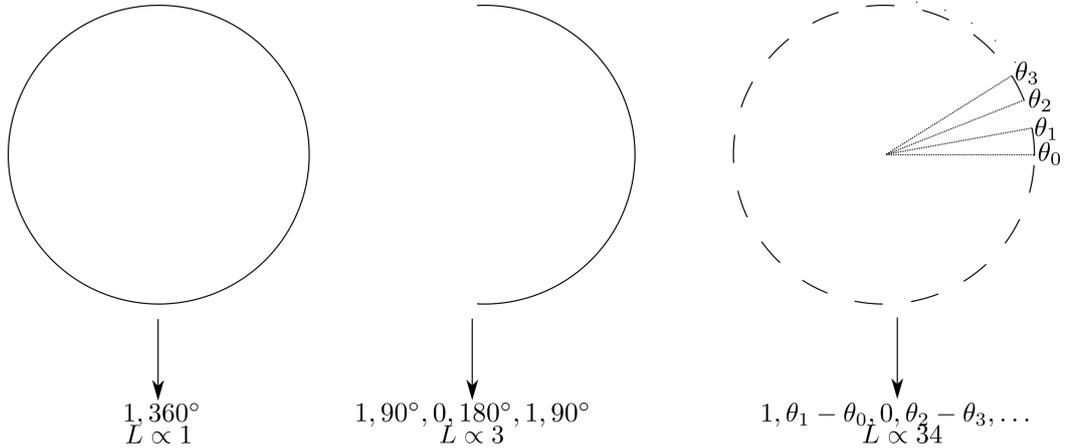


Figure 3.2: Circles represent visual states over  $[0, 360^\circ)$ , black regions are 1 and white/blank regions are 0. Here  $1, 90^\circ, 0, 180^\circ, 1, 90^\circ$  means a run of ones anti-clockwise from angle 0 to  $90^\circ$ , a run of zeros for  $180^\circ$  and finally another run of  $90^\circ$  back to the 0  $360^\circ$  boundary. The length of the run-length encoding is proportional to the number of boundaries on the visual state, if the  $0, 360^\circ$  boundary is always considered a boundary, and decimal run-lengths are approximated to a constant number of digits. Note the run-lengths in each encoding contain a list of bits and a corresponding angular length (decimal value) where this length must also be converted to a sequence of bits, implying a discretisation. I.e on a computer these decimal values are approximated as 32 or 64 bit floats, even analytically this process necessarily implies a discretisation (like the sensor model) at some level even if it is much finer than e.g 40 sensors, otherwise decimal and transcendental numbers will yield extremely large even infinite codes. E.g even picking a code word, say 00, to represent  $\pi$  requires storing the value of  $\pi$  once for later decoding which we can only approximate in reality.

Therefore, if we directly interpret the visual state  $\psi_i : [0, 2\pi) \mapsto \{0, 1\}$  as a source can we determine its entropy? Unfortunately we do not know the underlying distribution that generates any particular  $\psi_i$  so we cannot directly find the entropy analytically. However, we could try to compress  $\psi_i$ , losslessly, and use this as an approximate entropy. Consider any  $\psi_i$ , we know that it maps any given  $\chi$  to a bit ( $\{0, 1\}$ ), further  $\psi_i$  will be composed of different contiguous regions of 1's or 0's even if such a region is infinitely small. With this information we can trivially apply a run-length-encoding to a given  $\psi_i$ , up to the encoding of decimal (even transcendental) values, which as we will see require some approximation.

To do this start at  $\chi_0 = 0$  and note the parity  $\psi_i(\chi_0)$ , then find the next  $\chi_1 \in (0, 2\pi)$  such that  $\psi(\chi_1) \neq \psi(\chi_0)$  and note this value. Repeat this process until the limit of  $\chi_b \rightarrow 2\pi$  for some  $b \in \{1, 2, \dots\}$ . This scheme will convert the information in  $\psi$  down to a list of  $b$  decimal values (run lengths  $\chi_{i+1} - \chi_i$ , for  $0 \leq i \leq b$ ) and a parity bit. i.e a visual state  $\psi(\chi) = 1$  will be compressed to  $1, 2\pi$  (or 1 by convention) and for the visual state  $\psi(\chi) = \Theta[\cos(k\chi)], k \in \mathbb{N}$  where  $\Theta[x]$  is the Heaviside step function we would find a list of  $2k$  angular runs (of equal size) and the parity bit. This scheme is presented visually in figure 3.2.

Immediately we have two issues (1) the values of  $\chi_{i+1} - \chi_i$  are real numbers. This means to actually complete this compression (compress the run lengths like in the binary sequence example in the last section) we need to approximate the run-lengths, in principle to arbitrary numerical precision. For finite numerical approximation this necessarily implies a number of “sensors” i.e. taking decimal values at two decimal places gives a minimal interval of 0.01 radians implying  $\lceil \frac{2\pi}{0.01} \rceil = 629$  “sensors”. Here a higher number of “sensors” resolves the boundaries of the run  $\chi_i$  to higher accuracy, and does not present problems for calculating entropy, unlike  $n_s$  in equation 3.10 where  $n_s \rightarrow \infty$  actually leads to maximal entropy with a finite FST. Given this fact we would typically choose a very precise approximation, e.g. 32- or 64-bit floats that imply  $\approx 10^8$  or  $\approx 10^{16}$  [109] “sensors” (via the machine epsilon)

(2) As raised by the cosine example, if the angular sizes are in some regular pattern, we could compress even further due to the repetition of interval sizes by choosing an appropriate encoding of them. I.e. if  $\chi_{i+1} - \chi_i = \text{constant}$  for all intervals we could truncate the constant and define a code word to represent it,  $c$ . Then we could write a decoder to refer to and replace the code word  $c$  with the actual run-length. This would save repeating the same data many times. For a small number of repeating patterns we could do something similar for each. Applying this to the cosine example we could achieve a compression of length  $1 + b + k$  ( $+k$  since we do need to store the constant run-length at least once for decoding) as apposed to  $1 + b \cdot k$  where

$k$  is the length of the decimal approximation of the constant decimal run-length in bits. For simplicity, we will continue by taking just the RLE as outlined above without applying additional optimisations such as finding repeated run-lengths and optimising their code-words. One justification for this choice is that it otherwise implies a different decoding scheme on a per-example basis.

In general, the length of the run-length encoding of a visual state performed as above is exactly  $1 + bk$  where  $b$  is the number of boundaries in the visual state and  $k$  is the length of the encoding of the  $b$  (approximated) decimal run-lengths (truncated to  $k$ -bit representation). For example, we could use 64-bit floating point numbers to approximate interval sizes, giving  $k = 64$  and so  $\approx 10^{16}$  implied “sensors”. In any case the length of this run-length encoding of a visual state,  $L(\psi_i)$ , is directly proportional to the number of boundaries in the visual state, neglecting the constant 1. This is particularly interesting given the use of the boundary angles, the  $\chi_0, \chi_1, \dots, \chi_n$  used above, by Pearce *et al* [73] for their model of a bird flock.

To directly link  $L(\psi_i)$  to the Shannon entropy of a given visual state  $\psi_i$  would imply that this simple encoding is equivalent to entropy codes, such as Shannon codes or Huffman codes [105, 104]. Which is likely not the case. We can assume, by Shannon’s source coding theorem, that it is at least proportional i.e  $L(\psi_i) = cH(\psi_i)$  to the true entropy  $H$ , and use  $L(\psi_i)$  as an approximation where a larger  $L(\psi_i)$  means a larger entropy. Since we are ultimately going to be comparing (and maximising) entropies, a scalar proportionality constant will not affect our ability to judge agent actions by approximating  $H(\psi_i) \approx L(\psi_i)$ , i.e maximising  $L$  will maximise  $H$ .

### Path Entropy and Eye Resolution

The main issue with the compression approach is that it defines an entropy for a given visual state, and not for a distribution of states along a path. This would not be a problem for calculating path entropies if each  $\psi$  were independent and identically distributed, where we could sum the entropies, this is likely not the case since a state along path is in effect conditioned on the previous states within the same path, and all are conditioned on the root visual state. Mathematically, if we examine one branch of the FST, for an agent  $i$  at time  $t$ , by enumerating a set of all visual states on all paths along it,  $\mathbf{B}_{\beta_i}^t$ , we can write the entropy using the inequality

[108]

$$H_{\beta_i^t}^{CE} = H(\psi_1, \psi_2, \dots, \psi_{|\mathbf{B}_{\beta_i^t}|}) \leq \frac{1}{|\mathbf{B}_{\beta_i^t}|} \sum_{\psi \in \mathbf{B}_{\beta_i^t}} H(\psi) \quad (3.14)$$

$$= \frac{1}{|\mathbf{B}_{\beta_i^t}|} \sum_{\psi \in \mathbf{B}_{\beta_i^t}} L(\psi). \quad (3.15)$$

We can see the issue of converting entropies of states to an entropy of a path via the inequality. We hypothesise thought that maximising equation 3.15 will in turn maximise  $H_{\beta_i^t}^{CE}$  through the inequality.

There is one caveat to the hypothesis above, *a priori*; the maximum entropy of  $H(\psi)$ , roughly equivalent to the maximum number of boundaries, can be attained when all visual projections of the other  $N - 1$  agents are non-overlapping, that is when there are  $2(N - 1)$  boundaries on the visual state. This case can, and will occur, in groups that are completely non-cohesive even at an arbitrary distance from one another. This also presents a problem for the minimum timescale for visual states to change. That is for agents at arbitrary distance small changes in the configuration  $\mathbf{X}_i^t$  will largely not effect  $i$ 's visual state. Taking  $\tau \rightarrow \infty$  in theory may resolve this, but this is not computationally feasible.

Therefore, we secondly hypothesise we can resolve this by introducing a visual resolution parameter,  $s \in (0, 2\pi]$ , that defines how large a visual projection must be (in angular size) to be visible to an agent. This in effect implies a length-scale at which two agents are invisible to one another  $\arcsin \frac{a}{d} > s$  for agent radius  $a$  and inter-agent distance  $d$ . Further, given a marginally opaque (circular in 2D) flock at distance  $d$  and radial size  $r$  from an agent the flock as a whole becomes invisible at the length scale  $\arcsin \frac{r}{d} > s$ .

This is the point of “equivalence”, the choice of  $s$  does present interesting phases of behaviour as we will cover subsequently, but as it will turn out a choice of  $s \sim \frac{2\pi}{80}$  commensurate with  $n_s = 40$  in the discrete state entropy model, produces groups of high order  $\phi \sim 0.98$ . Even across the spectrum of behaviour governed by  $s$ , each implies an effective sensor discretisation. In particular a discretisation coarser than that implied by the approximation of decimal numbers, i.e a resolution of  $2\pi/80$  compared with  $\approx 10^{-16}$  for a 64 bit floating point approximation.

## 3.2 Both Path Entropy Methods are Phenomenologically Equivalent

We have now defined two methods of calculating an entropy of future paths, in this section we will show that both lead to spontaneous flocking behaviour. We will also compare the specific idiosyncrasies of using  $H_{B^t}$  (equation 3.10), which we will refer to as the State Count Entropy (SCE), and  $H_{\beta_i^t}^{CE}$  (equation 3.15) which we will call the Compression Entropy (CE), and what these properties mean physically.

In order to make our comparisons quantifiable we will focus on notions of order, density, opacity, morphology, “clustering”, and in some cases vorticity. Order for our purposes will be defined similarly to Vicsek *et al.* [4] namely

$$\phi = \left\langle \frac{1}{N} \left| \sum_{i=1}^N \hat{\mathbf{n}}_i^t \right| \right\rangle_t. \quad (3.16)$$

The global order is the normalised sum of director vectors  $\hat{\mathbf{n}}_i^t = [\cos(\theta_i^t), \sin(\theta_i^t)]^T$ , where  $\theta_i^t$  is the  $i$ -th agents orientation at time  $t$ . The order, density and opacity of flocks will also be of interest in quantifying behaviour, the density we will typically measure in the sense of the convex hull area fraction of area  $A^t$ . Density  $\rho^t$  is defined by

$$\rho = \frac{N\pi r^2}{\langle A^t \rangle}. \quad (3.17)$$

The density measures the coverage of the convex hull of agent positions by the agents, given that our model includes agents with physical size. We will also measure the opacity. For our model opacity is defined via the proportion of sensors activated in a visual state, averaged over realised visual states across agents and time. Or the proportion of 1’s and 0’s in the CE model. We identify this quantity as

$$\Theta = \left\langle \frac{1}{n_s} \sum_{s=1}^{n_s} \psi^s \right\rangle. \quad (3.18)$$

Marginal opacity of the field of view,  $\Theta \sim 0.5$ , is defined in analogy to opacity in bird flocks. The meaning being that a group of birds assembles in such a way to allow an observer to see substantial areas of clear sky and hence not other agents, measurements have indicated opacity of 0.25 to 0.6 [73].

To understand the behaviours we see in our model and their effects on ubiq-

uitous measurements in the collective motion literature (order, density, opacity) we will also quantify morphology, vorticity, and clustering. Morphology we define from the positions of agents at a given time by use of the Eigenvalues of the position covariance matrix,

$$\mathbf{K}_{xy} = \langle (x_i^t - \mu_x^t)(y_i^t - \mu_y^t) \rangle_{i,t}. \quad (3.19)$$

Where  $x_i^t$  and  $y_i^t$  denote the spatial positions in the  $x$  and  $y$  axis for agent  $i$  at time  $t$ , and  $\mu_x^t = \langle x_i^t \rangle_i$  and  $\mu_y^t = \langle y_i^t \rangle_i$  denote the average projected over each dimension. The Eigenvalues  $\lambda_1, \lambda_2$  of the matrix  $\mathbf{K}_{xy}$  are used to define the morphology

$$\mathcal{M}^t = \frac{\lambda_2}{\lambda_1}. \quad (3.20)$$

The indices 1 and 2 are chosen so that  $\lambda_1 \leq \lambda_2$ . For example a group with a circular shape will have  $\lambda_1 \sim \lambda_2$  and so  $\mathcal{M} \sim 1$ , a group stretched along one particular direction (a line at the extreme) will have  $\mathcal{M} \gg 1$ . To quantify vorticity we use the average cross-product of an agent's position in the centre of mass frame,  $\mathbf{x}_i^t - \langle \mathbf{x}_j^t \rangle_j$  and its current director  $\hat{\mathbf{n}}_i^t$ . The cross product in two-dimensions can be defined as the value  $\mathbf{a} \times \mathbf{b} = a_1 b_2 - a_2 b_1$  for vectors  $\mathbf{a} = [a_1, a_2]^T$  and  $\mathbf{b} = [b_1, b_2]^T$ . The metric captures the degree to which agents are circulating around some central point. The vorticity metric is

$$\nu = \left\langle \frac{1}{N} \sum_{i=1}^N \frac{(\mathbf{x}_i^t - \langle \mathbf{x}_j^t \rangle_j) \times \hat{\mathbf{n}}_i^t}{\|\mathbf{x}_i^t - \langle \mathbf{x}_j^t \rangle_j\|_2} \right\rangle. \quad (3.21)$$

Finally, and perhaps most importantly, we seek to measure a sense of clustering. Why clustering of agents is present in the model at all is first worth considering. Here agents choose actions associated with maximal future path entropy of visual states. The variation among visual states comes from different relative spatial positions of agents within the group, and a visualising agent. Given circular agents of radius  $r$ , and a sensor coverage threshold  $\theta_c$ , a length scale can be derived. Namely, the distance at which one agent may “sense” another. That is when a second agent is able to fill a sensor. The angular size of a circle at distance  $d$  is  $2 \arcsin r/2d$ . This means for an agent to be visible the distance must satisfy  $\theta_c \leq 2 \arcsin r/d$ . So the critical distance (using the small angle approximation  $\sin x \sim x, x \ll 1$ ) is  $d_{\text{vis}} = \frac{2r}{\theta_c}$ . For  $\theta_c = \pi/40$  as commonly used in this thesis,  $d_{\text{vis}} \approx 25$  radial units (in this thesis  $r = 1$ ). The same argument can be used for an agent's ability to “see” a group,

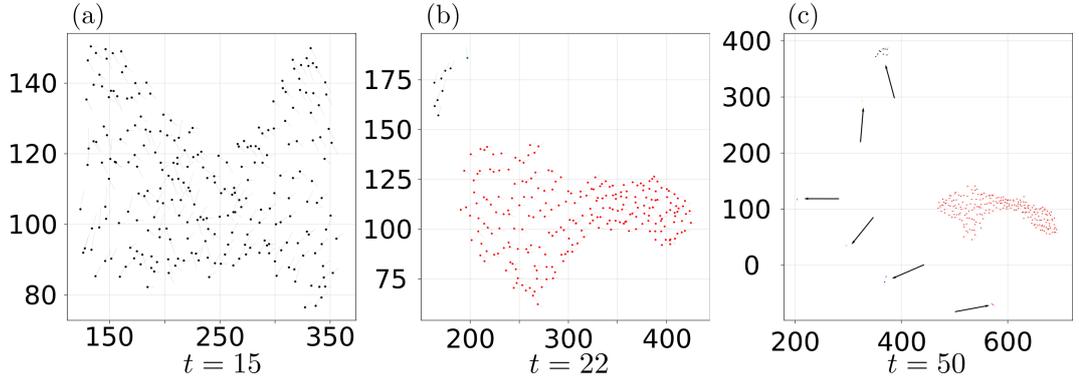


Figure 3.3: Example clustering of a group (in open space) using the method outlined below. (a) the group is cohesive (b) a smaller group splits from the main group and is identified as a second cluster (c) multiple groups have split from the main cluster, some are single-agent clusters. We use DBScan utilising “density” in space and orientation-space to detect order and cohesive groups as clusters. Lone agents are identified as clusters for accurate fragmentation statistics.

taking instead  $d_{\text{vis}} = \frac{2N}{\theta_c}$  for a group at marginal opacity with group radius scaling as  $N$  [73]. Agents far beyond this distance are unlikely to cluster with other except by random chance (if random decision are made with degenerate maxima). This is a simple consequence of an agent “seeing” little and therefore receiving no activated sensor input. For agents within this distance of some other (or others) a greater variety of visual states may be experienced by continuing to be close. This is the mechanism for clustering. Indeed, we hypothesised (and will investigate in section 3.2.2) for the case of compression entropy, where no such length scale exists *a priori* due to the lack of sensors, we must re-impose this length scale to see clustering. Otherwise, maximum entropy is trivially attained with other agents set at arbitrary distance.

In order to detect fragmentations in our data, i.e. when one agent or a number of agents leave the main group, we develop an algorithmic definition of a cluster. The goal of the method is to identify ordered and/or cohesive groups as clusters, e.g. figure 3.3 (a). For example a group of  $N$  agents with high order,  $\phi \sim 0.99$ , and high cohesion (a number density say of  $\sim 1/N$ , indicating a flock at marginal opacity of radial size  $N$ ) should be recognised as a single cluster. If however one agent leaves the group we should find two clusters, or if the group splits into two highly ordered and cohesive flocks moving apart we should also have two clusters, e.g. figure 3.3 (b). We also seek to allow for clusters of diverse morphologies, such as stretching along one direction or the formations seen in figure 3.3 (a), (b), and (c).

To determine clusters from data is difficult algorithmically, however intuitive it may be to the human eye. Often there is a requirement for the tuning of clustering algorithm hyperparameters. These parameters are either determined by domain knowledge of the data to be clustered, or by a data-driven discovery process, both leave room for bias and subjectivity. We would like to limit reliance on the need for a data-driven approach, and subjectivity.

We choose to use the density based clustering algorithm DBScan [110] to identify clusters using agent position and orientation data, since in essence a cluster in our data represents highly ordered and cohesive groups. Clearly a cohesive group will have higher (spatial) density than a fragmented one, and an ordered group will similarly have a high “angular density” (in orientation space accounting for the periodicity). What remains is to select hyperparameters informed by our application. The DBScan method has two hyperparameters **Eps** and **MinPts** (as named by M. Ester *et al*). Briefly **Eps** is a parameter which defines a distance cut-off as to whether points *can be* in the same cluster (this is a *transitive* relationship so two points may be mutually too far, but connected by an intermediate point), and **MinPts** is simply a number for the smallest cluster size. We aim to find these hyperparameters via a physical argument, i.e. using the domain knowledge approach. **MinPts** is simple to identify, we wish to identify single agents “clusters” in order to capture single agent fragmentations, so **MinPts** = 1. For **Eps** we must first define distance, and then a length scale cutoff separately for position and angular data. At a given time  $t$  in a simulation we take the agents positions  $\mathbf{x}_i^t = [x_i^t, y_i^t]^T$  and orientations  $\theta_i^t$  (interpreted in the range  $[0, 2\pi)$ ) to compute the distance matrix  $\mathbf{d}_{ij}^t$

$$\mathbf{d}_{ij}^t = \frac{1}{2}(\mathbf{d}\mathbf{x}_{ij}^t + \mathbf{d}\theta_{ij}^t). \quad (3.22)$$

Entries in the matrix are the average of the Euclidean distance for the positions,

$$\mathbf{d}\mathbf{x}_{ij}^t = \left( \frac{\sqrt{(x_i^t - x_j^t)^2 + (y_i^t - y_j^t)^2}}{\pi\sqrt{N}} \right), \quad (3.23)$$

scaled to units of the inter-agent distance at marginal opacity  $\sim \pi\sqrt{N}$  (equation 3.23). And the  $2\pi$ -Periodic Euclidean distance on the orientation data,

$$\mathbf{d}\theta_{ij}^t = \frac{\text{Min}(|\theta_i^t - \theta_j^t|, 2\pi - |\theta_i^t - \theta_j^t|)}{2\Delta\theta}, \quad (3.24)$$

scaled to units of the orientational move parameter:  $\Delta\theta$  (equation 3.24).

The reasoning for this choice is that the spatial part of our distance covers the

cohesive characteristic and the orientational part covers the ordered characteristic. In particular, as we will show, our algorithm targets marginal opacity, so we can use a scaling argument for the length scale cutoff in the spatial data. For the angular data we choose a particular parameter from our model which is more arbitrary (in general across different models or parameters), and if we change the  $\Delta\theta$  parameter in our model may need to be re-thought, but here  $\Delta\theta = 15^\circ$  is typically a constant, so we consider it a reasonable choice for the data we present.

An alternative method could take an order threshold, say  $\phi > 0.9$ , and use this to compute an approximate angle to use as a scaling from the definition of  $\phi$ . However, this method then falls to a justification of a particular  $\phi$ -value. For example taking small  $\theta_i^t \sim 0$  we could write the order as  $\phi \sim \langle \cos \theta_i^t \rangle \sim 1 - \frac{\langle (\theta_i^t)^2 \rangle}{2}$  so  $\sqrt{\langle (\theta_i^t)^2 \rangle} \sim \sqrt{2(1 - \phi)}$ , for  $\phi = 0.9$  this gives approximately  $25^\circ$ . Which is similar to taking  $2\Delta\theta$  anyway, which is normally  $30^\circ$ .

We choose to additively combine our distances in space and orientation (with a normalisation factor) and choose  $\mathbf{Eps} = 1$  when actually calling on the DBScan algorithm. We can now apply all group level statistics such as  $\phi, \Theta, \mathcal{M}$  etcetera to individual clusters, or average these over clusters, and detect/measure fragmentation. For example, we can calculate the order of a particular cluster  $C_\gamma^t$ , at time  $t$ , as

$$\phi(C_\gamma^t) = \frac{1}{|C_\gamma^t|} \left| \sum_{i \in C_\gamma^t} \hat{\mathbf{n}}_i^t \right|. \quad (3.25)$$

Similar equations apply for density, vorticity or other group measures. Throughout when reporting cluster level statistics we will make this clear in figures main text, when we are comparing cluster level (usually the largest cluster) to global statistics we will adopt a notation  $\phi_C, \rho_C$  to represent statistics computed over the largest cluster in the data, for order and density respectively. Mathematically these statistics make use of the output from the clustering methodology. This is a set of sets of indices for each time step  $t$ ,  $\mathcal{C}^t = \{C_1^t, C_2^t, \dots, C_n^t\}$  for indices  $n \in [1, 2, \dots, N]$ . Each set of indices  $C_\gamma^t \subseteq \{1, 2, \dots, N\}$  satisfies  $C_\gamma^t \cap C_{\gamma'}^t = \emptyset \ \forall \gamma \neq \gamma'$  and  $\bigcup_{\gamma=1}^N C_\gamma^t = \{1, 2, 3, \dots, N\}$ . We call each set of indices  $C_\gamma^t$  a cluster at time  $t$ . These indices can be summed over as in equation 3.25 to determine the average order of cluster a cluster over time.

Given the understanding of our statistics of comparison we will move on to examining our path entropy models.

### 3.2.1 State Count Path Entropy

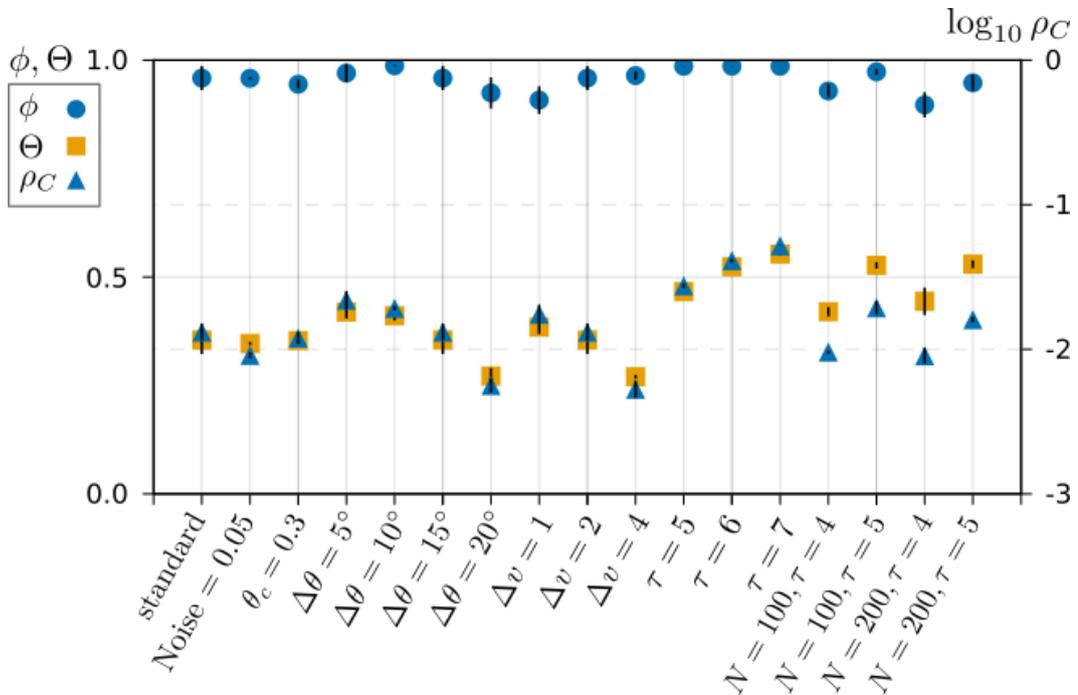


Figure 3.4: Order ( $\phi$ ), opacity ( $\Theta$ ), and time averaged convex hull number density ( $\rho$ ), equations 3.16, 3.18, 2.1 respectively, using the path entropy method (SCE) across a broad range of model parameters. Here note  $\rho$  is plotted on a  $\log_{10}$  scale. We find across this parameter set average order  $\sim 0.95$  (SD  $\sim 0.03$ ) with a minimum and maximum values of 0.86, 0.99 respectively. Lower orders are driven by fragmentations of the main group. Each data point corresponds to 3 simulations which were run for  $T = 1000$  time steps, statistics were calculated over time steps  $T = 500$  to  $T = 1000$ .

To begin we will look at our model’s behaviour over varying parameters sets. In figure 3.4 we find high order across a broad range of model parameters, in particular we see a minimum order of 0.86 in the  $N = 200$  and  $\tau = 4$  case. Notable reductions in order are seen for larger groups with lower numbers of future states ( $\tau = 4$ ) and for increased  $\Delta\theta = 20^\circ$  and decreased  $\Delta v$  (e.g.  $\Delta v = 1$ ). Opacity  $\Theta$  appears to be targeting marginal opacity  $\Theta \sim 0.5$  with increased  $\tau$  and increased  $N$ . We will explore the parameter space in  $\Delta\theta$ ,  $v_0$ , and  $\Delta v$  in latter sections.

For long timescales we find a slow rate of individuals fragmenting from the main group, which is increasing with group size and decreases with  $\tau$ . Figure 3.5 shows simulations ranging across time horizons  $\tau$  for two group sizes,  $N = 50$  (left) and  $N = 250$  (right). Three specific examples are shown in figure 3.6 for group sizes  $N = 50, 100, 250$ . Indicated on the plots are agent orientations and

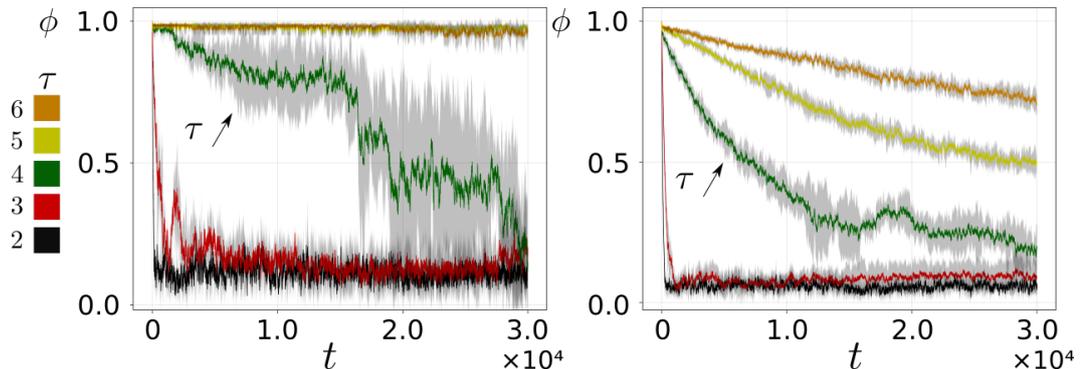


Figure 3.5: Long run-time simulations ( $T = 30000$ ) of (left)  $N = 50$  and (right)  $N = 250$  groups, in each case  $n_s = 40$ ,  $\Delta\theta = 15^\circ$ ,  $v_0 = 10$ ,  $\Delta v = 2$ , for increasing  $\tau$  values from  $\tau = 2$  to  $\tau = 6$ . We find a rapid increase in global order past  $\tau > 3$  caused by a decreased rate of fragmentation across time. Note that fragmentation is increased a large time-scales for large groups, this represents individual agents leaving the main group at a slow rate across time. Error bars indicate one standard deviation on three replicates.

the Eigenvectors used to calculate the morphology. Note the global order, density, and morphology statistics for (b) and (c) show decreased order and for  $N = 100$  (b) a band-like morphology. These values are due to fragmentations. In (b) there was a single fragmentation (over 1000 time steps) and in (c) there were  $\approx 7$ , note the statistics for the main cluster in (b) and (c) at the bottom of the caption. In particular morphology reduces from  $> 1000$  to  $\sim 8$  highlighting the importance of clustering in our data. A trend is beginning to emerge already in figure 3.6, namely with increasing group size a stretching of the main group appears to be present; a morphology change from marginal opacity. For the SCE model we have found broadly stable highly ordered and cohesive flocks over a range of model parameters. We have also probed the long-time behaviour of small and large systems revealing a slow rate of fragmentation in groups of increasing  $\tau$ .

### 3.2.2 Compression Entropy

We mentioned previously that in the CE model we find that a necessary additional parameter is required, namely a restriction on the minimum angular size an object can take to be visible. Recall that under CE the maximum entropy will be attained for a given visual state if that visual state has  $2(N - 1)$  visual boundaries.  $2(N - 1)$  comes from the fact  $N - 1$  agents project as at most 2 tangent lines from a given agent's perspective, and if there are no overlapping visual states this produces  $2(N - 1)$  visual state boundaries. This state can be attained at arbitrary spatial separation.

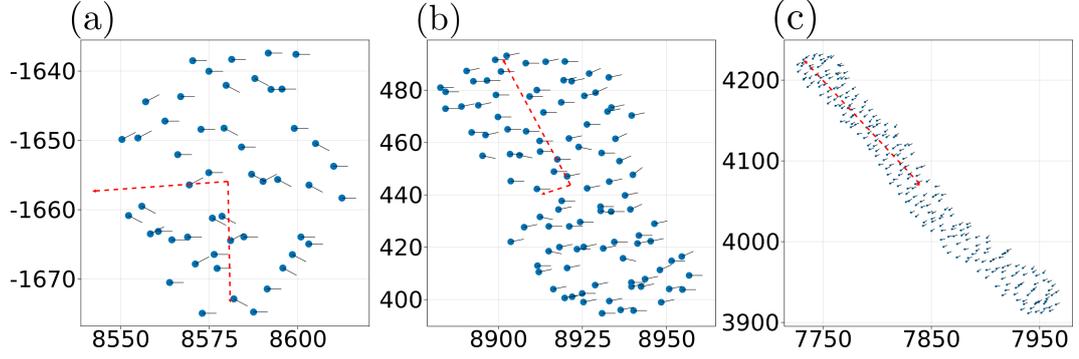


Figure 3.6: Three example simulations with  $n_s = 40, \Delta\theta = 15^\circ, v_0 = 10, \Delta v = 2, \tau = 6$  with (a)  $N = 50$  (b)  $N = 100$  (c)  $N = 250$ . In each case the Eigenvectors used to calculate the morphology are indicated in (dashed) red scaled by a common constant for visibility. (b) and (c) include some fragmentations ( $\approx 1$  and  $2$  respectively) not pictured. In these three cases the (global) order, morphology and density ( $\phi, \mathcal{M}, \rho$ ) are (a)  $0.986, 1.89, 0.092$  (b)  $0.975, 1168, 0.068$ , and (c)  $0.954, 57.23, 0.051$ . The reduction in order for (b) and (c) and band-like morphology in (b) is due to fragmentation of  $\approx 1$  and  $7$  agent/s from the main group respectively, accounting for these fragmentations by computing statistics on the largest cluster, gives (main cluster  $\phi, \mathcal{M}, \rho$ ) of (b)  $0.981, 8.96, 0.068$  and (c)  $0.98, 81.9, 0.05$ .

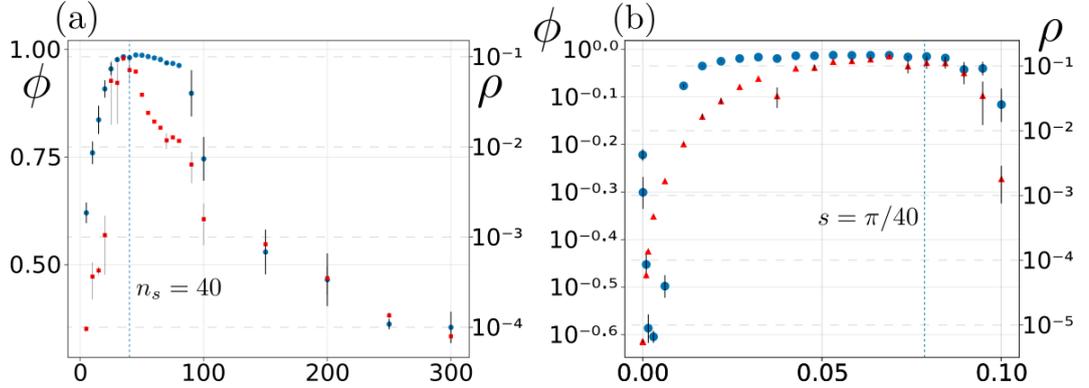


Figure 3.7: The order  $\phi$  (blue circles) and largest cluster convex hull density  $\rho_C$  (red triangles) depend on (a) the number of sensors  $n_s$  and (b) the resolution parameter for the unique state counts and compression entropy models respectively. Indicated as a dashed line is the value of  $n_s = 40$  and  $s = \pi/40$  which is indicative of the size of a projection (or union of projections) needed to activate a sensor in both the USC and SCE models (with discretised sensors) and a sensor count of  $n_s = 40$ . Here  $\tau = 6, N = 50$  and error bars correspond to one standard deviation of the order across 10 replicates. Note also density begins to increase before order in (b). This measurement indicates decorrelated motion but with a sense of group cohesion, like a swarm of insects for example.

In fact  $N - 1$  agents placed “at infinity”, with position vectors  $r[\cos(i2\pi/(N - 1)), \sin(i2\pi/(N - 1))]^T$  for agent index  $i \in 1, 2, \dots, N - 1$ , as  $r \rightarrow \infty$  will attain maximum visual state entropy. It is perhaps then unsurprising that agent’s target this state, see figure 3.7 (b) around  $s = 0$  and note the density in red squares. Compare this data with figure 3.7 (a) as  $n_s$  increases.

In the CE model, the resolution parameter  $s \in [0, 2\pi]$  defines a minimal angular size below which a visual projection is removed on an agents visual state. This encodes a constraint on a combination of object size and object distance, a close but small object will have a relatively large angular size compared to a much larger object if it is far enough away.

We see that in figure 3.7 that low values  $s \lesssim 0.01$  result in no onset of order. Intermediate  $0.01 \lesssim s \lesssim 0.8$  results in ordered motion. Large values of  $s$  will lead to highly degenerate visual states and therefore low order, i.e. at the extreme where  $s \rightarrow 2\pi$  the visual state will be completely barren containing no projections at all. It is worth comparing figure 3.7 (a) and (b) to see the effect of the number of sensors and the resolution parameter on order, note of course that  $n_s$  and  $s$  have an inverse relationship i.e a small  $n_s$  should be compared to a large  $s$ . Remember also the implicit criterion of a half-full sensor in (a) meaning an angular size constraint can be derived as  $\pi/n_s$  i.e. projections (or unions of them) smaller than this will not be visible. Given this data we now produce the same data as computed for figure 3.4 using the CE model with a parameter  $s = \pi/40$  as shown in figure 3.8. We find broad agreement with the SCE model across the parameters tested indicating that both models are largely equivalent in their outputs. There are a few deviations, for example with  $\Delta\theta = 5^\circ$  and all other parameters standard, the CE model appears to result in an improved order against SCE. Deviations in density can be noted with lower density in the CE model than in the SCE model.

### 3.3 Phenotypes

We have found a sizeable regime with minimal changes by varying  $\Delta\theta$ ,  $v_0$ , and  $\Delta v$  separately when comparing the SCE and CE models but have yet to explore combinations of these parameters. For motivation consider what we know *a priori* about these parameters. Changing  $v_0$  alone has the effect of increasing or decreasing how far agents explore in the future. In particular  $v_0 \rightarrow 0$ , but non-zero, implies that visual states on the FST are to within a small variance equal to the root’s (current time’s) visual state, meaning entropy will shrink to 0 in this limit. We might expect when  $v_0\Delta t\tau \sim r$ , the agent radius, that we find a low order but collision avoiding

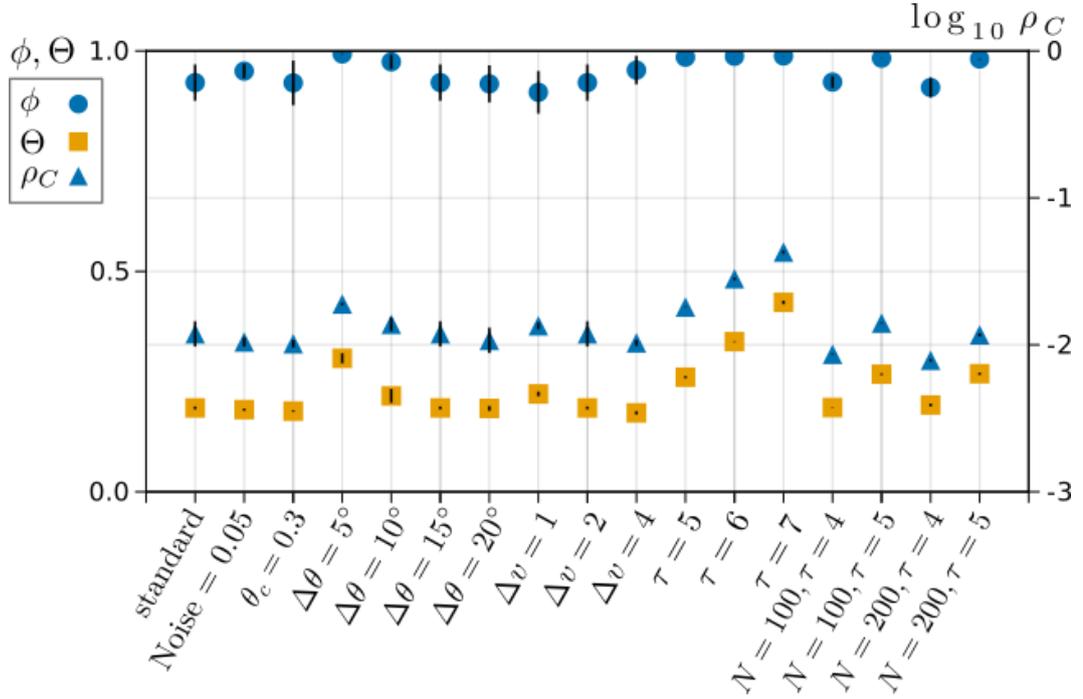


Figure 3.8: Parameter sweep for the CE model, confer with figure 3.4. Order ( $\phi$ ), opacity ( $\Theta$ ), and convex hull number density ( $\rho_C$ ) using the path entropy method (SCE) across a broad range of model parameters. Here note  $\rho_C$  is plotted on a  $\log_{10}$  scale. We find across this parameter set average order  $\sim 0.95$  (SD  $\sim 0.03$ ) with a minimum and maximum values of 0.86, 0.99 respectively. Lower orders are driven by fragmentations of the main group. Each data point corresponds to 3 simulations which were run for  $T = 1000$  time steps, statistics were calculated over time steps  $T = 500$  to  $T = 1000$ .

swarm, since agents are unable to reason much about the quality of far away future states, but can reason about collisions in the near future. Alternatively if we change  $\Delta v$  alone we can observe the limit of  $\Delta v \rightarrow 0$  with behaviour dependent on  $v_0$ , and  $\Delta v \rightarrow v_0$  or  $\Delta v > v_0$ . In the latter two cases we imply “stand still” and “reverse” actions, *a priori* it is unclear what this will mean for resultant dynamics. Finally, when changing  $\Delta\theta$  one scale of interest is the ratio  $180^\circ/\Delta\theta$ , in particular comparing this to  $\tau$ , for example with  $\Delta\theta = 15^\circ$  we need  $180/15 = 12$  same sign rotation actions in order to complete a full rotation, whereas with  $\Delta\theta = 30^\circ$  we need only 6 which makes this possible at  $\tau = 6$ . It is reasonable *a priori* that we might find increased rotation with increased  $\Delta\theta$  and maybe even a turning flock. However it seems unlikely the group will pick and maintain a direction of turning.

Clearly the interplay between these parameters could define varied behaviour in the model. To test this hypothesis we examine the effect of varying  $\Delta v \in [0, 2]$ ,

$v_0 \in [0, 5]$  and  $\Delta\theta \in \{15^\circ, 25^\circ, 30^\circ\}$  in figure 3.9 where each row shows the order, the largest cluster size, and order of the largest cluster for each  $\Delta\theta$  value. We find that increasing  $\Delta\theta$  decreases the order in the regime  $v_0 \gtrsim 3$ ,  $\Delta v \gtrsim 1$  which is commensurate with the data in figure 3.4. We also find that stepping along the  $v_0$  or  $\Delta v$  axes from the top right to the bottom-left corner leads to disorder, and the same is true along any vector from the top-right corner. However, the largest cluster size shows a different picture. The regime  $\Delta\theta = 15^\circ$  with  $v_0 \gtrsim 2$  and  $\Delta v \gtrsim 1$  shows the behaviour we expect if the disordering in  $\phi$  for the same range is due to fragmentation, i.e. the largest cluster size reduces, this is still the case for  $\Delta\theta = 25^\circ, 30^\circ$  in this regime. However, looking in the regime  $v_0 \lesssim 2$  with  $\Delta v > 0$  we are seeing a decreasing order  $\phi \rightarrow 0$  with higher largest cluster sizes, and with  $\Delta 25^\circ, 30^\circ$  the largest cluster size is actually bigger. This is indicative of a disordered yet cohesive swarm, which is further confirmed by the largest cluster order  $\phi_C \approx 0$ . Further we find the onset of this swarm at  $\Delta v \gtrsim v_0$  with  $\Delta\theta = 15^\circ$  but by  $\Delta\theta = 30^\circ$  we find the swarm existing with  $\Delta v \lesssim v_0$  as well.

Some examples of these behaviours are shown in figure 3.10. For example figure 3.10 (a) shows the familiar ordered, cohesive and travelling group, figure 3.10 (b) shows a circulating disordered by cohesive group which includes velocity reversals, figure 3.10 (c) involve a “stopping” action which does was less cohesive but not fully fragmented i.e. agents on the periphery returned during the simulation time ( $T = 1000$ ). Figures 3.10 (d,e) show the same parameters with  $N = 250, 50$  respectively, in this case  $\Delta v = 0.75$  whilst  $v_0 = 1$  we find a group which breaks and reforms across time swarming around a roughly stationary centre of mass point, (e) shows the group about to reform. Figure 3.10 shows a “balanced” action space with two speed and two orientation change moves which leads to another swarm like disordered group ( $v_0 = 1$ ), increasing  $v_0$  leads to behaviour like (a).

### 3.4 Conclusions

We have seen that agents deciding on actions by independently maximising future path entropy of visual states generates spontaneous collective motion. Further by attempting to move away from discrete states of  $n_s$  sensors by use of a compression entropy measure of entropy of a general visual state, we must re-impose an effective discretisation into sensors by an angular size constraint. The constraint meaning individual agents, or groups of them, projecting into a smaller angle than the constraint angle are removed from an agents visual state. We find these two methods, SCE and CE, are broadly equivalent in phenomenology. We therefore established

that a compression entropy model could be formulated, with the angular size constraint. We found that across a broad range of parameters the two approaches are largely equivalent in terms of order and density, with some change in the opacity values.

An important point with the SCE method is that the visual state is discretised into  $n_s$  sensors. While there is biological basis for sensors, no clear biological basis is apparent for the number  $n_s = 40$ , which is approximately optimal for e.g.  $\tau \approx 4, N \approx 50$ . One may note that a birds eye has far more than  $n_s = 40$  “sensors” making up its eye, if a “sensor” can be equated to the light sensitive cells in the eye. Even in compound eye insects like *Drosophila* there are 700 units making up the eye [103]. Fundamentally it is not too difficult to accept a discretisation, but it is difficult to explain  $n_s = 40$ .

Synthesising this chapter’s results we propose that the explanation for this difference is that the sensors themselves are not directly related to the number of “units in the retina”, but rather they represent (at least) one step up in the cognitive process; an abstract representation of the visual state. That is the visual state itself is a representation of raw retina activation, and the sensors are how the agent processes this information to a use-able state. Our evidence being that, computationally, the visual state is discretised by floating point numbers to the range of  $10^8$  and  $10^{16}$  for 32- and 64-bit representations, which might be better placed as the direct comparison to the “units in the retina” rather than the number of sensors. This is because coarse floating point representation will corrode visual information, i.e. tangent line projection lines for this model, but increasingly fine floating point representation will resolve this information more accurately. We also require an angular size threshold of coverage of a sensor at which it becomes activated. Further we saw that even without sensors, as in the CE method, we still require the imposition of this angular length scale, below which projections (or unions of them) are ignored.

Interpreting the angular length scale itself as an abstraction in the cognitive processing of the visual state, would mean a bird or insect sees the visual state, to a degree of precision dictated by the accuracy of the retina (floating point numbers computationally), and this is processed in such a way to ignore objects that are too small or too far (small in angular size) to be recognised. This could be due to a property of the cognitive process, not the resolution of the retina, discounting information of little perceived value as well as information which is too difficult to make out. I.e a far away or small object may not matter for decision-making even if it is perfectly visible and at high visual fidelity. Theoretically this could

be tested with actual biological examples. Take *Drosophila* as an example, how far away does one individual need to be from another before (1) it can no longer see the other [eye resolution] (2) its decisions are no longer impacted by the other individual [ignoring information of little value]. These length scales could be equal or unequal, to one another giving real evidence for the interpretation of sensors in our model. Experimentally these questions are likely more difficult to answer than stated here, if it were possible it presents an avenue to experimentally test the visual projection and sensor component of the model proposed, or at least the importance of the angular size in decision-making in the real world.

By varying the model parameters, in particular  $\Delta\theta$ ,  $\Delta v$  and  $v_0$ , we revealed a number of different phenotypes, qualitatively distinct dynamical modes, in our model. Firstly, a translating highly ordered parameter regime  $v_0 \gtrsim 2$ ,  $\Delta v \lesssim v_0$  e.g. figure 3.10 (a), secondly swarming groups characterised by low order but high cohesion around  $v_0 \lesssim 2$  and  $\Delta v \lesssim v_0$  e.g. figure 3.10 (d), and finally swarming groups with higher group rotation  $v_0 \lesssim 2$  and  $\Delta v > v_0$  e.g. figure 3.10 (e). This provides one avenue for fitting our model to real data, i.e. by fitting the action set parameters themselves. We shall return to this idea in the final chapter of this thesis, chapter 5, when we extend the action space to a continuous setting.

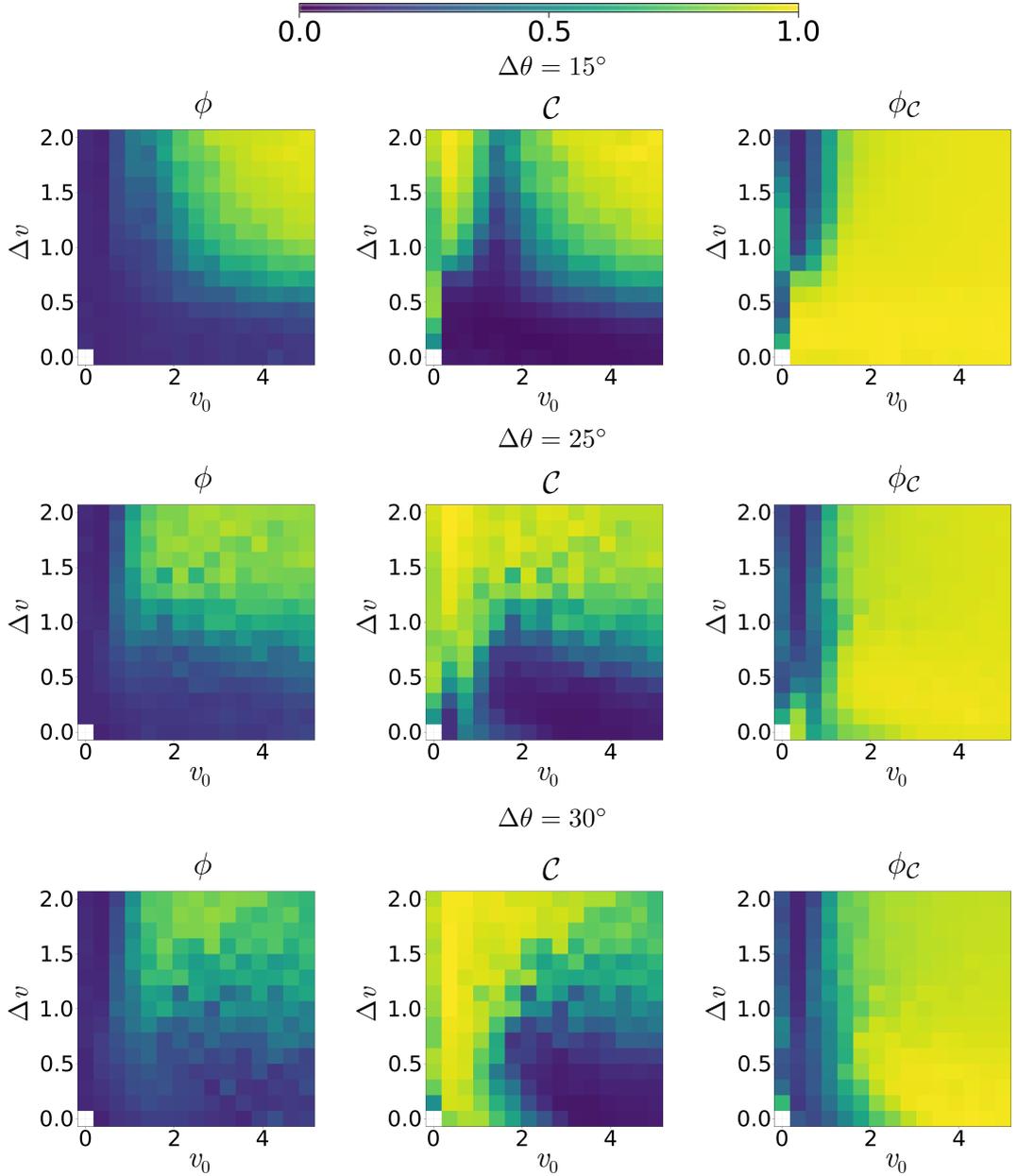


Figure 3.9: Order (first column), largest cluster size (second column), and largest cluster order (third column) for  $\Delta\theta = 15^\circ, 25^\circ, 30^\circ$  (across rows) for groups with  $\tau = 5$ ,  $N = 50$ ,  $n_s = 40$ . Each group was simulated for 5000 time steps with colour as the mean across three replicates, normalised according to the colour bar. Each statistic,  $\phi$ ,  $\mathcal{C}$ , and  $\phi_{\mathcal{C}}$  are defined in the range  $[0, 1]$ . We find two regions of high order and high cohesion and low order but still high cohesion. These are indicative of highly ordered cohesive groups with persistent directed motion (the former) and a swarming phenotype (the latter). The swarming phenotype is more stable across parameter space  $(v_0, \Delta v)$  at increased  $\Delta\theta \approx 30^\circ$ .

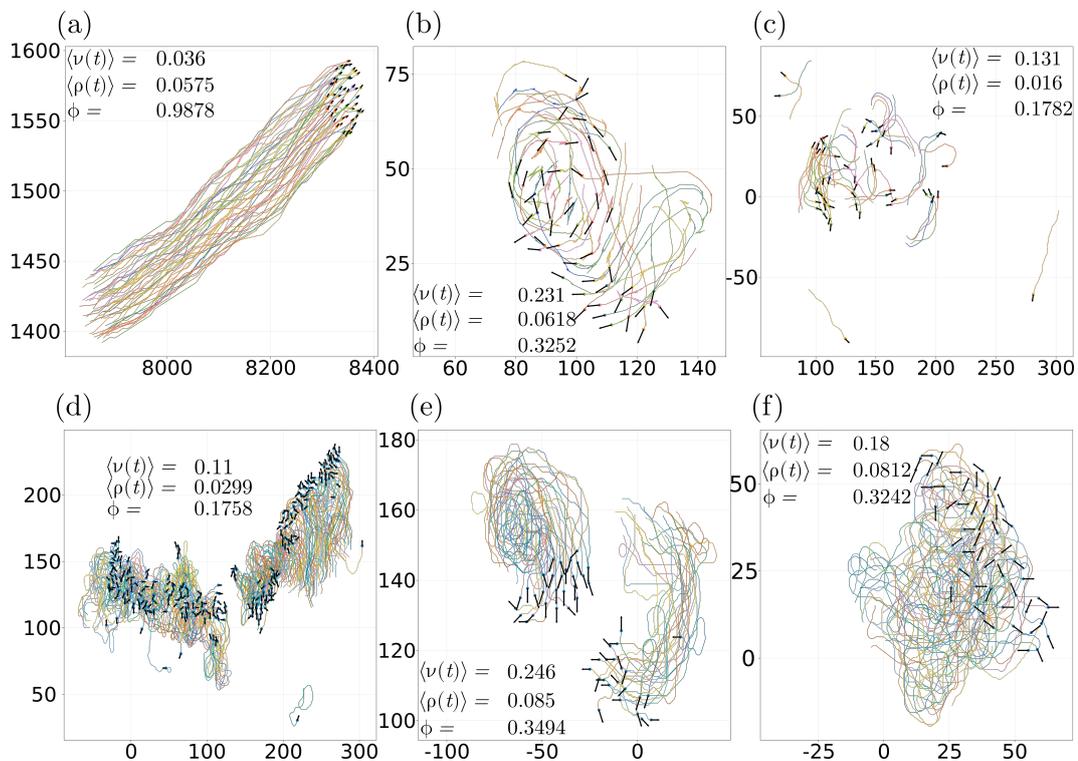


Figure 3.10: Various phenotypes observed by varying model parameters, common parameters are  $\tau = 5$  and  $n_s = 40$ , all include trials plotted for 100 timesteps. (a) the ordered and cohesive phenotype with parameters  $N = 50$ ,  $\Delta\theta = 15^\circ$ ,  $v_0 = 10$ ,  $\Delta v = 2$ . (b) Persistent rotation here with velocity reversals i.e.  $N = 50$ ,  $\Delta\theta = 15^\circ$ ,  $v_0 = 1$ ,  $\Delta v = 2$  implying speeds  $-1, 1, 3$ , rotation is long-lived but not always with the same sign across different initial conditions, and possibly run-time. (c) Not cohesive but not completely fragmented group,  $N = 50$ ,  $v_0 = 1$ ,  $\Delta v = 1$ ,  $\Delta\theta = 15^\circ$  (i.e. including a stopping action  $v^t = 0$ ). (d,e)  $N = 250, 50$  respectively with common parameters  $\tau = 5$ ,  $\Delta\theta = 30^\circ$ ,  $\Delta v = 0.75$ , and  $v_0 = 1$ . We see coexistence of multiple clusters breaking and forming whilst swarming around a largely stationary central point, in (e) the two clusters are about to reform and then break again, reducing  $\Delta\theta$  to  $15^\circ$  and lower results in large scale fragmentations within  $\sim 100$  time steps. (f)  $N = 50$ ,  $\Delta\theta = 30^\circ$  but with a different actions space, two speeds  $v^t = 1$  or  $v^t = 2$  and two angle increments  $\pm\Delta\theta$ .

## Chapter 4

# Visual Information and the Cognitive Process

We have seen how maximising environmental state entropy can result in spontaneous collective motion. In the first method, state count entropy (SCE), agents individually calculate hypothetical future visual states, accessible by an action set available to them and a model of other agents (typically ballistic). Each use this information to pick the action now which maximises the entropy along all hypothetical subsequent paths. The entropy being calculated using the empirical state count distribution along paths. The second, compression entropy (CE), instead defined an entropy for an arbitrary, non-discretised visual state by the length of an encoded representation of it. Agents then pick the action maximising the average compression length, across all subsequent paths. Both highlight the importance of the diversity and quality of visual information input in this decision-making environment, and both lead to emergent flocking behaviour.

One particular assumption has been implicit in our discussion of these methods thus far, all agents have complete and perfect information of the group dynamics. That is an agent knows *exact* positions and orientations of all other agents, and is able to use these to project “perfect” ballistic trajectories into the future, even if these trajectories are not accurate predictions of the future group dynamics. One can now ask what happens if we disrupt this perfect information, e.g. by introducing some noise - do we still get collective motion? Or said another way, how robust is this decision-making process with imperfect or noisy information? We will take two threads to attempt to answer this question using deterministic and stochastic perturbations respectively to disrupt the cognitive process.

For the deterministic approach we will develop an explicit occlusion model,

that is we will examine the case where if an agent  $j$  is obscured by another agent (or agents)  $k$  from the perspective of agent  $i$  then agent  $j$  is considered invisible to agent  $i$ . In particular this will mean agent  $i$  will not be able to use the information from agent  $j$ , its position, orientation etc., in its cognitive process; agent  $j$  will be absent from  $i$ 's visual state now and in the future projections. Key to this is the criterion by which “an agent  $j$  is obscured by another agent  $k$  from the perspective of agent  $i$ ”, we will make this precise using a geometric construction, involving a single control parameter. This occlusion control parameter will allow us to control how much another agent must be obscured to be discounted, in the sensory acquisition stage of the algorithm.

The stochastic approach involves considering a variety of random processes. In particular, we will examine the effect of adding a noise term *post-decision*. This will play the role of modelling uncertain decisions or simply decision error which will of course feed back into future decision-making. Secondly we will apply noise directly to the cognitive process itself. Here within each agent's cognitive process involves using trajectories of the other agents drawn from a stochastic process. In particular, we will use a process where actions, speeds and rotations, are drawn from normal distributions, independently, in place of the ballistic model for the hypothetical futures  $1, 2, \dots, \tau$ . We will use zero-mean normal distributions with two variances (for speeds and rotations separately) so that the mean trajectory is ballistic.

For this chapter our hypotheses are that the occlusion model will have a minor effect on the order and cohesion of groups, but may change morphology, density and opacity. We also expect any de-stabilising effects to diminish with increasing future time horizons  $\tau$ , since higher values of  $\tau$  generate more stable groups. In the presence of noise processes we expect to uncover some form of order-disorder transition in the presence of increasing noise.

## 4.1 Perturbing Visual Information Deterministically: Occlusion

In the real world visual occlusion would involve the obscuring of one object by another e.g, when a nearby object is directly in front of a more distant object. Whether the object is totally or partially obscured will depend on the relative sizes of the objects, their shapes, and relative positions. In two-dimensions with identical circular agents, using the visual projection method we can exactly identify occlusions by examining overlapping intervals on the visual state. As shown in the sketch in

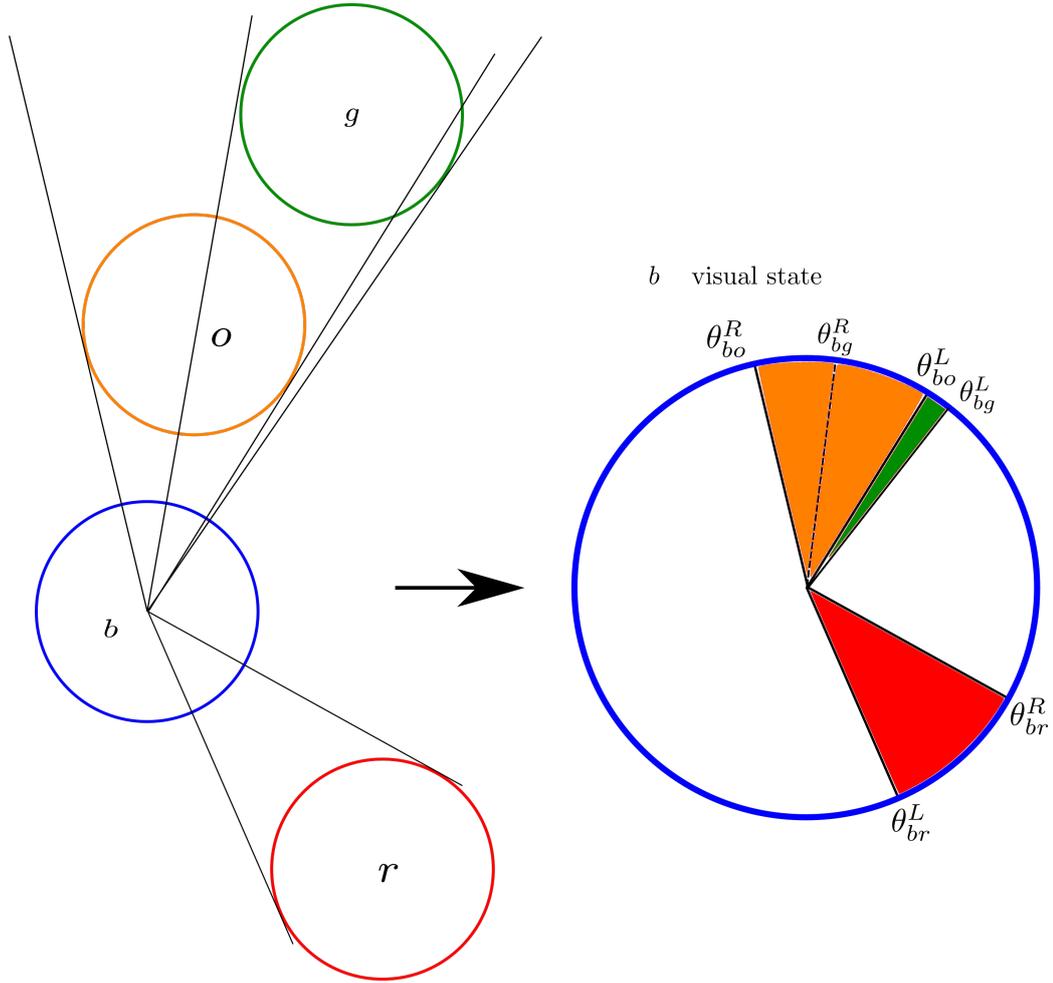


Figure 4.1: Schematic representation of the occlusion problem as overlapping visual projections, from the perspective of the blue agent ( $b$ ). The diagram on the left represent agent position and the diagram on the right the resultant visual state. Note the overlaps between the visual projection of the orange ( $o$ ) and green ( $g$ ) agents.

figure 4.1.

#### 4.1.1 Explicit Occlusion Model

To understand this consider an example of a visual state of an agent, figure 4.1. Here the blue agent ( $b$ ) see's three others: the green ( $g$ ) the orange ( $o$ ) and the red ( $r$ ) agents. Under the visual projection method these agents map to the intervals defined by the angles  $[\theta_{bg}^L, \theta_{bg}^R]$ ,  $[\theta_{bo}^L, \theta_{bo}^R]$ ,  $[\theta_{br}^L, \theta_{br}^R]$ . Where the index  $bo$  means the blues ( $b$ ) projection of the orange agent ( $o$ ), while  $L$  and  $R$  refer to the “left”=smallest and

“right”=largest angle for the interval. Of course the  $[0, 2\pi)$  boundary must be carefully handled, typically we split any projections crossing the boundary into two pieces to maintain the smallest-largest classification. Note that (1) the projections of the orange and green agents overlap (2) the green agent is further than the orange from blue. Naturally this means the orange agent is obscuring the view of the green agent, but not totally. The proportion of the obscurity is given by the ratio  $\frac{\theta_{bo}^L - \theta_{bg}^L}{\theta_{bg}^R - \theta_{bg}^L}$ , which represents contracting the projection interval of the green (furthest) agent, so it no longer overlaps with the orange, then calculating the percentage of the original size lost by this process. Clearly with one dimensional intervals like

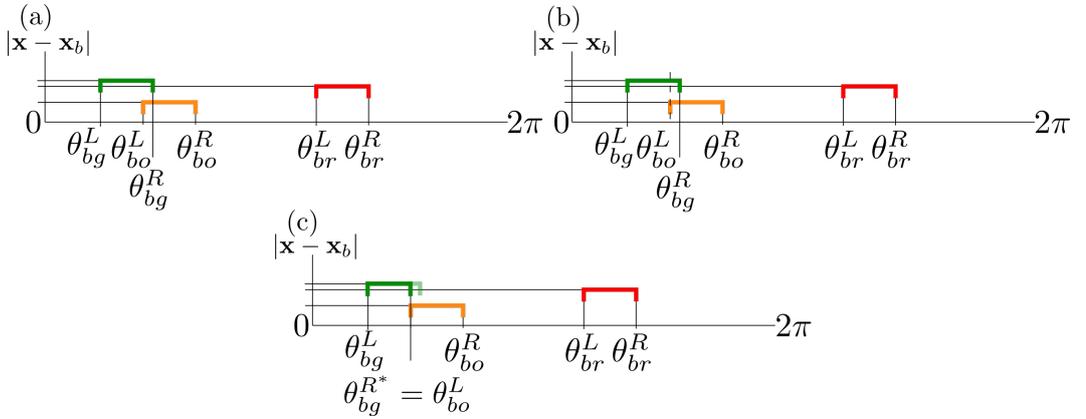


Figure 4.2: Sketch of the occlusion algorithm as applied to the situation in figure 4.1. (a) projection intervals are sorted by distance from the agent calculating the visual state. (b) overlaps are identified, i.e. by the dashed line covering the overlap between the orange and green intervals (c) for each overlap the furthest interval is contracted to remove the overlap, indicated by the faded region on the green interval.

this there are numerous cases of overlapping that can be found, e.g. one interval may be totally contained within another, or overlap on one side or the other. Since we have identical circles the overlaps we will get will not include “splitting”, where an agent’s projection is split into multiple pieces, since further agents always have smaller angular sizes than closer ones. With elliptical particles this will not be the case, e.g. if two ellipses, with 2:1 aspect ratio, lie on the same vector, relative to a visualising agent, the closest with its major axis parallel to the vector and the furthest with its minor axis parallel to the vector, the effect of occlusion on the furthest agent will be to split its projection into two pieces. We will assume identical circles here to avoid this problem.

The occlusion model then involves calculating the “visibility ratio”,

$$\gamma_{ij} = \frac{\theta_{ij}^{R^*} - \theta_{ij}^{L^*}}{\theta_{ij}^R - \theta_{ij}^L}. \quad (4.1)$$

Where the interval  $[\theta_{ij}^{L^*}, \theta_{ij}^{R^*}]$  represents the smaller contracted interval, or for non-occluded agents the original interval. The  $\gamma_{ij}$  should for some agent  $i$  then contain information on “how visible” the  $j$ 'th agent ( $i \neq j$ ) is from the perspective of the  $i$ 'th agent. I.e.  $\gamma_{ij} = 1$  implies agent  $j$  is fully visible to agent  $i$  ( $j$ 's projection interval is unchanged), and  $\gamma_{ij} = 0$  implies agent  $j$  is fully obscured to agent  $i$  ( $j$ 's projection interval is entirely redundant; it is contained within another projection or union of projections).

The problem of finding the  $\gamma_{ij}$ 's can be solved as a weighted partial set cover problem, where weights are the distance from the visualising agent, and the cost is, for example, amount of an interval contracted multiplied by the inverse distance. Therefore, “sets” also includes all possible contractions of the original sets (visual projection intervals), and “partial set cover” means covering the interval  $[0, 2\pi]$  as completely as possible. For illustration figure 4.2 (a) shows sorting the projections in figure 4.1 by distance from the reference agent (b, blue), figure 4.2 (b) shows an identified overlap (dashed line), and figure 4.2 (c) shows a contraction of the “furthest projection”.

Although considering all pairs of intervals, which is  $O(N^2)$  for  $N$  agents, appears computationally expensive, it does not necessarily mean poorer scaling with  $N$  (per time step) when compared with the non-occlusive case. is because occluded agents are excluded from decision-making. The decision-making itself relies on multiple  $O(N^2)$  operations for e.g. visual state calculations. Since the occlusion process leaves  $N_o \leq N$  agents to process in hypothetical futures, this fact can actually decrease run-time for moderate  $N \lesssim 1000$ .

The information in  $\gamma_{ij}$  allows us to define a criterion for occlusion. At what point is an agent unable to project another obscured agent into the future? Clearly a totally obscured agent is an obvious candidate for exclusion, but otherwise one can pick any number  $\mathcal{O} \in [0, 1]$  where an agent is discounted if its visibility is less than a chosen  $\mathcal{O}$ , i.e.

$$\gamma_{ij} \leq \mathcal{O}. \quad (4.2)$$

Taking  $\mathcal{O} = 0$  means all agents will be visible except for the fully obscured, taking  $\mathcal{O} = 1$  means all agents are considered invisible. We can also define a special value

$\mathcal{O}^* < 0$  which would mean all agents are visible regardless as to their actual visibility.

Before moving on we should note firstly that, in theory equation 4.2 can be extended to three-dimensional space. To see this recognise that  $\gamma_{ij}$  is a ratio of two angular sizes, in 3D this is readily computed using solid angles in place of these angular sizes. The difficulty of course lies in finding the overlaps of projections in 3D space, the added dimension of the visual projections (now spherical caps) creates an infinite number of different overlap types (curves on the sphere). This leads to finding the area of a “dissected” spherical cap; a spherical cap “dissected” by the curves of overlapping spherical caps. In principle this can be solved, although it is not clear if an analytic technique is possible or if an approximate method is required. We will not dwell on this in this thesis as the extension to 3D is left for the future.

Secondly, equation 4.2 defines a condition on matrix elements, this can be used as a natural connectivity matrix that varies temporally. In the context of the literature on consensus algorithms [111, 112] and their application to flocking phenomena [113], this connectivity matrix could be used. In this context agents interact through a connectivity matrix by e.g. averaging heading direction and speed. In particular, it is known that as long as the network is connected then consensus can be achieved, with a rate of convergence related to the degree of connectivity [111, 112]. For a flock modelled in this way this means agents attempt to reach consensus on velocity. As such it would be surprising if using the visibility matrix defined by equation 4.2 as e.g. the velocity or position graph used by Tanner *et al.*'s model or indeed as the only graph in a modified model, did not lead to robust flocking behaviour. For  $\mathcal{O} < 1$ .

#### 4.1.2 Effect of the Occlusion Threshold

To understand how taking into account occlusion in the visual state will affect the model, we will first examine the effect of occlusion on some visual state. Given a visual state of agent  $i$ ,  $\psi_i$ , the occlusion model will find the visibility values  $\gamma_{ij}$  for agent  $j$ . When calculating these values intervals are contracted so as to not overlap, i.e. redundancy is removed from the visual state. As such the contraction process itself will not change the appearance of the visual state at all. The next step is to ignore agents who are occluded, via equation 4.2. In future states occlusion is not applied, we assume the modelling agent can project future trajectories and therefore model agents even when they are, temporally, occluded from its perspective in hypothetical futures.

At this point there is a choice we can either take (1)  $\psi_i$  as is or (2) remove agents from the visual state that are considered occluded. Case (1) does have an

advantage in consistency, that is in hypothetical future states we do not apply the occlusion model whereas with (2) this removal process would not be completed in hypothetical futures. (1) does have the disadvantage that for small perturbations to the system configuration agent relative positions may have changed little, but on the removal of occluded agents the visual state one hypothetical step in the future will change relatively dramatically. This problem also is more severe for “sensitive” occlusion parameters e.g.  $\mathcal{O} \lesssim 1.0$ . However, removing projections on the visual state in (2) is somewhat unrealistic since by definition the removed sections are actually visible. For consistency in the visual state at time  $t$  and hypothetical visual states, we will opt for choice (1).

More complex criteria could be used to determine agent visibility. Here if an agent is obscured at time  $t$  it is not projected into hypothetical futures. We could define a “memory” parameter, say  $\Delta\tau \in [1, 2, \dots]$ , such that each agent can recall past positions for previous times  $[t - \Delta\tau, t]$ . Meaning that if an agent is obscured at time  $t$ , but was visible for at least one  $t' \in [t - \Delta\tau, t]$  the agent is projected into hypothetical futures. This memory could also be taken over discrete time points  $t' \in \{t - \Delta\tau, t - \Delta\tau + 1, \dots, t\}$  as apposed to the continuous interval. However, in defining agents with memory for the purposes of occlusion, we may also want to define how agents use that memory in their cognitive process in other ways. For example by fitting a heuristic model based upon each agent’s past positions and actions, to use in place of the ballistic model. For this thesis we will continue with memory-less agents, leaving this as a future research avenue.

Given this specification we can compute the effects of occlusion on the input to agents. At time  $t$  in the simulation an agent will find its current visual state  $\psi_i$  and which agents are visible given the occlusion parameter. Without occlusion the agent  $i$  calculates future states by taking actions and modelling agents. With occlusion any agents considered occluded will not be propagated into hypothetical futures, and therefore will not impact future visual states and decision-making. *A priori* this is all we can say as to how the occlusion model affects the model, we know the current visual state at some time  $t$  may be changed and that, potentially, the future states as well if occlusion is applied. We will now examine any changes in phenomenology via the usual statistics.

## Occlusion and Model Parameters

First looking at the order across various parameters in figure 4.3 we see the effect of varying occlusion. Figure 4.3 (a) is the non-occlusive case and figure 4.3 (b,c,d) show results of simulation obtained with  $\mathcal{O} = 0.0, 0.5, 0.95$  respectively. It appears

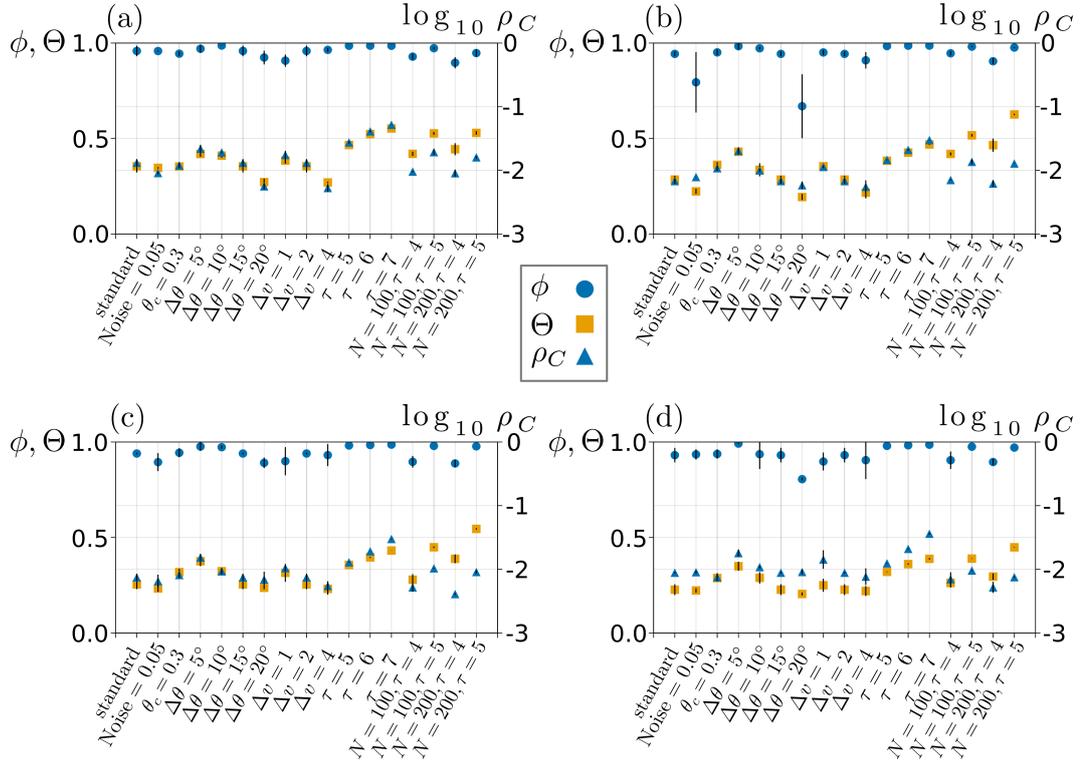


Figure 4.3: Parameter sweep for SCE using four occlusion thresholds, (a,b,c,d)  $\mathcal{O} = \mathcal{O}^*$  (no occlusion), 0.0, 0.5, 0.95 respectively. This data suggests (1) occlusion promotes order in larger groups ( $N = 200$ ) in this case (2) the standard set and systems with higher  $\tau = 5, 6, 7$  appear largely unchanged (3) the model appears less robust to changes in  $\Delta\theta, \Delta v$  and the presence of orientational noise post decision, but this decreases with  $\mathcal{O}$  increasing (4) density appears to have decreased slightly on average, but with significant variation. (5) opacity appears reduced on average but with significant variation.

that for  $N = 50$  and  $\tau > 4$  occlusion has little impact, although the order is so high it is perhaps somewhat unsurprising. However, for  $N = 200, \tau = 5$  we see an increase in order across all occlusion values, but with  $N = 200, \tau = 4$  we see the opposite. The standard parameter set appears largely unaffected. On average there appears to be a decrease in density and opacity for all occlusion values, but this does vary, e.g.  $N = 200, \tau = 5$  we see an increased opacity and decreased density at  $\mathcal{O} = 0.0$  figure 4.3 (b) but across (c-d) we find the opacity reduces and is lower than in (a) for (d). Changes in  $\Delta v$  and  $\Delta\theta$  result in (b) we see fragmentation in the orientational noise case and with  $\Delta\theta = 20^\circ$  resulting in large error bars that disappears in (c) and (d), but the reduced order remains. Changing the activation threshold to  $\theta_c = 0.3$  appears to be unaffected in each occlusive case where we see

a higher order, but this looks likely within one standard error.

### Varying Effects of Occlusion for Small and Large Groups

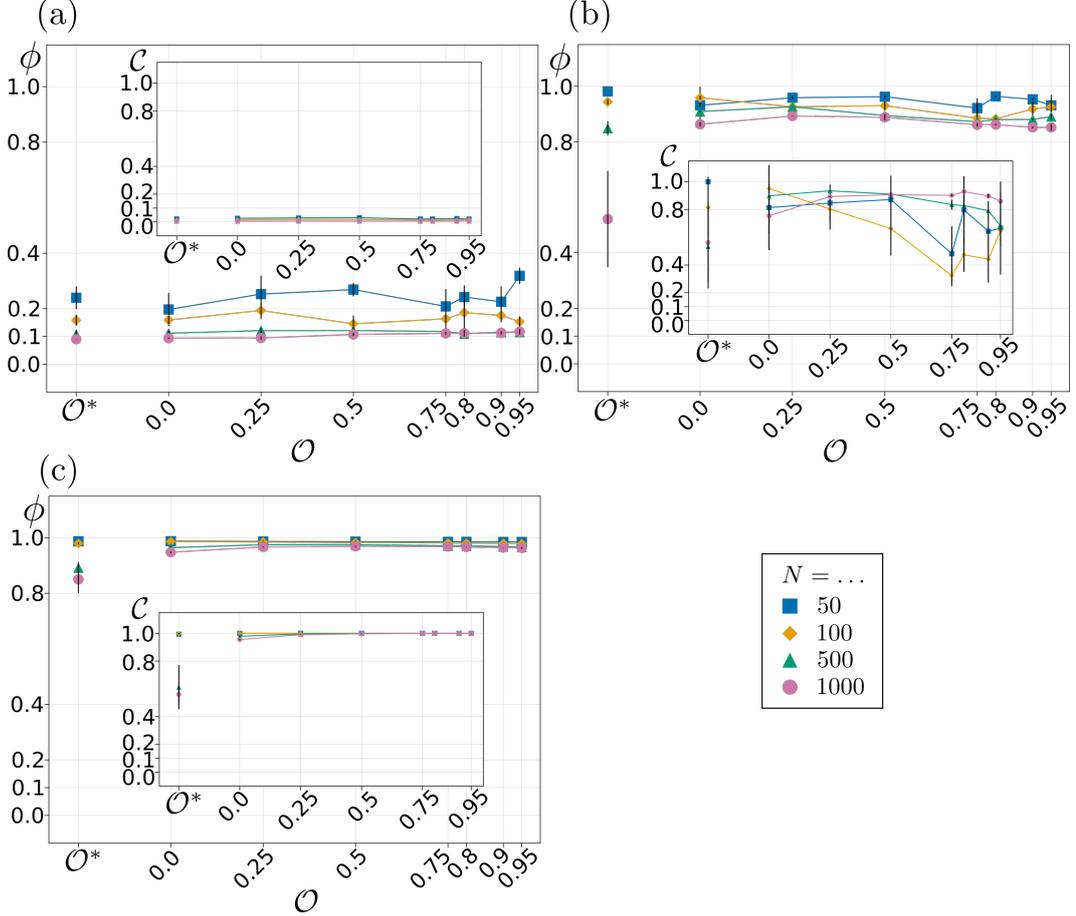


Figure 4.4: Varying occlusion threshold has decreasing effect for future time horizons  $\tau \gtrsim 4$ , except in larger groups  $N \gtrsim 500$ . Figures (a-c) show the order parameter, equation 3.16 resulting from simulations with  $\tau = 2, 4, 6$  futures respectively. Figures (a-c) insets show the time averaged size of the largest cluster as a fraction of  $N$ , a low value (approaching  $1/N$ ) indicates mass fragmentation which is usually associated with low order, a value of 1 indicates total cohesion. Error bars represent the standard deviation of 3 simulations, where agents are positioned uniformly at random in a box of side length  $L$  with initial number density is  $N/L^2 = 0.005$ , and orientations are drawn from a normal distribution with mean 0 and standard deviation of  $\Delta\theta^\circ$

Continuing with the order we will examine it more closely by varying  $N$  and  $\tau$ . Figure 4.4 shows the global order  $\phi$  and largest cluster size (inset) for the values  $\tau = 2, 4, 6$  and for  $N = 50, 100, 500, 1000$ . Clearly  $\tau = 2$  is not enough at

any occlusion values or system size, unsurprisingly. For  $\tau = 4$ , Figure 4.4 (b), we have the emergence of order, which reduces with system size due to fragmentations. The fragmentation rate is greatly reduced as occlusion is introduced, in particular we see a jump to a much larger order in the  $N = 500, 1000$  cases when occlusion is operating. This effect is diminished for smaller groups  $N = 50, 100$  giving way to a reduction in order, although for both this appears well within the error bars (one standard deviation here). For  $N = 500, 1000$  there appears to be some optimal values around  $0.25 \lesssim \mathcal{O} \lesssim 0.5$  for  $\phi$ , followed by reduced values for  $\phi$  past  $\mathcal{O} = 0.95$ . For  $N = 50, 100$  occlusion appears to reduce the order at all values, although in some cases the error bars are too large to be conclusive (e.g.  $N = 100, \mathcal{O} = 0.95$ ). For  $\tau = 6$  Figure 4.4 (c) the picture changes again, occlusion has almost no effect for  $N = 50, 100$ . Whilst for  $N = 500, 1000$  the increase from no occlusion to some occlusion appears reduced but still significant, and after  $0.25 \lesssim \mathcal{O}$  increasing occlusion has little to no effect. It appears then that occlusion has a stabilising effect on the order of large groups, perhaps on setting around  $N \approx 200, \tau \approx 4$  given the data in 4.3, and with  $\tau \gtrsim 6$  occlusion has little to no impact on order even with increasing  $N$ .

Examining the high order cases, by limiting the y-axis to  $\phi \geq 0.95$ , we see the small structure in that data in figure 4.5, we see that for  $N = 50$  occlusion actually causes a small decrease in the order. For  $N = 100$ , with increasing occlusion thresholds, the order declines, but all cases have a higher order than the no occlusion case.  $N = 500, 1000$  are more complicated showing a peak at an intermediate occlusion value, as mentioned before, followed by a similar reduction. As for significance of these data ( $\tau = 6$ ), we test how significant the increased average order is at the  $\mathcal{O} = 0.0$  case over the non-occlusive simulation against the null hypothesis that this increase is zero (i.e. the order is the same as the non-occlusive case). Applying these tests using a left tailed t-test (unequal variance form) we find p-values of  $\approx 0.238, 0.065, 0.010, 0.037$  for the cases  $N = 50, 100, 500, 1000$  respectively. This indicates no significant increase for  $N = 50, 100$  and significance at the 0.05 level for  $N = 500, 1000$  with the highest significance for  $N = 500$ . Comparing to the peak ( $\mathcal{O} = 0.25$ ) with  $N = 500, 1000$  gives p-values 0.010, 0.0272 respectively. Finally comparing to the final data points ( $\mathcal{O} = 0.95$ ) we obtain p-values 0.0172 (right-tailed), 0.381 (right-tailed), 0.013 (left-tailed) and 0.029 (left-tailed), meaning that for  $N = 50$  occlusion  $\mathcal{O} = 0.95$  lowered the order significantly compared to the non-occlusive case, for  $N = 100$  the test is inconclusive, and for  $N = 500, 1000$  the  $\mathcal{O} = 0.95$  case remains at a significantly increased order. The data then suggests that for sufficient  $\tau \gtrsim 4$  and  $N \gtrsim 1000$  occlusion plays a beneficial role, even when

severe, in maintaining an ordered group, but for small groups  $N \lesssim 50$  occlusions plays a negligible to obstructive role in maintaining ordered groups. We can also see from the inset for 4.4 (c) that the increase in order is likely a consequence of a decrease in fragmentation, indeed using another unequal variance t-test we find p-values 0.026 and 0.0036 that the reduction in fragmentation (increased  $\mathcal{C}$ ) between no occlusion and  $\mathcal{O} = 0.95$  for  $N = 500, 1000$  respectively. In fact note that this effect is large enough that fragmentation is almost completely eliminated in  $N = 500, 1000$ , for  $N = 1000$  in particular the average (using the data in 4.4 (c)) fragmentation rate (fragmentations per time step) was 0.005 for  $\mathcal{O} = 0.95$  up from 5.6 with no occlusion.

We also find in figure 4.5 (b) that the increased order persists for  $10^4$  time steps against the non-occlusive case for  $N = 500$  groups ( $n = 4$ ) with  $\tau = 6$

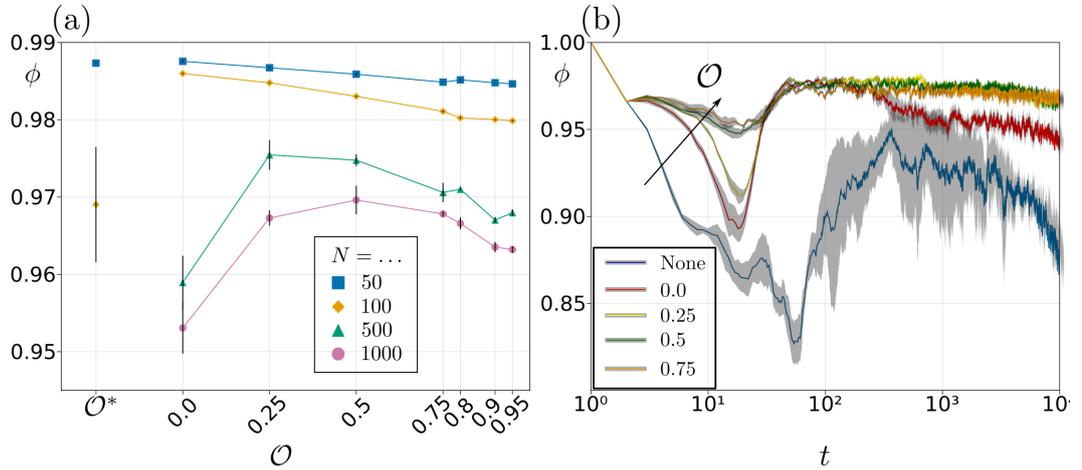


Figure 4.5: (a) Fine structure in the effect of occlusion for  $\tau = 6$ . Error bars represent the standard deviation of 3 simulations, where agents are positioned uniformly at random in a box of side length  $L$  with initial number density is  $N/L^2 = 0.005$ , and orientations are drawn from a normal distribution with mean 0 and standard deviation of  $\Delta\theta^\circ$ . (b) Long time (10,000 time steps) simulations for  $N = 500$ ,  $\tau = 6$ , individuals for varying occlusion thresholds, error bands indicate one standard error on the mean of 4 simulations. The increased order  $\phi$  is maintained at least up to  $t = 10,000$ , although all series continue to lose some order across time. Compare also with the inset of figure 3.5 (b).

Seeing a reduction in fragmentation and increasing order in large groups is one impact of accounting for occlusion, but the structure of the flock through the opacity, morphology and density of the group is another. Figure 4.6 details the impact of occlusion on morphology and opacity (insets) for  $\tau = 2, 4, 6$  and  $N = 50, 100, 500, 1000$ . Again  $\tau = 2$  is unsurprisingly indicative of large-scale

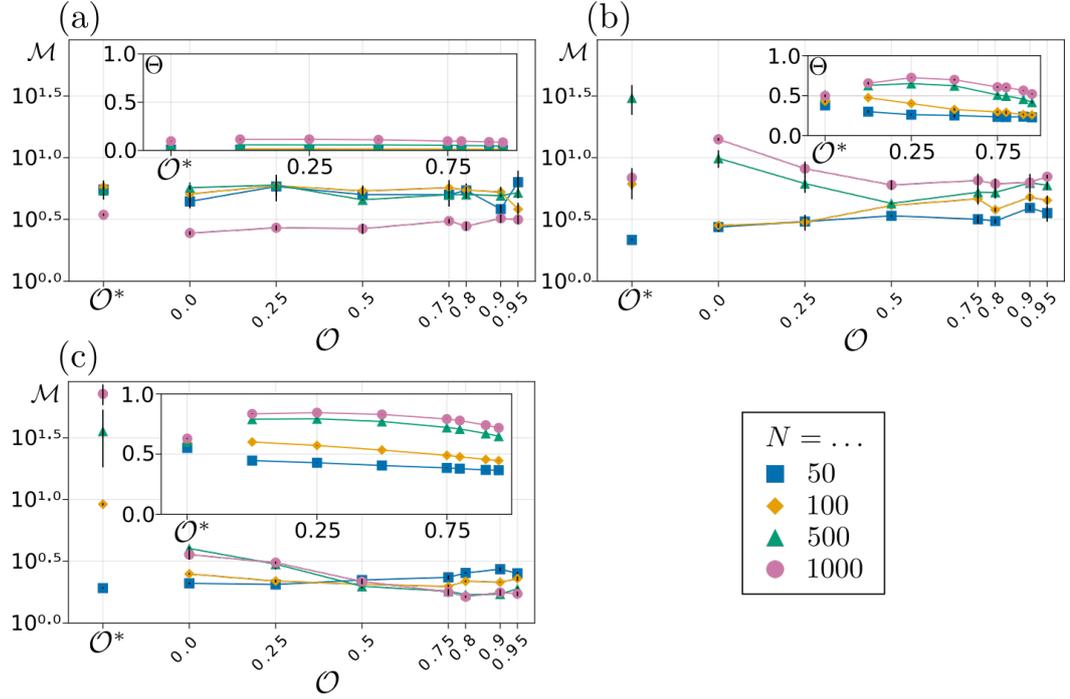


Figure 4.6: Figures (a-c) show the morphology parameter (equation 3.20)  $\log_{10}$  scale, the ratio of eigenvalues in equation 3.19, for the  $\tau$  values 2, 4, 6 respectively. Left to right on the  $x$  axis represents an increasingly strong occlusion threshold (from no occlusion). Figures (a-c) *insets* represent the opacity  $\Theta$  defined as the ratio of active sensors to inactive sensors. Error bars represent the averaging over 3 simulations, where agents are positioned uniformly at random in a box of side length  $L$  such that the initial number density is  $N/L^2 = 0.005$ , and orientations are drawn from a normal distribution with mean 0 and standard deviation of  $\Delta\theta^\circ$ .

fragmentation. For  $\tau = 4, 6$  both tell a similar story the morphology is at first quite extreme but with increasing occlusion tends towards a more circular group with the  $N = 50$  case being largely unchanged. Figure 4.8 (a) shows a typical example of a  $N = 50$  agent group, note the circular symmetry (non-occlusive case), whilst (b) and (d) detail an occlusive  $\mathcal{O} = 0.95$  and non-occlusive example with  $N = 1000, \tau = 6$  respectively. Note the extreme morphology in the non-occlusive case that gives way to a fully cohesive roughly circular flock. From figure 4.4 (c) inset that flocks do not fragment on average with these parameters and occlusion, but do show fragmentation at a large scale without. Accounting for occlusion results in a large-scale change in the morphology of the group and a large reduction in fragmentation in large groups. The change from a “line” formation to a roughly circular one is particularly striking. Finally, note also figure 4.8 (e) this example uses the same parameters as (b) but we do not discount future states subsequent to

a collision the morphology change is again drastic, we see no fragmentations (in this example), and the stretching is now along the direction of travel unlike (d) which is perpendicular to it. Similarly, the arrangement in (d) is quite stable whereas (e) undergoes quite extreme bending and flexing across time in a highly complex manner including breaking into two groups and then reforming.

Along with the substantial variance in morphology we see by accounting for occlusion, we also find significant changes in density. Consider figure 4.7 which shows the global ( $\rho$ ) and largest cluster  $\rho_C$  densities with (a,c) using  $\tau = 4$  and (b,d) using  $\tau = 6$ . For  $\rho$  we find at  $\tau = 4$  (4.7 (a)) that the density for  $N = 50$  decreases with increasing occlusion strength the opposite is true for the  $N = 1000$  case. We know that the order and morphology (circular) for  $N = 50$  are largely unaffected by occlusion (figures 4.4, and 4.6), and now we see that the density is decreasing with stronger occlusion. This means the flock is expanding roughly isotropically; the inter-agent distance is increasing. This makes reasonable sense, the agents are remaining cohesive but at a larger distance to retain information about the flock state in the presence of occlusion. For the large groups,  $N = 1000$ , we can understand the increasing density as a factor of the decreasing fragmentation we saw earlier. Moving to  $\tau = 6$  (4.7 (d)) the picture remains the same, except for  $N = 100, 5000$  which were marginal before. Now  $N = 500$  is approaching the same phenomenology as  $N = 1000$  and the same is true for  $N = 100$  and  $N = 50$ . For the largest cluster (4.7 (c,d))  $\tau = 4$  shows little effect given the error bars, whereas in  $\tau = 6$  we see decreasing density in all cases which are significant for  $N = 50, 100$  (p-values  $< 10^{-5}, 10^{-6}$  respectively) and not significant for  $N = 500, 1000$  (p-values  $\approx 0.065$ ) when comparing the non-occlusion case to the  $\mathcal{O} = 0.95$  case in an unequal variance t-test. Given the purpose of the largest cluster density is to remove fragmentations from the analysis the decrease makes sense according to the same argument for the  $N = 50$  case, i.e. the largest cluster expands in order to remain cohesive, but with a larger inter-agent distance so each agent retains information about the flock state.

### Understanding Occlusion by Ignoring Hypothetical Collisions

In the last section we have seen how occlusion appears to drive decreased fragmentation, decreasing largest cluster density, and a transition to circular morphology in large groups whilst only driving a decrease in density for small groups (all for sufficient  $\tau$ ). These changes may explain an increased order, that is if occlusion results in an increase in inter-agent spacing to promote visibility of the group as a whole, maybe this allows for better decision-making that targets order? This is possible,

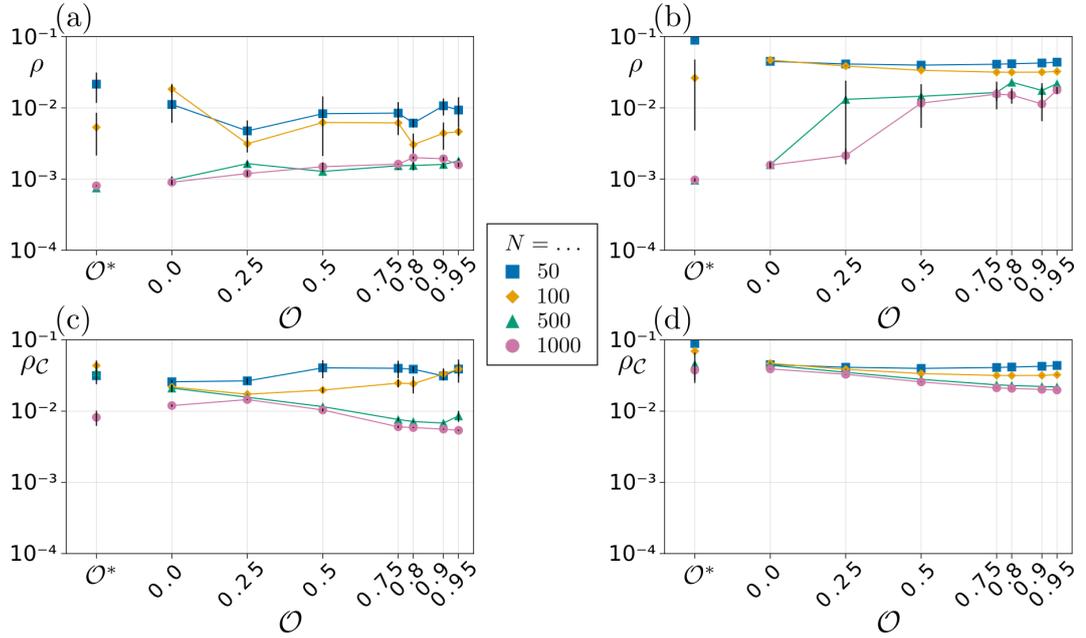


Figure 4.7: Density of the entire group  $\rho$  and of the largest cluster  $\rho_{c^*}$  for  $\tau = 4$  (a,c) and  $\tau = 6$  (b,d). Error bars are the standard error on the mean for three repeats.

but it is difficult to test this hypothesis. Another hypothesis is that the way in which occlusion aids large groups might be by reducing errors in each agent’s decision process. The ballistic trajectories of other agents are not accurate prediction of actual agent trajectories. *A priori* the effect of occlusion is two-fold (1) potentially reducing the number of ballistically modelled agents into the future (2) altering future visual states due to any ignored agents. The changes to the visual state are less easy to examine directly, but reducing the number of agents projected into the hypothetical future will also reduce the number of collisions an agent anticipates in large, relatively dense groups. An assumption in the decision process is that when an agent anticipates a collision, states subsequent to that hypothetical collision are not calculated. These hypothetical collisions may or may not be accurate predictions. If occlusion reduces the number of these hypothetical collisions (increasingly so in large groups), it is possible agents’ decision processes operate on more visual states. That is perhaps, occlusion, by eliminating ballistic trajectories, might increase order by giving agents more freedom in exploring the FST. We can test this hypothesis directly by counting states beyond hypothetical collisions. If the mechanism is due to collisions on the FST interfering with an agents decision process, we should see little effect of occlusion when discounting collisions.

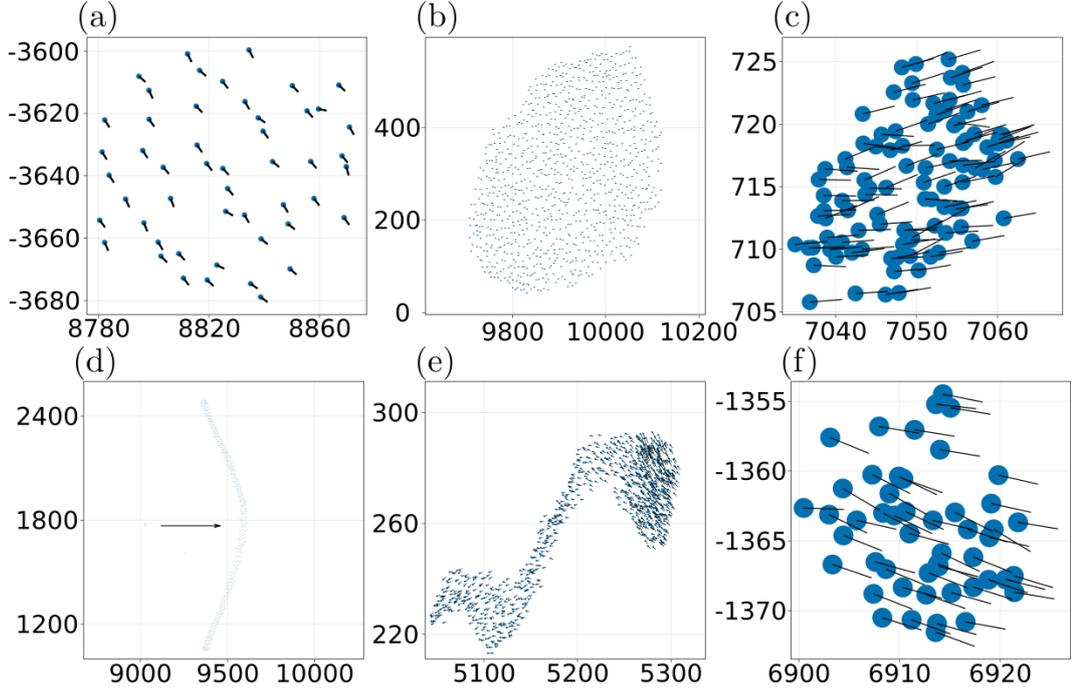


Figure 4.8: Snapshots for various parameters and group sizes showing the varied morphology we observe. (a) An  $N = 50$  group with  $\tau = 4$  futures and no occlusion. (b) A group of  $N = 1000$  agents with  $\tau = 6$  and an occlusion threshold of 0.95 (c)  $N = 100$  agents with an occlusion threshold of 0.5 and including states past future collisions (d)  $N = 1000$  agents with  $\tau = 6$  but occlusion off (e)  $N = 1000$  agents with  $\tau = 6$ , occlusion 0.95 and enumeration past collisions in the future states tree (collisions off) (f)  $N = 50$  and a occlusion threshold of 0.95. Note also the small fragmentation in (c) at approximate coordinates  $(x = 9000, y = 1800)$ . The arrow in (c) denotes the average heading direction of the group (by eye), observe the stretching in (d) is along the direction of travel (although this sometimes fluctuates in the simulation).

Figure 4.9 shows this data with (a) indicating collision branch cuts and (b) not cutting branches. If we compare figure 4.9 (a) directly to figure 4.9 (b) for  $N = 1000$  we find that the increase in the order data is significant for no occlusion (p-value 0.0055), marginal for  $\mathcal{O} = 0.0, 0.25$  (p-values  $\approx 0.03$ ), and inconclusive for  $\mathcal{O} = 0.5$  (p-value  $\approx 0.3$ ) when testing if the data with collision detection has a higher order (unequal variance t-test) data point by data point, but the reverse becomes significant for  $\mathcal{O} = 0.75, 0.8, 0.9, 0.95$  with p-values of orders  $10^{-6}, 10^{-8}, 10^{-8}, 10^{-7}$  in each case. That is the data with collision detection and high occlusion has a higher order. However, for the largest cluster size (insets) we find that the data with collision detection has higher average largest cluster size for  $\mathcal{O} = 0.0, 0.25$  (p-values

0.0014,  $1.3 \times 10^{-5}$  respectively), marginally so for  $\mathcal{O} = 0.5$  and no occlusion (p-values 0.013, 0.018) but is inconclusive otherwise. This data suggests that occlusion does play a role in promoting order whether we cut FST branches at collisions or not. But for low occlusion thresholds not cutting FST branches appears to suppress the largest cluster size with high statistical significance, and marginally so for suppressing order. In fact the lag we see for  $N = 1000$  in  $\mathcal{C}$  and  $\phi$  at  $\mathcal{O} = 0.25, 0.5$  when comparing (b) to (a) is some evidence for our hypothesis that occlusion is (at least partially) working via suppressing erroneous collisions on the FST. Since, when continuing past collisions on the FST this suppression has no effect, a significantly higher occlusion threshold is needed for  $N = 1000$  to order ( $\mathcal{O} \gtrsim 0.8$  as apposed to  $\mathcal{O} \gtrsim 0$  in figure 4.4).

We can also remark on the opacity and morphology. Without collision branch cuts the opacity should be understood as a product of greatly suppressed collision avoidance, and we find average opacity around  $\Theta \approx 0.9$  decreasing across occlusion values to around  $\Theta \approx 0.8$ . For example observe figure 4.8 (c,e,f) which are cases without the FST collision branch cutting, the overlapping means visual states are largely filled; at such an extreme, where two agent overlap, visual projection will leave large portions of an agents visual field obscured, hence the increase in opacity. Density is similar, recalling density units are scale by agent radius, at low occlusion values it is in excess of 1. Density decreases with occlusion in all cases to around  $\rho \approx 10^{-0.5}$  for  $N = 500, 1000$  and  $10^{-0.25}$  for  $N = 50, 100$ . The morphology is more complex in the large groups where unlike in the  $N = 50, 100$  cases where we see a denser circle, we find stretched and highly unstable examples such as figure 4.8 (d) which as we remarked upon earlier varies greatly across time (bending, splitting and reforming).

## 4.2 The Role of Noise in the Cognitive Process

Adding occlusion, in effect, deterministically changes the visual state of an agent  $\psi_i$ . This, in fact, deterministically changes the information an agent processes to make decisions. In this section we will examine the role of noise on the dynamics. Specifically we will examine (1) a Vicsek-like noise process by which agents make decisions deterministically but suffer a degree of noise on the actualised orientation update (2) a ‘‘cognitive’’ noise process by which agents model others as moving ballistically with an additional degree of noise on rotation and/or speed updates, such that the average projected future path is ballistic. With both noise processes we hope to understand how the dynamics change with the inclusion of noise processes.

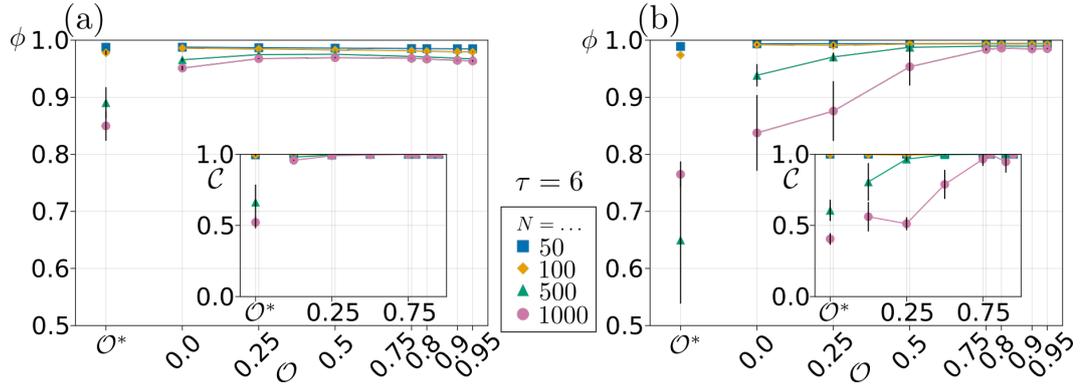


Figure 4.9: Order and largest cluster size for different groups sizes with  $\tau = 6$  for (a) collision branch cuts on the FST (b) no collision branch cuts on the FST. Data for the largest cluster sizes. Error bars indicate one standard error on the mean computed over 5 replicates for time steps 500 to 1000.

#### 4.2.1 Vicsek-Like Noise

In a Vicsek-like noise process we mean applying orientational noise after an agent has made its decision. Using equations 3.7 and 3.8, we can apply noise by re-writing the equations of motion as

$$\mathbf{x}_i^{t+1} = \mathbf{x}_i^t + \mathbf{v}_i^{t+1} \Delta t, \quad (4.3)$$

$$\mathbf{v}_i^{t+1} = v_{\alpha_i^t} R(\theta_{\alpha_i^t} + \eta_i^t) \hat{\mathbf{v}}_i^t. \quad (4.4)$$

That is agents select actions at the present time step  $t$ ,  $\alpha_i^t$ , and apply them to their velocity vector  $\mathbf{v}_i^t$ . In the presence of a Vicsek-like noise process the random variable  $\eta_i^t \sim \mathcal{N}(0, \eta^2)$  is applied to the selected orientation. This random variable is a mean zero Gaussian random number with variance  $\eta$  and zero correlation across time  $t$  or agent index  $i$ . The expectation is that low values of  $\eta$  will produce an ordered regime and high values will result in random dynamics dominated by the orientational noise term.

We expect there to be some critical value of the noise  $\eta$  separating an ordered and disordered phase. Vicsek *et al* find an almost completely disordered group (with  $N=10000$  agents) with values of noise  $\gtrsim \pi$  rad and a consistently ordered phase with noise  $\lesssim 0.5$  rad (note the use of a random increment sampled from a uniform distribution in [4]). Therefore, we might expect a comparable completely disordered phase with noise around  $180^\circ$  and an ordered regime ending somewhere around the  $\eta \approx 30^\circ$  mark. Our equation of motion is rather different to Vicsek *et al.*, meaning we do not expect any exact replication of results.

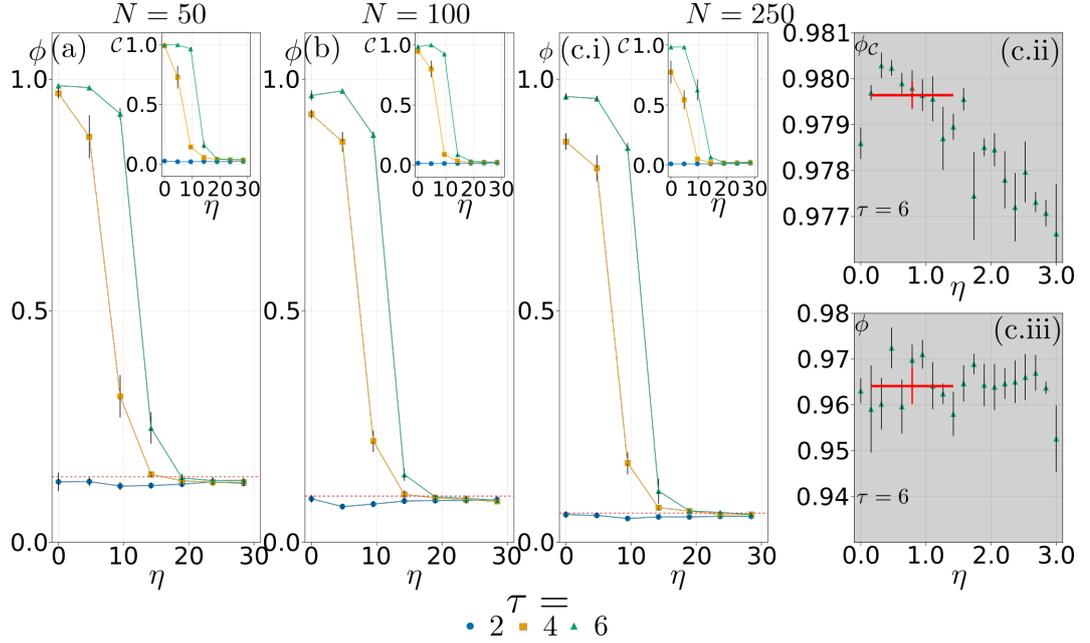


Figure 4.10: Effect of post-decision orientational noise on global order  $\phi$ , largest cluster size  $C$  (inset) for  $N = 50, 100, 250$  group sizes (a,b,c.i) and  $\tau$  values 2, 4, 6. In each case a rotational additive noise term is applied directly to the output of the FSM decision process sampled from a normal distribution with variance  $\eta^2$ , statistics are calculated after a burn-in of 500 time steps up to time step 1000.  $\tau = 2$  is generally not enough for ordered motion to arise, with  $\tau = 4, 6$  we see a transition from high order  $\phi \sim 0.98$  to disorder  $\phi \sim 1/\sqrt{N}$  (dashed line) with a slightly more robust ordered phase with  $\tau = 6$  compared to  $\tau = 4$ . We see from  $C$  that this is driven by large scale fragmentation. Error bars indicate 1 standard error computed from 3 replicates. We also plot the global order and order of the largest cluster ( $\phi_C$ ) for low noise values in the  $N = 250$  case (c.ii,c.iii).

### Order Increase With Small Noise Values

To test this we observe the data shown in figure 4.10 for group sizes  $N = 50, 100, 250$  across the  $\tau$  values 4, 6. As expected we do find the ordered regime around small noise values  $\eta \approx 0$  and ending with complete disorder ( $\phi \approx 1/\sqrt{N}$ ) by about  $\eta \approx 30^\circ$ . Figure 4.10 (c.i) shows an interesting peak in the order for  $\tau = 6$  at non-zero noise whereas for  $N = 50, 100$  the order is monotonically decreasing. We examine this data point further by plotting in figure 4.10 (c.ii) and (c.iii) the order and largest cluster order  $\phi_C$  for small noise values up to  $\eta = 3$ . We perform an upper tailed t-test to judge the significance of the difference in the mean order for the zeros noise case,  $\mu_0$ , and the cases  $0 < \eta \leq 1.5$  averaged,  $\mu$ , (shown by the red dashed lines), using the null hypothesis  $\mu_0 > \mu$ . This was done for both  $\phi$  and  $\phi_C$ , collecting enough

samples for the  $\eta = 0$  case to match the (63) total number of combined samples in the noise case, i.e. all data for the scatter points  $0 < \eta \leq 1.5$ .

For  $\phi$  we reject the null with a p-value  $< 10^{-13}$  (n=63). For  $\phi_C$  we reject with p-value  $< 10^{-13}$ . The tests confirm that for noise values  $0 < \eta \leq 1.5$  the average order (and order of the largest cluster) is higher than the case without noise,  $\eta = 0$ , with high significance.

The significance test for  $\phi$  suggests that the order does increase for small but non-zero noise values. But we also find an increase in the largest clusters and  $\phi_C$ . This suggests that the ordering effect of a small addition of post-decision noise, is not just due to increased cohesion. For the smaller group sizes we already know that at the zero-noise case groups of  $N = 50$  and  $N = 100$  remain stable and cohesive with  $\tau = 6$ , but that  $N = 250$  groups suffer persistent fragmentations, this is likely why we see the effect in  $N = 250$  but not the two smaller group sizes i.e. they are already cohesive so see little benefit. From our analysis of the role of occlusion we know that for large groups ( $N = 500, 1000$  for certain) occlusion promotes order in a similar fashion by reducing fragmentation and thereby increasing cohesion. Our hypothesis was that, since the ballistic modelling assumption is merely an approximation for true agent future states, the errors in each agent's cognitive process build with  $N$ . Occlusion has a "culling" effect on how many agents any particular agent models based upon vision. This reduces these errors leading to more robust decision-making. One explanation then is that, again for large groups suffering fragmentation, small amounts of noise improve decision-making based upon ballistic assumptions.

#### 4.2.2 Cognitive Noise

The cognitive process depends on the function mapping the current, or a previous, estimate of a future state into the next prediction an agent makes of the future which we took (implicitly) as ballistic previously. In particular, we will keep the function as mapping the other agents into ballistic trajectories on average by adding a noise process at each step. Now we add noise to the hypothetical trajectories of other agents.

We add both rotational noise  $\eta_{ij}^t$  with the same statistics as  $\eta_i^t$  in the Vicsek-like noise process but now also zero correlation across both indices  $ij$  and time. As well as a Gaussian speed noise  $\zeta_{ij}^{t'}$  with zero mean, variance  $\zeta^2$ , and zero correlation across  $ij$  and time. The random variables are added to hypothetical agent decisions such that  $p_{\beta_j^{t'}} = v_0 + \zeta_{ij}^{t'}$ , and  $\theta_{\beta_j^{t'}} = \eta_{ij}^t$ .

On average the model will apply a speed  $v_0$  and an identity rotation, i.e. ballistic future states, but now with two noise parameters to vary uncertainty in

each agent’s cognitive process. These parameters are  $\theta_\theta$  and  $\eta_v$  for rotation and speed standard deviations respectively. We plot this data in figure 4.11, in the same fashion as for the Vicsek-like noise process.

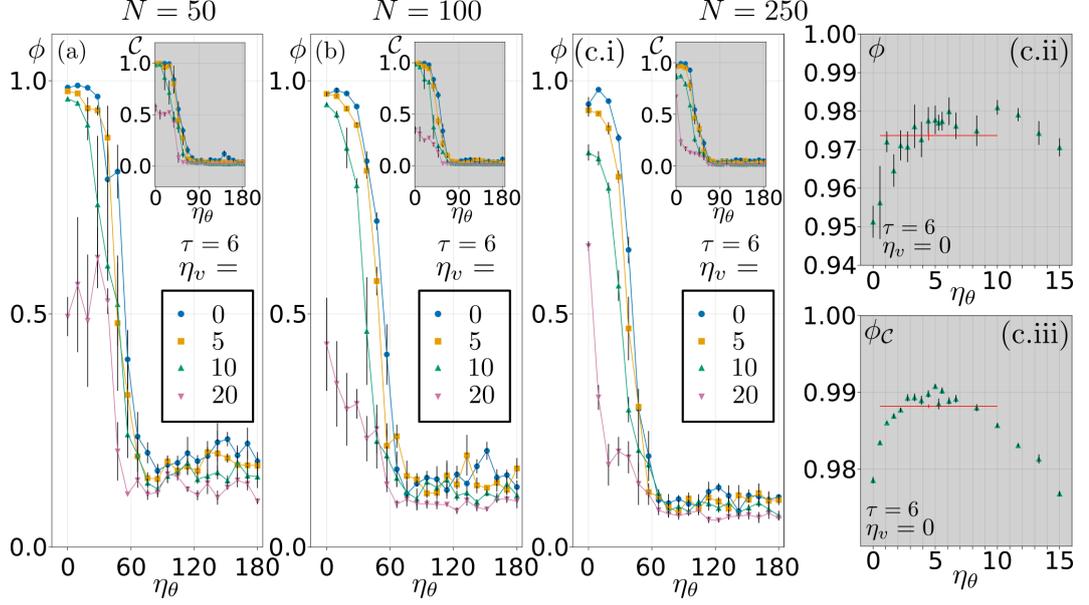


Figure 4.11: Impact of increasing “cognitive” noise on global order  $\phi$  and largest cluster size  $\mathcal{C}$  for groups of sizes  $N = 50, 100, 250$  (a,b,c.i), for each a burn-in of 500 was used with statistics calculated up to time step 1000. Figures c.ii and c.iii detail the effects with low rotational noise  $\eta$  and 0 speed noise  $\zeta$ . Cognitive noise is noise applied to each agent’s model of other agents one or both of orientational noise (Gaussian and white with variance  $\eta^2$ ) or speed noise (Gaussian and white with variance  $\zeta^2$ ) applied to the base ballistic model. i.e.  $\eta = \zeta = 0$  implies a ballistic model and non-zeros values indicate diffusive trajectories/ increasing uncertainty.

In each case however we only consider  $\tau = 6$  and instead vary the value of  $\eta_v$  and  $\eta_\theta$ . For no speed noise  $\eta_v = 0$ , there is an increase in global order. This appears generic for panels 4.11 (a,b,c) occurring around  $\eta_\theta = 15^\circ, \eta_v = 0$ , i.e. for all  $N = 50, 100, 250$ . We hypothesise this indicates a cognitive process with noise (as apposed to ballistic) is a better predictor of actual future states leading to greater order, this effect appears to disappear at  $\eta \gtrsim \Delta\theta = 15^\circ$ . We see a transition from high order to a disordered phase with increasing noise in both  $\eta_v$  and  $\eta_\theta$ . Note however that the plateau in  $\phi$  is slightly higher than  $1/\sqrt{N}$ , this is due to the fact that the fragmentations driving these transition lead to the formation of “trains” of agents (at  $\tau = 6$  shown here), i.e. groups of agents moving as a string one following the other. These trains are characterised by high internal order but with a greatly increased inter-agent spacing compared to marginally opaque groups, meaning they

are not detected as clusters. We expect with increased run time ( $T = 1000$  here) these trains would fragment.

For the global order  $\phi$  an upper tailed t-test on the difference of the means,  $\mu_0$ , in the noise-free case ( $\eta = 0$ ) and,  $\mu$ , the mean of the data with noise ( $0 < \eta \leq 10$ , dashed red cross) rejected the null hypothesis  $\mu_0 > \mu$ , p-value  $< 0.00016$  with  $n = 80$ . The same t-test for  $\phi_C$  at low noise was significant for the same range  $0 < \eta \leq 10$ , p-value  $< 10^{-45}$ . These results confirm that the increase in order  $\phi$ , with orientational noise in hypothetical agent trajectories, and largest cluster order  $\phi_C$  are both highly significant.

The data in figure 4.11 show that a small value of orientational noise in the cognitive process of agents, promotes order and largest cluster order, over a purely ballistic assumption. The peak increase appears to be around  $\eta = 5^\circ$  in the largest cluster order for  $N = 250$  and disappears by  $\eta = 15^\circ$ . We can compare this approximate peak with the characteristic orientational noise found in the ballistic model. That is we can approximate  $\langle \hat{v}_i^t \cdot \hat{v}_i^{t+\delta} \rangle \approx \langle 1 - \frac{1}{2}\vartheta^2 \rangle$  for a discrete time lag  $\delta \in \{1, 2, 3, \dots\}$ . Using this we can find  $\langle \vartheta^2 \rangle$ . If we interpret this as the result of summing  $\delta$  independent and identically distributed normal random variables with mean 0 and common variance  $\sigma^2$  we find for  $\delta = 6 = \tau$  and  $N = 250$  that  $\sigma = \sqrt{\frac{\langle \vartheta^2 \rangle}{6}} \approx 5.79^\circ$  (average of 10 repeats, standard deviation 0.022). This value is slightly larger than the approximate peak for  $\eta$ . It is possible then that the optimal noise value is one which emulates the natural fluctuations in the model without noise, which would be self-consistent.

### 4.3 Conclusions

We examined the effect of perturbing the model, first deterministically by introducing occlusion and secondly by introducing noise processes to agent actions and the cognitive processes. The occlusion model we introduce is based upon simple geometry (for circles) and defines a natural interaction network for our agent-based model. By varying the free parameter, occlusion strength  $\mathcal{O}$ , we found that the SCE model was to a first approximation, and low  $N \lesssim 100$ , robust to occlusion, which makes sense biologically due to the presence of occlusion in the real world, but at high  $N \gtrsim 100$  we found significant impact for the better. That is we found occlusion can actually promote order in our model. We hypothesised this was due to an increasing error from ballistic models as  $N$  increases, being offset by occlusion naturally ignoring larger quantities of agents as  $N$  increases. To test this hypothesis we found that by not cutting branches prematurely on the FST, when a collision

is detected between a ballistic model and an agent’s hypothetical future position, largest cluster size and order were initially reduced with statistical significance, but then both increased compared to cutting branches as occlusion became more severe. This effect occurred for  $N \gtrsim 500$ . The discounting of agent providing a benefit in large groups could be interpreted as a sign of diminishing returns, i.e. it is unnecessary or even harmful to model every member of the group. Given the success of topological rather than metric models of collective motion, and the evidence for the former in nature [39], it is perhaps unsurprising that we find this effect in our model.

Finally, we examined two phase transitions in our model governed by the introduction of noise. Firstly following closely that used in the Vicsek model, we introduced a re-orientational additive noise. Rotational noise was applied directly to agent dynamics. Secondly we replaced the ballistic model, used by agents to project hypothetical future states, by a Stochastic model. In this model agents select actions, speeds and rotations, sampled from mean-zero Gaussian distributions. In both cases we found a transition from an ordered to disordered state with increasing noise strength. Surprisingly we found that low values of noise actually increased order, both overall and realised in the largest cluster. For the added orientational noise in the cognitive process we interpreted this similarly to the impact of occlusion that is a noisy model, deviating slightly from a mean ballistic model, is more consistent with the actual output of the model in the deterministic case. In fact, we found a rough agreement between orientations fluctuations  $\langle \Delta(\theta_i^t)^2 \rangle \approx 5.79^\circ$  and the optimal rotational noise parameter in the ballistic model ( $\eta \approx 5^\circ$ ).

The two forms of perturbations we use, occlusion and noise, are not a complete list of possible confounding factors in our model. Future research could explore the impact of, for example, physical obstacles. That is how does a group react when obstacles occlude large portions of the group? Since agents within our model, project their flockmates into the future and seek maximum entropy of visual states. *A priori* it is possible when faced with an obstacle, the group at large moves to one side of the obstacle or the other. This might occur if a majority of the flock happen to be heading to one side of the obstacle over the other. An individual on the least favoured side, projecting others ballistically, may select to turn towards the favoured side of the group for maximum visual diversity in the future. On the other hand if an obstacle would split a flock exactly into two parts, the group may continue roughly ballistically.

## Chapter 5

# Monte-Carlo Sampling of Future Paths

In this chapter we will first discuss how FSM can be recontextualized as a Monte-Carlo tree search, a random search on a future states tree of finite size.

Next with a basic Monte-Carlo process on a finite action space we will see how we can extend to continuous action spaces, that is instead of  $|\mathcal{A}| < \infty$  discrete actions (in particular  $|\mathcal{A}| = 5$  as we used in previous chapters), we relax the set  $\mathcal{A}$  to some continuous region or distribution which instead of enumerating fully we sample from, or more specifically we sample paths up to the time horizon  $\tau$ .

We will then compare these approaches to the explicit enumeration methodology and discuss in the context of existing literature.

### 5.1 Monte-Carlo Future State Maximisation on a Finite Action Space

Up to this point FSM has been considered by way of explicitly enumerating all possible future states, given an action space  $\mathcal{A}$ , for all agents. This is computationally tractable using both a discrete action space, such as the default five actions selecting three speeds or two orientation increments:  $v_1 = v_4 = v_5 = v_0$ ,  $v_2 = v_0 + \Delta v$ ,  $v_3 = v_0 - \Delta v$  and rotations  $\theta_1 = \theta_2 = \theta_3 = 0$ ,  $\theta_4 = -\Delta\theta$  and  $\theta_5 = \Delta\theta$  which we have used up to now, and a modest number of future times to search ( $\tau \leq 6$  typically). Increasing the size of the actions space (up to and including an infinite size) and increasing the number of future states both lead to an exponential increase in computation time for an agent to reach a decision. This is clear from the number of paths in the future states tree,  $|\mathcal{A}|^\tau$ . Not only is this a computational problem, but

when considering real-world animal systems it is not plausible that a bird, insect or other animal could envisage the exponentially increasing number of future states without bound.

One way to reach greater depths is to sample futures rather than exhaustively enumerate them. Sampling future states up to time  $\tau > 0$  (0 being the present time), can be thought of as sampling a sequence of actions, a path, of length  $\tau$  from some distribution. In the explicit enumeration case, the sampler can be thought of as drawing from a uniform distribution (or any as long as the probability of sampling any particular path is not zero) over future paths without repeats, and taking exactly  $|\mathcal{A}|^\tau$  samples, one for each path. In general a sample path is defined as the sequence of actions

$$\mathbf{s}_{ik} = \{\beta_{ik}^{t'}\}_{t'=1}^\tau : \beta_{ik}^{t'} \in \mathcal{A}, \quad (5.1)$$

for an agent  $i$  which in turn are index by  $k$  which ranges up to some pre-defined number of samples  $S$ . Such that  $\mathbf{s}_{ik} \in \mathcal{A}^{N \cdot S}$ . Clearly enumerating paths and sampling without replacement will result in the same outcome i.e. all paths are sampled once. We could of course pre-define some distribution over paths, or define a distribution which changes as the number of samples progresses. We will examine some of these cases in this chapter.

### 5.1.1 Sampling Actions

Since sampling a path in the FST is equivalent to sampling a sequence of actions as defined in equation 5.1. We can think of the sampling process at the path or action level. At the path level we define some distribution,  $\mathbb{P}(\mathbf{s}_{ik})$ , over entire paths and sample these paths and the states associated with them. At the level of actions we imagine a sample beginning at the root of the FST and selecting an action  $\beta_{ik}^1 \in \mathcal{A}$  by some probability  $\mathbb{P}(\beta_{ik}^1 = \beta)$  at time  $t = 1$ , and at time  $t = 2$ ,  $\beta_{ik}^2$  is selected with probability (possibly dependent on  $\beta_{ik}^1$ )  $\mathbb{P}(\beta_{ik}^2 | \{\beta_{ik}^1\})$ , etc. This then, using the law of total probability, generates a path which is sampled with probability

$$\mathbb{P}(\mathbf{s}_{ik}) = \mathbb{P}(\beta_{ik}^1) \mathbb{P}(\beta_{ik}^2 | \{\beta_{ik}^1\}) \mathbb{P}(\beta_{ik}^3 | \{\beta_{ik}^1, \beta_{ik}^2\}) \cdots \quad (5.2)$$

$$= \prod_{t'=1}^\tau \mathbb{P}(\beta_{ik}^{t'} = \beta | \{\beta_{ik}^{t''}\}_{t''=1}^{t'-1}). \quad (5.3)$$

We use the convention that the empty sequence is identified with  $\{\beta_{ik}^t\}_{t=1}^0 = \emptyset$ . The nature of the distribution  $\mathbb{P}(\beta_{ik}^{t'})$  and its possible dependence on the path's

previous selections  $\{\beta_{ik}^{t''}\}_{t''=1}^{t'-1}$  defines the sampling scheme. For example, we can take  $\mathbb{P}(\beta_{ik}^t)$  to be Markovian and uniform over  $\mathcal{A}$  to obtain the sampling scheme corresponding to uniform path selection *with replacement*, and for sampling path *without replacement*,  $\mathbb{P}$  would have to also depend on the prior sampled paths,  $\mathbf{s}$ , in a way that is complicated to write down in explicit analytic form. In the uniform case, we can instead think of the sampler as selecting entire paths uniformly from those paths not already selected. That is

$$\mathbb{P}(\mathbf{s}_{i(k+1)} | \cup_{l=1}^k \mathbf{s}_{il}) = \begin{cases} \frac{1}{|\mathcal{A}|^{\tau-k}} & \mathbf{s} \notin \cup_{l=1}^k \mathbf{s}_{il} \\ 0 & \mathbf{s} \in \cup_{l=1}^k \mathbf{s}_{il} \end{cases}. \quad (5.4)$$

Dependence on the previously sampled paths  $\mathbf{S}^m$  is also useful if we want our sampling scheme to take advantage of the current state of information about the FST. That is we could imagine selecting actions based upon the current estimate of entropy along subsequent nodes given that action selection, or some other property such as prioritising branches with the most un-sampled states. As such we could use equations of motion, equation 3.7 and 3.8, to calculate the states implied by each path  $\mathbf{s}$  and additionally condition selection probabilities on these states.

### The naïve case

Let us begin by examining a base case. Here the simplest form of equation 5.3 is to drop dependence on the prior actions along the path being sampled,  $\{\beta_{i''}\}_{i''=1}^{t'-1}$ , the previously sampled path  $\mathbf{s}$  and their states. This leaves us with some distribution  $\mathbb{P}(\beta)$  over the actions  $\beta \in \mathcal{A}$  which is fixed for any agent and any path. In this situation any form for  $\mathbb{P}$  that is not uniform will bias sampling, so we take  $\mathbb{P}(\beta) = \frac{1}{|\mathcal{A}|}$  as the most basic case to examine.

The main test we wish to make is how many samples,  $s$ , does it take before the decisions returned based on samples approaches the decisions made on full enumeration. Or we might also ask, how many samples are needed to reach an ordered phase of motion? If by chance the sampling process samples every path in the tree exactly once the output will be identical, all else being equal in terms of implementation. But how likely is that to occur?

We can write down the process of sampling in this base case analytically. First note the sampling process can be translated to a set of walkers (samples) which descend a complete  $|\mathcal{A}|$ -ary tree (from the root node to the leaves) to depth  $\tau$ . In particular each walker, at the root or any internal node, will move to one of the available children with a uniform probability independently of the other walkers,

importantly always descending the tree, and upon reaching a leaf node terminates. Figure 5.1 shows an example of four walkers (four samples) traversing the complete binary (2-ary) tree of depth 2. In this particular case the sampling process reached each leaf, notice how reaching each leaf corresponds to full enumeration of all paths from the root node in the tree to the leaves. To formalise this process

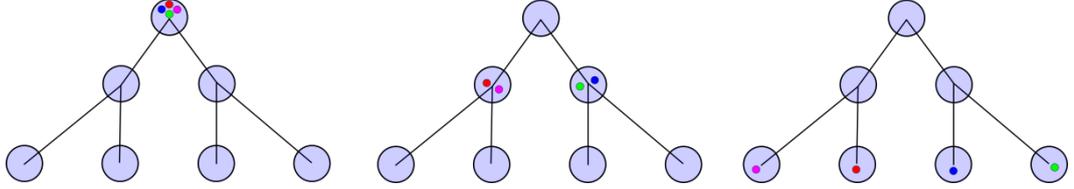


Figure 5.1: A visualisation of four walkers descending a binary tree of depth two. In this case each leaf has received at least one walker, the probability of the pictured event is  $\approx 0.09375$ , in contrast if the tree was only depth one (so the third row is cutoff and the second row becomes the leaves) then the probability is 0.875 of all leaves being visited.

there are  $w$  distinct walkers  $W_i^t$ , for  $i = 1, 2, \dots, w$ , where the notation  $W_i^t$  denotes walker  $i$ 's position (a node in the tree) at step  $t \in 0, 1, \dots, d - 1$  such that  $W_i^0 =$  the root node  $\forall i$ . The walkers traverse a complete  $n$ -ary tree of depth  $d$  (where the root node is counted as depth zero). The stochastic process a walker undergoes is Markovian and independent of the other walkers and lasts  $d$  steps  $t = 0, 1, \dots, d - 1$ . At each step  $t < d - 1$  a walker  $W_i(t)$  is at an internal node (or the root node)  $v$ , with exactly  $n$  children identified by  $u_1, u_2, \dots, u_n$ . The transition probability is

$$\mathbb{P}[W_i(t + 1) = u_k | W_i(t) = v] = \frac{1}{n} \quad \forall k = 1, 2, \dots, n, \quad t < d - 1. \quad (5.5)$$

The question is to work out, for a given  $n$  and  $d$ , how large  $w$  must be to insure all paths are enumerated with a certain probability.

### Probability of Visiting All Leaves

Since the process takes place on a complete  $n$ -ary tree visiting all nodes is equivalent to visiting all leaf nodes, using this we can ignore the tree and think of the problem as distributing  $w$  distinct balls into  $n^d$  distinct bins. The balls being walkers and the bins being the leaf nodes.

For  $w$  distinct balls and  $n^d$  distinct bins there are exactly  $(n^d)^w$  ways to distribute them in total, including empty bins. For non-empty (but indistinct) bins the num-

ber of ways to distribute the balls is expressed by the Stirling number of the second kind [114] (for  $n$  balls and  $k$  bins)

$$S_2(n, k) = \frac{1}{k!} \sum_{i=0}^k (-1)^i \binom{k}{i} (k-i)^n, \quad n \geq k. \quad (5.6)$$

Equation 5.6 can be converted to distinct bins by multiplying by a factor of  $k!$ . Combining these facts for the  $w$  walkers and  $n^d$  leaf nodes, the probability that all leaf nodes are non-empty after the stochastic process of equation 5.5 is given by

$$\mathbb{P}(\text{All leaves visited}) = \begin{cases} \frac{S_2(w, n^d) n^{d!}}{(n^d)^w} & w \geq n^d \\ 0 & \text{otherwise} \end{cases}. \quad (5.7)$$

### Infeasibility of the naïve Approach

Figure 5.2 (a) shows the order of a group following the Naïve approach (main plot) and (inset) equation 5.7 alongside a Monte-Carlo simulation of the walker process given by the transition probability 5.5. Clearly, in this naïve case the number of samples needs to be unreasonably large considering the that at  $\tau = 4$  in full enumeration we have 625 paths, but for the naïve sampling approach we find that 10000 samples is still not enough for a stable high order flock. The reduced order here is due to the flock fragmenting into multiple smaller, but highly ordered subgroups. There is significant variance in the count of these groups, and we see this reflected in the largest cluster size 5.2 (b). However, it is clear the order roughly converges much faster than the idealised case. This is quite encouraging. It means we likely do not have to fully sample the tree to achieve ordered motion.

#### 5.1.2 Sampling Paths Without Replacement

Sampling with replacement clearly is suboptimal for a deterministic FST as is the case here since re-sampling one path will always return an identical result. We say a deterministic FST given that in theory we could have a case where the FST changes on each sample, stochastically. In this noisy case one could re-sample the same sequence of action but obtain slightly perturbed visual states. For now, we will continue with a deterministic tree, in which case it makes sense to examine the sampling process without replacement. The hypothesis being that we achieve convergence to the full enumeration case much earlier than the naïve case, in fact we should obtain identical output when sampling enough paths (without replacement) to fully enumerate the FST at a given  $\tau$ .

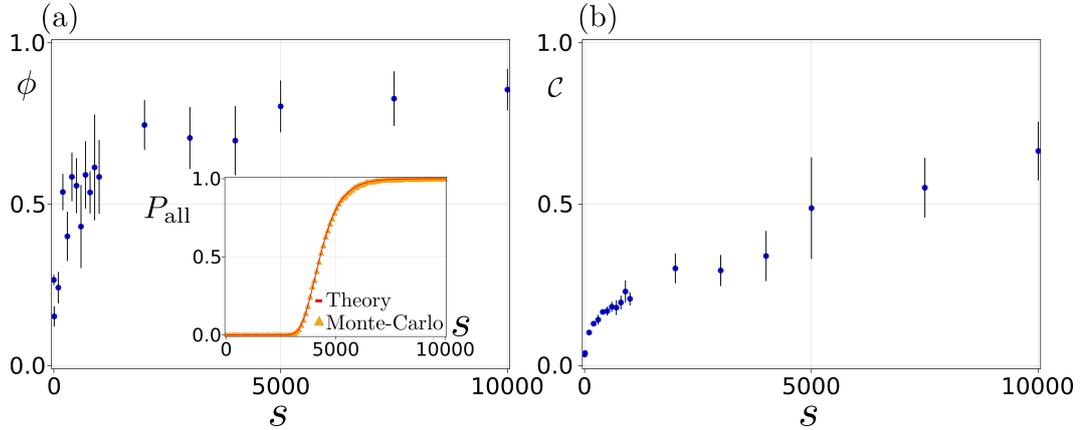


Figure 5.2: (a, Main plot) Global order increases with number of samples drawn with replacement. The order does not reach the same as in explicit enumeration due to sample degeneracy. Here  $N = 50$  agents, and  $\tau = 4$ , were simulated with differing numbers of walkers, but identical starting conditions. Data points and error bars correspond to the mean of 5 simulations, and one standard deviation of 5 simulations respectively. (a, Inset) Probability of all paths being sampled in a 5-ary tree of depth 4 with  $S$  walkers, this is the form of the future states tree for the default FSM parameter set. For reference 99% probability is achieved at approximately 6894 walkers. Theory indicates evaluating equation 5.7, and the simulations are by simulating the process according to equation 5.5.

To do this we simply compute the list of all possible paths, given  $\mathcal{A}$  and  $\tau$  and refactor the implementation to choose among possible paths at random uniformly from paths not yet sampled, by eliminating each path from the possible distribution once sampled. Doing this we can state with certainty that after exactly  $|\mathcal{A}|^\tau$  samples the output from the explicit algorithm and this path sampling algorithm without replacement are identical; the decision process functions on identical input. We are interested in two properties (1) how does the order of a group behave as a function of the number of paths sampled? (2) If we limit the number of samples to (possibly) cover depth  $\tau$  in the tree,  $S = |\mathcal{A}|^\tau$  but allow each path to continue to a higher depth  $\tau' > \tau$  how do these extra visual states effect the order? Question (1) serves largely as a check on the implementation of the sampling but will give us interesting information about how much of the FST needs to be known, for example if we eliminate a single path from the explicit enumeration does this dramatically effect order? Question (2) is more interesting because fully enumerating at depth  $\tau$  has a certain computational cost which scales as a power (in  $\tau$ ) of the number of actions with a coefficient that is the time taken to simulate one path. If however we can see a benefit (i.e. increased order and stability) from sampling say  $5^4$  paths but

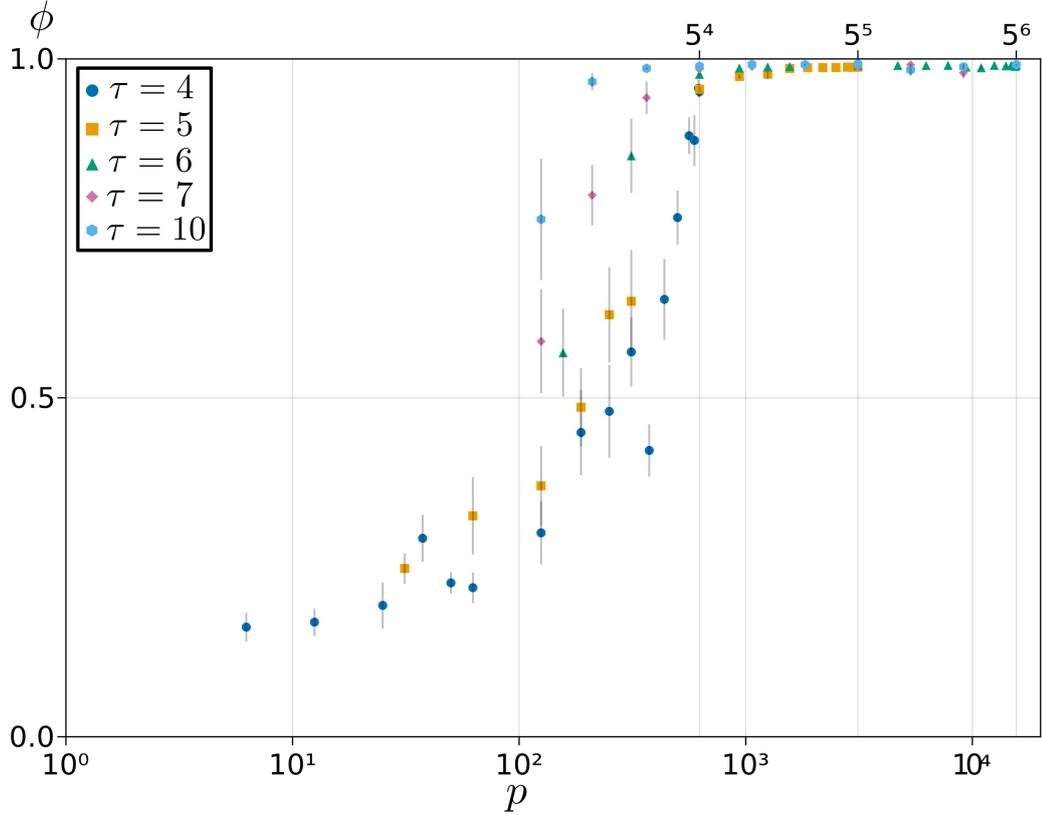


Figure 5.3: Global order,  $\phi$ , as a function of the number of paths (in the FST) sampled,  $p$ , for time horizons  $\tau = 4, 5, 6$ . Note the extra x-axis (top) marking  $S = 5^4, 5^5, 5^6$  show necessary but not always sufficient (depending on  $\tau$ ) conditions for complete enumeration when  $\tau = 4, 5, 6$  respectively. In particular, we see that for a common value of  $p$  and increased horizon  $\tau$  promotes order. In particular observe the increases in order across 100 to 625 paths for increasing  $\tau$ . For each scatter point  $N = 50$  agents were simulated for 1000 time steps, with standard parameters and paths were sampled uniformly at random without replacement. The error bars indicate one standard error on the mean over  $n = 5$  replicates. Statistics were accumulated for time steps 500 to 1000.

extending each to depth  $\tau = 5$  (one extra visual state per path), we can realise some of the benefit from sampling at higher depths for lower computational cost.

Figure 5.3 shows this data for samples in the range  $[7, 15625]$ . The experiment conducted was to sample fixed percentages of paths from the total number of paths at depth  $5^\tau$  for  $\tau$  values 4, 5, 6, 7, 10. We find the onset of order around  $5^4$  sampled paths as expected, regardless the value of  $\tau \lesssim 10$ . But for  $\tau = 10$  we see the onset of order at lower than  $5^4$  sampled paths. This reflects the fact that fully sampling the FST at the  $\tau = 4$  depth is sufficient, and sampling further increase order, but for

example  $\tau = 3$  is known to be insufficient in explicit enumeration. These findings help us in two ways. Firstly we have a baseline of the effectiveness of sampling confirming the importance of  $\tau = 4$  futures in the model. Secondly we can see the effect of sub-sampling the tree at e.g. depth  $\tau = 4$  but continuing these samples to larger  $\tau$  values. For example consider the data points for  $\tau = 5, 6, 7, 10$  but where  $p \leq 5^4$ . Sampling  $5^4$  paths to depth  $\tau > 4$  is *necessary* but *not sufficient* number for fully enumerating the tree at depth  $\tau = 4$ , however this potential loss of information is offset by the additional information gained by the FST nodes sampled by the continuation above  $\tau = 4$ . Importantly if the sample happened to fully enumerate all paths at the  $\tau = 4$  level we should expect that the additional information gained from the states at  $\tau > 4$  might promote ordered motion. We can see this effect in the  $\tau > 4$  data points.

The significance of the increased order obtained through sampling less paths to larger time horizons is the benefit of sampling. In short, we would like to be able to sub-sample the FST but retain the order motion characteristic of the high  $\tau$  full enumeration of paths case. We see this already appearing in the baseline case, in subsequent sections, encouraged by this, we shall spend time developing more efficient sampling schemes to hopefully exploit these properties further.

## 5.2 Partially Sampling Deeper Tree Leads to Higher Order

Being able to sample the FST, instead of fully enumerating it, and still producing spontaneous collective motion can incur a number of advantages. Firstly if less samples are required to reach an ordered phase, than full enumeration implies, practically the algorithm is faster. Secondly sampling allows us to *sub-sample* higher tree depths for linear (in the number of visual state evaluations) computational cost in tree depth. It is possible we could see an onset of order with fewer samples, sub-sampling higher depths, over fully enumerating shallower depths. This second point if true would imply something about the problem we study, i.e. if sub-sampling longer future times increases order or “stability” of the group more generally over fully enumerating shallower depths this would predict flocks in the real world should care more about the distant future (likely up to some limit) than the immediate future.

We have seen that for a constant number of paths, 625 say, we obtain an increased order if we extend samples to higher values of  $\tau$  beyond the possibility of full enumeration. E.g. for 625 paths we will sample (without replacement) all

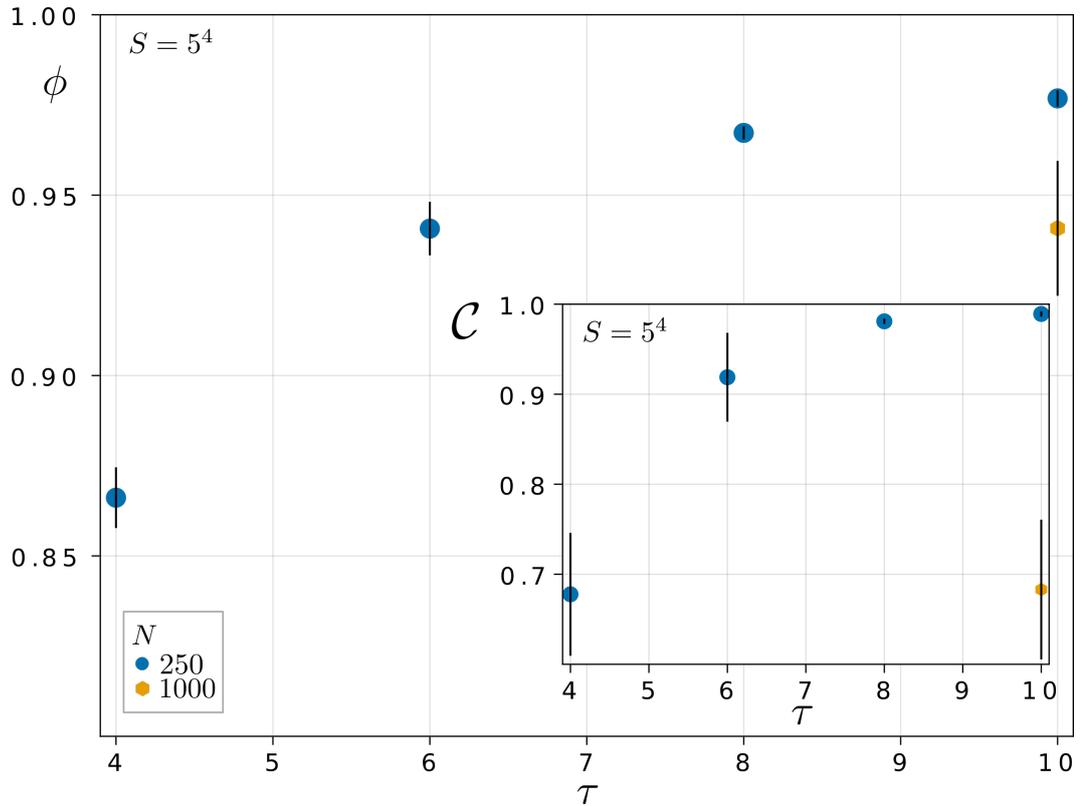


Figure 5.4: Order is increased due to a decrease in fragmentation in large groups ( $N = 250$  and  $N = 1000$ ) when extending a constant number of samples,  $S = 5^4$  here, to deeper tree depths. Paths were sampled without replacement, error bars show one standard error on the mean over six replicates. The inset shows the largest cluster sizes for the same simulations

paths at  $\tau = 4$ , but if we allow each sample to continue on to  $\tau = 10$  we can realise a benefit from high tree depths without the burden of fully enumerating at depths  $4 < \tau' \leq 10$ . Indeed, the added computational load is linear ( $\sim S\tau'$ ) in the number of futures  $\tau'$  we allow samples to extend past  $\tau$ , all else being constant. For example  $\tau = 10$  is not feasible for  $N = 50$  and 1000 time steps given full enumeration, but we can sample hundreds or thousands of paths up to  $\tau = 10$  without much more cost than full enumeration at  $\tau = 4$ . For larger groups this effect is most apparent due to the dramatic reduction in fragmentation seen. Figure 5.4 shows a constant number of samples  $S = 5^4$ , enough to possibly fully enumerate at depth  $\tau = 4$ , extended to increasing tree depths up to  $\tau = 10$ . We see an order of approximately (mean, standard deviation) 0.977, 0.006 and a largest cluster size of 0.99, 0.007. If we compare this to the mean order and largest cluster size for fully enumeration simulation of  $N = 250$  and  $\tau = 6$  (means and standard deviations (0.939, 0.0157)

and (0.956, 0.0152 respectively for  $n = 132$  replicates) a t-test for the null hypothesis that the order data for the path-sampled  $\tau = 10$  simulations in figure 5.4 comes from a distribution with mean 0.939 is rejected with p-value (right tailed)  $\approx 7 \times 10^{-6}$  indicating the data has a statistically significant increased mean order over the null. The same methodology to the cluster size mean yields a p-value of  $\approx 5 \times 10^{-5}$  which is also significant. For  $N = 1000$  we performed the same analysis but by simulating only the  $\tau = 10$  case, due to the lengthy computational time, for 8 replicates, and we obtain a p-value of 0.0035 indicating sampling at  $\tau = 10$  with  $S = 625$  paths does significantly increase the average order against full enumeration at  $\tau = 6$ .

The significance of these results is that fully enumerating at  $\tau = 6$  implies  $5^6 = 15625$  visual state calculations, implying a longer run-time, but we achieve more stable flocks with 40% of the visual state evaluation if we sample 625 paths extended to  $\tau = 10$  where 625 samples extended to depth  $\tau$  evaluate  $S\tau = 6250$  visual states. Therefore, *sub-sampling* deeper trees can be more beneficial than fully enumerating shallower tree; agents should prioritise a *depth-first* tree search over a *breadth first* tree search. We will remark again on this in the closing notes of this chapter as a falsifiable prediction of our model.

### 5.3 Sampling Actions: Applying a Monte-Carlo Tree Search

We have seen that sampling paths in the FST without replacement is key to reaching the same output as the explicit enumeration approach in the same (or less) computational load. There are however two questions to answer about this approach (1) is sampling paths uniformly without replacement the most efficient method? Or can we use the partial information as the FST is updated to improve sampling efficiency? (2) How can this methodology function with extremely larger values of  $\tau$  and with continuous action spaces? The first question we will explore by using the idea of a Monte-Carlo tree Search in this section, after which we shall return to the second question.

Monte-Carlo Tree Search (MCTS) [77] was famously used by Deep Mind's Alpha Go agent [78]. MCTS is usually defined in the context of a game allowing for a sequence of actions followed by a pay-off. The structure of MCTS in this context is to (1) search recursively in a game state tree to find a leaf, which can be totally at random or guided (2) expand from the leaf node by taking an action in the game to reach a new leaf-node state (3) simulate from this new leaf to an end state (4) back propagate the payoff by updating the node values. This is the basic structure, each

step can and has been modified for different applications. For FSM we will use the backpropagation and recursive searching chiefly, to determine where on the FST to sample next.

The core issue with applying MCTS to SCE is this back-propagation, since this will inform the searching. Previously when sampling we recorded samples and computed the entropy data once at the end of the sampling process, just like explicit enumeration. For back-propagation after each sample we update information at each node in the tree using the most recent sample. For example if we wish to use the current approximation of the entropy along branches to guide future samples we will need to find an efficient manner to update the entropy’s along nodes. Ideally we would want to be able to define an “online” method [115] for calculating the entropy. For example consider finding the average of a stream of numbers, if  $k$  numbers,  $x_i$ , have been seen then  $k$ th mean is simply  $\mu_k = \frac{1}{k} \sum_{i=1}^k x_k$ , when we are given the  $k + 1$ th number we can update the mean by  $\mu_{k+1} = \frac{k\mu_k + x_{k+1}}{k+1}$ , rather than re-doing the whole calculation on the new set of  $k + 1$  numbers. The utility is that when back-propagating, the value at a node can be trivially updated without having to start the calculation from scratch.

For our case we are calculating the entropy of a categorical distribution with  $2^{n_s}$  possible categories but only a small number being seen <sup>1</sup>. Each sample we add new states to the FST and therefore need to update counts and entropies. For example when a new node in the tree is sampled all nodes prior to it (existing along some path from the root to the new node) need to account for the new state. The state counts  $n(\psi)$  and therefore probabilities and entropies can be updated in an online fashion using prior entropy values, see Appendix A for the details. The problem however, is that doing this requires knowing if the new state is unique or not. This presents a slight problem since whenever we update a node we must check if the new state is already part of that node’s distribution by direct comparison before apply the update rule. In SCE this is done for the root node itself, 1 node, at the end of the “sampling” process, now we would need to do it for all nodes along a sampled path ( $\tau$  nodes) each sample. This comparison will scale linearly in the number of unique states being compared to, e.g. if at some node we have seen states  $\{\psi_1, \psi_2, \psi_3\}$  with counts 1, 4, 2 respectively, for a new state  $\psi$  we compare it to the three seen states regardless as to their degeneracy. This backpropagation is done after each sample by back-propagating from the final node in the sample. In comparison, for the CE case the quality is the average of visual state entropies

---

<sup>1</sup>In particular the  $|\mathcal{A}|^\tau$  leaf nodes may all be unique, and there are  $|\tau|\mathcal{A}|^\tau$  in total across paths. Internal nodes will not be unique by construction.

$H(\psi)$  along branches, therefore backpropagation is simply a case of the online mean algorithm without needing the uniqueness check.

### 5.3.1 FSM and Monte-Carlo Tree Searching

Similar to Fractal AI [79] and MCTS more generally [77] our tree search algorithm judges which action to take (which child node to move to) by comparing a notion of quality between the candidate child nodes. For us quality could take many forms. Ideally we would like a self-similar quality, meaning that the sampler judges which child to take by the same algorithm as decision are made by the agent at the end of sampling; by maximising entropy. If we only maximise entropy then the first sample will be uniformly distributed among the total set of paths. This sample will update entropies along the chosen path. If the “null entropy value” at un-sampled nodes is 0 the second then will always choose the *same path* unless it is completely degenerate with 0 entropy. This is the exploitation-exploration problem. If the default value were 1 for the un-known entropies, this would mitigate this problem somewhat (for entropies normalised to  $[0, 1]$ , see appendix B), but still paths with high entropy will be re-sampled with high probability. A naïve solution to this is to simply count the number of un-sampled nodes subsequent to a given node and use this with the entropy to decide on child nodes. For example, we can take the default entropy at un-sampled nodes as 1 (maximal) but their quality should be further weighted by the count of un-sampled subsequent nodes (with leaves given a count of 1). Then during sampling, entropies and counts are updated, and we judge which path to take based on the combination of these.

We do this to hopefully take advantage of the information gained by recording future entropies and un-sampled counts at the internal nodes, that is when sampling a path we choose the next action in the path informed by these values. Here we will do this by Boltzmann factors of the form

$$p(\beta_{ik}^t | \{\beta_{ik}^1, \beta_{ik}^2, \dots, \beta_{ik}^{t-1}\}) = \frac{C(\beta)e^{-\frac{1}{T}\Delta\hat{H}(\beta)}}{\sum_{\beta' \in \mathcal{A}} C(\beta')e^{-\frac{1}{T}\Delta\hat{H}(\beta')}}. \quad (5.8)$$

Where  $p(\beta_{ik}^t | \{\beta_{ik}^1, \beta_{ik}^2, \dots, \beta_{ik}^{t-1}\})$  is the probability for agent  $i$  sampling action  $\beta_{ik}^t$  at future time  $t$ , given the previous actions  $\beta_{ik}^1, \beta_{ik}^2, \dots, \beta_{ik}^{t-1}$  taken at *previous future* times  $1, 2, \dots, t - 1$ . The value of  $\Delta\hat{H}(\beta)$  is the *estimated* gain (or loss) in *quality* for sampling action  $\beta$  at time  $t$  after sampling actions  $\{\beta_{ik}^1, \beta_{ik}^2, \dots, \beta_{ik}^{t-1}\}$ . The pre-factors  $C(\beta)$ , which are optional in the case of a stochastic FST, down-regulate sample degeneracy for the reasons discussed above, and depend (as  $\Delta\hat{H}$  does) on the history of samples  $\cup_{l=1}^k \mathbf{s}_{il}$  codified in the FST as samples are evaluated. Finally,

we interpret the incorporated temperature parameter  $T$  in each Boltzmann factor as a means to manipulate the behaviour of the algorithm i.e. to control the balance between exploration and exploitation. In fact this parameter facilitates a similar role to the counts  $C$ , low temperatures will lead to greedy selection of high entropy paths and high temperatures lead to increasing uniform selection. Considering this as an optimisation-like algorithm literature surrounding simulated annealing [116], particularly the “cooling schedule” of  $\beta$  is highly relevant in its interpretation and practical implementation in our sampler.

---

**Algorithm 4:** FSM Monte-Carlo Tree Search

---

**Result:** Agent Path Samples

```

1 while  $s < S$  do
2   sampleStates[ $s$ ]  $\leftarrow \emptyset$ ;
3   Node  $\leftarrow$  Root;
4   sampleStates[ $s$ ].add(Node. $\psi$ );
5    $\alpha \leftarrow$  sampleAction(Node,FST);
6   initialAction[ $s$ ] $\leftarrow \alpha$ ;
7   Node  $\leftarrow$  nextNode(Node, $\alpha$ ,FST);
8   isCollided  $\leftarrow$  collided(Node,agentModel);
9   while  $\neg$ leaf(Node)  $\wedge$   $\neg$ isCollided do
10    if empty(Node) then
11      /* Only need to fill each node once */
12      Node. $\psi \leftarrow$  getVisualState(Node);
13      Node.add( $\psi$ );
14      /* future states */
15      /* Record path entropy */
16      sampleStates[ $s$ ].add(Node. $\psi$ );
17       $\alpha \leftarrow$  sampleAction(Node,FST);
18      Node  $\leftarrow$  nextNode(Node, $\alpha$ ,FST);
19      isCollided  $\leftarrow$  collided(Node,agentModel,1);
20    end
21    backpropagate(Node);
22   $s \leftarrow s + 1$ ;
23 end

```

---

**Algorithm 5:** Backpropagating Entropies

---

**Result:** Entropy Backpropagation

```

1 Function backpropagate(Node, $\psi$ ) is
2   if Node == Root then
3     | return;
4   Node  $\leftarrow$  parent(Node);
5   Node.add( $\psi$ );
6   backpropagate(Node);
7 end

```

---

---

**Algorithm 6:** Boltzmann Sampling

---

**Result:** sampled action

```
1 Function sampleAction(Node, FST) is
2   for  $\alpha \in \mathcal{A}$  do
3      $u \leftarrow \text{nextNode}(\text{Node}, \alpha, \text{FST});$ 
4     if  $\text{leaf}(u) \vee \text{isCollided}(\text{Node}, \text{agentModels})$  then
5        $b[\alpha] \leftarrow 0;$ 
6       /* uniform among leaves */
7     else
8        $\Delta H \leftarrow \text{Node.quality} - u.\text{quality};$ 
9        $b[\alpha] \leftarrow u.\text{FutureCount} \cdot \exp(-\Delta H/T);$ 
10    end
11  if  $\text{normalise}(b) == 0$  then
12    /* All options are leaves or collided, random uniform action */
13     $\text{return sample}(\text{ones}(b.\text{length}) \cdot 1.0/b.\text{length});$ 
14  return  $\text{sample}(b);$ 
15 end
```

---

To lay this out practically algorithm 4 shows the basic structure of how agents sample the future states tree. The `sampleAction` and `backpropagate` function is defined in algorithms 6 and 5 respectively. For the utilities: the `nextNode` function moves to the next node in the FST, by action  $\alpha$ , `leaf(Node)` returns true if Node is collided or at future time  $> \tau$  (a leaf in the *full* tree), `empty(Node)` indicates whether the node has been visited before, `parent` obtains the unique parent node (or a null value for the root node), `exp` is the exponential function  $\exp(x)$  and `normalise(b)` normalises vector `b` in-place, finally `sample(b)` samples from a probability distribution given by the elements of `b`. Algorithm 4 represents one sample *path* in the FST, in one run up to  $\tau - 1$  nodes will have new visual states calculated and the `backpropagate` function will update nodes on paths leading to it by moving up parent nodes. During one sample we always begin at the root node, which is always added to a samples states (lines 2-4 algorithm 4). Next sample an initial action (lines 5-8 4) using the Boltzmann sampler in algorithm 6. For a given node the sampler calculates a probability distribution for selecting actions given by equation 5.8 optionally weighted by the number of unsampled nodes subsequent to taking each action (thereby potentially down regulating degenerate samples). The returned action is then actioned to give the next state. Lines 9-17 in algorithm 4 follow that same process moving down the tree and collecting states into the sample distribution. In line 12 of algorithm 4 and line 5 of algorithm 5 the nodes states and back-propagated states are added to the node for use in the sampler. The optional future counts can also be updated in an efficient manner similar to the backpropagation of states given that at the first sample it is simple combinatorics to write

down the number of unsampled states after each node in the tree, once a new node is sampled we simply decrement these initial counts by 1 for all parent nodes <sup>2</sup>.

This methodology gives us two new parameters, the inverse temperature  $\beta$  and the number of samples  $S$ .  $\beta$  controls how greedy or random the sampling of actions to make paths is and the number of samples controls how many *paths* are sampled. Note that taking  $\beta \rightarrow 0$  implies random selection as expected and  $\beta \rightarrow \infty$  selects the best value (or among the degenerate maxima) when searching the FST. We can even identify  $\beta \rightarrow \infty, S \rightarrow \infty$  as the SCE algorithm, if we are careful with degenerate samples. To explain consider the following two wrinkles to the above algorithm: (1) because our sampling scheme samples paths by sampling actions we don't sample *paths* without replacement, the inverse temperature  $\beta$  and the use of the un-sampled counts will alleviate this, but degenerate samples will likely still occur. In order to achieve *exact* convergence to explicit enumeration in the limit of  $S \rightarrow \infty$  we can further record an identity for each path, namely its final node, which uniquely identifies it. This identity will allow us to find and ignore degenerate samples when calculating the count distributions at the decision step. For example say we sample the same path  $s$  twice, this would mean in the final distribution used for an agent's decision all states in the path  $s$  will be double counted, they will all be degenerate. But it may actually be the case that the final state in  $s$  (assuming no collisions), being a leaf node, might be unique and further the other states in  $s$  may be rare in the rest of the tree. With even more degenerate samples of the same path, the entropy distribution used for decision-making will diverge from the true distribution under explicit enumeration and in fact in the limit of  $S \rightarrow \infty$  all states will be infinitely degenerate.

We can resolve this, even without backpropagation of counts of entropies, by using the identity mentioned above. When we accumulate the final distribution of states along each branch we keep track of which paths have been accounted for; if a sample represented a path already accounted for we ignore it for the purposes of decision-making. Not doing this will mean an erroneous degeneracy of visual states will be collected just like the data in figure 5.2 versus figure 5.3. Note we still have multiply counted states along paths, this is a simple fact of the structure of paths on an FST. But in this sampling scenario it is possible for two samples to be identical; be the same path. In this sense the algorithm can produce erroneous degeneracy.

Of course, interestingly, the problems of degeneracy among samples is a factor of the deterministic nature of the FST in our problem. If the FST were

---

<sup>2</sup>Collisions can be handled in the same manner but of course require decrementing parent counts by larger values i.e. the number of nodes lost by the branch cut.

itself Stochastic, degenerate sampled paths might aid decision-making. For example consider situations such as the cognitive noise model presented in chapter 3 section 4.2.2, where each agent’s model of the others is on average ballistic but subject to a noise process. Two samples taking identical paths through the FST will, dependent on the noise strength, result in different visual states being collected along each sampled path despite the same actions being taken. We will return to this when comparing the sampling algorithm presented here to explicit enumeration.

### 5.3.2 Does Uniform Action Sampling Lead to Ordering?

To see if we can benefit from the Boltzmann sampling setup, or in particular whether a uniform action selection distribution is justifiable we can vary the parameter  $\beta$  in algorithm 6. High values of  $\beta$  indicate uniform sampling. This data is shown in figure 5.5 (a) and (b) where we use the future SCE change (normalised) to judge future actions and in (c) and (d) using the number of visual state boundaries (with a resolution parameter  $s = \pi/40$ ). In each case we vary  $\beta \in [1, 10, 100, 1000]$  and  $\tau$  as 6 and 10. For the future entropy we find an unclear relationship with  $\tau = 6$  and a marginal benefit of uniform sampling at  $\tau = 10$ . However, using the number of boundaries we find low  $\beta = 1$  to be both consistent and superior to all other cases except for  $S \approx 625$  where higher  $\beta \approx 1000$  values are as good and better. For future entropy it appears we can choose uniform sampling for large enough  $S \gtrsim 625$  and  $\tau \gtrsim 10$ . Note though that we can get an order of  $\phi \approx 0.95$  with  $S = 125$  and  $\tau = 10$  if we use the average number of visual state boundaries method, this presents a very significant computational saving at the expense of stability in the group i.e. the order of  $\phi \approx 0.95$  is due to fragmentations (usually into 2 groups in these data points).

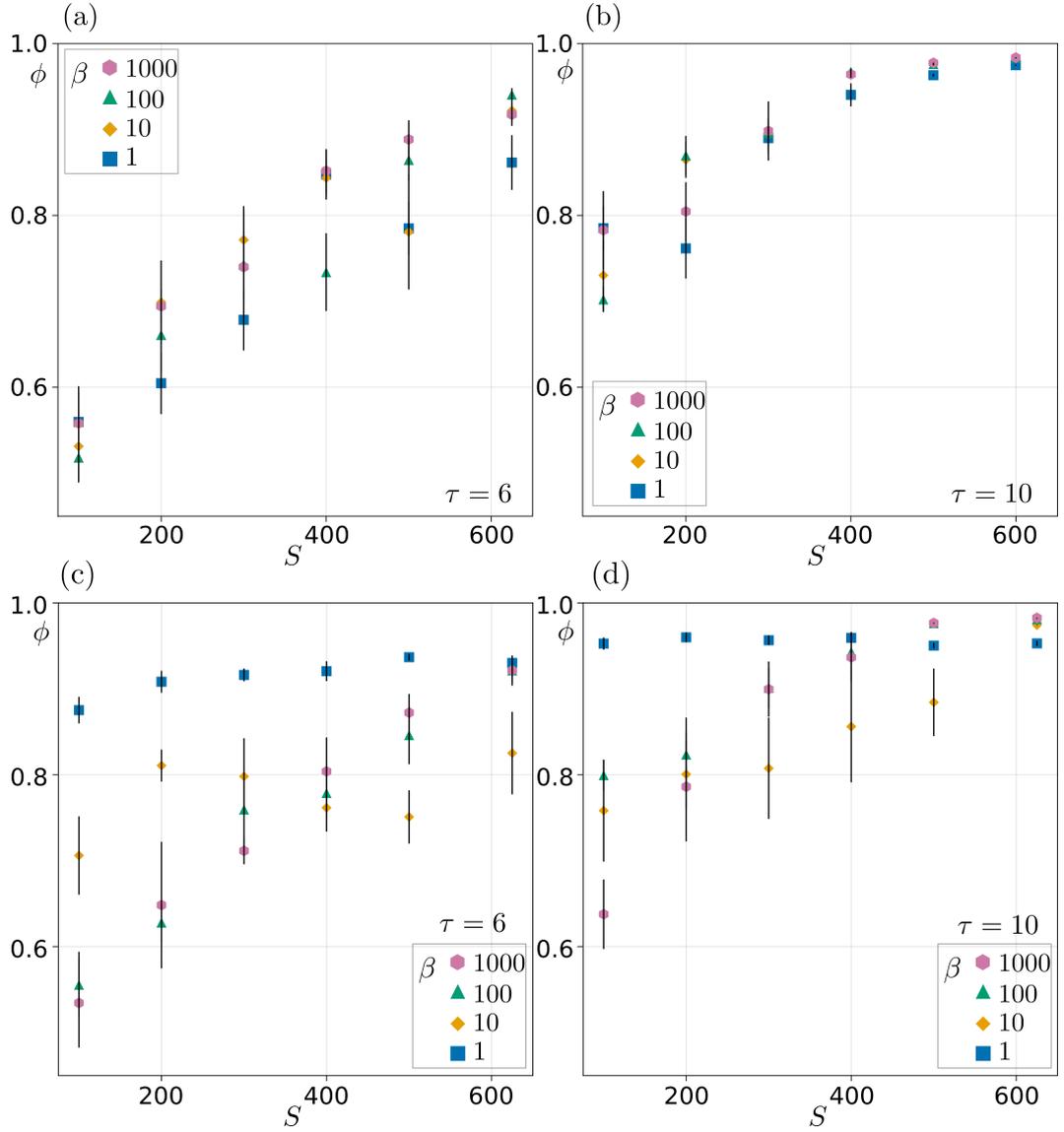


Figure 5.5: Order for  $N = 50$  agents using the Boltzmann sampler with future entropy to judge actions to sample. (a) and (b) use  $\tau = 6$  and  $\tau = 10$  respectively. (c) and (d) use the average count of boundaries on future states with  $\tau = 6$  and  $\tau = 10$  respectively. Error bars indicate one standard error on the mean for 6 repeated simulations. Increasing  $\beta$  has an unclear impact at  $\tau = 6$ . With  $\tau = 10$  increased  $\beta$  appears to have a slight benefit implying uniform sampling is advantageous.

## 5.4 Continuous Action Spaces

To extend to continuous actions spaces a sampler is needed in practice if we are to perform FSM, since it is impossible to fully enumerate a continuous space com-

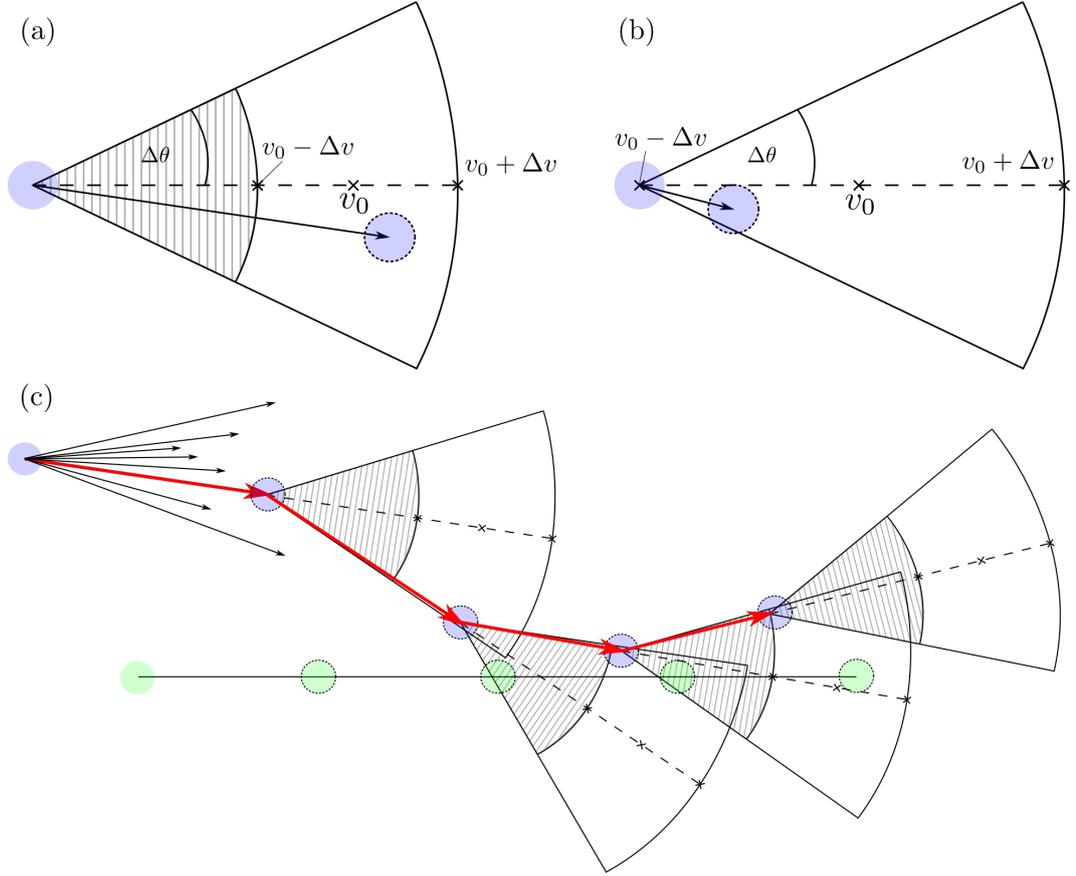


Figure 5.6: Two examples of a continuous action space in our context. (a) a model where agents may increment their orientation by any angle  $\theta \in [-\Delta\theta, \Delta\theta]$  and choose speeds in the range  $v \in [v_0 - \Delta v, v_0 + \Delta v]$ . (b) the same as in (a) but now since  $\Delta v = v_0$  the speed can be any value in  $[0, 2v_0]$ . (c) An example sampled path to depth  $\tau = 4$  (red, thick arrows) starting with a set of  $n_a = 8$  initial actions to choose from (black, thin arrows with first red thick arrow).

putationally. First though we must formalise what a continuous action is in this context. In the discrete case we have defined rules which increment orientation by a magnitude  $\pm\Delta\theta$  or select speeds  $v_0$  and  $v_0 \pm \Delta v$ . This means that each agent's speed is always contained within the set  $\{v_0 - \Delta v, v_0, v_0 + \Delta v\}$  and its orientation within  $\{n\Delta\theta \bmod 2\pi\}$  for integers  $n \in \mathbb{Z}$ . To move to a continuous analogue of this we can proceed in a number of ways, one such example is shown in figure 5.6. We consider  $v_0, \Delta v$  and  $\Delta\theta$  as physical constraints on an agent such that in time  $\Delta t$  an agent can at most modify its heading by a magnitude of  $\Delta\theta$ , i.e a continuous increment in angle from the range  $[-\Delta\theta, \Delta\theta]$ , and select speeds within the continuous range  $[v_0 - \Delta v, v_0 + \Delta v]$ . This creates a continuous action space for speed

and orientation. One subtle point is how  $\theta_i^t$  and  $v_i^t$  are handled, i.e. we can choose  $\theta_i^t \in [0, 2\pi)$  and consider actions as increments to  $\theta_i^t$  which we do, for the speed however we must be careful. If we choose  $v_i^t \in \mathbb{R}$  meaning the speed action being interpreted as an increment to agent speed our model may result in infinite speeds or negative speeds without further limitations, to eliminate this we take, as above, speed actions to be selections from a continuous range rather than selection of *increments* from a continuous range or selection from discrete range. Finally, note that our actions are defined as vectors taking an agent from its current position to any of the allowed positions by  $\Delta\theta, v_0$  and  $\Delta v$ . To be precise we can define an action as a vector  $\alpha_i^t \in [-\Delta\theta, \Delta\theta] \times [v_0 - \Delta v, v_0 + \Delta v]$  interpreted in a local polar coordinate system  $(\phi, r)$  defined from the position  $\mathbf{x}_i(t)$  and orientation  $\theta_i(t)$  of the agent under consideration, i.e.  $(\phi = 0, r = 0)$  is the position  $\mathbf{x}_i(t)$  and vectors  $(0, r)$  point along agent  $i$ 's heading direction at time  $t$ .

We wish to use the above continuous action space in our SCE model. To do so we have some practical difficulties. An agent cannot enumerate all possible paths, so we must sample, but importantly how we sample the initial actions along paths matters. We could simply select  $S$  initial actions, one for each sampled path  $1, 2, \dots, S$ , but this would mean each path only contains  $\tau$  visual states. In terms of the entropy in the SCE method paths sampled in this way would amount to low entropy values for small  $\tau$ . In fact in doing this we find high order cohesive groups only with  $\tau \approx 512$ , an extreme value, as we will see shortly. Recall in the SCE model with the discrete action space we naturally have  $|\mathcal{A}|$  actions which our agent chooses from, in analogy to this and to combat the problem just stated we define a new parameter  $n_a \leq S$  which defines the number of *initial actions* an agent considers taking now, i.e. the set  $\mathcal{A}_i^0 = \{\beta_{i1}, \beta_{i2}, \dots, \beta_{ina}\}$  where each is sampled from some probability distribution (say the same as for selecting actions along future paths; a uniform distribution). Using  $n_a$  we can range up to the pathological case mentioned above,  $n_a = S$ , and also consider  $n_a = 5$  in analogy to the standard 5 action space we have used in the SCE model in previous chapters, and anywhere in between.

Agents following this model will sample  $n_a \leq S$  *possible* initial actions. Then for each path  $1, 2, \dots, S$  will first select an initial action  $\beta_{il} \in \mathcal{A}_0$  at random, then select a further  $\tau - 1$  actions  $\beta_i^2, \dots, \beta_i^\tau$  uniformly, each from the whole space  $[-\Delta\theta, \Delta\theta] \times [v_0 - \Delta v, v_0 + \Delta v]$  (i.e. *not* limited to  $\mathcal{A}_0$ ) to define the sampled paths. Along the  $S$  paths will be grouped together into distributions defined by the initial actions. Then an agent picks the initial action  $\alpha_i^t \in \mathcal{A}_0$  maximising the future path entropy computed analogously to the SCE method. One example path is visualised

in figure 5.6 (c). The equations of motion are

$$\mathbf{x}_i^{t+1} = \mathbf{x}_i^t + \mathbf{v}_i^t \Delta t, \quad (5.9)$$

$$\mathbf{v}_i^{t+1} = \alpha_v R(\alpha_\theta) \hat{\mathbf{v}}_i^t. \quad (5.10)$$

Where we use the notation  $\alpha_v$  and  $\alpha_\theta$  to denote the two (polar magnitude and angle) components of the selected action  $\alpha$  which maximises path entropy across samples. The model reproduces similar output to the discrete case, for example in figure 5.7 we see one particular examples of an  $N = 50$  group following the “default” parameters we have used previously in this thesis, where  $n_a = 5$  and  $S = 625$  to emulate the 5 actions in the discrete case. Here the flock remains cohesive with  $\phi = 0.974$ . The distributions for chosen speed and rotations can both be fitted by truncated normal distributions with means  $v_0$  and 0 respectively, with standard deviations around  $\approx 4$ , and  $\approx 19^\circ$ , in the discrete case action choices are roughly uniformly distributed across time. Given we obtain no surprises for  $n_a = 5$  and standard parameters we

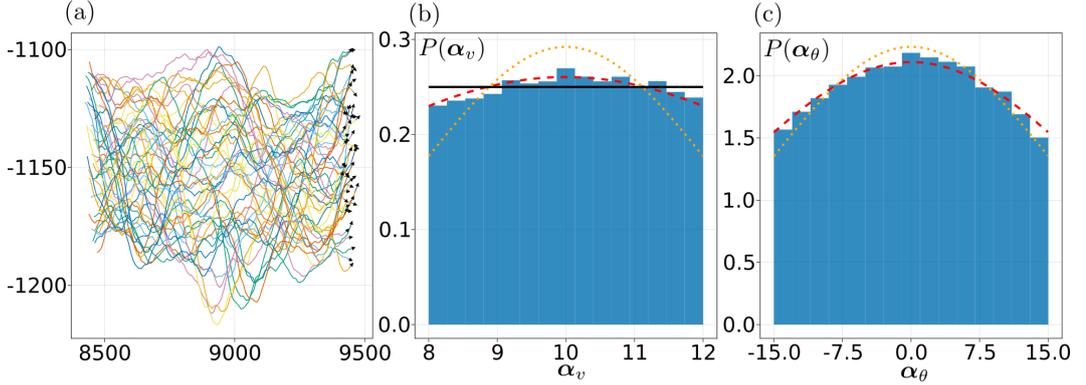


Figure 5.7: Example output from the continuous model with  $N = 50$ ,  $\tau = 5$ ,  $n_a = 5$ ,  $\Delta\theta = 15^\circ$ ,  $\Delta v = 2.0$ ,  $v_0 = 10.0$  (a), we find  $\phi = 0.974$ . The distribution of realised actions is shown in (b) for the speed and (c) for the orientation, in each case the truncated normal distributions  $\alpha_v \sim \mathcal{N}_{v_0-\Delta v}^{v_0+\Delta v}(v_0, \Delta v^2)$  and  $\alpha_\theta \sim \mathcal{N}_{-\Delta\theta}^{\Delta\theta}(0, \Delta\theta^2)$  are plotted as orange dotted lines, in (b) the black line indicates a uniform distribution over the range. The red dashed lines indicate truncated normal’s with the same means but different variances:  $\alpha_v \sim \mathcal{N}_{v_0-\Delta v}^{v_0+\Delta v}(v_0, 4)$  and  $\alpha_\theta \sim \mathcal{N}_{-\Delta\theta}^{\Delta\theta}(0, 361)$  which are better (b) and arguably better (c) fits. For flocks fragmenting both  $\alpha_\theta$  and  $\alpha_v$  tended to be uniform distributions. Statistics for (a) and (b) were collected for all time-steps (1-1000) in (a).

examine the role of  $n_a$  in figure 5.8. We find that  $n_a$  depends on the number of paths sampled, and we see the problem case  $n_a = S$  as mentioned above where the order  $\phi$  drops. It appears  $n_a = 5$  is not optimal, which is not so surprising since

for FSM more actions are exponentially more difficult to compute and that choice was made for computational reasons mainly. For increased samples however we can retain high order across higher  $n_a$  actions that agents select from. The data reflects the trade of between how many samples are taken and how many actions there are to choose from, more actions to choose from than samples means some actions are completely unsampled, both equal and an agent “spreads itself thinly” by not building much knowledge about each possible action (even probably not sampling some at all), with  $n_a \ll S$  an agent builds a more uniform and accurate knowledge of how each actions plays out, but this limits available choices.

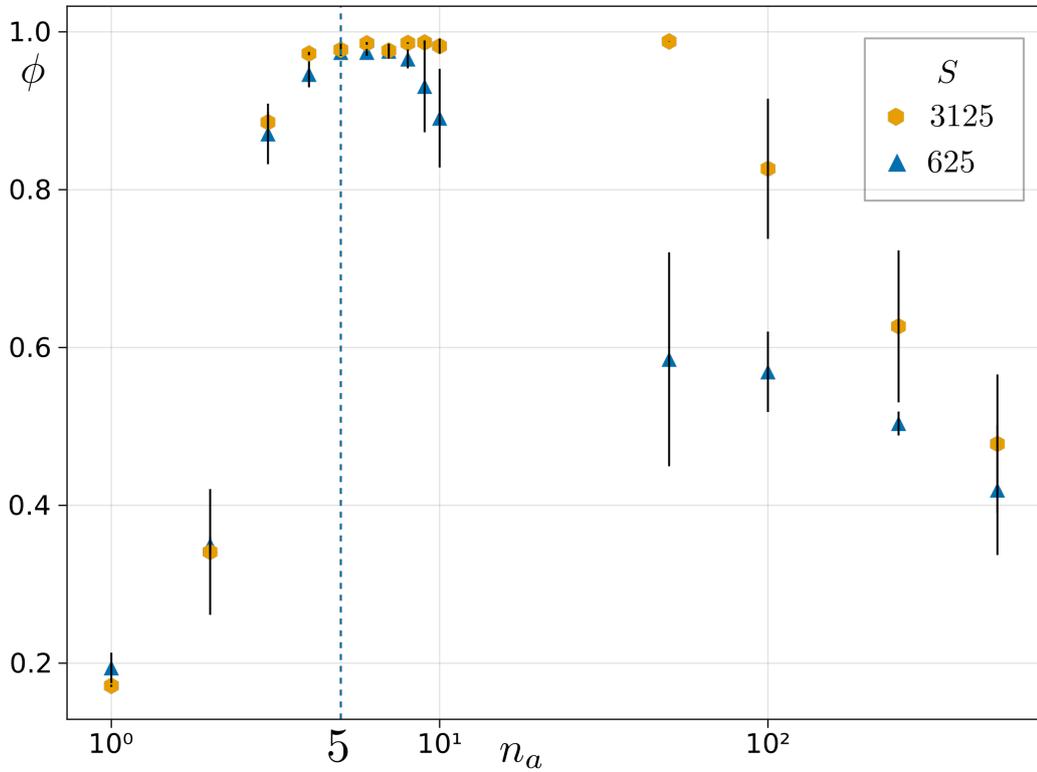


Figure 5.8: Order of an  $N = 50$  group with  $\tau = 5$  dependent on the number of initial action  $n_a$  an agent selects from for 625 and 3125 sampled paths. Here  $v_0 = 10$ ,  $\Delta v = 2$  and  $\Delta\theta = 15^\circ$  to compare directly with the SCE model with discrete action space.  $n_a = 5$  is marked for comparison. There is a trade-off between how many initial actions an agent selects from and versus how many paths need to be sampled to effectively judge between them.

### 5.4.1 Phenotypes

As in the discrete action space model we can also explore the action space by varying the parameters  $\Delta\theta$ ,  $v_0$ , and  $\Delta v$ . We show this data in figure 5.9 (c.i) and (c.ii) as well as four examples from that data in 5.9 (a.i-a.iv). We find a transition largely controlled by decreasing  $v_0$  from cohesive, translating and ordered motion to swarm behaviour at  $v_0 = 1$ . Behaviour is largely unchanged in  $\Delta v/v_0$  with the exception of values around  $\Delta v \sim v_0 = 1$  which appears to show a reduction in the size of the largest cluster at  $v_0 = 1$ . At intermediate values  $1 \lesssim v_0 \lesssim 8$  we find more frequent fragmentations but with many highly ordered and cohesive sub-fragments as well as isolated single agents.

Additionally, we can also, in the continuous model, modify the action distribution by for example biasing the rotation selection in sign. Up to now we have taken the actions sampled uniformly and independently in each component i.e.  $\beta_\theta \sim \mathcal{U}([-\Delta\theta, \Delta\theta])$ ,  $\beta_v \sim \mathcal{U}([v_0 - \Delta v, v_0 + \Delta v])$  with  $\beta$ 's to denote unrealised actions (i.e. on the FST, and/or being considered for actual realisation). To bias  $\beta_\theta$  we could sample from a truncated normal distribution  $\beta_\theta \sim \mathcal{N}_{-\Delta\theta}^{\Delta\theta}(\theta_\mu, \theta_\sigma^2)$  for some mean  $\theta_\mu \neq 0$  and variance  $\theta_\sigma^2$  with  $\mathcal{N}_{-\Delta\theta}^{\Delta\theta}$  indicating truncation at  $\pm\Delta\theta$  to keep the sampled rotation within those bounds. By increasing  $\theta_\sigma$ , all else being constant, the distribution approaches a uniform distribution across its support. Controlling the variance allows us to see the behaviour across more severe or less severe orientational biases. We show one example in figure 5.10.

## 5.5 Conclusions

In this chapter we have examined how to apply a sampling approach to the SCE model. In particular, we began by discussing the importance of sampling paths along the FST *without* replacement in the case of a deterministic FST where there was no noise, e.g. on the ballistic models of other agents. We found that by sampling paths without replacement by selecting  $S$  paths at random from the list of all possible paths given by the action space  $|\mathcal{A}|$  and the time horizon  $\tau$  a benefit can be achieved for sampling deeper in  $\tau$  without fully enumerating. That is we found by sampling a lower number of paths but extending each path to longer tree depths (e.g.  $\tau = 10$ ) we can achieve the same order and stability as fully enumerating shallower trees (e.g.  $\tau = 5$ ).

Encouraged by this we applied a Monte-Carlo Tree search algorithm to our problem by considering the future path entropy at nodes in the FST. That is our MCTS builds sample paths by selecting  $\tau$  actions from a probability distribution

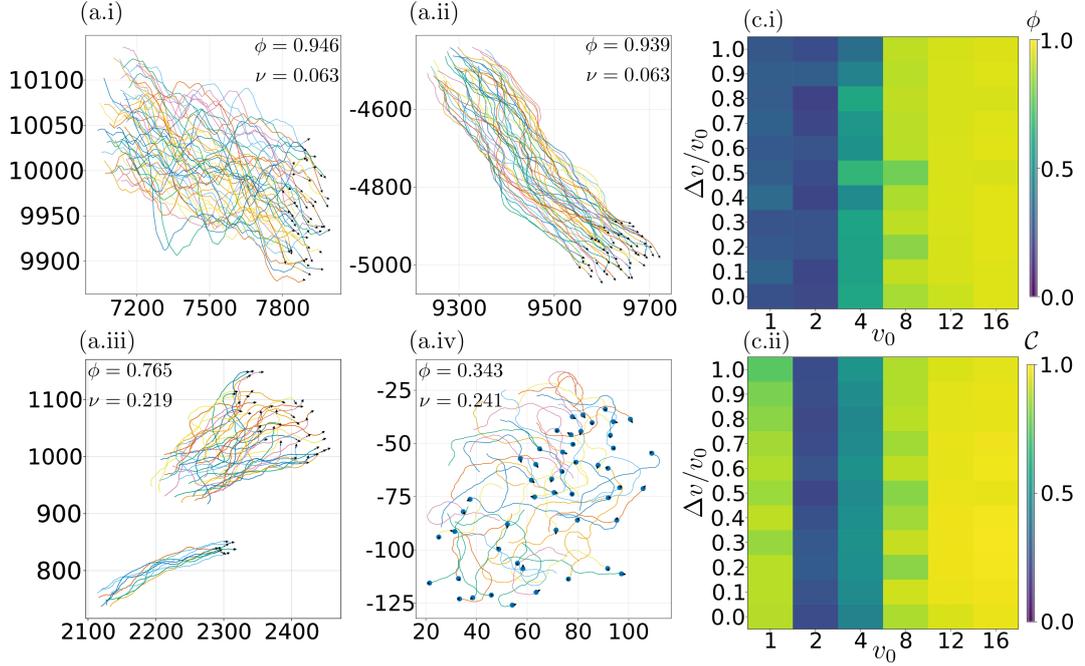


Figure 5.9: Phenotypes in the continuous model, we see swarming (a.iv) and highly ordered cohesive groups (a.i,a.ii) for low and high values of  $v_0$  respectively, with fragmenting groups (a.iii) at intermediate values of  $v_0$ .  $\Delta v/v_0$  modifies this only slightly at  $v_0 \approx 1$  (c.i) and (c.ii). In (b) we see the result of biasing the orientation selection to a positive sign. In all cases  $N = 50$  and  $v_0 = 10$ ,  $\Delta v = 2$ ,  $\Delta\theta = 45^\circ$ ,  $S = 625$ ,  $n_a = 5$ , and  $\tau = 10$  unless stated otherwise.

formed by Boltzmann factors proportional to  $\exp(-\Delta Q/\beta)$  where  $Q$ , the increase in “quality” by choosing an action is computed by the change in future entropy by taking that action. We found however that, especially for larger  $\tau \gtrsim 10$  and  $S \gtrsim 625$  sampling using an approximately uniform distribution  $\beta \approx 1000$  was sufficient for ordered and cohesive flocks to emerge. We did however find that for larger groups,  $N \gtrsim 250$  sampling deeper into the FST is better than fully enumerating lower tree depths. In particular, we found that for  $N = 250$  and  $\tau = 6$  (15625 paths fully enumerated) the order was lower (statistically significantly) than when sampling less paths (625) but to depths  $\tau = 10$ . Importantly this meant we could achieve a “better result“ for less computational cost (6250 visual states versus 15625 visual states).

Finally, using the knowledge we gained from developing a sampling approach to the SCE model, we applied this to the development of a continuous action space version of the model. In this case we defined agents able to increment their orientation and select their speeds from continuous sets. We found, just like in the

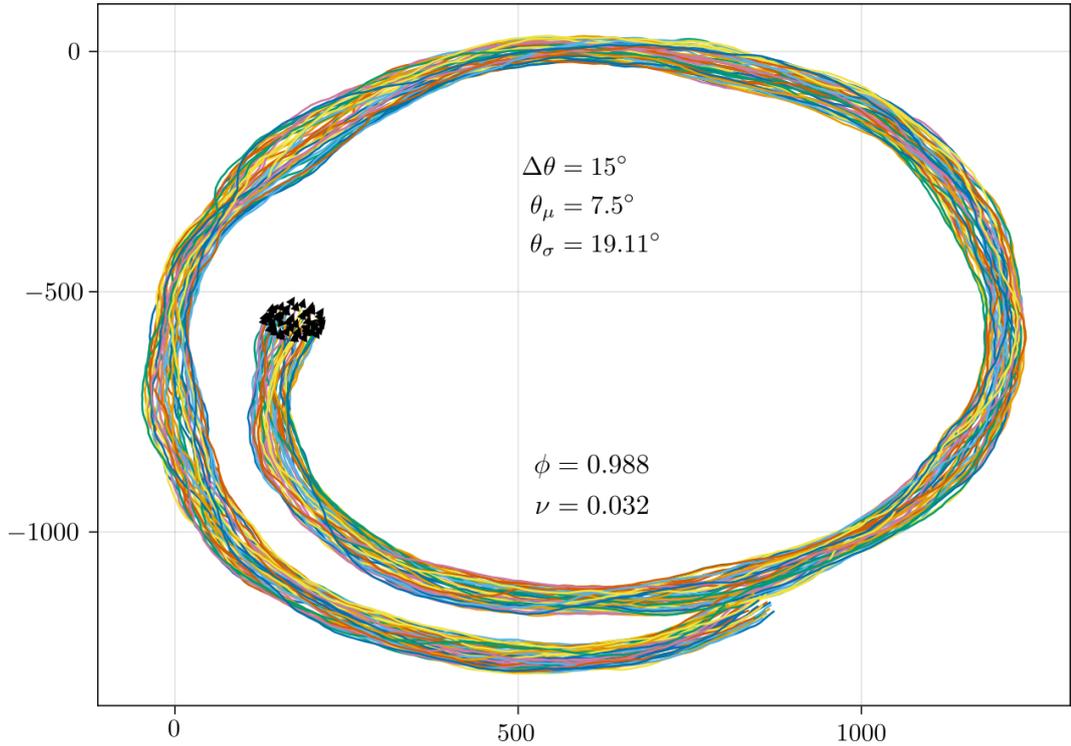


Figure 5.10: Bias applied to orientation leads to a cohesive flock following a circular path.  $N = 50$  and  $v_0 = 10$ ,  $\Delta v = 2$ ,  $\Delta\theta = 45^\circ$ ,  $S = 625$ ,  $n_a = 5$ , and  $\tau = 10$  unless stated otherwise.

SCE sampling results, that selecting actions uniformly when searching the future achieved stable, highly ordered and cohesive flocks. But we had to introduce a new parameter,  $n_a$ , the number of actions an agent select from initially, but no effecting future action imagined on the FST after this initial action. For finite samples  $S$  there is a trade-off between how many initial actions can be selected from versus the number of samples distributed among them, when  $S = n_a$  we only have a single path associated with each decision, when  $n_a \ll S$  we have many paths associated with each decision. This is reflective of the fact that if each agent is to select from a large number of possible actions, we need to ensure that there are enough samples to gain a clear picture of the future path entropy subsequent to them. Computationally this quickly becomes infeasible.

## Chapter 6

# Conclusions and Future Work

In this thesis we have covered a data driven top-down approach to a problem in active matter, Whirligig beetle motion, and in detail developed a bottom-up approach from the principle of maximising future path entropy.

In chapter 2 we used experimental data to analyse trajectories of individual beetles as part of larger collectives. By doing so we defined a methodology for estimating local density and used this to uncover a power law speed density relationship, as well as finding minor inertial effects. We investigated how we could explain the power law decay using inertial ABPs with some success, but we are still left with the problem of behavioural effects that we cannot eliminate in our analysis. From a top-down perspective unexpected results, at least no predicted results, are often not fully explainable without additional experiment to rule out certain confounding factors. In our case we cannot rule out the possibility of beetle interactions driving the self-propulsion mechanism as apposed to it being the result of purely repulsive interaction in an inertial setting, or even hydrodynamic effects. We found that our active particle model with re-orientational torque included does allow use to replicate the bi-modal density distribution seen in the data by repulsive interactions plus this reorientation. We concluded this is evidence for a MIPS like effect seen in the experimental data that drives this separation into high and low density regions with corresponding low and high self propulsion speed. Aside from conducting novel experiments into Whirligig beetle behaviour, either to test the re-orientational torque term directly or to determine casual effects generating the speed-local density relationship we uncover, future experiments can and should follow the bottom-up data driven approaches used to model other similar systems. Such approaches attempt to determine individual force responses to re-occurring situations and use these to inform equations of motion. We would argue that as

unbiased an approach as possible should be taken, in particular we suggest that learning a force response via machine learning methodology could prove a fruitful avenue.

In Chapter 3 we introduced a model of collective motion capturing different phenotypes based upon maximisation of future path entropy. We developed this model from the extant literature on intrinsically motivated collective motion, that maximises the count of unique future states inspired by the principle of future state maximisation. We also found in an attempt to move away from a discrete visual state, by defining a measure of path entropy based upon compression, that we need to re-impose a form of effective discretisation into the model in order to achieve spontaneous collective motion. We interpreted this as implying the number of sensors should be compared not the count of cells in the retina, which is far larger in biological systems than approximately the optimal 40 sensors in our model, but to an abstracted representation of the raw visual state further up the cognitive process. That is how near should a bird or insect be to another object in order to use valuable processing time to consider it in its estimation of the future? Our model answers this question with a range of values codified by the number of sensors in the discrete case, or the resolution parameter  $s$ . In particular for our circular particles we find a value of  $\arcsin 1/d \sim \pi/40 = 4.5^\circ$  to be an approximately optimal value of this angular size constraint for inter agent spacing  $d$ . Real biological experiments could be attempted to unpick whether there is real world evidence for such a cognitive discounting, where an objects angular size is too small to bother thinking about but large enough to make-out, and if any link can be made to the value we predict for our circular 2D agents. Whirligig beetles would make for an example group that could be probed by an experimental setup not so dissimilar to the quasi-two dimensional setup we saw in chapter 2. Confounding factors however will be that visual input is not necessarily the only way in which beetle interact with one another, for example hydrodynamic effect such as individuals sensing ripples will likely confound any data. We also determined one way in which our model could be useful in real world experiments. By tuning the parameters of orientation and speed in the action set we could drastically affect the outcome of model. In particular we uncovered a transition from order and cohesive motion to disorder but cohesive motion resembling swarming systems not unlike Whirligig beetles. Future research could attempt to fit these and other parameters to experimental data.

In chapter 4 we probed our models by modifying the cognitive process using a deterministic explicit occlusion model as well as stochastic noise process on model outputs and agent predictions of their group-mates. We determined in each case our

model is quite robust, especially to occlusion in small groups. But unexpectedly we found our model is improved, resulting in more stable groups, with large numbers of individuals (1000s). This we interpreted as an interplay between occlusion pruning the number of group-mates an agent predicts the future path of and the errors made by this prediction process. Colloquially an “analysis paralysis” effect. Obviously in the real world occlusion is natural. For us, it is likely we have this effect due to the long-ranged nature of our interactions. Unlike Couzin or Vicsek like models our agents (without occlusion) can interact over large distances. Or more accurately an agent can interact with a large group over long ranges. That is from the sensor size,  $2\pi/n_s$  and 50% activation threshold, we have  $\arcsin \frac{r}{d} \sim \frac{\pi}{2n_s}$ , then individuals interact up to a scale of  $d \sim \frac{2n_s r}{\pi}$  for agent radius  $r$  or  $d \sim n \frac{2n_s r}{\pi}$  for an  $n$  agent group treated as a circle of radius  $n$ . This gives us approximate length scale (for  $n_s = 40$ ) of  $\approx 25n$  units (in terms of agent radius) as an interaction radius. Occlusion naturally reduces long-range interaction in our model as further away agents are more likely to be occluded, however this is inextricably linked to the opacity of the group. Since our model does target marginal opacity we can expect some long-ranged interactions to persist. We introduced cognitive noise by replacing the ballistic prediction heuristic by a stochastic process and uncovered a significant increase in both global and largest cluster order, in particular this increase was associated with apply rotational noise to the prediction heuristic rather than a noisy speed. The fact we also measure a fluctuation (without any noise in the model) of a similar scale ( $5^\circ$  to  $5.79^\circ$ ) encouraged us to interpret this optimal noise as a self-consistent one. That is applying some rotational noise, commensurate with the actual model output, in the cognitive heuristic prediction results in a better prediction of the future.

We closed this thesis, in chapter 5 by relaxing the complete enumeration of the future to a sampling process of the future. Initially we focused on sampling paths and actions to build paths in the discrete action setting. Our major result was that it is beneficial in our model to search distant futures sparsely over searching near futures completely; a depth first tree search instead of a breadth first tree search. This predicts that in the real world collectively moving individuals targeting highly ordered and cohesive groups should care more about incomplete knowledge of the medium term future over complete knowledge of the near future. This prediction is perhaps somewhat difficult to test experimentally, but it is another point in our model that offers a falsifiable test. Whatever the case this does offer an avenue for obtaining more stable groups with less evaluation of visual states, which for practical implementations of our model is undoubtedly useful. We secondly find that it is better to search the future in a uniformly sampled manner (without replacement),

or at least using the information in the future states tree as in a monte-carlo tree search provides marginal benefit. Armed with this knowledge we defined a novel model using a fully continuous action space by interpreting the value of  $\pm\Delta\theta$  as a bound on a continuous selected angle increment, and  $v_0$  with  $\Delta v$  as a bound on a continuous set of speeds an agent can select. This continuous model necessitates sampling the future. We find that the model provides commensurate output to the discrete action space setting with the added benefit of more control in the action set. We propose that this extra degree of control over the actions could be used to fit the continuous model to experimental data more readily. That is one could statistically determine the action distribution from real world data, and see if our model is able to recover the same phenomenology.

Overall in this thesis we have investigated two complementary approaches to AM research. Namely, the bottom-up and top-down approaches. We defined a top-down approach as one imposing phenomenology directly in the modelling approach, e.g. specific behavioural rules. The bottom-up approach we took to mean defining a principle from which phenomenology emerges naturally. The terms themselves are relative to the modelling task, physical system, and specific phenomenology addressed. The gain is to understand not just the effect of a given phenomenology, but develop an understanding of its emergence. Here we applied these concepts to chiefly swarming, ordered motion, and collision avoidance. In chapter 2 we took a top-down approach by assuming a phenomenological torque term. By which individual beetles seek to re-orientate back to the group at large. By assuming this phenomenology and fitting data to a model we were able to investigate MIPS in the context of Whirligig beetles. We could also take a bottom-up approach by asking by which principle Whirligig beetles cluster. We may argue individuals gain safety within the group, or that perhaps that as part of a dynamic group food may be drawn up as in Phalaropes [65]. A principle such as minimising neighbour loss could be explored here, as has been done recently as a bottom-up approach to ordered motion [117]. For example a guiding principle for Whirligig beetles may be to keep neighbours close. This could be codified in a visual state by maximising coverage of the visual field. Whether this naturally leads to a re-orienting torque is speculation. But the difference is clear. Here we impose the phenomenon, but we could and should ask why the phenomenon exists at all. We can fit a model to data, but we should always ask why a particular model should fit.

The proposed framework of maximising future path entropy could further the field in a number of ways. Firstly, the stipulation of future path entropy *over visual states* is not a requirement. Any notion of state could be used within the

developed framework. This allows for the application of the framework to systems with any notion of state. For example at the microscale visual input may not make sense, but Bio-chemical input might. The difference lies purely in the state-space of the model. In a practical sense the frame specifically applied to visual processing can be cast as simple mapping between visual input and action output. Here we discretise visual input into a Boolean vector. This can be fed as input into a neural network trained to reproduce actions as per the actual model. Such a methodology drastically decreases the simulation time between visual input and agent action. These ideas could be applied in any context in which collective motion is desired. For example in video game, animation, or reinforcement learning settings. Similarly to the ubiquitous application of Reynolds' flocking algorithm in these contexts since its inception [3]. Although the question of a three-dimensional application to flocking is still open.

# Bibliography

- [1] S. Ramaswamy. “Active matter”. In: *Journal of Statistical Mechanics: Theory and Experiment* 2017.5 (2017), p. 054002.
- [2] M. Rubenstein, A. Cornejo, and R. Nagpal. “Programmable self-assembly in a thousand-robot swarm”. In: *Science* 345.6198 (2014).
- [3] C. W. Reynolds. “Flocks, Herds and Schools: A Distributed Behavioral Model”. In: *SIGGRAPH Comput. Graph.* 21.4 (Aug. 1987), pp. 25–34.
- [4] T. Vicsek et al. “Novel Type of Phase Transition in a System of Self-Driven Particles”. In: *Phys. Rev. Lett.* 75 (6 Aug. 1995), pp. 1226–1229.
- [5] A. Cavagna et al. “Scale-free correlations in starling flocks”. In: *Proceedings of the National Academy of Sciences* (2010).
- [6] A. Attanasi et al. “Information transfer and behavioural inertia in starling flocks”. In: *Nature physics* 10.9 (2014), pp. 691–696.
- [7] H. Hildenbrandt, C. Carere, and C. Hemelrijk. “Self-organized aerial displays of thousands of starlings: a model”. In: *Behavioral Ecology* 21.6 (Oct. 2010), pp. 1349–1359.
- [8] M. Nagy et al. “Hierarchical group dynamics in pigeon flocks”. In: *Nature* 464.7290 (2010), p. 890.
- [9] R. Gerum et al. “The origin of traveling waves in an emperor penguin huddle”. In: *New Journal of Physics* 15.12 (2013), p. 125022.
- [10] A. C. Gallup et al. “Visual attention and the acquisition of information in human crowds”. In: *Proceedings of the National Academy of Sciences* 109.19 (2012), pp. 7245–7250.
- [11] M. Moussaïd, D. Helbing, and G. Theraulaz. “How simple rules determine pedestrian behavior and crowd disasters”. In: *Proceedings of the National Academy of Sciences* (2011).

- [12] W. L. Romey and A. R. Lamb. “Flash Expansion Threshold in Whirligig Swarms”. In: *PLOS ONE* 10.8 (Aug. 2015), pp. 1–12.
- [13] W. L. Romey, A. L. Smith, and J. Buhl. “Flash expansion and the repulsive herd”. In: *Animal Behaviour* 110 (2015), pp. 171–178.
- [14] D. Calovi et al. “Swarming, Schooling, Milling: Phase diagram of a data-driven fish school model”. In: *New Journal of Physics* 16 (Aug. 2013).
- [15] J. Gautrais et al. “Deciphering interactions in moving animal groups”. In: (2012).
- [16] H. C. Berg. *E. coli in Motion*. Springer Science & Business Media, 2008.
- [17] I. H. Riedel, K. Kruse, and J. Howard. “A Self-Organized Vortex Array of Hydrodynamically Entrained Sperm Cells”. In: *Science* 309.5732 (2005), pp. 300–303.
- [18] B. Friedrich and F. Jülicher. “The stochastic dance of circling sperm cells: Sperm Chemotaxis in the plane”. In: *New Journal of Physics* 10 (Dec. 2008).
- [19] I. Buttinoni et al. “Active Brownian motion tunable by light”. In: *Journal of Physics: Condensed Matter* 24.28 (2012), p. 284129.
- [20] J. R. Howse et al. “Self-Motile Colloidal Particles: From Directed Propulsion to Random Walk”. In: *Phys. Rev. Lett.* 99 (4 July 2007), p. 048102.
- [21] H.-R. Jiang, N. Yoshinaga, and M. Sano. “Active Motion of a Janus Particle by Self-Thermophoresis in a Defocused Laser Beam”. In: *Phys. Rev. Lett.* 105 (26 Dec. 2010), p. 268302.
- [22] D. Nishiguchi and M. Sano. “Mesoscopic turbulence and local order in Janus particles self-propelling under an ac electric field”. In: *Phys. Rev. E* 92 (5 Nov. 2015), p. 052309.
- [23] A. Ghosh and P. Fischer. “Controlled propulsion of artificial magnetic nanostructured propellers”. In: *Nano letters* 9.6 (2009), pp. 2243–2245.
- [24] A. Bricard et al. “Emergence of macroscopic directed motion in populations of motile colloids”. In: *Nature* 503.7474 (2013), pp. 95–98.
- [25] S. Thutupalli, R. Seemann, and S. Herminghaus. “Swarming behavior of simple model squirmers”. In: *New Journal of Physics* 13.7 (2011), p. 073021.
- [26] C. Bechinger et al. “Active particles in complex and crowded environments”. In: *Reviews of Modern Physics* 88.4 (2016), p. 045006.
- [27] J. S. Olafsen and J. S. Urbach. “Clustering, Order, and Collapse in a Driven Granular Monolayer”. In: *Phys. Rev. Lett.* 81 (20 Nov. 1998), pp. 4369–4372.

- [28] I. S. Aranson, D. Volfson, and L. S. Tsimring. “Swirling motion in a system of vibrated elongated particles”. In: *Phys. Rev. E* 75 (5 May 2007), p. 051301.
- [29] J. Deseigne, O. Dauchot, and H. Chaté. “Collective motion of vibrated polar disks”. In: *Physical review letters* 105.9 (2010), p. 098001.
- [30] Y. Lanoiselée et al. “Statistical analysis of random trajectories of vibrated disks: Towards a macroscopic realization of Brownian motion”. In: *Phys. Rev. E* 98 (6 Dec. 2018), p. 062112.
- [31] C. Scholz et al. “Inertial delay of self-propelled particles”. In: *Nature communications* 9.1 (2018), p. 5156.
- [32] A. Murali et al. “Geometric constraints alter the emergent dynamics of an active particle”. In: *Phys. Rev. Research* 4 (1 Feb. 2022), p. 013136.
- [33] G. A. Patterson et al. “Clogging Transition of Vibration-Driven Vehicles Passing through Constrictions”. In: *Phys. Rev. Lett.* 119 (24 Dec. 2017), p. 248301.
- [34] J. D. Davidson et al. “Collective detection based on visual information in animal groups”. In: *Journal of The Royal Society Interface* 18.180 (2021), p. 20210142.
- [35] P. M. Chaikin, T. C. Lubensky, and T. A. Witten. *Principles of condensed matter physics*. Vol. 10. Cambridge university press Cambridge, 1995.
- [36] I. D. Couzin et al. “Collective Memory and Spatial Sorting in Animal Groups”. In: *Journal of Theoretical Biology* 218.1 (2002), pp. 1–11.
- [37] H. Hildenbrandt, C. Carere, and C. Hemelrijk. “Self-organized aerial displays of thousands of starlings: a model”. In: *Behavioral Ecology* 21.6 (Oct. 2010), pp. 1349–1359.
- [38] E. Soria, F. Schiano, and D. Floreano. “The Influence of Limited Visual Sensing on the Reynolds Flocking Algorithm”. In: *2019 Third IEEE International Conference on Robotic Computing (IRC)*. 2019, pp. 138–145.
- [39] M. Ballerini et al. “Interaction ruling animal collective behavior depends on topological rather than metric distance: Evidence from a field study”. In: *Proceedings of the National Academy of Sciences* 105.4 (2008), pp. 1232–1237.
- [40] F. Ginelli and H. Chaté. “Relevance of Metric-Free Interactions in Flocking Phenomena”. In: *Phys. Rev. Lett.* 105 (16 Oct. 2010), p. 168103.

- [41] D. S. Lemons and A. Gythiel. “Paul Langevin’s 1908 paper “On the Theory of Brownian Motion” [“Sur la théorie du mouvement brownien,” C. R. Acad. Sci. (Paris) 146, 530–533 (1908)]”. In: *American Journal of Physics* 65.11 (1997), pp. 1079–1081.
- [42] G. Volpe, S. Gigan, and G. Volpe. “Simulation of the active Brownian motion of a microswimmer”. In: *American Journal of Physics* 82.7 (2014), pp. 659–664.
- [43] P. Pearle et al. “What Brown saw and you can too”. In: *American Journal of Physics* 78.12 (2010), pp. 1278–1289.
- [44] U. Erdmann et al. “Brownian particles far from equilibrium”. In: *The European Physical Journal B-Condensed Matter and Complex Systems* 15.1 (2000), pp. 105–113.
- [45] M. E. Cates. “Diffusive transport without detailed balance in motile bacteria: does microbiology need statistical physics?” In: *Reports on Progress in Physics* 75.4 (2012), p. 042601.
- [46] F. Peruani et al. “Collective Motion and Nonequilibrium Cluster Formation in Colonies of Gliding Bacteria”. In: *Phys. Rev. Lett.* 108 (9 Feb. 2012), p. 098102.
- [47] V. Narayan, S. Ramaswamy, and N. Menon. “Long-Lived Giant Number Fluctuations in a Swarming Granular Nematic”. In: *Science* 317.5834 (2007), pp. 105–108.
- [48] Y. Fily and M. C. Marchetti. “Athermal Phase Separation of Self-Propelled Particles with No Alignment”. In: *Phys. Rev. Lett.* 108 (23 June 2012), p. 235702.
- [49] M. E. Cates and J. Tailleur. “Motility-Induced Phase Separation”. In: *Annual Review of Condensed Matter Physics* 6.1 (2015), pp. 219–244.
- [50] J. Tailleur and M. E. Cates. “Statistical Mechanics of Interacting Run-and-Tumble Bacteria”. In: *Phys. Rev. Lett.* 100 (21 May 2008), p. 218103.
- [51] I. Buttinoni et al. “Dynamical Clustering and Phase Separation in Suspensions of Self-Propelled Colloidal Particles”. In: *Phys. Rev. Lett.* 110 (23 June 2013), p. 238301.
- [52] F. Ginot et al. “Nonequilibrium Equation of State in Suspensions of Active Colloids”. In: *Physical Review X* 5 (Nov. 2014).
- [53] M. J. Schnitzer. “Theory of continuum random walks and application to chemotaxis”. In: *Phys. Rev. E* 48 (4 Oct. 1993), pp. 2553–2568.

- [54] H. Löwen. “Inertial effects of self-propelled particles: From active Brownian to active Langevin motion”. In: *The Journal of Chemical Physics* 152.4 (2020), p. 040901.
- [55] H. Mukundarajan et al. “Surface tension dominates insect flight on fluid interfaces”. In: *Journal of Experimental Biology* 219.5 (Mar. 2016), pp. 752–766.
- [56] J. Rabault, R. A. Fauli, and A. Carlson. “Curving to Fly: Synthetic Adaptation Unveils Optimal Flight Performance of Whirling Fruits”. In: *Phys. Rev. Lett.* 122 (2 Jan. 2019), p. 024501.
- [57] D. Breoni, M. Schmiedeberg, and H. Löwen. “Active Brownian and inertial particles in disordered environments: Short-time expansion of the mean-square displacement”. In: *Phys. Rev. E* 102 (6 Dec. 2020), p. 062604.
- [58] S. Mandal, B. Liebchen, and H. Löwen. “Motility-Induced Temperature Difference in Coexisting Phases”. In: *Phys. Rev. Lett.* 123 (22 Nov. 2019), p. 228001.
- [59] A. Suma et al. “Motility-induced phase separation in an active dumbbell fluid”. In: *EPL (Europhysics Letters)* 108.5 (Dec. 2014), p. 56004.
- [60] T. J. Pitcher and C. J. Wyche. “Predator-avoidance behaviours of sand-eel schools: why schools seldom split”. In: *Predators and prey in fishes: Proceedings of the 3rd biennial conference on the ethology and behavioral ecology of fishes, held at Normal, Illinois, U.S.A., May 19–22, 1981*. Ed. by D. L. G. Noakes et al. Dordrecht: Springer Netherlands, 1983, pp. 193–204.
- [61] M. Zheng et al. “Behavior pattern (innate action) of individuals in fish schools generating efficient collective evasion from predation”. In: *Journal of Theoretical Biology* 235.2 (2005), pp. 153–167.
- [62] S.-H. Lee, H. Pak, and T.-S. Chon. “Dynamics of prey-flock escaping behavior in response to predator’s attack”. In: *Journal of Theoretical Biology* 240.2 (2006), pp. 250–259.
- [63] Y. Inada and K. Kawachi. “Order and Flexibility in the Motion of Fish Schools”. In: *Journal of Theoretical Biology* 214.3 (2002), pp. 371–387.
- [64] A. Ancel et al. “Energy saving in huddling penguins”. In: *Nature* 385.6614 (1997), pp. 304–305.
- [65] B. S. Obst et al. “Kinematics of phalarope spinning”. In: *Nature* 384.6605 (Nov. 1996), pp. 121–121.
- [66] A. Eriksson et al. “Determining interaction rules in animal swarms”. In: *Behavioral Ecology* 21.5 (Sept. 2010), pp. 1106–1111.

- [67] R. Lukeman, Y.-X. Li, and L. Edelstein-Keshet. “Inferring individual rules from collective behavior”. In: *Proceedings of the National Academy of Sciences* 107.28 (2010), pp. 12576–12580.
- [68] Y. Katz et al. “Inferring the structure and dynamics of interactions in schooling fish”. In: *Proceedings of the National Academy of Sciences* 108.46 (2011), pp. 18720–18725.
- [69] J. E. Herbert-Read et al. “Inferring the rules of interaction of shoaling fish”. In: *Proceedings of the National Academy of Sciences* 108.46 (2011), pp. 18726–18731.
- [70] A. Zienkiewicz et al. “Leadership emergence in a data-driven model of zebrafish shoals with speed modulation”. In: *The European Physical Journal Special Topics* 224.17 (2015), pp. 3343–3360.
- [71] H. J. Charlesworth and M. S. Turner. “Intrinsically motivated collective motion”. In: *Proceedings of the National Academy of Sciences* 116.31 (2019), pp. 15362–15367.
- [72] H. J. Charlesworth. “Future state maximisation as an intrinsic motivation for decision making”. July 2019.
- [73] D. J. G. Pearce et al. “Role of projection in the control of bird flocks”. In: *Proceedings of the National Academy of Sciences* 111.29 (2014), pp. 10422–10426.
- [74] P. Kunszt, A. Szalay, and A. Thakar. “The Hierarchical Triangular Mesh”. In: Feb. 2006, pp. 631–637.
- [75] R. S. Sutton and A. G. Barto. *Reinforcement learning: An introduction*. MIT press, 2018.
- [76] I. Goodfellow, Y. Bengio, and A. Courville. *Deep learning*. MIT press, 2016.
- [77] G. M. J.-B. Chaslot et al. “Progressive Strategies for Monte-Carlo Tree Search”. In: *New Mathematics and Natural Computation* 04.03 (2008), pp. 343–357.
- [78] D. Silver et al. “Mastering the game of Go with deep neural networks and tree search”. In: *nature* 529.7587 (2016), pp. 484–489.
- [79] S. H. Cerezo and G. D. Ballester. “Fractal AI: A fragile theory of intelligence”. In: *arXiv preprint arXiv:1803.05049* (2018).
- [80] A. D. Wissner-Gross and C. E. Freer. “Causal Entropic Forces”. In: *Phys. Rev. Lett.* 110 (16 Apr. 2013), p. 168702.

- [81] ImageMagick Studio LLC. *ImageMagick*. Version 7.0.10. Jan. 4, 2023.
- [82] S. Kumar et al. “Detecting and tracking high density homogeneous objects in low framerate videos”. In: *in preparation* (2021).
- [83] L. Huang et al. “Densebox: Unifying landmark localization with end to end object detection”. In: *arXiv preprint arXiv:1509.04874* (2015).
- [84] A. Pérez-Escudero et al. “idTracker: tracking individuals in a group by automatic identification of unmarked animals”. In: *Nature methods* 11.7 (2014), pp. 743–748.
- [85] R. E. Kalman. “A new approach to linear filtering and prediction problems”. In: (1960).
- [86] Li Zhang, Yuan Li, and R. Nevatia. “Global data association for multi-object tracking using network flows”. In: *2008 IEEE Conference on Computer Vision and Pattern Recognition*. 2008, pp. 1–8.
- [87] H. W. Kuhn. “The Hungarian method for the assignment problem”. In: *Naval research logistics quarterly* 2.1-2 (1955), pp. 83–97.
- [88] K. Bernardin and R. Stiefelhagen. “Evaluating Multiple Object Tracking Performance: The CLEAR MOT Metrics”. In: *J. Image Video Process.* 2008 (Jan. 2008).
- [89] R. T. Rockafellar. *Convex Analysis*. Princeton University Press, 2015.
- [90] H. Edelsbrunner, D. Kirkpatrick, and R. Seidel. “On the shape of a set of points in the plane”. In: *IEEE Transactions on Information Theory* 29.4 (1983), pp. 551–559.
- [91] H. Edelsbrunner and E. P. Mücke. “Three-dimensional alpha shapes”. In: *ACM Transactions on Graphics (TOG)* 13.1 (1994), pp. 43–72.
- [92] M. M. G. Sosna et al. “Individual and collective encoding of risk in animal groups”. In: *Proceedings of the National Academy of Sciences* 116.41 (2019), pp. 20556–20561.
- [93] F. I. Pelupessy and W. Schaap. “Density estimators in particle hydrodynamics-DTFE versus regular SPH”. In: *Astronomy & Astrophysics* 403.2 (2003), pp. 389–398.
- [94] H. L. Devereux et al. “Whirligig beetles as corralled active Brownian particles”. In: *Journal of The Royal Society Interface* 18.177 (2021), p. 20210114.

- [95] C. B. Barber, D. P. Dobkin, and H. Huhdanpaa. “The Quickhull Algorithm for Convex Hulls”. In: *ACM Trans. Math. Softw.* 22.4 (Dec. 1996), pp. 469–483.
- [96] W. F. Spitz. “High-order compact finite difference schemes for computational mechanics”. PhD thesis. The University of Texas at Austin, 1995.
- [97] P. E. Kloeden and E. Platen. *Numerical Solution of Stochastic Differential Equations*. Springer Berlin Heidelberg, 1992.
- [98] N. Grønbech-Jensen and O. Farago. “A simple and effective Verlet-type algorithm for simulating Langevin dynamics”. In: *Molecular Physics* 111.8 (2013), pp. 983–991.
- [99] F. Nogueira. *Bayesian Optimization: Open source constrained global optimization tool for Python*. 2014–.
- [100] S. H. Rudy et al. “Data-driven discovery of partial differential equations”. In: *Science Advances* 3.4 (2017), e1602614.
- [101] R. T. Chen et al. “Neural ordinary differential equations”. In: *Advances in neural information processing systems* 31 (2018).
- [102] B. Tzen and M. Raginsky. “Neural stochastic differential equations: Deep latent gaussian models in the diffusion limit”. In: *arXiv preprint arXiv:1905.09883* (2019).
- [103] R. Cagan. “Principles of Drosophila eye differentiation”. In: *Current topics in developmental biology* 89 (2009), pp. 115–135.
- [104] C. E. Shannon. “A mathematical theory of communication”. In: *The Bell System Technical Journal* 27.3 (1948), pp. 379–423.
- [105] D. A. Huffman. “A Method for the Construction of Minimum-Redundancy Codes”. In: *Proceedings of the IRE* 40.9 (1952), pp. 1098–1101.
- [106] A. Robinson and C. Cherry. “Results of a prototype television bandwidth compression scheme”. In: *Proceedings of the IEEE* 55.3 (1967), pp. 356–364.
- [107] P. Elias. “Universal codeword sets and representations of the integers”. In: *IEEE Transactions on Information Theory* 21.2 (1975), pp. 194–203.
- [108] M. Thomas and A. T. Joy. *Elements of information theory*. 2006.
- [109] *IEEE Standard for Floating-Point Arithmetic*. 2019, pp. 1–84.
- [110] M. Ester et al. “A density-based algorithm for discovering clusters in large spatial databases with noise.” In: *kdd*. 34. 1996, pp. 226–231.

- [111] R. Olfati-Saber. “Flocking for multi-agent dynamic systems: algorithms and theory”. In: *IEEE Transactions on Automatic Control* 51.3 (2006), pp. 401–420.
- [112] R. Olfati-Saber, J. A. Fax, and R. M. Murray. “Consensus and Cooperation in Networked Multi-Agent Systems”. In: *Proceedings of the IEEE* 95.1 (2007), pp. 215–233.
- [113] H. G. Tanner, A. Jadbabaie, and G. J. Pappas. “Flocking in fixed and switching networks”. In: *IEEE Transactions on Automatic control* 52.5 (2007), pp. 863–868.
- [114] R. L. Graham, D. E. Knuth, and O. Patashnik. *Concrete Mathematics: A Foundation for Computer Science*. Reading: Addison-Wesley, 1989.
- [115] R. M. Karp. “On-line algorithms versus off-line algorithms: How much”. In: *Algorithms, Software, Architecture: Information Processing 92: Proceedings of the IFIP 12th World Computer Congress*. Vol. 1. 1992, p. 416.
- [116] D. Bertsimas and J. Tsitsiklis. “Simulated Annealing”. In: *Statistical Science* 8.1 (1993), pp. 10–15.
- [117] M. Durve, F. Peruani, and A. Celani. “Learning to flock through reinforcement”. In: *Phys. Rev. E* 102 (1 July 2020), p. 012601.

## Appendix A

# Online Entropy Calculation

The Monte-Carlo tree search we define in algorithm 4 utilises a backpropagation technique (algorithm 5) to efficiently add visual states to previous nodes when sampling a path. Whilst doing this we also update node quality (entropy). When the entropy is simply the number of boundaries on the visual states' entropy is the number of boundaries on the visual state, this can be done via a trivial online average/accumulation algorithm. Updating the entropy as calculated by the empirical count distribution, can also be done online.

To see this consider a categorical distribution computed from a data stream,

$$C_k = \{C_k^{(1)}, C_k^{(2)}, \dots, C_k^{(N)}\}, \quad (\text{A.1})$$

where  $k$  is an integer index denoting the number of elements seen in the data stream.  $N$  denotes the number of categories seen in the stream. And  $C_k^{(i)}$  is the count of element  $i$  in the stream after seeing  $k$  items. We assume the number  $N$  of possible items is unbounded. Using these definitions the entropy of the categorical distribution implied by the counts  $C_k$  is

$$H_k = - \sum_i \frac{C_k^{(i)}}{\sum_j C_k^{(j)}} \log_b \frac{C_k^{(i)}}{\sum_j C_k^{(j)}} = - \sum_i p_k^{(i)} \log p_k^{(i)}, \quad (\text{A.2})$$

for some base  $b$ , where we define  $p_k^{(i)} = \frac{C_k^{(i)}}{\sum_j C_k^{(j)}}$  as the probability of item  $k$  given by  $C_k$ .

When one new item arrives we have two cases (1) the item has not been seen previously (2) the item has been seen previously. The entropies in each case can be

written as

$$H_{k+1} = \begin{cases} \frac{N}{N+1} H_k + \log_b(N+1) - \frac{N}{N+1} \log_b N & (1) \\ \frac{N}{N+1} (H_k + p_k^{(l)} \log_b p_k^{(l)}) - \frac{N}{N+1} \log_b \frac{N}{N+1} (1 - p_k^{(l)}) - \frac{N p_k^{(l)} + 1}{N+1} \log_b \frac{N p_k^{(l)} + 1}{N+1} & (2) \end{cases} \quad (\text{A.3})$$

Of course the equation for (2) can be used in both cases, noting that setting  $p_l = 0$  means case (2) is case (1). For (1) we can write the entropy as

$$H_{k+1} = - \sum_i \frac{N}{N+1} p_k^{(i)} \log_b \frac{N}{N+1} p_k^{(i)} - \frac{1}{N+1} \log_b \frac{1}{N+1}, \quad (\text{A.4})$$

since the single new element is unique. Then we remove the factors of  $\frac{N}{N+1}$  from the sum to obtain

$$H_{k+1} = - \frac{N}{N+1} \left[ \sum_i p_k^{(i)} \left( \log_b p_k^{(i)} + \log_b \frac{N}{N+1} \right) \right] - \frac{1}{N+1} \log_b \frac{1}{N+1}, \quad (\text{A.5})$$

$$= \frac{N}{N+1} H_k - \frac{1}{N+1} \left( N \log_b \frac{N}{N+1} + \log_b \frac{1}{N+1} \right), \quad (\text{A.6})$$

$$= \frac{N}{N+1} H_k - \frac{1}{N+1} \left( N \log_b N - (N+1) \log_b(N+1) \right). \quad (\text{A.7})$$

Which yields (1) after dealing with the constant via logarithm identities. For (2) begin by noting that only for one  $l \in 1, 2, \dots, N$  is  $C_{k+1}^{(l)} \neq C_k^{(l)}$ , as only one new element arrives, so we can write the entropy as,

$$H_{k+1} = - \sum_{i \neq l} \frac{N}{N+1} p_k^{(i)} \log_b \frac{N}{N+1} p_k^{(i)} - \frac{N p_k^{(l)} + 1}{N+1} \log_b \frac{N p_k^{(l)} + 1}{N+1}. \quad (\text{A.8})$$

The next step is to remove the  $\frac{N}{N+1}$  factors in the sum resulting in

$$- \sum_{i \neq l} \frac{N}{N+1} p_k^{(i)} \log_b \frac{N}{N+1} p_k^{(i)} = - \frac{N}{N+1} \left[ \sum_{i \neq l} p_k^{(i)} \left( \log_b p_k^{(i)} + \log_b \frac{N}{N+1} \right) \right], \quad (\text{A.9})$$

$$= \frac{N}{N+1} (H_k + p_k^{(l)} \log_b p_k^{(l)}) - \frac{N}{N+1} \log_b \frac{N}{N+1} (1 - p_k^{(l)}). \quad (\text{A.10})$$

Which when used with equation A.8 yields (2).

For completeness the counts and probabilities can be trivially updated by

$$p_{k+1}^{(i)} = \begin{cases} p_k^{(i)} & \text{i not observed at step k} \\ \frac{Np_k^{(i)}+1}{N+1} & \text{i observed at step k} \end{cases}, \quad (\text{A.11})$$

$$C_{k+1}^{(i)} = \begin{cases} C_k^{(i)} & \text{i not observed at step k} \\ C_k^{(i)} + 1 & \text{i observed at step k} \end{cases}. \quad (\text{A.12})$$

In terms of complexity starting from scratch each time requires 1 integer addition,  $2(N+1)$  floating point multiplications and  $N$  float-logarithm evaluations when the new element is unseen. That is each probability needs to be updated as  $(Np_k^{(i)})/(N+1)$ , with the new  $p_k^{(i)} = N/(N+1)$ , with  $N/(N+1)$  calculated once, plus the multiplication via each logarithm. Compared to 1 integer addition, 3 floating point multiplications and 2 float-logarithm evaluations for case A.3 (1). With a previously seen new element, naïvely, we need 1 integer addition, 1 float addition,  $2(N+1)$  floating point multiplications and  $N$  float-logarithm evaluations. Compared to 1 integer addition, 8 floating point multiplications, 4 float-additions, and 3 float-logarithm evaluations for case A.3 (2).

## Appendix B

# Maximum Discrete State Entropy

Calculating maximum entropy for both the CE and SCE models may be useful practically for normalisation purposes. For example within the sampling algorithm, algorithm 4. Additionally, when these maxima occur is of theoretical interest to the models, and their predictions.

For the CE model the maximum entropy is easy to find since there can be at most  $2N$  angles making up  $N$  agent visual projections on  $[0, 2\pi)$  even in the case of the  $0-2\pi$  boundary. Since the entropy we calculate is simply the number of boundaries on the visual state  $2N$  is the maximum value. Along any one branch then we can use this result to normalise entropies.

For the SCE model the maximum value of the entropy along a branch can be found by noting that the nodes at depth  $1 \leq t \leq \tau$  feature in  $|\mathcal{A}|^{\tau-t}$  paths reflecting the  $\tau - t$  possible action choices made along a path (so leaf nodes at  $t = \tau$  appear in only 1 “path” taking  $x^0 = 1$ ) and there are  $|\mathcal{A}|^{t-1}$  nodes at depth  $t$ . The product of these numbers means that all the nodes at depth  $1 \leq t \leq \tau$  contribute  $|\mathcal{A}|^{\tau-1}$  visual states to the count distribution. The visual state from the root node (the current visual state) features in  $|\mathcal{A}|^{\tau-1}$  paths, just like the depth  $t = 1$  nodes since we consider one branch at a time.

Since there are  $\tau$  depths and a root node this gives  $\sum_{\psi} n(\psi) = (\tau + 1)|\mathcal{A}|^{\tau-1}$  visual states in total along one branch. Given the assumption all visual states are unique (when considering nodes not paths)  $|\mathcal{A}|^{\tau-1}$  of these visual states (the leaf nodes) may be unique on the branch, with all the others having degeneracy  $|\mathcal{A}|^{\tau-t}$  for  $1 \leq t \leq \tau$  and  $|\mathcal{A}|^{\tau-1}$  for the root. Since a node at depth  $t$  features in  $|\mathcal{A}|^{\tau-t}$  paths, but it’s visual state,  $\phi$ , no-where else then  $\frac{n(\phi)}{\sum_{\psi} n(\psi)} = \frac{|\mathcal{A}|^{\tau-t}}{(\tau+1)|\mathcal{A}|^{\tau-1}} = \frac{|\mathcal{A}|^{1-t}}{1+\tau}$ .

From this the maximum entropy is

$$\max(H_{\beta_i^t}) = -\frac{1}{\tau+1} \log_2\left(\frac{1}{\tau+1}\right) - \sum_{t=1}^{\tau} \frac{1}{\tau+1} \log_2\left(\frac{|\mathcal{A}|^{1-t}}{\tau+1}\right), \quad (\text{B.1})$$

where the first term corresponds to the root node. The sum in equation B.1 is taken over depths in the tree and the coefficient is obtained from the count  $|\mathcal{A}|^{t-1}$  of nodes at depth  $1 \leq t \leq \tau$  multiplied by their shared degeneracy  $\frac{|\mathcal{A}|^{1-t}}{1+\tau}$ .

Equation B.1 is also useful for sampling in the SCE model case since we can use the same logic for any internal node we simply need to be careful what  $t$  and  $\tau$  are replaced with. For example if the tree has depth  $\tau = 6$ , and we want to know the maximum possible entropy back-propagated to a node a depth  $t = 5$  we can make use of equation B.1 by considering  $t' = 0$  the future time 5 and then summing up to a new  $t' = 1$  implicitly the horizon  $\tau' = 1$ . This fact could be used to judge actions as sampling occurs in the MCTS algorithm, algorithm 4.