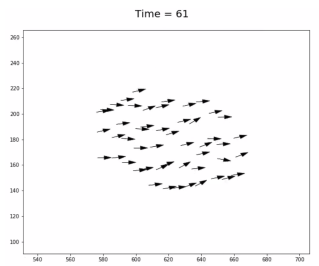
Supplementary Information

Henry J. Charlesworth and Matthew S. Turner

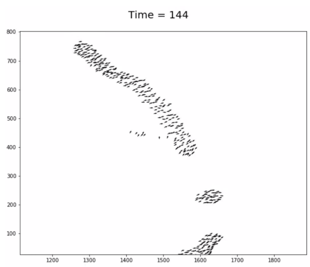
May 4, 2019

1 Supporting Video legends

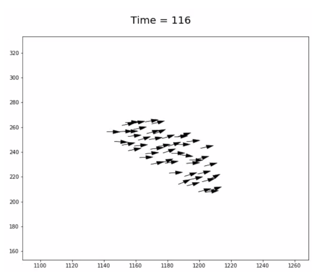
1. **SI Movie 1** Dynamical trajectories that emerge spontaneously for agents moving according to FSM, i.e. maximising their future accessible states. Here we use a standard set of parameters $N = 50, \tau = 4, n_s = 40, \Delta\theta = 15°, v_0 = 10, \Delta v = 2$. The agents are initially placed at random in a 100×100 box (all lengths in units of the particle size) and aligned with the x-axis up to a Gaussian-distributed noise on each agents orientation, with standard deviation $\sigma = \Delta\theta$. When modelling future trajectories each agent here assumes that all other agents will continue to move ballistically, i.e. at the same orientation at nominal speed v_0.



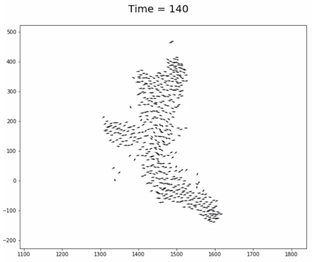
2. **SI Movie 2** Dynamical trajectories under FSM as described in caption to Movie S1 but with $N = 500, \tau = 6$.



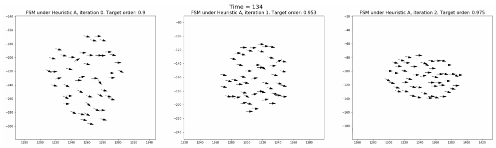
3. **SI Movie 3** Dynamical trajectories obtained under FSM with standard parameter values but where the visual difference between states is computed using a continuous sensor model, as described in the next section below. The trajectories are qualitatively similar to those shown in Movie S1, with a slightly higher density.



4. **SI Movie 4** Dynamical trajectories using FSM as described in caption to Movie S3, but with $N = 500, \tau = 5$.

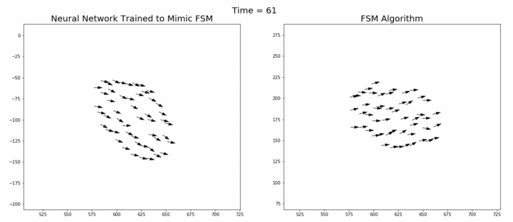


5. **SI Movie 5** Showing the first three iterations of FSM dynamics arising under Heuristic A, as reported in Fig 3 of the main text. Parameter values otherwise as described in caption to Movie S1. Stability improves with increasing self-consistency (order).



6. **SI Movie 6**

   A comparison of the trained neural network (left) mimicking the full FSM algorithm (right). The neural network is trained on 10 million training examples from the FSM algorithm over a variety of initial conditions with standard parameter values as given above and under the ballistic assumption for the future trajectories of other agents.



2 A continuous measure of visual degeneracy

Variations of our model are able to treat the overlap between visual states in a more continuous manner. In the main text we adopt a model that has the benefit of extreme simplicity — states are included simply according to whether they are distinct from all other states accessible on the decision tree and discarded if they are identical to any one of them. As before we define the visual state in terms of the geometric projection of the other agents but now assume a continuous sensor (infinite n_s). We then define a real-valued difference between any two visual states $f_i(\theta)$ and $f_j(\theta)$, each a function on $[0, 2\pi)$ that takes value 0 or 1, depending on whether the projection of (any) other agent occupies the sensor angle θ or not. The difference d_{ij} between these two states is defined as the fraction of the angular view over which they have different values, as shown in figure 1.

$$d_{ij} = \frac{1}{2\pi} \int_0^{2\pi} \left[f_i(\theta) \left(1 - f_j(\theta)\right) + f_j(\theta) \left(1 - f_i(\theta)\right) \right] d\theta \qquad (1)$$

We then generalize the FSM dynamics by using exactly the same algorithm outlined in the main text but rather than counting the **number** of **distinct** visual states accessible on each branch of the decision tree (with each realized state therefore counted as 1, if distinct, or 0, if degenerate) we instead assign a weight to each node that is the **average difference between this and every**

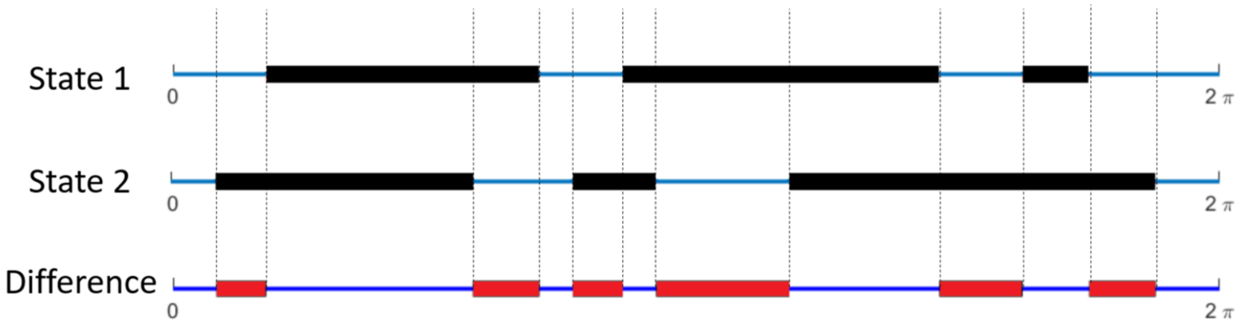


Figure 1: (SI) The difference between two continuous visual states. The continuous visual field of an agent can be constructed by taking a continuous projection of the other agents. This is similar to that shown in Fig 1(b) in the main text but without the discretisation that divides the angular field of view into a fixed number of sensors, i.e. the sensor is continuous. We then define a measure of the difference d_{ij} between any two visual states (such as states 1 and 2 shown) as the fraction of the angular field $[0, 2\pi)$ where they have different values (1/0 or 0/1 respectively), as shown (red).

other state accessible on branch α. When summed over all nodes on that branch this gives the weight of that branch, W_α. With n_α the number of nodes on the α branch of the decision tree (after truncation on/after collisions, as before) this is therefore defined as

$$W_\alpha = \frac{1}{n_\alpha} \sum_{i=1}^{n_\alpha} \sum_{j=1}^{n_\alpha} d_{ij} \tag{2}$$

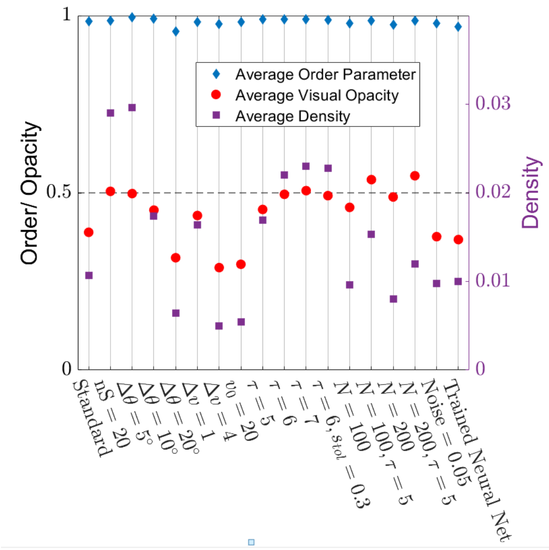
FSM dynamics then involves the agents choosing whichever move in the present time t corresponds to the branch of the decision tree with the highest score W_α. Movies showing the trajectories that emerge under this modification to our algorithm are shown in SI Movies 3 and 4. For smaller swarms the dynamics can be seen to be qualitatively extremely similar to those realized under the discrete version of FSM analysed in the main text. However, for larger swarms this continuous measure gives rise to much richer dynamics and is more robust to fragmentation.

3 Calculation of the correlation length

A correlation length is reported in Fig 2(b) in the main text. As in [7] we calculate the correlation function of the velocity in the centre of mass frame as a function of the distance between agents. That is, for each agent we define its velocity difference $\mathbf{u_i} = \mathbf{v_i} - \langle \mathbf{v} \rangle$ where $\mathbf{v_i}$ is the agent's instantaneous velocity and $\langle \mathbf{v} \rangle$ is the instantaneous average velocity of the centre of mass of all agents. We then compute the pairwise average of $\mathbf{u_i}.\mathbf{u_j}$ and record this as a function of the distance between all pairs of agents. This is negative at large distances, becoming positive with decreasing separation. The *correlation length* is then defined as the point that this correlation function changes sign. It is this value that is reported in Fig 2(b) in the main text.

4 Robustness and collisions

In figure 2 we report the swarm's average order, opacity, density, and collision rates for a variety of parameter variations of the base model, as described in the main text.



Parameters	Order	Density	Opacity	Diffusion Constant	Collisions per Step /N	Collisions per Step MFT / N
Standard	0.9832	0.0107	0.3887	4.3	0.00048	0.2202
n_s = 20	0.9849	0.0290	0.5043	4.0	0.0278	0.5984
$\Delta\theta$ = 5°	0.9958	0.0296	0.4979	2.6	0.0011	0.3976
$\Delta\theta$ = 10°	0.9914	0.0174	0.4513	2.85	0.00066	0.2462
$\Delta\theta$ = 20°	0.9555	0.0064	0.3169	14.1	0.00046	0.4170
Δv = 1	0.9820	0.0164	0.4360	5.2	0.00134	0.4214
Δv = 4	0.9753	0.0050	0.2891	8.3	0.00016	0.1889
v_0 = 20	0.9812	0.0054	0.2985	17.2	0.0008	0.4248
τ = 5	0.9900	0.0169	0.4531	1.92	0.0041	0.1607
τ = 6	0.9899	0.0220	0.4958	1.98	0.012	0.2202
τ = 7	0.9888	0.0230	0.5062	2.27	0.017	0.2648
τ = 6, s_{tol} = 0.3	0.9878	0.0228	0.4921	2.77	0.0166	0.3201
N = 100	0.9778	0.0096	0.4590	9.9	0.0010	0.4513
N = 100, τ = 5	0.9863	0.0153	0.5373	4.67	0.0063	0.3512
N = 200	0.9743	0.0080	0.4883	15.5	0.0011	0.5814
N = 200, τ = 5	0.9850	0.012	0.5482	5.9	0.0057	0.3417
Noise = 0.05	0.9787	0.0098	0.3764	4.81	0.00082	0.2237
Trained Neural Net	0.9680	0.0100	0.3682	5.17	0.0066	0.2462

Figure 2: (a) Robustness to variation of simulation parameter values. The visual opacity is the average fraction of sensors reading 1 and remains near 0.5; the density is N divided by the area within a convex hull containing the agents and remains a few %; the order is the average alignment and remains within a few % of perfect alignment. The "standard" set of parameters are $N = 50$, $\tau = 4$, $n_s = 40$, $\Delta\theta = 15°$, $v_0 = 10$, $\Delta v = 2$, $s_{tol} = 0.5$ and noise $= 0$. The entries are labelled (bottom) with any parameters that differ from these values. Here s_{tol} refers to the fraction of a sensor which needs to be filled for it to read 1 instead of 0. The noise value is the standard deviation of a Normal random variable with mean zero, which is added to both the angular orientation and the speed of each agent at each time step after it has selected a preferred move under FSM. This represents the only occasion that any extrinsic noise is introduced anywhere in this work. (b) Table showing average measures of collective FSM dynamics under various parameter values. The average order, density and opacity values correspond to the values shown in (a). We estimated the average diffusion constant of the agents relative to the centre of mass D_{eff} by fitting the average mean-squared centre of mass frame displacement after time t to $4D_{\text{eff}}\, t$ (in the linear regime). We then report a mean field estimate for the rate of collisions that we would expect for agents moving randomly with this diffusion constant at the average density realised under these conditions. This represents a control value for the collision rate under such a Mean Field Theory (MFT) and is shown in the last column. Also reported is the actual rate of collisions realized in the FSM algorithm, shown in the penultimate column. Here collisions are assumed to occur in continuous time. Trajectories of each particles are interpolated between discrete time points and solid body overlap between any pair of agents (unit radius disks) at any moment in time is deemed to constitute a collision. In both columns the collision rate is the total number of collisions divided by the number of agents N, i.e. the collision rate per agent per time step. Note the substantial reduction in actual collisions from the MFT estimate, indicating natural collision avoidance under FSM.