Article

Adaptive Behavior

Adaptive Behavior
2022, Vol. 30(1) 19–35
© The Author(s) 2020
Article reuse guidelines:
sagepub.com/journals-permissions
DOI: 10.1177/1059712320930418
journals.sagepub.com/home/adb
SAGE

# Experimental capabilities and limitations of a position-based control algorithm for swarm robotics

Yating Zheng[1,2], Cristián Huepe[1,3,4] and Zhangang Han[1]

## Abstract

Achieving efficient and reliable self-organization in groups of autonomous robots is a fundamental challenge in swarm robotics. Even simple states of collective motion, such as group translation or rotation, require nontrivial algorithms, sensors, and actuators to be achieved in real-world scenarios. We study here the capabilities and limitations in controlling experimental robot swarms of a decentralized control algorithm that only requires information on the positions of neighboring agents, and not on their headings. Using swarms of e-Puck robots, we implement this algorithm in experiments and show its ability to converge to self-organized collective translation or rotation, starting from a state with random orientations. Through a simple analytical calculation, we also unveil an essential limitation of the algorithm that produces small persistent oscillations of the aligned state, related to its marginal stability. By comparing predictions and measurements, we compute the experimental noise distributions of the linear and angular robot speeds, showing that they are well described by Gaussian functions. We then implement simulations that model this noise by adding Gaussian random variables with the experimentally measured standard deviations. These simulations are performed for multiple parameter combinations and compared to experiments, showing that they provide good predictions for the expected speed and robustness of the self-organizing dynamics.

**Keywords**

Collective motion, decentralized control, position-based models, self-organization, swarm robotics

Handling Editor: Georg Martius, Max Planck Institute for Intelligent Systems, Germany

## 1. Introduction

One of the first challenges of swarm robotics is to develop decentralized control algorithms that can lead groups of autonomous robots to self-organize into coherent states of collective motion (Brambilla et al., 2013). A variety of such algorithms have been implemented in simulations, mostly trying to reproduce the collective dynamics observed in biological systems. They have tried to imitate, for example, the dynamics of bacterial colonies (Zhang et al., 2010), ant groups (Mersch et al., 2013), fish schools (Calovi et al., 2014; Gautrais et al., 2008; Tunstrøm et al., 2013), and bird flocks (Bhattacharya & Vicsek, 2010; Cavagna et al., 2013; Toner & Tu, 1995). These bioinspired algorithm could help develop a set of simple core rules for controlling robot swarms and performing other predetermined collective tasks, such as path planning (Sartoretti et al., 2014), spatial formations (Kushleyev et al., 2013; Mathews et al., 2017; Rubenstein et al., 2014; Wang et al., 2014), or collective decision making (Vigelius et al., 2014).

Many of the current collective motion algorithms have been strongly influenced by the Vicsek model (Chaté et al., 2008; Vicsek et al., 1995; Vicsek &

[1] School of Systems Science, Beijing Normal University, Beijing, China
[2] IRIDIA, Université Libre de Bruxelles, Brussels, Belgium
[3] ESAM and Northwestern Institute on Complex Systems, Northwestern University, Evanston, IL, USA
[4] Cristian Huepe Labs, Chicago, IL, USA

**Corresponding authors:**
Yating Zheng, School of Systems Science, Beijing Normal University, Beijing 100875, China.
Email: zhengyating@mail.bnu.edu.cn

Zhangang Han, School of Systems Science, Beijing Normal University, Beijing 100875, China.
Email: zhan@bnu.edu.cn

Zafeiris, 2012). This is a minimal model that is purely based on alignment interactions, where each agent advances at a fixed speed and tends to align to the mean heading direction of its neighbors. When simulated in a two-dimensional periodic box with noise levels below a critical value, all agents will become approximately aligned and thus move in a common direction. More complex models that use a similar alignment-based mechanism to achieve self-organization have also been introduced. The model in Couzin et al. (2002), for example, considers a three-dimensional system and adds attraction–repulsion interactions to generate group aggregation and avoid collisions between agents. More recently, other models have experiments to address how interactions could depend on other factors, such as agent speed or the choice of interacting neighbors (Bialek et al., 2012; Gautrais et al., 2012; Katz et al., 2011). Despite the differences between these models, they all strongly rely on explicit, Vicsek-like aligning interactions to self-organize into collective motion.

In the context of swarm robotics, a strong reliance on alignment interactions can have several disadvantages. For example, components that detect relative orientations are less common than those measuring relative positions. In addition, each robot will typically have to obtain information not only on the headings of other agents (to implement the alignment-based algorithm) but also on their relative positions (in order to avoid collisions and group dispersion). This implies that all robots must have the necessary hardware to detect both quantities, which increases their cost and complexity. Furthermore, although aligning interactions are effective in achieving self-organized collective translational motion, they typically cannot produce other states of coherent collective motion, such as group rotation[1] (Gautrais et al., 2012). It would therefore be beneficial to implement in robot swarms control algorithms that can achieve collective motion without requiring aligning interactions, that is, without requiring the exchange of orientation information between agents. One such algorithm[2] is the Active-Elastic (AE) model, which was recently proposed theoretically in Ferrante et al. (2013a, 2013b) and first tested experimentally in Ferrante et al. (2012).

In this work, we implement the AE model as a position-based decentralized motion control algorithm for a collective robotics experiment, exploring its advantages and disadvantages in a real-world setting. We show that a system of up to seven robots initially in a nonaligned state can robustly self-organize to a common heading direction and achieve collective motion, even in an experimental setup with significant sources of noise and long processing time-delays. Despite this success in achieving collective motion, we also observe that the AE algorithm is intrinsically prone to small-scale oscillations, which can eventually lead to instabilities. We will identify the origin of these oscillations with a simple analytical calculation that considers only two agents. We then characterize the experimental noise and processing times introduced by real-world limitations, showing that we can closely reproduce the robot dynamics when these are added to our numerical simulations. We thus demonstrate that it is possible to predict the optimal experimental parameter combinations leading to self-organization by simulating the observed noise. Finally, we show that a small variation of the AE algorithm can achieve instead self-organization into collective rotation.

The article is organized as follows. In Section 2, we describe the AE algorithm in its dimensional and non-dimensional forms, its self-organizing mechanism, and the order parameters used to monitor its collective states. Section 3 presents our experimental setup and the tests we performed for its validation. In Section 4, we describe the typical self-organizing dynamics observed in our robotic system and characterize its stability and experimental noise. We then compare in Section 5 the typical collective dynamics in our experiments and simulations for different regions of the parameter space. In Section 6, we introduce a variation of the AE algorithm that leads to collective rotation in our experiments. Finally, Section 7 discusses the capabilities and limitations of the AE model for controlling robot swarms, in light of our experimental and numerical results, and presents our conclusions.

## 2. Position-based control algorithm

In this section, we describe the AE control algorithm and order parameters used in our swarm robotics experiments.

### 2.1. The AE model

We implemented a decentralized swarm robotics control algorithm based on the AE model introduced in Ferrante et al. (2013a, 2013b). This minimal model was shown to produce self-organized collective motion, even in the presence of noise, when starting from a group of agents with random initial headings. It only requires the exchange of positional information between agents, without requiring any explicit alignment interaction or the exchange of orientation or velocity information.

We used as the control algorithm a noiseless AE model and defined all the interaction forces between agents as having the same strength $k_0$ and equilibrium distance $l_0$. With these simplifications, the AE model is defined as follows

$$\dot{\vec{x}}_i = \left(v_0 + \alpha \vec{F}_i \cdot \hat{n}_i\right)\hat{n}_i \tag{1}$$

$$\dot{\theta}_i = \beta \vec{F}_i \cdot \hat{n}_i^{\perp} \tag{2}$$

with

$$\vec{F}_i = \frac{k}{l_0}\sum_{j\in S_i}\left(\|\vec{r}_{ij}\| - l_0\right)\frac{\vec{r}_{ij}}{\|\vec{r}_{ij}\|}. \tag{3}$$

Here, $\vec{x}_i$ and $\theta_i$ represent the position and heading direction of agent $i$, respectively. Unit vector $\hat{n}_i$ points parallel to the agent's heading direction and $\hat{n}_i^{\perp}$ points perpendicular to it. Self-propulsion at the preferred speed $v_0$ is imposed for all agents.

Equations (1) and (2) determine the forward/backward speed and the angular velocity of each agent, respectively. The system is overdamped: each agent moves faster or slower than its preferred speed $v_0$ by a factor proportional to the projection of forces parallel to its direction of motion, while turning with an angular velocity proportional to the projection perpendicular to it (with $\alpha$ and $\beta$ the respective proportionality constants). In equation (3), $\vec{r}_{ij} = \vec{x}_j - \vec{x}_i$ is the distance between agent $i$ and agent $j$. The total force $\vec{F}_i$ over each agent $i$ is computed as the sum of linear spring-like forces with spring constant $k/l_0$ and natural length $l_0$. This sum is over all neighboring agents (represented by the $S_i$ sets), which we connect through "virtual springs" at the beginning of each run. This choice of interacting agents remains fixed throughout the experiment. The interaction network is therefore unchanged from the beginning, regardless of how much the distance between robots may change during the run. The use of a fixed interaction network has advantages and disadvantages for real-world applications. On one hand, it allows for individual positions in predetermined formations to be robust, even under strong perturbations. On the other hand, such formation rigidity can be a disadvantage for certain applications and may present implementation problems, when the inter-robot distances display large changes throughout the dynamics. In practice, however, the use of a fixed interaction network does not affect the results of the experiments and simulations presented in this article, since we only consider strong attraction–repulsion forces. The typical distances between agents will thus not change significantly, and interacting neighbors should remain the same, even without a fixed network.

We confirmed in preliminary analyses that AE models with attraction–repulsion potentials that allow changes of interacting neighbors produce similar self-organizing dynamics as the fixed interaction models studied here, but introducing additional complications. If the interaction range is too short, connectivity is broken and the swarm can disband; if it is too long, interactions with second neighbors can produce excessive aggregation. Further studies that go beyond the scope of this article should therefore be carried out to determine the proper nonfixed interaction rules for each scenario. For simplicity and robustness, we will thus only consider here fixed interaction networks.

## 2.2. Nondimensional form

In order to reduce the dimensionality of the parameter space, we write the nondimensional form of equations (1), (2), and (3) in terms of the system's natural length and time units, $L = l_0$ and $T = l_0/v_0$, respectively. We thus obtain

$$\frac{d\tilde{\vec{x}}_i}{d\tilde{t}} = \left[1 + \tilde{\alpha}\sum_{j\in S_i}\left(\| \tilde{\vec{r}}_{ij} \| - 1\right)\frac{\tilde{\vec{r}}_{ij}\cdot\hat{n}_i}{\| \tilde{\vec{r}}_{ij} \|}\right]\hat{n}_i \tag{4}$$

$$\frac{d\theta}{d\tilde{t}} = \tilde{\beta}\sum_{j\in S_i}\left(\| \tilde{\vec{r}}_{ij} \| - 1\right)\frac{\tilde{\vec{r}}_{ij}\cdot\hat{n}_i^{\perp}}{\| \tilde{\vec{r}}_{ij} \|}. \tag{5}$$

Here, the variables with tilde are nondimensional (expressed in units of $L$ and $T$) and we have defined the nondimensional effective coupling constants $\tilde{\alpha} = \alpha k/v_0$ and $\tilde{\beta} = \beta k l_0/v_0$. Note that the angular variable $\theta$ has no tilde, since it is always nondimensional.

## 2.3. Self-organizing mechanism

Given that the AE model has no explicit local alignment interactions that could directly lead to heading consensus, a more subtle mechanism for its self-organization was unveiled in Ferrante et al. (2013a, 2013b). This mechanism is based on a rearrangement of individual headings that favors the injection of self-propulsion energy into lower "elastic" modes of the collective structure formed by all interacting agents. As in standard damped elastic systems, higher energy modes tend to decay faster than lower ones, since they are harder to excite. In this active case, however, each agent is continuously injecting energy through its self-propulsion, so all modes cannot be fully dampened. Instead, through the coupling in equation (2), higher elastic modes decay by steering agents away from them. Self-propulsion thus feeds more energy into lower modes, which have larger regions of coherent motion, eventually achieving self-organization into collective motion at the scale of the whole system. A more detailed explanation of this mechanism was given in Ferrante et al. (2013a, 2013b). In the remainder of this article, we will explore the capabilities and limitations of this mechanism when applied to real robotic systems.

It is important to point out that the scalability of the AE algorithm, for achieving collective motion in systems with a large number of agents, has already been demonstrated numerically in Ferrante et al. (2013a, 2013b). In these papers, simulations with over 3000 agents are shown to self-organize into collective motion. The approaches presented here could therefore, in principle, be applied to larger robotic systems, although further research is needed to show that the real-world conditions explored in this article are also scalable.

### 2.4. Order parameters

We define here two order parameters that will help us monitor the degree of translational and rotational order in the system, as in Couzin et al. (2002).

First, we consider a quantity that can monitor to what extent robots are aligned and moving in a common direction. The polarization order parameter is thus defined as

$$P(t) = \frac{1}{N}\left\|\sum_{i=1}^{N}\hat{n}_i(t)\right\|. \tag{6}$$

where $N$ is the total number of agents. Second, we define a quantity that shows when the robots are rotating as a group. The rotational order parameter is thus given by

$$M(t) = \frac{1}{N}\left\|\sum_{i=1}^{N}\hat{R}_i(t)\times\hat{n}_i(t)\right\|. \tag{7}$$

Here, $\hat{R}_i(t) = \vec{R}_i(t)/\parallel\vec{R}_i(t)\parallel$ is a unit vector pointing from the group centroid to each agent, defined in terms of

$$\vec{R}_i(t) = \vec{x}_i(t) - \vec{X}_{\mathrm{ctd}}(t) \tag{8}$$

where $\vec{X}_{\mathrm{ctd}}$ is the position of the group centroid, given by

$$\vec{X}_{\mathrm{ctd}}(t) = \frac{1}{N}\sum_{i=1}^{N}\vec{x}_i(t). \tag{9}$$

Note that, with these definitions, the values of $P(t)$ and $M(t)$ will both range between 0 and 1. A case with $P(t) = 1$ and $M(t) = 0$ corresponds to perfect parallel group translation and a case with $P(t) = 0$ and $M(t) = 1$, to perfect rotation.

## 3. Experimental setup and validation

We describe in this section the details of our experimental setup and three tests that we performed to validate it.

### 3.1. Agents, arena, and time-step

Our experimental setup is presented in Figure 1. We chose the e-Puck robots as our agents (Mondada et al., 2009) because these small differential wheeled robots can be easily manipulated and programmed. They also display the smooth forward, backward, and turning motion required by our algorithm. On top of each e-Puck, we added a Wi-Fi communication module (Hlk-wifi-m04), connected to the processor through its serial port. We can then link a router or a wireless exchange board to all Wi-Fi modules to achieve one-to-all broadcast communication. This allows each robot to receive the relative positions of its neighbors, as well as its own heading, through wireless information from an external computer outside of the arena. Each robot then computes the control algorithm on its onboard processor and moves accordingly. Note that these relative positions and heading could have been, in principle, measured onboard. The only reason for obtaining them off-board in our experiments is that e-Pucks have no onboard compass and very imprecise range and bearing sensors. Our setup thus implements a control algorithm that could be fully decentralized and autonomous (with all sensors and computations onboard) for robots with better sensing capabilities.

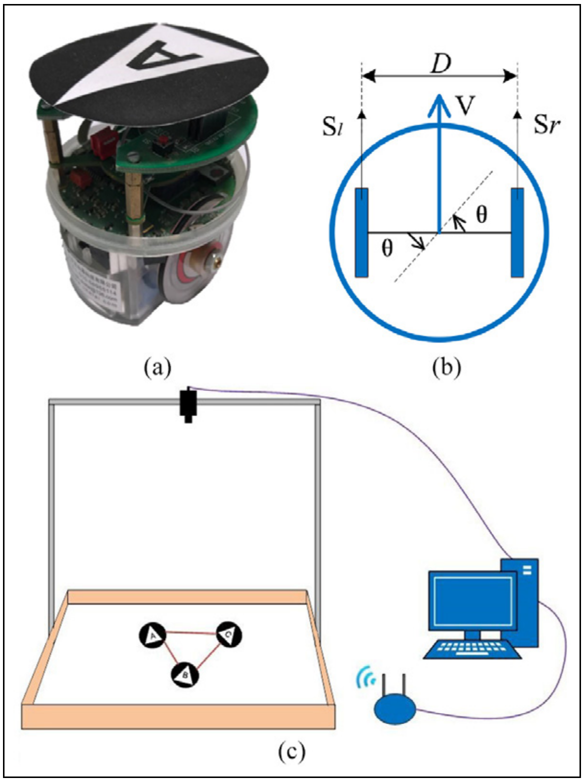


**Figure 1.** Robots and arena setup used in our swarm robotics control experiments. Groups of e-Puck robots, detailed in Panels (a) and (b), are placed on the arena sketched in (c). Each agent receives the position of its neighbors and its own orientation from an external computer, using this information to compute and execute the control algorithm on board: (a) View of a modified e-Puck robot used in our experiments. (b) Diagram of the e-Puck robot structure, viewed from above. (c) Arena and position acquisition system. The overhead camera and the computer detect agent positions, relaying them to the robots via Wi-Fi.

The details of the external sensing system are as follows. We placed a tag on top of each robot with a triangle pointing in its heading direction and a letter (see Figure 1(a)). These visual cues were used to determine the position, orientation, and identity of each robot. We set up a high resolution camera (1928 × 1448 pixels, FLIR Systems, Point Gray Research, model GS3-U3-41C6C) over the arena, pointing down, and linked it to an external computer (see Figure 1(c)). The camera was set at an approximate height of 1.5 m, in order to cover an area of approximately 1 × 1.5 m of the arena (which had a total size of 2 × 2 m), within which the robots were placed and all the experimental dynamics were set to occur. Using this setup, snapshots were taken throughout each experimental run and analyzed by an image processing software. The positions and orientations of all robots were then immediately broadcasted, with each robot receiving through its Wi-Fi module only its own orientation and the relative positions of its neighbors. At every update, the onboard control algorithm thus generated new $\vec{x}$ and $\dot{\theta}_i$ values for each robot, which were converted through a simple geometrical calculation into the speeds of the left and right wheels, $S_l$. and $S_r$, respectively. As shown in Figure 1(b), to do this, we first define the forward/backward speed as $V = \dot{\vec{x}}_i \cdot \hat{n}_i$ and the angular velocity as $\Omega = \dot{\theta}_i$. We then write $V = S_l + S_r$ and $\Omega = 2(s_r - s_l)/D$, where $D = 5.5$ cm is the distance between wheels. From here, we can find the expressions $S_l = V - D\Omega/4$ and $S_r = V + D\Omega/4$ for the speed of each wheel.

Despite requiring additional hardware and a special arena, the external sensing system in our experiments is ideally suited for testing the capabilities and limitations of the AE model. Indeed, since the tracking algorithm and information broadcasting take some time and are simultaneous, they force the $\dot{\vec{x}}_i$ and $\dot{\theta}_i$ update in control equations (1) and (2) to be discrete and synchronous. This update time thus becomes equivalent to the time-step used in our numerical integration, allowing us to implement a simple computer simulator that models not only the idealized motion but also the imprecisions of each robot, as discussed in the following sections. Although the processing time is not constant (ranging from 200 to 300 ms, depending on image complexity), we will use below a fixed time-step $\Delta t = 240$ ms, which corresponds to the mean processing time.

### 3.2. Validation in a two-robot system

We validated our experimental setup by testing three simple cases of the two-robot dynamics for which we know the theoretical results. In all these tests, we set the equilibrium distance between robots to $l_0 = 22.5$ cm and the preferred speed to $v_0 = 2.6$ cm/s.

Our first test consisted in placing the two robots side-by-side, heading in the same direction, and at the equilibrium distance $l_0$ at which the $\vec{F}_i$ forces in equation (3) vanish. Since the agents are set to advance in parallel, their distance should change only due to experimental errors so forces should remain close to zero. The result of this test is presented as curve A in Figure 2, which displays the distance $R_{12}$ between the two robots as a function of time (see also Video SV1 in the Supplementary Material). We observe that $R_{12}$ oscillates around the preferred equilibrium distance $l_0$, thus confirming that this is a stable configuration.

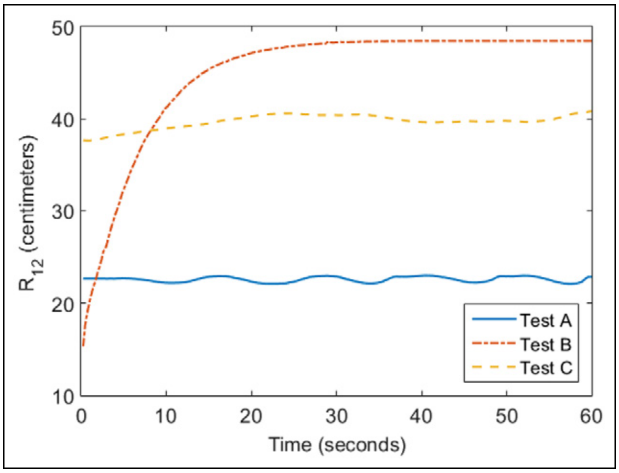


**Figure 2.** Relative distances, as a function of time, between the two robots used in three simple tests performed to validate our experimental setup (see Section 3.2).

The second test was to place the robots back to back, pointing in opposite directions. In this test, the robots will move away from each other until they reach an equilibrium distance $R_{\max}$, where their self-propulsion is balanced by their interaction force. Using equations (1) and (3), this distance is easily computed as

$$R_{\max} = l_0\left(1 + \frac{v_0}{\alpha k}\right). \tag{10}$$

The result of this experiment is presented as curve B in Figure 2 (see also Video SV2 in the Supplementary Material). Given that $\alpha k = 2.25$ cm/s, the theoretical equilibrium distance here is $R_{\max} = 48.5$ cm, which is within 1 mm of the final $R_{12}$ value reached by the experimental curve.

In our third and final test, we placed the robots side-by-side, pointing in parallel but opposite directions. We set them at a distance $R_{\text{rot}}$ from each other, which we chose so that both agents follow stationary circular trajectories about a common center of rotation midway between them (see Video SV3 in the Supplementary Material). We can compute $R_{\text{rot}}$ by imposing that the angular velocity of each robot around its own axis is equal to the angular velocity of its circular trajectory around the center of rotation. This condition is given by $2v_0/R_{\text{rot}} = \beta k(R_{\text{rot}} - l_0)$, from where we find that the distance at which stationary circular trajectories should be observed is

$$R_{\text{rot}} = \frac{1}{2}\left(l_0 + \sqrt{l_0^2 + \frac{8v_0}{\beta k}}\right). \tag{11}$$

Since we used $\beta k = 0.01\ \mathrm{s}^{-1}$ in our experimental test, we find from here $R_{\mathrm{rot}} \approx 37$ cm. Curve C in Figure 2 shows that, when starting from the initial condition described above, with $\mathrm{R}_{12} = R_{\mathrm{rot}}$, the robots trace quasi-stationary circles about a common center or rotation, as expected. Indeed, this curve shows that the distance $\mathrm{R}_{12}$ remains reasonably constant, without displaying the strong drifts observed when starting at any other distance (data not shown).

The three tests detailed above confirmed that our control algorithm works as expected in simple two-robot scenarios. They also displayed evidence of various sources of experimental noise that can lead to small differences with respect to our theoretical results. We point out, however, that these tests are not a systematic analysis of the precision or reliability of the two-robot system. They only describe our verification method for showing that our setup can match the expected theoretical dynamics. We found that our two-robot trials were always successful when there were no failures in the robots, tracking, or Wi-Fi system (such as mechanical, position detection, or communication issues). They therefore allowed us to debug our experimental setup. In scenarios where the focus is on the performance of a specific system for engineering applications, a systematic study of the two-robot system could provide critical information that can be extrapolated to larger robot swarms.

## 4. Translational collective motion

In this section, we study the self-organizing dynamics that leads to collective translational motion in our experimental system. We begin by examining the typical dynamics of a group of seven robots in a hexagonal formation and with initial random headings. We then focus on the oscillations in the heading direction that spontaneously appear after the collective translation state is reached. These result in sinusoidal trajectories that are only partially aligned. Finally, we end the section by characterizing in detail the experimental noise measured in our system.

### 4.1. Typical self-organizing robot dynamics

Figure 3 displays snapshots of the typical self-organizing dynamics observed in our experiments (see also Video SV4 in the Supplementary Material). We overlaid here blue lines on all panels to show which pairs of robots are interacting. We also added large green arrows to Panels (b) and (d) to indicate the group's heading direction. In addition to these snapshots, Figure 4 plots the corresponding values of the polarization and rotational order parameters, as defined in equations (6) and (7), for the same experiment.

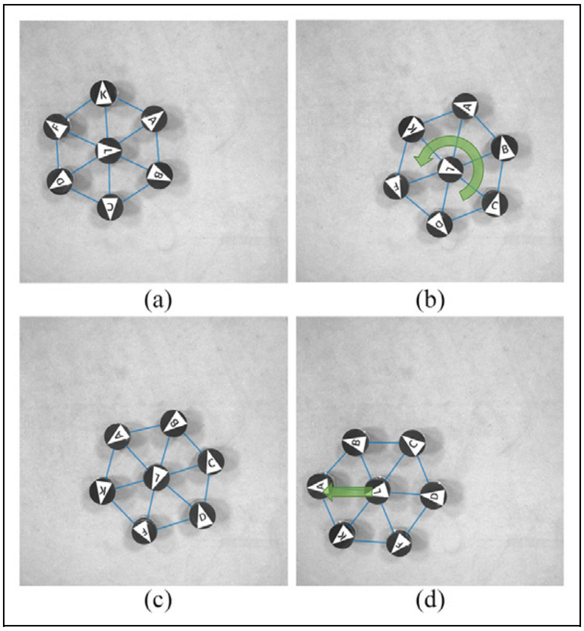


**Figure 3.** Snapshots of a robot swarm experiment that implements the position-based decentralized control algorithm specified in equations (1)—(3). We observe its typical self-organizing dynamics toward translating collective motion (see video SV4 in the Supplementary Material). The overlaid blue lines indicate which robots are interacting. (a) Standard initial condition used in our experiments and simulations (t = 0 s); robots are placed in a perfect hexagonal configuration with all but the central agent pointing radially outwards. (b) Transient rotating state (t = 108 s); as the system self-organizes, it sometimes first visits a metastable rotational state, as displayed in this panel. (c) Transition to translating state (t = 155 s); the group eventually leaves the rotational state, converging towards translational motion. (d) Final self-organized aligned state (t = 216 s); the system achieves translational collective motion and will move together until it reaches the edge of the experimental frame.

Figure 3(a) shows our standard initial condition; seven robots placed in an hexagonal configuration where all interacting pairs are at the equilibrium distance $l_0$. Note that, although the system can self-organize into translating collective motion starting from any set of initial robot orientations, the convergence time and specific dynamics will strongly depend on this initial condition. We therefore used the same *standard initial condition*, displayed on Figure 3(a), in all experiments performed for the parameter space exploration presented below. This initial condition consists of placing all robots pointing radially outward (except for the central one, for which this direction is undefined). This guarantees that, both, the polarization and the rotational order parameters will be close to zero at the beginning of each run (see Figure 4). As time moves forward, each robot advances in the direction indicated by the arrow on top of it until the elastic

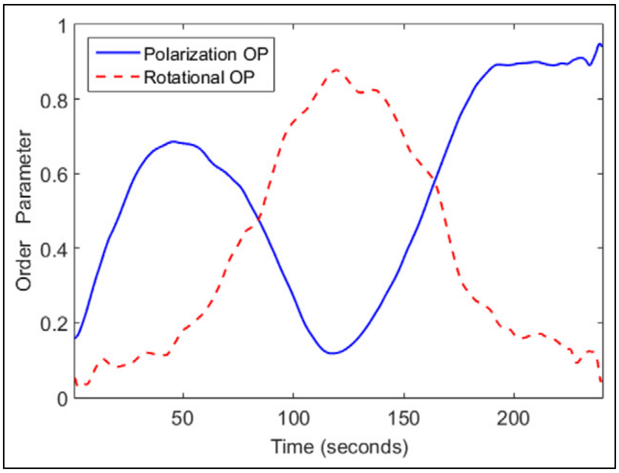


**Figure 4.** Polarization and rotational order parameters as a function of time for the experiment presented in Figure 3. At $t = 0$, both order parameters display low values. In the $t \approx 100$ to $t \approx 150$ interval, the robots reach a metastable rotating state with low polarization and large rotational order parameter. Finally, for $t > 200$, the system converges to translating collective motion, displaying high polarization and low rotational order parameter.

forces balance self-propulsion. It then starts oscillating forward and backward while slowly turning toward a collective state. Figure 3(b) displays a metastable collective rotational state that is often visited by the system before aligning (Ferrante et al., 2013b). It is characterized by the high rotational order parameter and low polarization order parameter (observed in the $t \approx 100$ to $t \approx 150$ interval in Figure 4). The system typically leaves this state after approximately 50 s, as the central robot continues to push forward, impeding stationary rotation. In Figure 3(c), the group starts reorganizing from rotational to translational collective motion. Finally, in Figure 3(d), it reaches an aligned state where all agents advance together, in which the polarization is almost 1 and the rotational order parameter is almost 0.

## 4.2. Stability analysis

In the experimental runs described above, we observed a phenomenon that was already apparent in the sinusoidal nature of Curve A in Figure 2; individual headings often display recurrent, persistent oscillations with respect to the mean heading direction after achieving translational collective motion. These oscillations can be seen, for example, in Videos SV1 and SV4 of the Supplementary Material. To understand the origins of this phenomenon, we carried out a simple linear stability analysis, which we detail below.

We consider a system of two robots advancing together side by side at the equilibrium distance $l_0$, following the AE control equations (1) and (2). This setup is equivalent to Test A in Section 3.2, but with zero noise. If we impose complete symmetry in the initial conditions, the dynamics will remain symmetric, so the motion of one robot will be the mirror image of that of the other. Under these conditions, this system can be described in terms of the distance $\Delta x$ and angular difference $\Delta\theta$ between robots (see Figure 5). We find that these quantities evolve following the equations

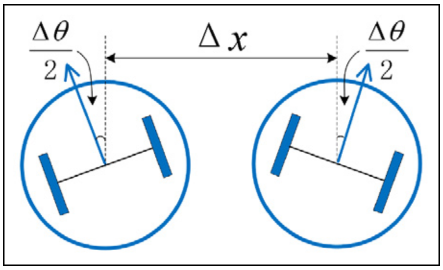


**Figure 5.** Sketch of the minimal two-robot fully symmetric setup for studying the linear stability of the aligned state. Perturbation variables $\Delta x$ and $\Delta\theta$ represent, respectively, small differences in the distance and relative heading angle between the robots with respect to their equilibrium values.

$$\frac{\mathrm{d}}{\mathrm{d}t}\Delta x = 2\left(v_0 - \alpha k \Delta x \sin\frac{\Delta\theta}{2}\right)\sin\frac{\Delta\theta}{2} \tag{12}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\Delta\theta = -2\beta k \Delta x \cos\frac{\Delta\theta}{2}. \tag{13}$$

If we then linearize these expressions for small $\Delta x$ and $\Delta\theta$ perturbations, we obtain

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{pmatrix}\Delta x\\ \Delta\theta\end{pmatrix} = \begin{bmatrix}0 & v_0\\ -2\beta k & 0\end{bmatrix}\begin{pmatrix}\Delta x\\ \Delta\theta\end{pmatrix}. \tag{14}$$

We can then compute the eigenvalues of the stability matrix in this equation, which are given by

$$\lambda_{\pm} = \pm\sqrt{-2\beta k v_0}. \tag{15}$$

Since both eigenvalues are purely imaginary, we conclude that this simple two-robot toy system displays marginal linear stability and that linear perturbations will not dampen out, producing instead persistent oscillations.

Although our experimental system contains many more degrees of freedom than the toy case described above, this analysis suggests that the AE control algorithm does not dampen out linear perturbations. Instead, it will sustain small oscillations that can only decay due to nonlinear interactions. This appears to explain the observed small, persistent heading oscillations, suggesting that the AE algorithm may have fundamental limitations in achieving and maintaining perfect alignment.

### 4.3. Experimental noise analysis

Our experimental control system can have multiple sources of error. We will refer to these, generically, as *noise*. An interesting advantage of our setup is that the information required by each agent to determine its next linear and angular speeds is measured and broadcasted simultaneously to all robots. This allows us to characterize the experimental noise by comparing their predicted and observed values at every update step.

Panel (a) of Figure 6 displays a scatter plot of the difference at every update step between predicted and observed angular and linear speed values, labeled $\Delta\omega$ and $\Delta v$, respectively. We carried out a set of experiments similar to those in Figure 3, where seven robots in a hexagonal configuration self-organize starting from our standard initial condition. We then computed $\Delta\omega$ and $\Delta v$ at every update throughout the trajectories of four of the robots, each in a different experimental run, and plotted the resulting values. We observe that all four scatter plots follow a similar distribution. Their approximate specular symmetry with respect to both axes is evidence of no significant systematic errors in the linear or angular speed of the robots. The nonisotropic shape implies nontrivial correlations between $\Delta\omega$ and $\Delta v$. If we neglect these correlations and plot both errors independently, however, we observe that they follow relatively narrow Gaussian-like distributions, as seen in Panels (d) and (e). It is apparent in these figures, however, that these experimental noise distributions also include rare (but large) long-tail events that cannot be described by a simple Gaussian function.

Rather than attempting to capture all the subtleties of the experimental noise, we will focus here on finding a minimal description of the noise distribution that reproduces the main qualitative properties of the observed dynamics. We therefore search for Gaussian functions that match only the central regions of the $\Delta\omega$ and $\Delta v$ distributions. To do this, we generate Q-Q (quantile–quantile) plots for all the $\Delta\omega$ and $\Delta v$ data against the quantiles of two normal distributions. We then find the standard deviation values for which their central regions match the diagonal, obtaining $\sigma_{\Delta\omega} \approx 0.03$ rad/s and $\sigma_{\Delta v} \approx 0.1$ cm/s, respectively. We confirm in Panels (b) and (e) of Figure 6 that the resulting Gaussian functions are a good approximation of the central regions (for $|\Delta\omega|<0.05$ and $|\Delta v|<0.2$, respectively) of the experimental noise.

The analysis presented above shows that our noise distributions can be reasonably well approximated by Gaussian functions. We can therefore simulate the experimental noise by adding normally distributed random variables to the speed and angular velocity of each agent. We will implement this approach below to study how this noise and the control parameters affect self-organization.

## 5. Dependency on control parameters

In this section, we will study how the self-organizing dynamics and the stability of polarized states depend on the AE algorithm parameters $\alpha$, $\beta$, $v_0$, $k$, and $l_0$.

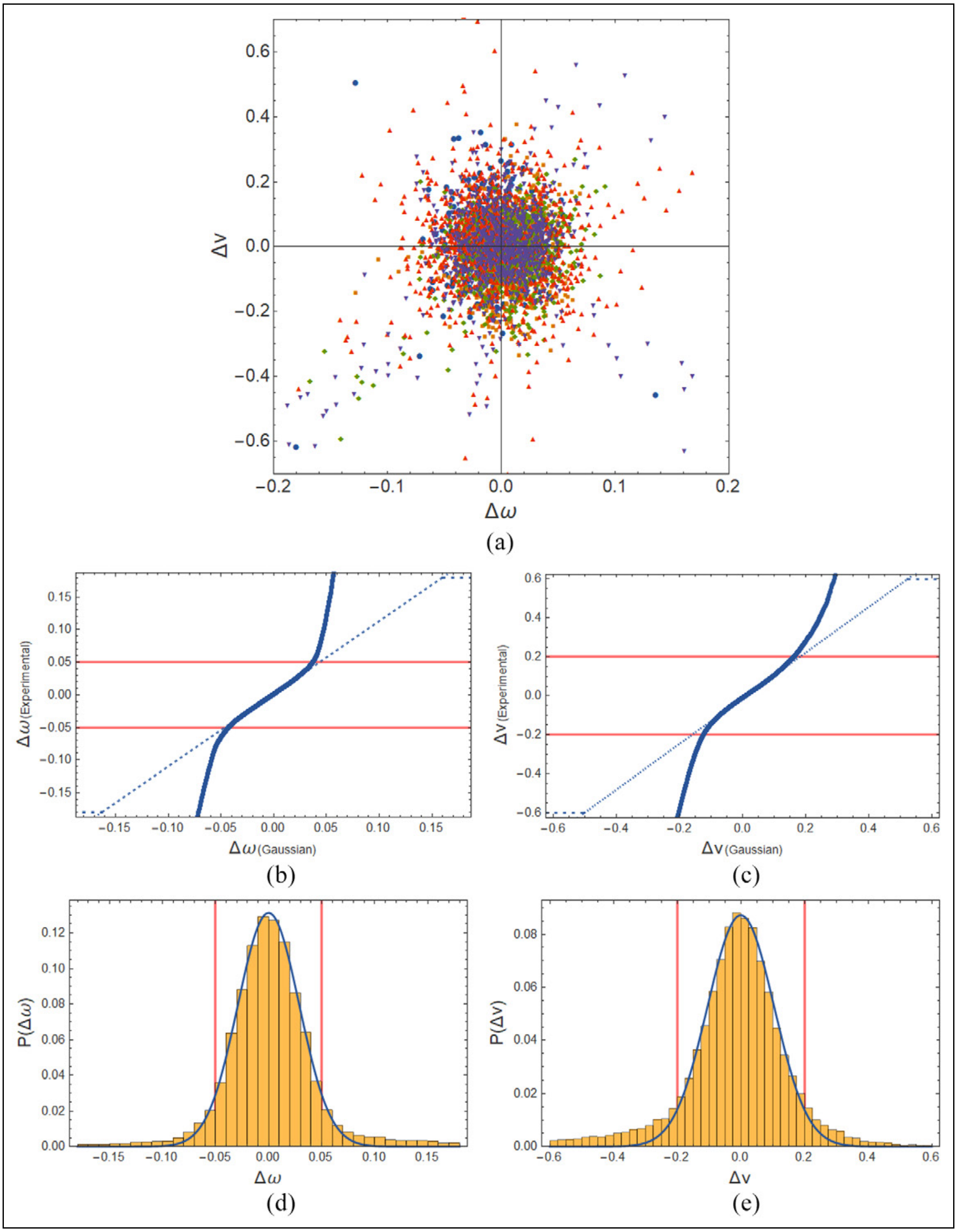
In order to reduce our parameter space to two dimensions, we will use the non dimensional quantities $\tilde{\alpha}$ and $\tilde{\beta}$, introduced in equations (4) and (5), as control parameters. This formulation captures all possible solutions of the original equations (1) and (2), but expressed in non dimensional units of space and time ($L = l_0$ and $T = l_0/v_0$, respectively). The actual spatio-temporal scales are relevant in real-world systems, however, since they are typically constrained by experimental conditions. In our case, spatial scales are bounded by the minimum distance between robots and by the size of the arena. Robot speeds are in turn limited by their operational range. Indeed, we found that e-Pucks could not be controlled with precision when they moved slower than ~0.2 cm/s and that our tracking system produced significant errors when they moved faster than ~1 cm/s. Given these constraints, we fixed our dimensional values to $v_0 = 0.5$ cm/s, $l_0 = 15$ cm and $k = 0.015$ in all experiments below, and then scanned the parameter space as a function of $\tilde{\alpha}$ and $\tilde{\beta}$.

The simulations implemented for this section were designed to closely mimic our experiments. In order to reproduce the effects of the experimental update step, we used a forward Euler method with numerical time-step $\Delta t = 240$ ms, which we found to be the mean processing time of our position acquisition system. We simulated the experimental noise by adding, at every time-step, a normally distributed independent random variable (with the standard deviation values, $\sigma_{\Delta\omega}$ and $\sigma_{\Delta v}$, computed in Section 4.3) to the linear and angular speeds of each agent, that is, to the right-hand side of equations (1) and (2).

Following the approaches described above, we carried out a set of experimental and numerical runs using for the same parameter combinations, starting from our standard initial conditions, and mimicking numerically the experimental update step and noise. We characterize below the resulting dynamics and self-organizing properties.

### 5.1. Experimental and numerical dynamics

Figure 7 displays examples of the typical polarization order parameter dynamics $P(t)$ in experiments and simulations, for four selected nondimensional parameter combinations that cover our experimental operational range. These were chosen from the experiments and simulations that were performed to generate the four corners of the diagrams presented in Figures 8 and 9. We will analyze here these four representative examples of the dynamics, rather than their mean



(a)

(b)

(c)

(d)

(e)

**Figure 6.** Analysis of the experimental noise in the angular and linear speeds of each robot. All plots display the difference between their predicted and measured values after every experimental update, labeled $\Delta\omega$ and $\Delta v$, respectively. Panel (a) shows a scatter plot of the angular and linear speed errors ($\Delta\omega$, $\Delta v$) for four different robots in different experimental runs (distinguished by different symbols and colors). Panels (b) and (c) present the Q-Q (quantile-quantile) plots of the angular speed noise $\Delta w$ and linear speed noise $\Delta\omega$ error data (obtained in five experiments for all seven robots in the hexagonal configuration) against the quantiles of two Gaussian distributions that match their central regions. Panels (d) and (e) display the $\Delta\omega$ and $\Delta v$ distributions with their corresponding Gaussians overlaid as blue curves. The red lines indicate the central regions ($|\Delta\omega| < 0.05$ and $|\Delta v| < 0.2$) where these provide a good approximation. Speeds are measured in cm/s and angles in rad/s.

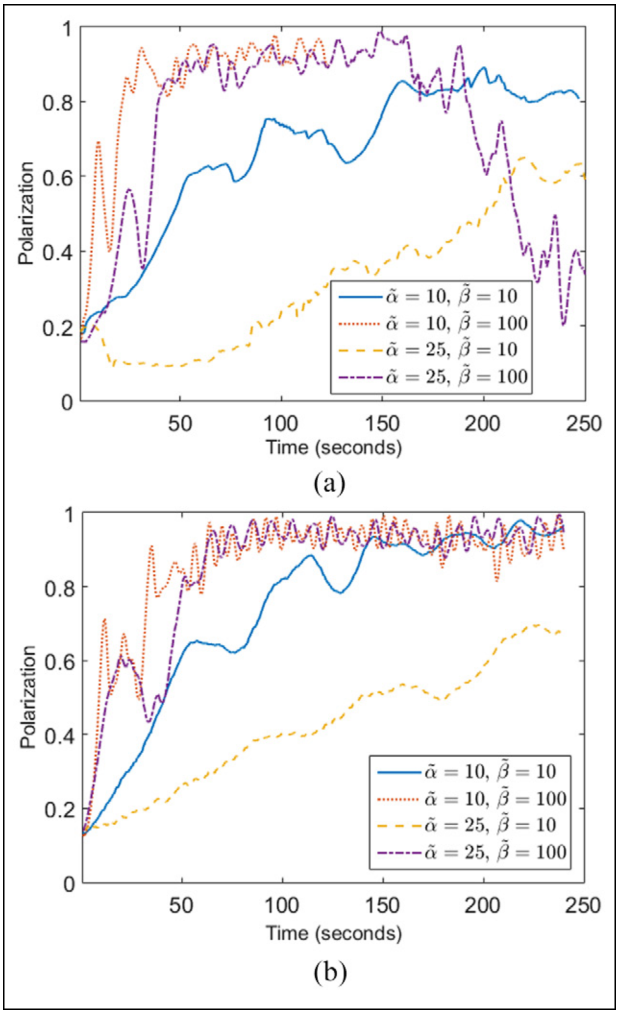


**Figure 7.** Polarization order parameter as a function of time for selected experimental and numerical runs with the same non-dimensional parameter combinations ($\tilde{\alpha}$, $\tilde{\beta}$): (a) Polarization order parameter in experimental data. (b) Polarization order parameter in numerical data. Each curve corresponds to a run with seven robots in a hexagonal configuration, starting from our standard initial conditions (see Figure 3). Dimensional parameters were fixed to $v_0$ = 0.5 cm/s, $l_0$ = 15 cm, k = 0.015, and $\Delta t$ = 0.240 s for all runs. The experimental and numerical curves display similar features, including their convergence times and the level of self-organized polarization they reached. Experimental curves display larger fluctuations, however, which tend to destabilize their ordered state, as in the $\tilde{\alpha}$ = 25, $\tilde{\beta}$ = 100 case. Note that some of these end early, when robots exit the arena.

characteristics over all available data, in order to compare the multiple features that they display, albeit qualitatively. An advantage of carrying out this type of initial analysis is that many of these features, such as the details of the observed oscillations, could be lost when averaging over specific characterizing quantities. We will then perform a quantitative analysis of the mean behavior of two selected features in Section 5.2.

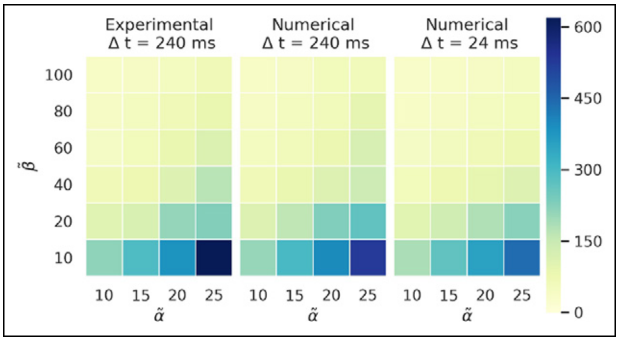


**Figure 8.** Time $\tau$ required to first reach a self-organized state with polarization $P \geqslant 0.5$, as a function of nondimensional control parameters $\tilde{\alpha}$ and $\tilde{\beta}$. Experimental results (left-side panel) are compared to simulations with two different numerical time-steps: $\Delta t = 240$ ms (central panel) and $\Delta t = 24$ ms (right-side panel). We observe that experiments and simulations with $\Delta t = 240$ ms behave almost identically, whereas the $\Delta t = 24$ ms time-step produces faster convergence times. In all cases, the fastest convergence is for low $\tilde{\alpha}$ and high $\tilde{\beta}$ values. Each unit in the color scale corresponds to 240 ms.

The figure shows that the experimental and numerical curves display similar features. For the same $\tilde{\alpha}$, $\tilde{\beta}$ values, the typical time it takes to reach the polarized state is almost identical. The maximum polarization reached and type of oscillations displayed by the different curves are also very similar. A difference, however, is that the noise appears to more easily destabilize the self-organized polarized state in the experimental case. This is apparent in the $\tilde{\alpha} = 25$, $\tilde{\beta} = 100$ curve, where noise-driven oscillations destabilize the ordered state after ~150 s. These oscillations also seem to be responsible for the lower level of order reached by the $\tilde{\alpha} = 10$, $\tilde{\beta} = 10$ experiment.

We hypothesize that the difference that we observe between the experimental and numerical dynamics is mainly due to the long tails displayed by the noise distributions in Figure 6. These are not captured by our Gaussian approximations and reflect rare but strong perturbations that can push the swarm away from its ordered state. We note, however, that this difference could also be due to other properties of the noise not included in our model, such as systematic hardware errors or correlations between noise sources. In order to examine the potential role of these different factors in mimicking numerically the observed dynamics, future works will have to develop a more detailed description of the noise (including its systemic components for different specific hardware pieces and its multiple underlying correlations) and quantitative measures for characterizing the self-organizing dynamics of the specific robot swarm under consideration.

### 5.2. Phase diagrams

We will now study how self-organization into translating collective motion depends on the control parameters $\tilde{\alpha}$ and $\tilde{\beta}$. We refer to the resulting figures as "phase diagrams," in a loose analogy with the plots used in physics to represent changes in the state of a given system as a function of its control parameters.

We present below two phase diagrams, each corresponding to a different order parameter. The first-order parameter, $\tau$, is defined as the amount of time it takes for the group to first reach a polarized state with $P(\tau) \geqslant 0.5$, starting from our standard initial condition in Figure 3. The second one, $\Psi = \langle P(t) \rangle_{t > \tau}$, is defined as the mean polarization from time $\tau$ until the end of the experiment or simulation. With this definition, $\Psi$ can be computed for runs of any duration, which is required because the length of the $P(t)$ data set will be different in each experiment, since it is determined by the amount of time the group remains within the tracking region of the arena. Note that a low $\Psi$ value can reflect either that the ordered state is not strongly aligned or that order was lost after time $\tau$. An example of the latter can be found in the $\tilde{\alpha} = 25$, $\tilde{\beta} = 100$ curve of Figure 7, which yields a low $\Psi$ value despite temporarily reaching a high degree of order.

Figures 8 and 9 present the experimental and numerical phase diagrams for $\tau$ and $\Psi$, respectively, as a function of $\tilde{\alpha}$ and $\tilde{\beta}$. These results were obtained by sampling 24 experimental parameter combinations, corresponding to four values of $\tilde{\alpha}$ (10, 15, 20 and 25) and six values of $\tilde{\beta}$ (10, 20, 40, 60, 80 and 100). For each parameter set, we performed five repetitions of the experiment, starting from our standard initial condition displayed on Panel (a) of Figure 3 (with all agents pointing radially outward, except for the central one for which this direction is undefined). We then selected the three experimental runs with the smallest trajectory variations (measured as the difference between polarization curves) to discard outlier experimental runs. For each parameter set, we also performed 20 numerical runs with different noise seeds.

The left-side panel of Figure 8 displays the mean value of $\tau$, averaged over the three selected experimental runs. The central and right-side panels present $\tau$ averaged over all 20 simulations, using time-steps $\Delta t = 240$ ms and $\Delta t = 24$ ms, respectively. The diagram shows that our simulations closely predict the experimental convergence times when we set the numerical time-step to 240 ms, in order to match the mean update time measured in our experiments. When we consider a shorter time-step of $\Delta t = 24$ ms, all convergence times are significantly reduced, but the structure of the phase diagram remains the same. For all $\Delta t$ values, the low $\tilde{\alpha}$ and high $\tilde{\beta}$ region displays the shortest convergence times.

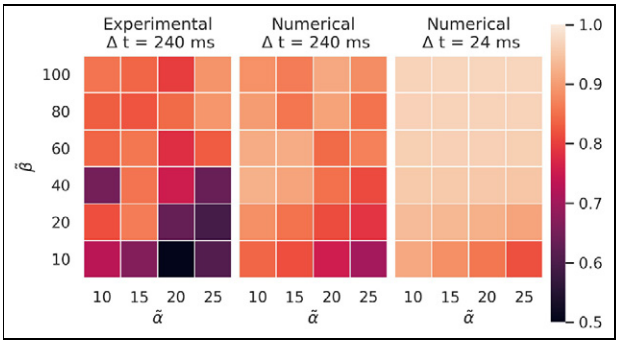


**Figure 9.** Mean polarization $\Psi$ after first reaching a self-organized state with $P \geqslant 0.5$, as a function of nondimensional control parameters $\tilde{\alpha}$ and $\tilde{\beta}$. Experimental results (left-side panel) are compared to simulations with two different numerical time-steps: $\Delta t = 240$ ms (central panel) and $\Delta t = 24$ ms (right-side panel). Here, the agreement between experiments and simulations is much less than for $\tau$ (see Figure 8), but all cases still tend to display higher order for low $\tilde{\alpha}$ and high $\tilde{\beta}$ values. In contrast to Figure 8, the shorter numerical time-step ($\Delta t = 24$ ms) produces here very different results, displaying a much higher level of self-organized polarization.

Figure 9 presents the mean value of $\Psi$, averaged over the same experiments and simulations used in Figure 8. The left-side and central panels show that in this case, even when using the same $\Delta t = 240$ ms as in experiments, we observe much lower experimental polarization values than predicted by simulations. We also note that these phase diagrams display no clear structure, showing instead strong fluctuations between regions with neighboring $\tilde{\alpha}$, $\tilde{\beta}$ values. Only when a much smaller $\Delta t = 24$ ms is used, a cleaner phase diagram emerges, where the highest $\Psi$ values are reached for low $\tilde{\alpha}$, high $\tilde{\beta}$ parameter combinations.

In order to better understand the structure of these phase diagrams and their dependency on $\Delta t$, we generated more detailed versions using only numerical simulations, which we present in Figures 10 and 11. These new diagrams display results for 90 parameter combinations, using $\Delta t = 240$ ms in Panel (a) and $\Delta t = 24$ ms in Panel (b). They show the same structures as Figures 8 and 9. The phase diagrams obtained for $\tau$ with different time-steps are again very similar, showing only slightly shorter convergence times in the small $\Delta t$ case. The diagram for $\Psi$ is again very noisy for large $\Delta t$ and only displays a clear structure for small $\Delta t$. A detailed analysis of the numerical convergence curves showed that the main cause for this is that larger $\Delta t$ values produce much less stable polarized states. This appears to be related to the persistent oscillations that develop after the agents align, which stems from the marginally stable dynamics of the AE algorithm that we showed in Section 4.2. Indeed, the stability of the converged state

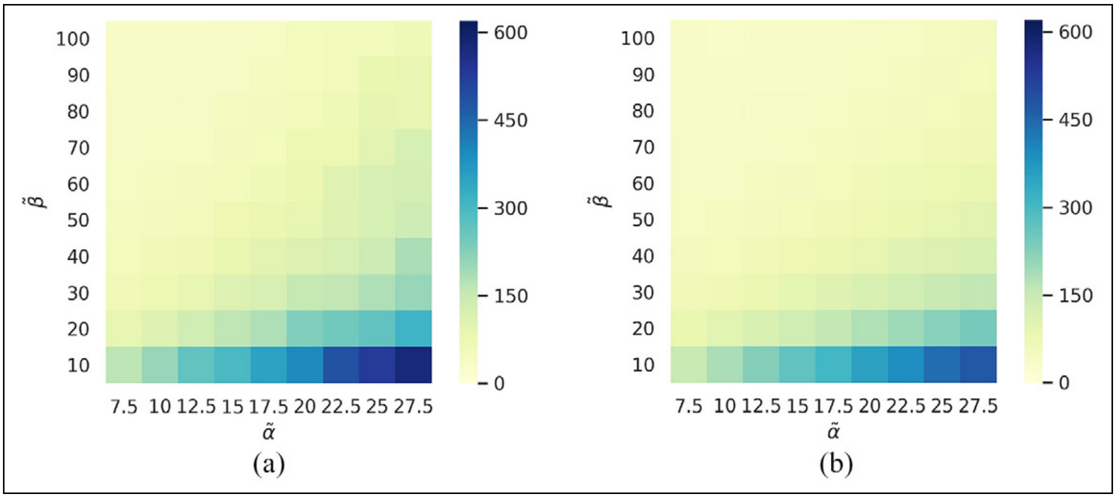


**Figure 10.** Time $\tau$ required to first reach a self-organized state with polarization $P \geqslant 0.5$, as a function of nondimensional control parameters $\tilde{\alpha}$ and $\tilde{\beta}$, in numerical simulations with two different values for the time-step $\Delta t$: (a) Simulations with time-step $\Delta t = 240$ ms. (b) Simulations with time-step $\Delta t = 24$ ms. Whereas convergence is faster for the smaller $\Delta t$, the difference is small and the structure of the phase diagram is the same. The diagram predicts that swarms with low $\tilde{\alpha}$ and high $\tilde{\beta}$ values will produce the fastest self-organizing dynamics. Each unit in the color scale corresponds to 240 ms.

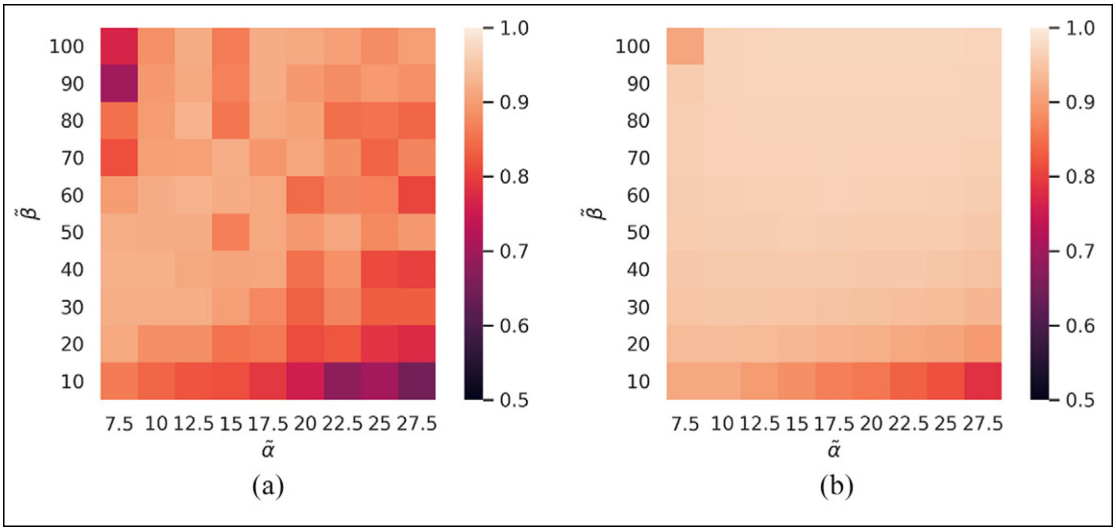


**Figure 11.** Mean polarization $\Psi$ after first reaching a self-organized state with $P \geqslant 0.5$, as a function of nondimensional control parameters $\tilde{\alpha}$ and $\tilde{\beta}$, in numerical simulations with two different values for the time-step $\Delta t$: (a) Simulations with time-step $\Delta t = 240$ ms. (b) Simulations with time-step $\Delta t = 24$ ms. In contrast to Figure 10, a smaller $\Delta t$ produces here a very different, much less noisy diagram with significantly lower $\Psi$ values. The figure predicts that swarms with $\tilde{\alpha} \approx 10$ and $\tilde{\beta} \approx 80$ will reach the highest $\Psi$ values.

would be greatly reduced when using large time-step values in this scenario, since these can easily destabilize marginally stable systems by amplifying small oscillations. This would also explain why $\Psi$ is becoming smaller for $\tilde{\alpha} = 7.5$ and $\tilde{\beta} = 100$, since this parameter combination will also enhances the destabilizing effect of these oscillations.

By combining the information in these types of phase diagrams, it is possible to predict the region of parameter combinations that should favor fast and robust self-organization in a given experimental system. After inspecting Figures 10 and 11, for example, we can estimate that in our specific setup, $\tilde{\alpha} \approx 10$ and $\tilde{\beta} \approx 80$ appear to be a good compromise region for achieving rapid convergence to a self-organized state that has a relatively high and stable level of order. We can also predict that the experimental update time must be smaller than 240 ms to achieve a self-organized state that is reliably robust, but it does not need to be smaller than 24 ms. Finally, these phase diagrams also show that even a simple Gaussian approximation of the measured experimental noise yields good predictions for the expected dynamics. We note, however, that although these simulations typically mimic the expected type of trajectories, these can sometimes display significant differences with the experiments because they fail to capture important details of the real-world noise, such as correlations between error sources or rare, but large, non-Gaussian fluctuations.

## 6. Rotational collective motion

Collective rotation is another state of collective motion found in nature (Calovi et al., 2014; Pitcher, 1983; Radakov, 1973; Tunstrøm et al., 2013), in which agents self-organize to orbit around a central point. In this section, we will show that our AE algorithm can be modified to produce self-organization into this rotating state.

In order to achieve collective rotation, we made a small modification in the AE equations, introducing a different $v_0$ for each agent. We thus set $v_0(R)$ as linearly proportional to the distance $R$ between the corresponding robot and the collective center of rotation, which we place at the centroid of the group. With this change, each robot can advance at its preferred speed while the group rotates collectively as a rigid body, without changing the distance between agents. Note that, as shown in Figure 3, the standard AE algorithm can also reach collective rotation, but only as a metastable state. By increasing $v_0$ with the distance to the center of rotation, we stabilize this rotating state. It thus becomes the final result of the self-organizing process, in which all robots can remain at their equilibrium distances while advancing at their preferred speeds.

We carried out proof-of-concept experiments to test whether real-world rotational collective motion can be achieved with our modified AE algorithm. We performed two sets of five repetitions of the experiments detailed below, always obtaining the same qualitative outcome. For these tests, we set up an elongated structure, rather than a hexagon, using the same seven robots. This structure increases the difference between the largest and smallest preferred $v_0$ values, thus favoring collective rotation. We note that our modified AE algorithm does not aim to self-organize the system into this initial condition, given here by the agent positions within the elongated formation and their corresponding $R$ values. Other decentralized control algorithms in the literature could be used to reach this state, however, such as the algorithm used in Rubenstein et al. (2014), which can also provide the initial $R$ values needed to compute $v_0$.

Figure 12 displays snapshots of our proof-of-concept experiment. As shown in Panel (a), all robots were initially placed at a distance $l_0$ = 15 cm from their nearest neighbors, with random headings. The experimental parameters were set to $\alpha = 5$, $\beta = 30$, $k = 0.015$. The preferred speed for each robot $i$ was defined as $v_0(i) = 2R_i/(\sqrt{3}\,l_0)$ cm/s, where $R_i$ is its distance to the centroid of the structure (which we chose to be the center of rotation). With this definition, the preferred speed of the robot labeled D in Panel (a) was set to $v_0$ = 0.5 cm/s. Panel (b) shows the group starting to self-organize, as it arranges into circular headings that are approximately tangential to the line connecting each robot to the center of rotation. In Panels (c) and (d), we confirm that a persistent state of collective rotation is established.

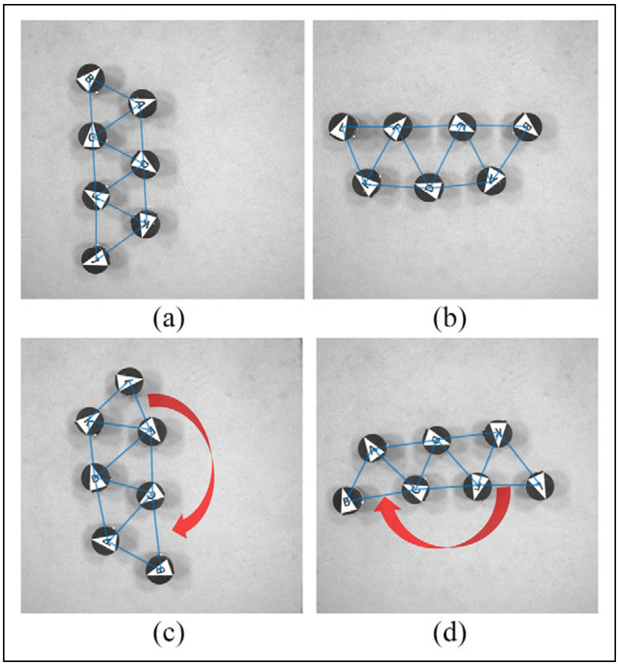


**Figure 12.** Snapshots of a proof-of-concept robot swarm experiment that self-organizes into rotational collective motion by following the control algorithm in Section 6 (see video SV5 in the Supplementary Material). The overlaid blue lines indicate which robots are interacting. (a) Initial condition with all robots at natural distance $l_0$ from their neighbors and random headings (t = 0 s). (b) Self-organization into circular headings (t = 47 s); as the system starts self-organizing, robots reorient into circular headings while the group rotates its first 90 degrees. (c) Structure starts rotating (t = 107 s); rotating motion is established. (d) Collective rotation continues until the end of the experimental run (t = 124 s).

The ability demonstrated above by our modified AE algorithm to self-organize agents into a rotating state highlights the many potential applications of our AE model in robot swarm control. Indeed, as discussed in the "Introduction" section, most decentralized control algorithms rely on explicit alignment, and can therefore only produce collective translation. The AE model will instead lead a group of agents to the lowest accessible elastic mode of their collective structure, which can be designed through small modifications to correspond to any number of collective states. For example, given that our elastic forces can be established permanently between any combination of robots (not only all nearest neighbors), one can imagine defining interaction networks that lead to self-organized states with various locally rotating substructures.

## 7. Discussion and conclusions

In this work, we have analyzed for the first time the capabilities and limitations of the AE model as a real-world decentralized control algorithm in a swarm robotics experiment. In contrast to its only previous implementation, in Ferrante et al. (2012), we focused here on characterizing the dependency of the self-organizing dynamics on model parameters and the effects of the experimental noise. We thus showed that it can effectively lead robots to self-organize into collective translating or rotating motion. Since the AE algorithm relies only on the positional information of neighbors, our experiments showed that robots do not require hardware that detects the headings of other robots to achieve self-organized collective motion.

Our control algorithm was fully implemented on board, but it was fed the robot's own orientation and neighbor positions by an external tracking system set up above the arena. Despite its current use of external information, this implies that the AE algorithm could be executed fully autonomously in other robots, if they have sensors capable of determining this information individually and with enough precision. Note that, in this case, each robot would not require additional hardware to determine its own heading direction, since the position of its neighbors would already be measured with respect to the orientation of its own reference frame.

It is important to point out that a fully onboard implementation of the AE algorithm (including all sensing) would result in several differences when comparing with our experiments. The most notable practical difference is that no external position acquisition capabilities would be needed, thus allowing the implementation of the AE algorithm in a broader range of conditions. Regarding algorithm performance, the difference will depend on the precision and processing times of the onboard sensors when compared with our arena. On one hand, these sensors will typically be less precise than our external system, and could perform badly when confronted with occlusions or large inter-agent distances. On the other hand, our external system is based on image analysis, which has its own limitations. For example, the processing time is relatively long (and increases with the number of robots) and the position detection fails at the edges of the arena. The effects on noise and performance of using onboard sensors will thus depend on how they behave under the required operational conditions and for the desired swarm dynamics.

An important difference between onboard and external position acquisition systems that can be addressed in a general framework stems from the synchronous or asynchronous nature of their updating processes. Indeed, our arena acquisition system measures and broadcasts the required position information to all agents almost simultaneously. This implies that all robots update their control algorithms practically at the same time. In contrast, onboard sensors are typically not synchronized, so individual robot dynamics will be updated asynchronously. In order to test the effects of this difference, we performed preliminary simulations that show that the asynchronous and synchronous dynamics follow similar trends (data not shown). The asynchronous simulations appear to produce more noisy trajectories, however, especially in the high $\beta$ cases, although this difference is reduced for smaller $\Delta t$ values. We thus find that similar analyses and overall conclusions could be applied to the asynchronous case, although its complete understanding will require systematic studies that are left for future work.

The use of an external tracking system allowed us to study in detail the properties of the experimental noise. Indeed, the fact that the information required to execute the control algorithm is collected and broadcasted simultaneously to all robots makes our experiment equivalent to a numerical simulation with synchronous update. We are thus able to analyze the experimental noise by comparing the predicted and observed robot states after every time-step. We found that the noise distribution of both the angular and linear speeds can be well fitted by Gaussians. However, we also observed small deviations from this distribution in the form of long-tails that correspond to strong perturbations with low probability. Despite these differences, we were able to successfully model the effects of noise by adding to the linear and angular speed of each agent a Gaussian-distributed random variable with the corresponding, experimentally measured, standard deviation.

By simulating the effects of noise, we found that we can numerically predict the approximate dynamics observed in experiments. The speed of self-organization and the stability of the ordered states are slightly lower in experiments than in simulations, however, which is likely due to the observed non-Gaussian fluctuations and potential underlying correlations of the noise distributions. This could be verified in future work by using random variables with the exact experimental noise distributions. Despite these small differences, our results show that numerical simulations that approximately include the experimental noise can be used to determine the optimal operational regime of a real-world robot swarm.

We also found in this article that the AE algorithm produces a self-organized translating state in which robots display small persistent oscillations about the aligned state. We studied the source of these oscillations by carrying out a simple two-robot analytical calculation and showed that they are a generic feature of the AE control algorithm, resulting from the marginal linear stability of the aligned state.

Taken together, the results highlighted above suggest various ways in which our work could be generalized to other swarm robotic systems. First, our experimental analyses could be useful for the future implementation of position-based algorithms in any context where exchanging orientational information is impossible or inefficient (for example, if the robots have very limited sensing capabilities). Second, our studies lay the groundwork for extending this type of algorithms to other experimental robotic swarms, such as unmanned underwater or aerial vehicle swarms, since the AE model could be extended to three dimensions and to include the effects of inertia, winds, currents, and so on. Third, our characterization of the noise dynamics and of the response to parameter changes under real-world conditions could be generalized to other types of swarms. Fourth, we expect that an intermediate, compromise $\alpha$ and $\beta$ parameter region (where convergence is fast but also stable to large perturbations), as well as the observed persistent oscillations (which we identified as a potential problem), should be present in other similar position-based systems. Finally, our analysis methods for characterizing noise distributions could be extended to other robot swarms, and in some real-world scenarios where the control update can be synchronous, we showed that these distributions can be directly incorporated into simulations to predict optimal parameter ranges.

In future work, we plan to further develop the noise characterization carried out in this article, in order to better match the theoretically predicted and the experimentally observed robot trajectories. We also plan to consider model modifications that could correct the marginal stability identified in the aligned state, which should improve its robustness. Furthermore, given that we demonstrated that rotational collective motion can be achieved by simply imposing different preferred speeds to different robots, we plan to test other simple modifications that could achieve new self-organized states, going beyond the usual collective translation or rotation. Finally, it would be interesting to extend our AE algorithm to three-dimensional systems, in order to test its effectiveness for Unmanned Aerial Vehicle (UAV) swarm control.

An interesting question that was not addressed in this article is how a group of robots that follow the AE algorithm would interact with obstacles. This is a complex matter, since the answer will strongly depend on their shape and their type of interactions with the robots. We can argue, however, that the AE algorithm has interesting potential capabilities in this regard. Indeed, since the model already includes attraction–repulsion forces, it provides a natural setting for designing different obstacle collision avoidance rules that can address specific swarm-obstacle interaction challenges. For example, if obstacles are treated as purely repulsive forces that are strong at short range but weak at intermediate (order $l_0$) distances, small obstacles could pass through the group without colliding with individual robots and without significantly affecting its structure. In contrast, if the obstacles are treated as having strong forces at scales larger than $l_0$, they would act as reflective barriers for the whole group. These examples show that the choice of the robot–obstacle interactions is critical in determining the resulting group dynamics. Therefore, developing an understanding of the capabilities of the AE algorithm in this context will require a systematic exploration of multiple scenarios. This interesting and worthy new research will also be left for future work.

## Acknowledgements

We are grateful for the help provided by Professor Zhongqi Sun (Beijing Institute of Technology) in implementing the Wi-Fi communication module that they designed and by Professor Lei Liu (University of Shanghai for Science and Technology) in debugging the tracking system.

## Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

## Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported by China National Natural Science Foundation (Grant No. 61374165) and China Scholarship Council (CSC) (Grant No. 201806040106).

## ORCID iD

Yating Zheng https://orcid.org/0000-0002-0261-9742

## Supplemental material

Supplemental material for this article is available online.

## Notes

[1] Note that the model in Couzin et al. (2002), for example, only achieves collective rotation due to the presence of long-range attraction forces that act against the alignment interactions by generating an effective force towards the center of the group.

2 Another algorithm known to achieve collective motion without orientation information exchange was proposed in Szabó et al. (2006).

## References

Bhattacharya, K., & Vicsek, T. (2010). Collective decision making in cohesive flocks. *New Journal of Physics*, *12*(9), Article 093019.

Bialek, W., Cavagna, A., Giardina, I., Mora, T., Silvestri, E., Viale, M., & Walczak, A. M. (2012). Statistical mechanics for natural flocks of birds. *Proceedings of the National Academy of Sciences of the United States of America*, *109*(13), 4786–4791.

Brambilla, M., Ferrante, E., Birattari, M., & Dorigo, M. (2013). Swarm robotics: A review from the swarm engineering perspective. *Swarm Intelligence*, *7*(1), 1–41.

Calovi, D. S., Lopez, U., Ngo, S., Sire, C., Chaté, H., & Theraulaz, G. (2014). Swarming, schooling, milling: Phase diagram of a data-driven fish school model. *New Journal of Physics*, *16*(1), Article 015026.

Cavagna, A., Giardina, I., & Ginelli, F. (2013). Boundary information inflow enhances correlation in flocking. *Physical Review Letters*, *110*(16), Article 168107.

Chaté, H., Ginelli, F., Grégoire, G., & Raynaud, F. (2008). Collective motion of self-propelled particles interacting without cohesion. *Physical Review E*, *77*(4), Article 046113.

Couzin, I. D., Krause, J., James, R., Ruxton, G. D., & Franks, N. R. (2002). Collective memory and spatial sorting in animal groups. *Journal of Theoretical Biology*, *218*(1), 1–11.

Ferrante, E., Turgut, A. E., Dorigo, M., & Huepe, C. (2013a). Collective motion dynamics of active solids and active crystals. *New Journal of Physics*, *15*(9), Article 095011.

Ferrante, E., Turgut, A. E., Dorigo, M., & Huepe, C. (2013b). Elasticity-based mechanism for the collective motion of self-propelled particles with springlike interactions: A model system for natural and artificial swarms. *Physical Review Letters*, *111*(26), Article 268302.

Ferrante, E., Turgut, A. E., Huepe, C., Stranieri, A., Pinciroli, C., & Dorigo, M. (2012). Self-organized flocking with a mobile robot swarm: A novel motion control method. *Adaptive Behavior*, *20*(6), 460–477.

Gautrais, J., Ginelli, F., Fournier, R., Blanco, S., Soria, M., Chaté, H., & Theraulaz, G. (2012). Deciphering interactions in moving animal groups. *PLOS Computational Biology*, *8*(9), Article e1002678.

Gautrais, J., Jost, C., & Theraulaz, G. (2008). Key behavioural factors in a self-organised fish school model. *Annales Zoologici Fennici*, *45*(5), 415–428.

Katz, Y., Tunstrøm, K., Ioannou, C. C., Huepe, C., & Couzin, I. D. (2011). Inferring the structure and dynamics of interactions in schooling fish. *Proceedings of the National Academy of Sciences of the United States of America*, *108*(46), 18720–18725.

Kushleyev, A., Mellinger, D., Powers, C., & Kumar, V. (2013). Towards a swarm of agile micro quadrotors. *Autonomous Robots*, *35*(4), 287–300.

Mathews, N., Christensen, A. L., OGrady, R., Mondada, F., & Dorigo, M. (2017). Mergeable nervous systems for robots. *Nature Communications, 8*, Article 439.

Mersch, D. P., Crespi, A., & Keller, L. (2013). Tracking individuals shows spatial fidelity is a key regulator of ant social organization. *Science*, *340*(6136), 1090–1093.

Mondada, F., Bonani, M., Raemy, X., Pugh, J., Cianci, C., Klaptocz, A., Magnenat, S., Zufferey, J.-C., Floreano, D., & Martinoli, A. (2009). The e-puck, a robot designed for education in engineering. In P. J. S. Gonçalves, P. J. D. Torres, & C. M. O. Alves (Eds.), *Proceedings of the 9th conference on autonomous robot systems and competitions* (*Vol. 1*, pp. 59–65). Instituto Politécnico de Castelo Branco.

Pitcher, T. J. (1983). Heuristic definitions of fish shoaling behaviour. *Animal Behaviour*, *31*(2), 611–613.

Radakov, D. V. (1973). Schooling in the ecology of fish. John Wiley & Sons.

Rubenstein, M., Cornejo, A., & Nagpal, R. (2014). Programmable self-assembly in a thousand-robot swarm. *Science*, *345*(6198), 795–799.

Sartoretti, G., Hongler, M.-O., de Oliveira, M. E., & Mondada, F. (2014). Decentralized self-selection of swarm trajectories: From dynamical systems theory to robotic implementation. *Swarm Intelligence*, *8*(4), 329–351.

Szabó, B., Szöllösi, G. J., Gönci, B., Jurányi, Z., Selmeczi, D., & Vicsek, T. (2006). Phase transition in the collective migration of tissue cells: Experiment and model. *Physical Review E*, *74*(6), Article 061908.

Toner, J., & Tu, Y. (1995). Long-range order in a two-dimensional dynamical XY model: How birds fly together. *Physical Review Letters*, *75*(23), 4326–4329.

Tunstrøm, K., Katz, Y., Ioannou, C. C., Huepe, C., Lutz, M. J., & Couzin, I. D. (2013). Collective states, multistability and transitional behavior in schooling fish. *Plos Computational Biology*, *9*(2), Article e1002915.

Vicsek, T., Czirók, A., Ben-Jacob, E., Cohen, I., & Shochet, O. (1995). Novel type of phase transition in a system of self-driven particles. *Physical Review Letters*, *75*(6), 1226–1229.

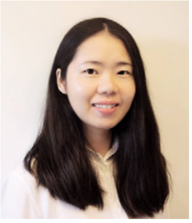
Vicsek, T., & Zafeiris, A. (2012). Collective motion. *Physics Reports, 517*(3–4), 71–140.

Vigelius, M., Meyer, B., & Pascoe, G. (2014). Multiscale modelling and analysis of collective decision making in swarm robotics. *PLOS ONE*, *9*(11), Article e111542.

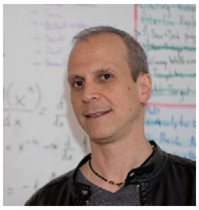
Wang, C., Xie, G., & Cao, M. (2014). Controlling anonymous mobile agents with unidirectional locomotion to form formations on a circle. *Automatica*, *50*(4), 1100–1108.

Zhang, H.-P., Béer, A., Florin, E.-L., & Swinney, H. L. (2010). Collective motion and density fluctuations in bacterial colonies. *Proceedings of the National Academy of Sciences of the United States of America*, *107*(31), 13626–13630.

## About the Authors



**Yating Zheng** is a PhD student at the School of Systems Science, Beijing Normal University, China, under the supervision of Professor Zhangang Han since September 2015. She is also a visiting PhD student at IRIDIA, Université Libre de Bruxelles, Belgium from September 2018  till now. Her supervisor is Professor Marco Dorigo. She has received a B.Sc. in 2015 from the School of Information and Control in Nanjing Information and Engineering University, China.



**Cristián Huepe** is the founder and director of the CHuepe Labs in Chicago, an external researcher at the School of Systems Science in Beijing University (China), the head research director at the SoL-UC lab in Santiago (Chile), and an adjunct research professor at ESAM and NICO in Northwestern University (USA). He received a Ph.D. in physics from University of Paris 7 and École Normale Supérieure de Paris and his B.Sc. in physics from Universidad de Chile. His research has spanned various fields in nonequilibrium nonlinear dynamics, statistical physics, and complex systems. He currently works in active matter, collective motion, swarm robotics, self-organization, emergent phenomena, complex networks, opinion dynamics in social networks and the interface between art and science.

**Zhangang Han** is a Professor and the Associate Dean at the School of Systems Science, Beijing Normal University, China. He received his PhD degree from Institute of Mathematics in Chinese Academy of Sciences, Beijing, 1997. He was a visiting scholar at the Institute of Solvay, Université Libre de Bruxelles, Belgium from 1997 to 1999, and at University of California, Los Angeles from 2003 to 2005.