

# Impressionist Algorithms for Autonomous Multi-Robot Systems: Flocking as a Case Study

Florian Berlinger<sup>1</sup>, Julia T. Ebert<sup>2</sup>, and Radhika Nagpal<sup>1</sup>

**Abstract**—Robot swarms have the potential to revolutionize areas ranging from warehouse management and agriculture to underwater and space exploration. However, there remains a substantial gap between theory and robot implementation. While algorithms might assume reliable communication, perfect sensing, and instantaneous cognition, most robots have lossy or even no communication, imperfect sensing, and limited cognition speed. In our previous work on implicit vision-based coordination, we demonstrated autonomous three-dimensional behaviors underwater by removing the need for radio communication between robots. Here we explore impressionist algorithms, capable of working with even more minimal information where traditional algorithms are prone to fail. Our case study focuses on classic flocking behaviors, where a robot swarm must coordinate group motion. We demonstrate that reliable alignment, dispersion, and milling can be achieved with only infrequent and imperfect sensory impressions. In simulation studies and theoretical analyses, we investigate the effect of systematically reducing spatial and temporal fidelity of individual information on the success metrics for the group; we also demonstrate physical experiments with Blueswarm robots using simple color detection. Our results show the potential of impressionist algorithms that operate on simpler neighborhood-awareness metrics and still achieve desired global goals.

## I. INTRODUCTION

Most future applications of robots involve large numbers working together to achieve high parallelism, robustness, and scalability. From robotic warehouses to agricultural monitoring and self-driving cars, robots will need to coordinate their behavior with the rest of the system to achieve global goals. Many coordination algorithms involve robots interacting with their local neighbors, allowing the distributed system to scale well as the number of robots increases. Nevertheless, even local-neighbor algorithms can make unrealistic assumptions, such as perfect and instantaneous communication and sensing of neighbor position. In contrast, most robots have limitations on sensing accuracy and computation speed, and many environments can be challenging for communication.

One such application is underwater robot swarms to investigate reefs, search for people, or patrol infrastructure. However, wireless communication performs poorly and localization methods such as GPS are unavailable underwater. Instead of explicit wireless communication, bio-inspired implicit coordination algorithms rely on passive neighbor ob-

This work was supported by the Office of Naval Research (ONR Award No.: N00014-22-1-2222), the Wyss Institute for Biologically Inspired Engineering, and an Amazon AWS Research Award. The supplementary material is available at [https://bit.ly/impressionist\\_algos](https://bit.ly/impressionist_algos). The authors are with the Department of Mechanical and Aerospace Engineering at Princeton University, Princeton, New Jersey<sup>1</sup>, and the John A. Paulson School of Engineering and Applied Science at Harvard University, Cambridge, Massachusetts<sup>2</sup>. E-mail: fcberlinger@gmail.com.

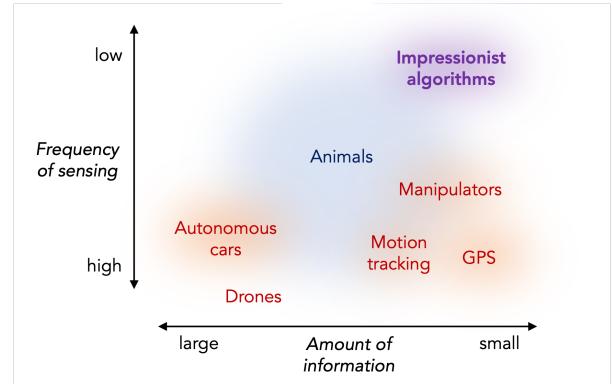


Fig. 1. Perception as a combination of spatial and temporal impressions. Typically, faster systems require higher sensing frequencies, and less structured environments larger amounts of information. Many animals, however, are able to cope with noisy, incomplete, and infrequent information when it comes to decision making. Inspired by this, impressionist algorithms are designed to lessen the perception and computation burden on robots while providing extra robustness by relying on fewer data points.

servations only. Fish, for instance, use implicit coordination for complex group tasks such as migration or foraging. We recently demonstrated the potential of implicit algorithms on Blueswarm, a platform for underwater collective behaviors [1]. Compared to theoretical implicit algorithm models that assume ideal and instantaneous sensing, Blueswarm robots have several limitations in neighbor observations, e.g., visual errors in parsing neighborhood information and limited perception speed. This motivates the question, how much spatial and temporal accuracy is really necessary?

In this paper we introduce the concept of impressionist algorithms and define them as a class of local-neighborhood algorithms that enables autonomous robots to collectively coordinate based on limited spatial and temporal resolution, i.e., incomplete neighbor information and limited reaction speed (Fig. 1). This contrasts with ideal local-neighborhood algorithms that assume full and perfect local neighbor information (i.e., position and velocity) and instantaneous reactions (i.e., infinite sensing-actuation frequency).

We evaluate impressionist algorithms for three tasks that are important for collective navigation: heading alignment so that all robots are moving in the same direction, dispersion to create a spread-out but connected sensor network, and milling which is a circular flocking motion (Fig. 2). Using simulations, we study and modify existing implicit coordination algorithms for alignment, dispersion, and milling in [1] and [2] to show the extent to which they can function as impressionist algorithms. We investigate the effect of

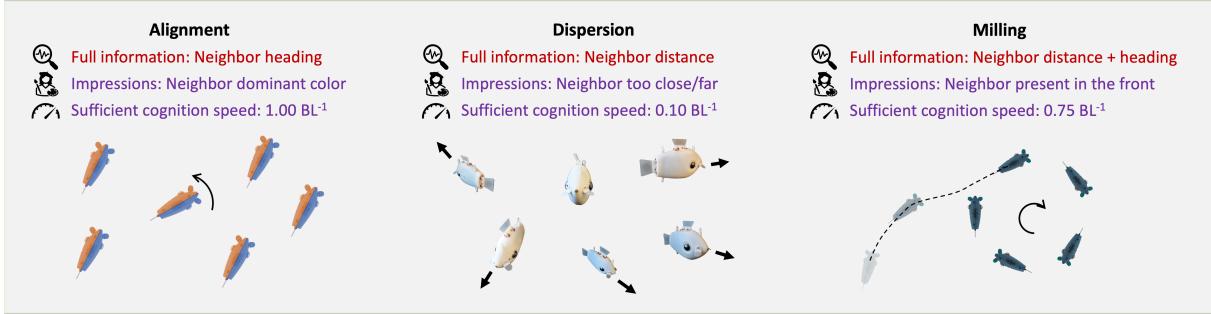


Fig. 2. **Impressionist flocking behaviors.** Previously, alignment, dispersion, and milling required high resolution headings or distances. With impressionist algorithms, the same behaviors can be achieved from rough color, size, or even presence estimates instead, and work at exceptionally low cognition speeds.

systematically reducing spatial and temporal fidelity of individual information to determine the thresholds on spatial and temporal resolution at which global behavior deteriorates.

Dispersion and milling can be achieved by impressionist algorithms that can only detect the bearing of neighbors and operate at low temporal resolutions. We demonstrate both tasks on physical experiments with uni-colored robots, and provide some theoretical insights into the success of the impressionist algorithm for milling. For alignment we propose an impressionist algorithm that uses binary body coloring instead of the heading of fellow robots, which is difficult to detect visually with high accuracy; we evaluate this algorithm’s performance using simulation and theoretical models. Our results show the potential of impressionist algorithms that operate on simpler neighborhood perception and still achieve desired global goals.

## II. RELATED WORK

Many flocking behaviors seen in natural and robotic collectives — including alignment [3], [4], dispersion [5], [6], and milling [1], [7] — can be described with simple local rules. A wealth of theoretical and simulation work exists that investigates and in some cases proves abstract versions of flocking [8]–[10]. In most cases, instances of flocking are studied in isolation. It is possible, however, to achieve multiple different flocking behaviors from adaptations to a single underlying rule or restrictions on the perception [11].

Inevitably, simulation models often make strong assumptions on available sensory data and locomotive capabilities, which complicates their transfer onto robotic systems. Consequently, few autonomous demonstrations of flocking exist in the robotics domain and most are restricted to two-dimensional space [1], [2], [12]–[15]. More typically, robot swarms rely on assistive technologies (e.g., base stations or external position information), which provide them with high frequency and fidelity data updates [16]–[22].

The domain of robot swarms deliberately designed to cope with low sensing frequency and small amounts of data is largely unexplored (Fig. 1). In many top-down approaches, algorithms require more data than robots can gather and make them reliant on external assistance (e.g., [21]). One way to address this gap is to find algorithms amenable to the real communication, sensing, and dynamics with evolutionary or machine learning (e.g., [23]). Such algorithms,

however, often lack interpretability and theoretical guarantees. In contrast, bottom-up approaches assume a priori that robots have very minimal sensors, and explore what group behaviors are possible given such strong assumptions. For instance, robots with a line sensor only can achieve circular milling motion [7]. This is in contrast to traditional bio-inspired milling behaviors that assume full knowledge of neighbor positions and headings [24], [25].

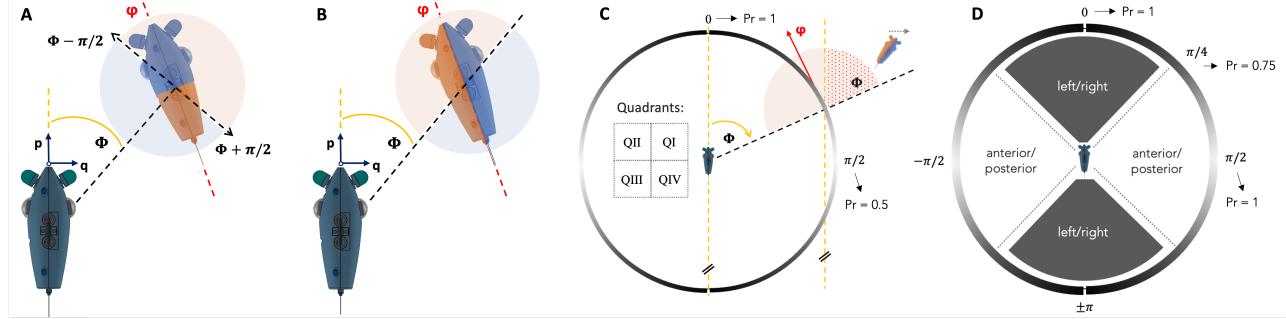
## III. TASKS, METRICS, AND ALGORITHMS

In this paper, we achieve impressionist algorithms by systematically reducing information to existing algorithms while aiming to maintain group performance. We tackle three autonomous flocking tasks with existing implicit coordination algorithms: alignment, dispersion, and milling (Fig. 2). We evaluate spatial resolution: can the algorithms perform without full neighbor position information (bearing, distance, heading); and also temporal resolution: can the algorithms operate at low cognition speeds, where cognition speed is defined as the ratio of perception frequency to motion speed.

For simulations, we used Bluesim [2], a custom simulator that closely matches the perception and locomotion of Bluebots. Robots of size  $\rho = 80\text{mm}$  were initialized uniformly at random over a surface square area of size  $1000\text{mm} \times 1000\text{mm}$  within an unbounded environment. By default, they had a perception frequency and swimming speed of 2Hz and  $130\text{mm/s} \equiv 1\text{BL/s}$  (body length/s). For robotic demonstrations, we used Blueswarm, a versatile platform for self-organized collective behaviors [1], [2], biomimetic actuation [26], [27], and fish swimming [28], [29]. Bluebots had four independently controllable fins to move in 3D space [30] and two cameras to detect fellow robots [1]. The laboratory tank was of size 1.78m by 1.78m by 1.17m.

### A. Alignment

Alignment is a task in which all robots must match their heading direction. We defined the quality of alignment by the final circular standard deviation of robot headings  $\sigma_\phi$  and declared alignment stable if  $\sigma_\phi \leq 0.5$  [2]. A classic alignment algorithm involves distributed averaging; robots track the headings of neighbors and rotate toward the average, which guarantees convergence [31]. We propose and compare impressionist versions of this algorithm where robots can only infer a lower resolution heading from body colorings.



**Fig. 3. Geometry of alignment with colored robots.** A colored robot with actual heading  $\varphi$  is detected at an angle  $\phi$ . Red is the salient color if the robot's heading  $\varphi$  is within the red semicircle. **(A)** Anterior/posterior colored: red corresponds to  $\phi - \pi/2 \leq \varphi < \phi + \pi/2$ . **(B)** Left/right colored: red corresponds to  $\phi - \pi \leq \varphi < \phi$ . **(C)** The probability of assigning a new heading  $\varphi$  that matches a robot's cardinal direction (left/right) depends on  $\phi$  and is color-coded on the unit circle: darker segments correspond to higher probabilities for left/right (and lower probabilities for anterior/posterior) colored robots. For instance, in the first quadrant when seeing more red on a left/right colored robot, the probability is  $1 - |\phi|/\pi$ ; robots in the dotted area get assigned a heading  $\varphi$  that does not match their cardinal direction. The average probability of a matching heading for bi-colored robots is 0.75. **(D)** The anterior/posterior and left/right colorings work well in complementary angular regions. When combined on quad-colored robots, the average probability of a heading assignment that matches a robot's cardinal direction increases to 0.875.

### B. Dispersion

Dispersion is a task where the robots must spread out to create a sensor network with some target distance  $d_t$  between neighboring robots. A classic algorithm for achieving dispersion is based on the Leonard-Jones potential, where robots use potential-based attractive and repulsive virtual forces derived from the distance of each neighbor (cf. [1]). We evaluate modified impressionist algorithms where robots can only sense binary (too close/far), or tertiary (too far, ok, too close) distance estimates and compare their performance to using exact distances by measuring the average robot distance as a function of the target distance  $d_t$ .

### C. Milling

Milling is a dynamic task where robots swim in a clockwise (or counterclockwise) circular formation. Many algorithms have been proposed for milling that rely on perfect neighbor positions [24], [25]; recently, a minimalist algorithm for milling with ground robots was proposed where robots only pay attention to the presence of robots in a frontal view [7]. This algorithm is inherently impressionist, and we previously demonstrated its success on seven physical Bluebot robots [1]. Here we dig in deeper to better understand the spatial and temporal limits, and provide some theoretical insights into the robustness of this impressionist algorithm.

We define the success by how circular the formation is. To determine the circle position and radius, we repeatedly fitted circles with center and radius ( $\mathbf{c}_{fit}, r_{fit}$ ) through the  $N$  simulated robots, using linear least squares (illustrated in the supplement). As a first metric, we required the distribution  $\sigma$  of the robots along the circle to be low, which, due to symmetry, is the case if robots are uniformly distributed; as a second metric, we required the closest ( $r_{min}$ ) and furthest ( $r_{max}$ ) robot from the circle center to be within a tight bound, which ensures that all robots are close to the circle:

$$\sigma = \left\| \sum_{i=1}^N \mathbf{p}_i - N\mathbf{c}_{fit} \right\| < 1.5r_{fit} \quad \& \quad \frac{r_{max}}{1.1} < r_{fit} < \frac{r_{min}}{0.9}, \quad (1)$$

where  $\mathbf{p}_i$  are the  $xy$ -coordinates of robot  $i$  and  $\mathbf{c}_{fit}$  are the  $xy$ -coordinates of the fitted circle center. The convergence time  $t_{conv}$  is the first instance which satisfies both metrics.

## IV. ALIGNMENT RESULTS

Our previous demonstrations of alignment relied on headings derived from LED positions on the Bluebots, and could be achieved by a simple averaging algorithm [2]. We showed that simulated alignment worked even if inferred robot headings were reduced to two cardinal directions (left/right). In other words, robots capable of estimating whether fellow robots are pointed to their left or right can align (Fig. 5A). Here we introduce a more natural and impressionist alignment algorithm that observes multiple body colors on each fellow robot, similar to schooling marks seen in reef fish. A fellow robot's orientation is estimated based on color salience, then fed to the same averaging procedure for alignment as before. In the following, we analyze different colorings and present simulation results on their effectiveness with regards to successful alignment.

### A. Anterior/posterior colored robots

It is challenging to conclude whether a fellow robot definitely faces to the left or right solely based on a body coloring rather than reliable heading information. However, by looking at a robot body, it is possible to decide whether the robot moves toward or away from oneself. If the anterior and posterior ends were colored differently as in Fig. 3A, one simply has to decide which color is salient. A robot swimming away ( $\phi - \pi/2 \leq \varphi < \phi + \pi/2$ ) is assigned a radially outward heading  $\varphi$  that corresponds to the angle of detection  $\phi$ ;  $\pi$  is added (a  $180^\circ$  rotation) if the robot is swimming toward (Eq. 2 and Fig. 3A). The planar direction is known from the  $pq$  coordinates as  $\phi = \text{arctan2}(q, p)$  [1].

$$\begin{aligned} \varphi &\leftarrow \phi && \text{if away (red posterior end)} \\ \varphi &\leftarrow \phi + \pi && \text{o/w} \end{aligned} \quad (2)$$

Simulations showed that alignment following this protocol also leads to convergence in robot headings (Fig. 4).

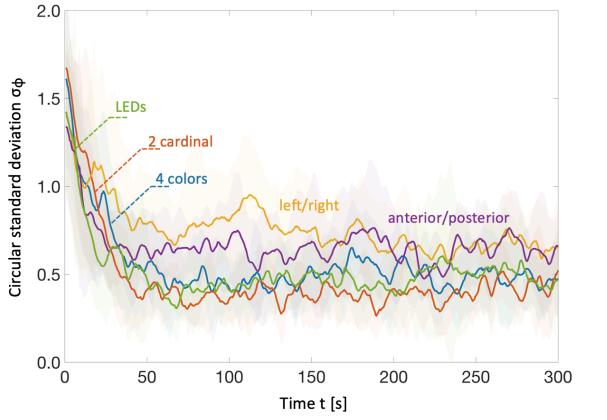


Fig. 4. **Simulations for alignment with multi-colored robots.** Robots with 4 colors aligned roughly as well as robots with exact (LEDs) and binary (2 cardinal) headings. Robots with anterior/posterior or left/right colorings had a 40% higher circular standard deviation after 300 s.  $N = 7$  robots were used, and data points are presented as the mean (solid lines) and standard deviation (shades) from  $n = 10$  simulation runs.

#### B. Left/right colored robots

Judging whether a robot is swimming away or toward — or which of two colors is salient — can be difficult, especially if one is presented a side view of that robot. Given the shape of a Bluebot, it is substantially easier to decide which lateral side one faces if they are colored differently, e.g., red on the left and blue on the right (Fig. 3B). The following equation assigns headings  $\phi$  perpendicular to the angle of detection  $\phi$  based on which side is more visible:

$$\begin{aligned} \varphi &\leftarrow \phi - \pi/2 && \text{if more red (left lateral side)} \\ \varphi &\leftarrow \phi + \pi/2 && \text{o/w} \end{aligned} \quad (3)$$

Alignment simulations showed convergence in this case as well (Fig. 4). The difference between the anterior/posterior and left/right colorings (cf. equations 2 and 3) is simply a  $90^\circ$  rotation. It is easy to see that two color segments could be arranged at any angle around the vertical axis of the robot. However, the left/right arrangement is the simplest for deciding which color is salient with this robot shape.

#### C. Probabilistic analysis

In the following, we analyze why alignment with bi-colored robots did not converge as well as alignment based on two cardinal directions (Fig. 4). In the case of two cardinal directions (the benchmark), the robot heading  $\phi$  is approximated to  $\pm\pi/2$  but the direction (left/right) is always assigned correctly (Fig. 5A). With bi-colored robots, the probability  $\Pr(\phi)$  of assigning an approximated orientation  $\phi$  that matches the robot's left/right direction depends on the angle of detection  $\phi$  (Fig. 5B). For a left/right colored robot,  $\Pr(\phi) = 1$  for  $\phi = \{0, \pi\}$ ; e.g., at  $\phi = 0$  and as long as the exact robot heading has a leftward component, red is salient. The observing robot sees red and assigns a correct leftward heading. However, this probability drops to  $\Pr(\phi) = 0.5$  for  $\phi = \pm\pi/2$  (Fig. 3C and Eq. 4). For instance at  $\phi = \pi/2$ , red is equally likely to be salient whether the exact heading

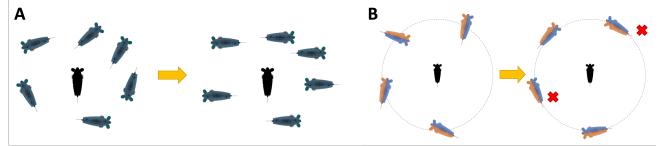


Fig. 5. **Alignment with discretized headings.** (A) All headings are correctly grouped into one of two cardinal directions: left or right. (B) Based on salient color, headings are set perpendicular to the planar direction  $\phi$  at which a robot appears. The headings of the two robots marked with a red cross do not match their correct cardinal directions.

has a left- or rightward component. Conversely, for an anterior/posterior colored robot,  $\Pr(\phi) = 0.5$  for  $\phi = \{0, \pi\}$  and  $\Pr(\phi) = 1$  for  $\phi = \pm\pi/2$ .

$$\begin{aligned} \Pr(\phi) &= 1 - |\phi|/\pi && \text{if } -\pi/2 \leq \phi < \pi/2 \\ \Pr(\phi) &= |\phi|/\pi && \text{o/w} \end{aligned} \quad (4)$$

To improve alignment, we weighted the inferred angles  $\varphi$  proportionally to their probability  $\Pr(\phi)$  of being correct (supplement). Intuitively, alignment with bi-colored robots should converge because the probability of assigning the correct cardinal (left/right) direction is always at least 0.5. Averaged over time and robot detections, however, the probability of assigning the correct direction is only 0.75; therefore, bi-colored convergence is inferior to cardinal convergence.

#### D. Robots with more than two colors

Because headings inferred from anterior/posterior and left/right colorings have complementary probabilities of pointing into the correct cardinal direction, it is possible to combine them to improve alignment (Fig. 3D). For instance, on a quad-colored robot with distinguishable anterior (e.g., light), posterior (e.g., dark), left and right (red and blue) sides, we would use the left/right distinction for forward ( $-\pi/4 \leq \phi < \pi/4$ ) and backward ( $3\pi/4 \leq \phi < \pi$  or  $-\pi \leq \phi < -3\pi$ ) facing directions, and the anterior/posterior distinction for left and right facing directions. The resultant average probability of a heading assignment that matches a robot's cardinal direction is now 0.875 (compared to 0.75 with bi-colored robots). Simulations of such alignment converged substantially better than bi-colored alignment and were roughly on par with cardinal and exact LED-based alignment (Fig. 4). Therefore, given accurate color detection, similar alignment results could be expected for real robot experiments with this simpler impressionist algorithm.

#### E. Sufficient cognition speed $f_c$

We used a cognition speed of  $2BL^{-1}$  for above experiments. Simulation results from our previous research showed that alignment starts to work at a cognition speed of  $1BL^{-1}$  ( $\sigma_\phi \approx 0.6$ ) and stops improving further at  $2BL^{-1}$  ( $\sigma_\phi \approx 0.3$ ) [2]. During successful evasive maneuvers with Bluebots which were part of the same study [2], we measured actual cognition speeds ranging from  $0.96BL^{-1}$  to  $3.42BL^{-1}$ .

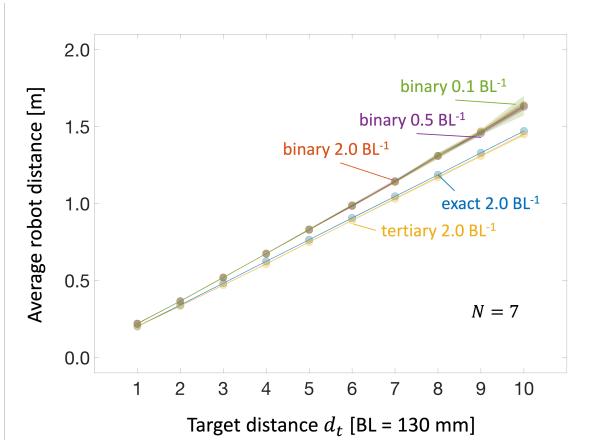


Fig. 6. Dispersion with binary (too close/far) and tertiary (too close, ok, too far) classifications of neighbor distances enables predictable average robot distances. Moreover, dispersion is robust against low cognition speeds.  $N = 7$  robots were used, and data points are presented as the mean (colored dots, often overlapping) and standard deviation (shades) from  $n = 10$  simulation runs. Exact  $2.0\text{BL}^{-1}$  is considered the reference case.

## V. DISPERSION RESULTS

We previously demonstrated potential-based dispersion [1]. The attractive or repulsive forces of fellow robots depended on their distance, inferred from the posterior LEDs on a Bluebot. The overall spread of the collective was controlled with a single parameter, the target distance  $d_t$ .

Here we introduce impressionist algorithms that do not rely on exact perception of neighbor distances, and also show how attractive and repulsive forces can be used for collective obstacle avoidance or leader following. In the case of binary distances, Bluebots classified fellow robots as either too close or too far. All robots closer than the target distance  $d_t$  were treated as if at 80% of  $d_t$ , and robots further than  $d_t$  at 120% of  $d_t$ . As a result, we only used two virtual forces as opposed to using forces proportional to the exact inter-robot distances as in previous work. In the case of tertiary distances, we added a satisfactory and force-free zone located within  $\pm 10\%$  of  $d_t$ . Closer robots were again treated as if at 80% of  $d_t$ , and further robots as 120% of  $d_t$ .

### A. Sufficient perception of distances

Estimating the exact position of neighboring robots requires accurate sensing and expensive image processing. Instead, by only having to estimate whether fellow robots are closer, further, or approximately at the target distance, Bluebots can still achieve predictable spacing. This can be done, for instance, by looking at the blob size of a neighboring robot, where the nominal blob size is fixed at the target distance. In our experiments with binary and tertiary distances, these impressionist forms of sensing led to the same linear scaling between the chosen target distance and the resulting average robot distance (Fig. 6). Dispersion with binary distances (too close/far) can overshoot the expected average neighbor distance due to the nonlinear potential function, by which closer robots exert stronger forces than further robots. Alternative potential functions or different

classification distances (e.g., 90% and 130% of  $d_t$  instead of the symmetric 80% and 120%) may be less prone to overshooting. By addition of a non-control zone, dispersion with tertiary distances followed exact dispersion very closely.

### B. Sufficient cognition speed $f_c$

The nominal cognition speed for dispersion was  $2\text{BL}^{-1}$ . For binary dispersion, the most impressionist version, we lowered cognition speed down to  $0.1\text{BL}^{-1}$  without observing significant changes in performance (Fig. 6). Dispersion gets away with severely limited amounts of information (too close/far) and low frequencies of sensing because individual robot errors are averaged out across the collective and time.

### C. Minimalist perception for Bluebot experiments

Dispersion can also be achieved with robots moving toward or away from the *unweighted* average, i.e., only having to consider the direction of fellow robots but *not* their distance. However, with this simpler unweighted dispersion it is not possible to set a target distance by which the spread of the collective can be controlled — both dispersion and aggregation are infinite (or up to the limit of the visual range). To limit dispersion and avoid partitioning of the collective, other means would have to be employed, such as setting a required minimum number of visible fellow robots.

Unweighted dispersion solely requires the detection of neighboring robots (but not their distance or heading). Here we use the same computationally inexpensive approach for detection as outlined in the milling section below. Snapshots from a color-based dispersion experiment with four black robots are shown in Fig. 7.

### D. Adaptation 1: Dynamic obstacle avoidance

A robot collective can dynamically avoid obstacles while maintaining cohesion by assigning a large repulsive force to unknown objects. To this end, a robot must be able to distinguish between other members of the collective and unknown objects. Here we present three simulated scenarios of such dynamic obstacle avoidance: a collective of 10 robots navigating a static, oncoming, or traversing obstacle (supplementary video). Each robot ran standard weighted dispersion with a target distance  $d_t$  of  $5\text{BL} \equiv 5 \cdot 130\text{mm}$ . The potential function for unknown objects was cropped at  $d_t$  such that they exerted the same repulsive force as fellow robots but never attract a robot. Our results show that this dispersion protocol allows a robot collective to circumnavigate obstacles while maintaining cohesion.

### E. Adaptation 2: Leader following

We can induce leader following by assignment of strong super-attraction to a super-robot (i.e., the leader). Here we present a simulated scenario with a collective of 10 robots. 9 robots ran standard weighted dispersion; the 10<sup>th</sup> robot operated as the leader, swimming a straight line. The potential function for the leader was cropped at  $d_t$  and magnified such that it exerted a 10-fold attractive force but no repulsive force. Our results with three different target

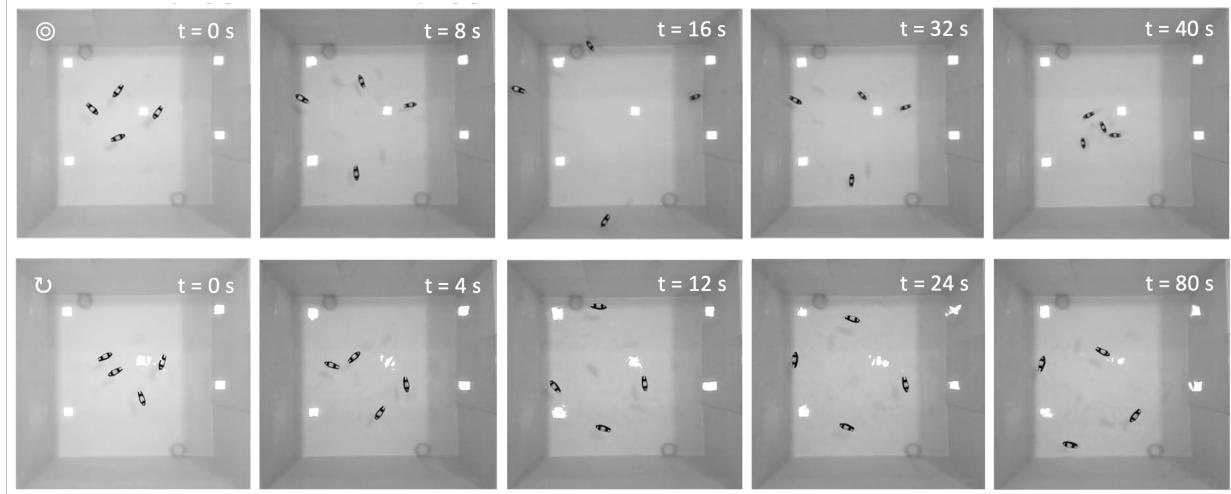


Fig. 7. **Flocking with four colored Bluebots.** Top: Dispersion and aggregation (timed to switch at  $t = 20\text{s}$ ). Bottom: Formation of a clockwise circle and sustained milling. Both behaviors are shown in the supplementary video.

distances  $d_t$  of 1, 5, and 10BL showcase that such dispersion protocol allows a robot collective to follow a designated leader while maintaining cohesion (supplementary video). Here, the robots followed as a flock whereas in the milling-related leader following (below), they followed in line.

## VI. MILLING RESULTS

We previously demonstrated dynamic circle formation and milling based on binary sensing of neighbor presence [1]. Each robot's field of view was simplified to a 3D triangular prism with opening angle  $2\alpha$  (which reduces to a plane of sight in the case of  $\alpha = 0$ ). If any neighbor was detected within this field of view, a robot turned counterclockwise; otherwise, it turned clockwise. We proved that the emergent circle formation is the unique steady state and can be maintained indefinitely. We further derived the following formula for the circle radius  $R$ , given  $N$  participating robots with approximately circular bodies of radius  $\rho$ :

$$R = \frac{\rho}{\cos \alpha - \cos(\frac{2\pi}{N} - \alpha)} \quad (5)$$

This milling algorithm was already impressionist, depending only on detecting presence or absence of neighbors in a front-facing view. Here we explain how the viewing angle parameter in conjunction with the number of robots affects the final formation. We also investigate how the algorithm performs under lower cognition speeds. Finally, we provide new theoretical insights into how and why circular milling emerges from this impressionist algorithm. The robot dynamics were simplified to clockwise or counterclockwise turning at a fixed and constant radius  $r_{\text{swim}} \approx 270\text{mm}$  and velocity  $v \approx 130\text{mm/s}$  (both values resemble actual Bluebot swimming). These simplified dynamics were validated against fully simulated robot dynamics (supplement).

### A. Viewing half-angle $\alpha$ and circle radius $R$

Looking at the denominator of Eq. 5, we observe three cases for the circle radius  $R$ :

$$R \in \begin{cases} \mathbb{R}^+, & \text{if } \alpha < \pi/N \\ \infty, & \text{if } \alpha = \pi/N \\ \mathbb{R}^-, & \text{if } \alpha > \pi/N \end{cases} \quad (6)$$

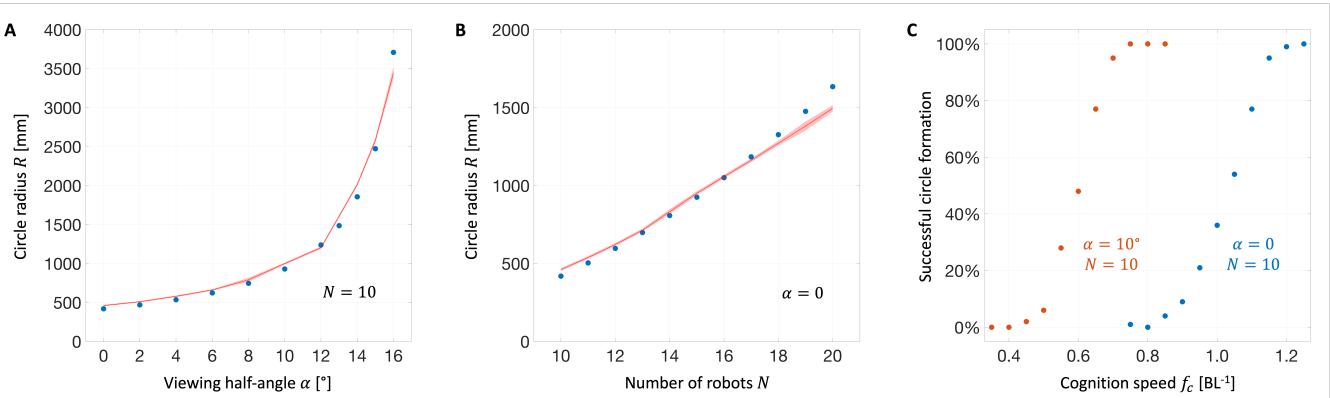
Intuitively, the more robots  $N$  form a circle, the smaller the viewing half-angle  $\alpha$  has to be in order *not* to see some fellow robot at all times (supplement). In practice and given robots with approximately circular bodies of radius  $\rho$ , the viewing half-angle  $\alpha$  has to be chosen below  $\pi/N$  to maintain a circle of radius  $R$ .

To study the effect of the viewing half-angle  $\alpha$  on the circle radius  $R$ , we ran 1100 experiments in which 10 robots successfully formed and maintained a circle. The radii of final circles had a low standard deviation and matched well with the theoretically expected values as per Eq. 5 (Fig. 8A). The convergence time varied significantly across simulation runs but the mean speed of circle formation was not affected largely by different viewing half-angles  $\alpha$  (supplement).

In theory, the circle radius rapidly approaches  $\infty$  around  $\alpha = \pi/N$  and becomes negative for  $\alpha > \pi/N$ . In practice, the robots kept forming circles of continuously growing radii for  $\alpha \geq \pi/N$ . For instance, circles with an  $\alpha$  of  $18^\circ$  and  $20^\circ$  had radii of  $13445\text{mm}$  and  $47862\text{mm}$  after  $10000\text{s}$ . In case of  $\alpha \gg \pi/N$ , all robots cannot fit on the circle without seeing some fellow robot at all times. As a result, we observed circles formed by the majority of robots with a few robots remaining trapped within the circle, rotating in the opposite direction; for instance, eight robots circling and two robots trapped for  $\alpha = 2\pi/N = 36^\circ$  (supplementary video).

### B. Number of robots $N$ and circle radius $R$

To study the effect of the number of robots  $N$  on the circle radius  $R$ , we ran another 1100 experiments with robots given forward looking plane-of-sight sensors ( $\alpha = 0$ ). In all experiments, the robots successfully formed and maintained a circle. The radii of final circles had a low standard deviation and matched well with the theoretically expected



**Fig. 8. Milling simulation results.** (A-B) The circle radius grows with the viewing half-angle and the number of robots, respectively, and can be predicted accurately. Mean and standard deviation of circle radii from  $n = 100$  experiments per data point (red line and shade) as well as theoretically expected radii (blue dots). (C) Cognition speed and viewing angle can be traded off. Percentage of successful circle formations from  $n = 100$  experiments per data point with  $\alpha = 10^\circ$  (red) and  $\alpha = 0$  (blue).

values as per Eq. 5 (Fig. 8B). The convergence time varied significantly across simulation runs (supplement). In general, the more robots  $N$  the larger the circle and the more distance to be travelled. However, the mean convergence time was only marginally affected because robots take parallel action.

#### C. Sufficient cognition speed $f_c$

In the previous experiments, robots had a cognition speed of  $f_c = 2\text{BL}^{-1}$  and successfully formed circles. Here we investigated the critical range of cognition speeds at which the quality of circle formation degraded until successful convergence, as defined by Eq. 1, was no longer possible.

For  $N = 10$  simulated robots with forward looking plane-of-sight sensors ( $\alpha = 0$ ) moving at a velocity of roughly 130 mm/s, a minimum cognition speed of  $f_c = 1.25\text{BL}^{-1}$  was required to enable circle formation (Fig. 8C, blue). At lower cognition speeds, neither the average robot distance from the fitted circle (i.e., the per-robot-error) nor the robot distribution  $\sigma$  (cf. Eq. 1) converged over time (supplement).

The minimum cognition speed can be lowered if more information is available each cycle. We increased the viewing half-angle from  $\alpha = 0$  to  $\alpha = 10^\circ$ . This in turn decreased the chance of missing a robot in the front. As a result, the new minimum cognition speed was  $f_c = 0.75\text{BL}^{-1}$  (Fig. 8C, red). This trade-off exemplifies our definition of perception as a combination of temporal and spatial impressions.

In practice, circle formation can be successful as long as the ratio of cognition to locomotion speed is high enough. Slowing down locomotion can improve success. A better option is to use simulation to find this ratio and design impressionist robots with sufficient cognition.

#### D. Minimalist perception for Bluebot experiments

Perception was the most computationally expensive task on the actual Bluebots. Our previous work relied on camera-based neighbor detection, made fast enough by reducing each robot to three bright LEDs [1]. To detect LEDs reliably, we had to create a dark underwater environment. Here we present a computationally inexpensive approach to Bluebot detection without LEDs and controlled ambient lighting. To

this end, we colored the robots black such that they were clearly distinguishable from the white tank environment. In order to apply our rapid blob detection in the usual way [1], the raw images were masked (the image area above the water surface was ignored) and negated, such that the robots appeared bright and the background dark (supplement). Snapshots from a color-based milling experiment with four black robots illustrate that circle formation is possible without LEDs (Fig. 7).

#### E. Adaptation 1: Leader following

A single robot programmed to behave differently, i.e., to move on a prescribed trajectory, leads any other robots that apply the circle formation rule. Such leader following works from random initialization, from existing circle formations, and also when an external leader is introduced to a group of circling robots (supplementary video).

#### F. Adaptation 2: Dispersion

Changing the counterclockwise and clockwise turning radii to  $r_0 = 0$  and  $r_1 = \infty$  makes any robot turn on the spot if they do not see another robot, or swim straight toward any other robot they see. As a result, robots aggregate (and form a tiny circle). If we move the field of view to the posterior, i.e., to detect presence in the back instead of in the front, the same robots disperse (supplementary video).

#### G. Theory of circle formation

We provide new analysis and theoretical evidence to show why circles are formed reliably from random initialization. Detailed explanations and assumptions are laid out in the supplement; here we describe the intuition behind circle formation, which is rooted in four observations: (1) A robot rotates clockwise until it detects another robot. (2) Then, the alternate clockwise and counterclockwise movement effected by the circle formation rule leads to a following/homing behavior. (3) A robot never follows multiple robots simultaneously. Multiple robots cannot permanently follow a single without their line-of-sights intersecting. (4) As a result, a scenario in which every robot follows and is followed by

exactly one other robot emerges. In the absence of any leading robot, a circle forms.

## VII. CONCLUSION

We designed impressionist algorithms, capable of coping with infrequent and incomplete sensory updates, for robotic flocking behaviors. For alignment, we conceived an algorithm based on body colorings that does not require the heading of fellow robots. For dispersion, we showed that exact distances to fellow robots are not required to achieve predictable average robot distances. For milling, our simulations confirmed the theoretical scaling of the circle radius with regards to the viewing angle and number of robots. We identified the minimally required cognition speed for successful circle formation and showed how it depends on the viewing angle. And we provided new theoretical insights in how the milling behavior forms. Finally, we demonstrated milling and dispersion with colored Bluebots, and showed multiple adaptations of both behaviors in simulation.

Comparing the impressionist traits, we observe that all the studied flocking behaviors can cope with limited data quality. The metrics for successful alignment and milling are more stringent and the behaviors rely on higher individual robot accuracy than dispersion. During dispersion, robot errors can cancel out and time acts as a low pass filter, rendering exact information unnecessary and allowing for exceptionally low sensing frequencies. We conclude that impressionist algorithms are particularly applicable to collective behaviors expressed at a group level which do not rely on group members making exact decisions at all times.

Overall, our experimental results show that alignment, dispersion, and milling can be achieved with reduced amounts of sensory data, in many cases stemming from sensory impressions rather than exact measurements. Our impressionist algorithms work where traditional algorithms are prone to fail due to lack of sensory data.

## REFERENCES

- [1] F. Berlinger, M. Gauci, and R. Nagpal, “Implicit coordination for 3d underwater collective behaviors in a fish-inspired robot swarm,” *Science Robotics*, vol. 6, no. 50, 2021.
- [2] F. Berlinger, P. Wulkop, and R. Nagpal, “Self-organized evasive fountain maneuvers with a bioinspired underwater robot collective,” in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 9204–9211.
- [3] M. Adioui, J. Treuil, and O. Arino, “Alignment in a fish school: a mixed lagrangian-eulerian approach,” *Ecological Modelling*, vol. 167, no. 1-2, pp. 19–32, 2003.
- [4] E. Carlen, M. C. Carvalho, P. Degond, and B. Wennberg, “A boltzmann model for rod alignment and schooling fish,” *Nonlinearity*, vol. 28, no. 6, p. 1783, 2015.
- [5] C. W. Reynolds, “Flocks, herds and schools: A distributed behavioral model,” in *Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, 1987, pp. 25–34.
- [6] T. Vicsek, A. Czirók, E. Ben-Jacob, I. Cohen, and O. Shochet, “Novel type of phase transition in a system of self-driven particles,” *Physical review letters*, vol. 75, no. 6, p. 1226, 1995.
- [7] M. Gauci, J. Chen, T. J. Dodd, and R. Groß, “Evolving aggregation behaviors in multi-robot systems with binary sensors,” in *Distributed autonomous robotic systems*. Springer, 2014, pp. 355–367.
- [8] R. Olfati-Saber, “Flocking for multi-agent dynamic systems: Algorithms and theory,” *IEEE Transactions on automatic control*, vol. 51, no. 3, pp. 401–420, 2006.
- [9] H. G. Tanner, A. Jadbabaie, and G. J. Pappas, “Stable flocking of mobile agents, part i: Fixed topology,” in *42nd IEEE International Conference on Decision and Control (IEEE Cat. No. 03CH37475)*, vol. 2. IEEE, 2003, pp. 2010–2015.
- [10] N. Moshtagh and A. Jadbabaie, “Distributed geodesic control laws for flocking of nonholonomic agents,” *IEEE Transactions on Automatic Control*, vol. 52, no. 4, pp. 681–686, 2007.
- [11] D. Strömbom, “Collective motion from local attraction,” *Journal of theoretical biology*, vol. 283, no. 1, pp. 145–151, 2011.
- [12] S. Li, R. Batra, D. Brown, H.-D. Chang, N. Ranganathan, C. Hoberman, D. Rus, and H. Lipson, “Particle robotics based on statistical mechanics of loosely coupled components,” *Nature*, vol. 567, no. 7748, pp. 361–365, 2019.
- [13] M. Rubenstein, A. Cornejo, and R. Nagpal, “Programmable self-assembly in a thousand-robot swarm,” *Science*, vol. 345, no. 6198, pp. 795–799, 2014.
- [14] I. Slavkov, D. Carrillo-Zapata, N. Carranza, X. Diego, F. Jansson, J. Kaandorp, S. Hauert, and J. Sharpe, “Morphogenesis in robot swarms,” *Science Robotics*, vol. 3, no. 25, 2018.
- [15] M. Brambilla, E. Ferrante, M. Birattari, and M. Dorigo, “Swarm robotics: a review from the swarm engineering perspective,” *Swarm Intelligence*, vol. 7, no. 1, pp. 1–41, 2013.
- [16] J. A. Preiss, W. Honig, G. S. Sukhatme, and N. Ayanian, “Crazyswarm: A large nano-quadcopter swarm,” *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3299–3304, 2017.
- [17] A. Kushleyev, D. Mellinger, C. Powers, and V. Kumar, “Towards a swarm of agile micro quadrotors,” *Autonomous Robots*, vol. 35, no. 4, pp. 287–300, 2013.
- [18] A. Weinstein, A. Cho, G. Loianno, and V. Kumar, “Visual inertial odometry swarm: An autonomous swarm of vision-based quadrotors,” *IEEE Robotics and Automation Letters*, vol. 3, pp. 1801–1807, 2018.
- [19] G. Vásárhelyi, C. Virág, G. Somorjai, T. Nepusz, A. E. Eiben, and T. Vicsek, “Optimized flocking of autonomous drones in confined environments,” *Science Robotics*, vol. 3, no. 20, 2018.
- [20] G. Vásárhelyi, C. Virág, G. Somorjai, N. Tarcai, T. Szörényi, T. Nepusz, and T. Vicsek, “Outdoor flocking and formation flight with autonomous aerial robots,” *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3866–3873, 2014.
- [21] S. Hauert, S. Leven, M. Varga, F. Ruini, A. Cangelosi, J.-C. Zufferey, and D. Floreano, “Reynolds flocking in reality with fixed-wing robots: communication range vs. maximum turning rate,” in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2011, pp. 5015–5020.
- [22] K. McGuire, C. De Wagter, K. Tuyls, H. Kappen, and G. C. de Croon, “Minimal navigation solution for a swarm of tiny flying robots to explore an unknown environment,” *Science Robotics*, vol. 4, 2019.
- [23] E. Soria, F. Schiano, and D. Floreano, “The influence of limited visual sensing on the reynolds flocking algorithm,” *IEEE International Conference on Robotic Computing (IRC)*, pp. 138–145, 2019.
- [24] D. S. Calovi, U. Lopez, S. Ngo, C. Sire, H. Chaté, and G. Theraulaz, “Swarming, schooling, milling: phase diagram of a data-driven fish school model,” *New journal of Physics*, vol. 16, no. 1, 2014.
- [25] R. Sepulchre, D. A. Paley, and N. E. Leonard, “Stabilization of planar collective motion with limited communication,” *IEEE Transactions on Automatic Control*, vol. 53, no. 3, pp. 706–719, 2008.
- [26] K. Soltan, J. O’Brien, J. Dusek, F. Berlinger, and R. Nagpal, “Biomimetic actuation method for a miniature, low-cost multi-jointed robotic fish,” *OCEANS 2018 MTS/IEEE Charleston*, pp. 1–9, 2018.
- [27] F. Berlinger, M. Duduta, H. Gloria, D. Clarke, R. Nagpal, and R. Wood, “A modular dielectric elastomer actuator to drive miniature autonomous underwater vehicles,” *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3429–3435, 2018.
- [28] F. Berlinger, M. Saadat, H. Haj-Hariri, G. V. Lauder, and R. Nagpal, “Fish-like three-dimensional swimming with an autonomous, multi-fin, and biomimetic robot,” *Bioinspiration & Biomimetics*, vol. 16, no. 2, p. 026018, 2021.
- [29] M. Saadat, F. Berlinger, A. Sheshmani, R. Nagpal, G. V. Lauder, and H. Haj-Hariri, “Hydrodynamic advantages of in-line schooling,” *Bioinspiration & Biomimetics*, vol. 16, no. 4, p. 046002, 2021.
- [30] F. Berlinger, J. Dusek, M. Gauci, and R. Nagpal, “Robust maneuverability of a miniature, low-cost underwater robot using multiple fin actuation,” *IEEE Robotics and Automation Letters*, vol. 3, no. 1, 2017.
- [31] A. Jadbabaie, J. Lin, and A. S. Morse, “Coordination of groups of mobile autonomous agents using nearest neighbor rules,” *IEEE Transactions on automatic control*, vol. 48, no. 6, pp. 988–1001, 2003.